

Cisco Spaces: Connector 3 Configuration Guide

First Published: 2022-06-24

Last Modified: 2026-08-03

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface ix

Audience ix

Conventions ix

Related Documentation x

Communications, services, and additional information x

Cisco Bug Search Tool xi

Documentation feedback xi

CHAPTER 1

Overview 1

Introduction to Connector 3 1

PART I

Getting Started 3

CHAPTER 2

Prerequisites 5

Prerequisites for Configuring Connector 3 5

Recommended Deployment Architecture - Advanced2 6

Switch from E1000 Adapter to VMXNET3 Adapter 6

CHAPTER 3

Scale Specifications 9

Cisco Spaces: Connector 3 use case and capacity details 9

CHAPTER 4

Open Ports (Wireless) 13

Information About Open Ports (Wireless) 13

OpenRoaming Firewall Rules 14

CHAPTER 5

Open Ports (Wired) 17

Open ports for IoT service (wired) 17

CHAPTER 6 Open Ports (Location Service) 19

Open Ports for Location Service 19

CHAPTER 7 Open Ports (Gateway Service) 21

Open Ports for Gateway Service and Sensor Connect 21

PART II Configuration 25

CHAPTER 8 Initial Setup 27

Initial Setup of Cisco Spaces: Connector 27

Activating Connector 3 on Cisco Spaces 28

Upgrading the Connector from Cisco Spaces Dashboard 35

Upgrading the Connector Using CLI 39

Upgrading the Connector Using Downloaded Connector Package 40

CHAPTER 9 Cisco Spaces: Connector AMI 43

Launch Connector 3 as an EC2 Instance from AMI 43

CHAPTER 10 Cisco Spaces: Connector Hyper-V 53

Creating a Virtual Switch 53

Downloading and Deploying HYPER-V 60

CHAPTER 11 Cisco Spaces: Connector Azure VMware 71

Cisco Spaces: Connector Azure VMware 71

Creating an Azure VMware solution (or Private Cloud) 72

Creating an Azure Virtual Network 76

CHAPTER 12 Cisco Spaces: Connector OVA 83

Deploying the Connector 3 OVA (Single Interface) 83

Deploying the Cisco Spaces: Connector OVA (Dual Interface) 91

Using Snapshots for Backup 99

CHAPTER 13 **Cisco Spaces: Connector VM on Nutanix Environment** 101
Cisco Spaces: Connector Nutanix cloud platform support 101
Set Up Connector VM on Nutanix Environment 101

CHAPTER 14 **Cisco Spaces: Connector VM on Azure Environment** 107
Cisco Spaces: Connector Azure cloud platform support 107
Set Up Connector VM on Azure Environment 107

CHAPTER 15 **Cisco Spaces: Connector VM on OpenShift Platform** 115
Connector VM on OpenShift Platform 115
Deploy a Connector VM on the OpenShift Platform 115

CHAPTER 16 **Connector on Cisco Spaces** 123
Activating Connector 3 on Cisco Spaces 123
Monitor the Status of Service Installation 131
Bulk Credential Update 132
Bulk Update Wireless Controller Credentials 133
Bulk Update Switch Credentials 134

CHAPTER 17 **Connector GUI** 137
Connector GUI 137
Configuring Privacy Settings 138

CHAPTER 18 **Proxy** 139
Configure a Proxy 139
Configure a Transparent Proxy 141

CHAPTER 19 **High Availability** 145
Configuring Connectors as VIP Paired 145
Connector Active-Active 150
Restrictions for Active-Active 151
Configuring Connectors in Active-Active 151

CHAPTER 20 **Cisco Spaces: FIPS Support in Connector** 155

FIPS Overview 155

Protocol Requirements 155

FIPS Support 156

Enabling FIPS 156

Disabling FIPS 157

Adding a FIPS-Enabled Catalyst 9800 Controller to a FIPS-Enabled Connector 159

Enabling FIPS 159

Configuring FIPS Authorization Key in the Controller 160

Obtaining a Certificate from the Connector 160

Importing a CA Certificate into the Trustpool 161

Reinitiating the NMSP Connection on the Controller 162

PART III **Troubleshooting** 163

CHAPTER 21 **Troubleshooting Tools** 165

Enable Debug Logs 165

Resetting the Password 166

Monitor Service Metrics 166

CHAPTER 22 **Troubleshooting Scenarios** 169

Connectivity Issues Between Connector and Cisco Spaces 169

Unresponsive Connector, or Failure of SSH to Connector 173

Instance is Corrupted or Deleted 176

Service Crash, or Restart Services 176

Upgrade has Failed, or How To Forcibly Push Configurations to Instances 177

Managing Weak Algorithms from SSHD Configurations 178

Weak SSHD Key Exchange (kex) Algorithms 178

Disable Weak SSHD Key Exchange (kex) Algorithms 178

Weak Host Key (hostkey) Algorithms 179

Disable Weak Host Key (hostkey) Algorithms 179

Weak Cipher Algorithms 180

Disable Cipher Algorithms 181

Weak Message Authentication Code (MAC) Algorithms 181
Disable Message Authentication Code (MAC) Algorithms 182
Weak Public Key (pubkey) Algorithms 182
Disable Public Key (pubkey) Algorithms 183

PART IV **Services** 185

CHAPTER 23 **Gateway Service with Sensor Connect** 187
Gateway Service overview and operating model 187
Gateway Service Operating Principles 187
Configure Gateway Service with Sensor Connect 188
Gateway Service Components 189

CHAPTER 24 **Location Service** 191
Compatibility Matrix for Cisco Spaces: Connector: Location service 191
Open Ports for Location Service 195

CHAPTER 25 **IoT Service (Wireless)** 197
Overview of Cisco Spaces: IoT Service (Wireless) 197
Components of Cisco Spaces: IoT Service 198
Compatibility Matrix for IoT Service (Wireless) 201
Prerequisites of IoT Service (Wireless) 203
Open Ports for IoT Service (Wireless) 203
Configure IoT Service (Wireless) 204
Verify IoT Streams for Catalyst 9800 Controller 206
Verify access points 207

CHAPTER 26 **Connector Gateway with Sensor Connect** 211
Connector Gateway with Sensor Connect 211
Create a Connector with Sensor Connect 211
Add a Controller for Sensor Connect 212
Install Sensor Connect 212
Manage Sensor Connect After Installation 213
View Sensor Connect Metrics 213

CHAPTER 27 IoT Service (Wired) 215
Overview 215
Overview of IoT Service (Wired) 215
Compatibility matrix for IoT Service (Wired) 216
Prerequisites for Cisco Spaces: IoT Service (Wired) 217
Prerequisites for Cisco Spaces: IoT Service (Wired) 219
Open ports for IoT service (wired) 221
Configure IoT Service (Wired) 222
Verify if Cisco Catalyst 9300 and 9400 Series Switches are Added to the Connector 231

CHAPTER 28 Hotspot Service 233
Configure Hotspot Service 233
Connector Dashboard: Hotspot service 235
Open Ports for Hotspot Service 235

CHAPTER 29 Local Firehose 237
Local Firehose Service 237
Configure Local Firehose Service 237
Connector Dashboard: Local Firehose Service 240

APPENDIX A Connect Connector to Cisco AireOS Wireless Controller 243
Configure and Test Connectivity Between a Connector and AireOS Controller 243

APPENDIX B Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers 249
Configure and Test the Connection Between Connector and Catalyst 9800 Controller 249

APPENDIX C Connect Connector to Cisco Catalyst 9300 or 9400 Series Switches 257
Connecting a connector to Cisco Catalyst 9300 and 9400 Series Switches 257



Preface

- Audience, on page ix
- Conventions, on page ix
- Related Documentation, on page x
- Communications, services, and additional information, on page x

Audience

This document is meant for Cisco Spaces network and IT administrators who deploy Cisco Spaces to monitor, manage, and optimize usage of assets in an organization.

Conventions

This document uses the following conventions.

Table 1: Conventions

Convention	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string. Otherwise, the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
<>	Nonprinting characters such as passwords are in angle brackets.
[]	Default responses to system prompts are in square brackets.

Convention	Indication
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.



Note Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.



Tip Means the following information will help you solve a problem.



Caution Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

Related Documentation

Cisco Spaces

Cisco Spaces Configuration Guide

Connector

- *Cisco Spaces: Connector Configuration Guide*
- *Cisco Spaces: Connector Command Reference Guide*
- *Release Notes for Cisco Spaces: Connector*

IoT Service

- *Cisco Spaces: IoT Service Configuration Guide (Wireless)*
- *Cisco Spaces: IoT Service Configuration Guide (Wired)*

Communications, services, and additional information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions, and services, visit Cisco DevNet.

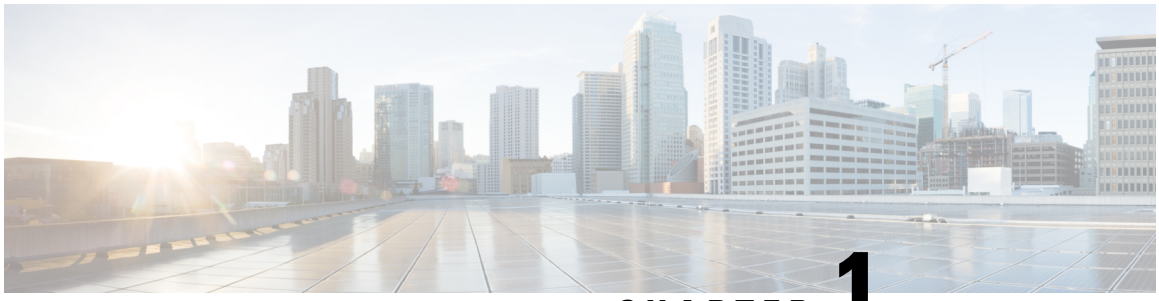
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a gateway to the Cisco bug-tracking system, which maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. The BST provides you with detailed defect information about your products and software.

Documentation feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.



CHAPTER 1

Overview

Note Starting from December 2023, Cisco Spaces: Connector 2.x has entered maintenance mode, and only security updates will be available up to June 2024. Extended support is limited to critical bug fixes, offered until October 2024. We strongly recommend that you upgrade to connector 3. To migrate from Connector 2.x to Connector 3, see Migrate from Connector 2.x to Connector 3.

- Introduction to Connector 3 , on page 1

Introduction to Connector 3

Cisco Spaces: Connector Release 3 (subsequently referred to as Connector 3) is a fully redesigned version of the Cisco Spaces: Connector Release 2.x, with the capability to efficiently manage multiple services that connect to different network devices such as wireless controllers, access points (APs), and switches. connector gathers and aggregates data from these devices and sends the data to Cisco Spaces.

With connector 3, you can do the following:

- Add or remove new services from Cisco Spaces.
- Perform advanced troubleshooting with the debugging, log upload, and restart functionalities in Cisco Spaces.
- Obtain detailed metrics for each service, such as, CPU, memory, connectivity, and up or down status.
- Configure Virtual IP address (VIP) pairs or active-active pairs that allow for high availability. You can view details of each instance that is a part of a high-availability pair.
- Monitor connector 3 and device status that are aggregated from each instance of connector.
- View how services are running on each instance, their upgrade status, and so on.
- Perform actions on an instance, such as restarting of services.
- Configure instances for connector. Device status is aggregated from each connector instance for monitoring.

Important If you are using the HUGS solution, you should continue to use Connector 3.2 while validation for Securitas is pending

Connector 3 sends data to Cisco Spaces over HTTPS; a proxy can also be used to route data.

See Initial Setup, Upgrading the Connector, and Migrating from Connector 2.x to Connector 3.

Note The term wireless controller is used in this document to collectively refer to the following:

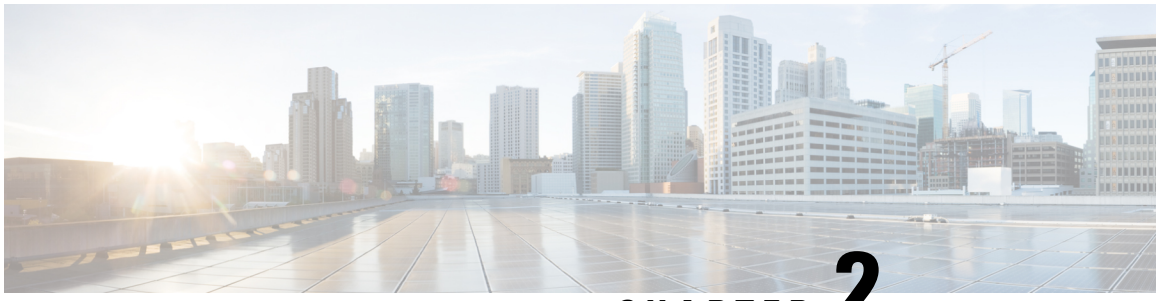
- Cisco AireOS Wireless Controller or AireOS controller
- Cisco Catalyst 9800 Series Wireless Controller or Catalyst 9800 controller
- Cisco Embedded Wireless Controller on Cisco Catalyst Access Points (Cisco EWC-AP)



PART I

Getting Started

- Prerequisites, on page 5
- Scale Specifications, on page 9
- Open Ports (Wireless), on page 13
- Open Ports (Wired), on page 17
- Open Ports (Location Service), on page 19
- Open Ports (Gateway Service), on page 21



CHAPTER 2

Prerequisites

- Prerequisites for Configuring Connector 3, on page 5
- Recommended Deployment Architecture - Advanced2, on page 6
- Switch from E1000 Adapter to VMXNET3 Adapter, on page 6

Prerequisites for Configuring Connector 3

- Make sure you allow access to necessary endpoints based on the region of your Cisco Spaces account. Refer to the following table for the endpoints that must be enabled:

Table 2: Enable Endpoints

Cisco Spaces Account	Endpoint to be Enabled
https://dnaspaces.io	https://connector.dnaspaces.io
https://dnaspaces.eu	https://connector.dnaspaces.eu
https://ciscospaces.sg	https://connector.ciscospaces.sg
https://ciscospaces.io	https://connect.ciscospaces.io
https://ciscospaces.eu	https://connect.ciscospaces.eu
https://ciscospaces.sg	https://connect.ciscospaces.sg

- Allow firewall to access these IP's addresses based on the respective regions.

	Primary IP Address	Primary IP Address (ciscospaces domain endpoints)	Disaster Recovery
US Setup	• 52.20.144.155 • 34.231.154.95	• 34.231.191.140 • 35.168.215.234	• 54.175.92.81 • 54.183.58.225
EU Setup	• 63.33.127.190 • 63.33.175.64	• 99.81.112.199 • 54.77.157.59	• 3.122.15.26 • 3.122.15.7
Singapore Setup	• 13.228.159.49	• 54.254.46.220	• 3.214.251.223

	Primary IP Address	Primary IP Address (ciscospaces domain endpoints)	Disaster Recovery
	• 54.179.105.241	• 47.130.91.208	• 3.122.57.46

- The Cisco Spaces: Connector works as expected when two tokens are generated and configured on two separate instances. However, if an additional token is generated and configured on a third connector instance, a certificate will not be issued, causing the hotspot to fail.

  Ensure that no more than two tokens are generated for each Connector and configured on the Cisco Spaces: Connector instances to prevent hotspot failures.

- Connector needs to be able to reach a Domain Name System (DNS) server. If you set up an explicit proxy, ensure that Connector 3 maintains the ability to communicate through this proxy.
- VMware ESXi 7.0, 8.0 or 9.0.1.0.
- VMware vCenter 7.0 or 8.0
- Virtual machine size: Advanced 2 (Recommended)
- Minimum bandwidth required: 4 Mbps

Note Make sure that the x86-64-v2 CPU is available for Enterprise Linux 9. Also, ensure that the x86-64-v2 CPU supports the following flags: SSE3, SSE4_1, SSE4_2, and SSSE3.

Recommended Deployment Architecture - Advanced2

This is the recommended deployment architecture for IoT and location combined use case:

- Virtual machine size (vCPU): 8
- Memory: 16 GB
- Hard Disk: 120 GB

Note We recommend that you use specifications of 8 vCPUs and 16 GB of RAM for optimal performance and efficiency.

Switch from E1000 Adapter to VMXNET3 Adapter

Use this procedure to improve network performance and avoid retokening the Connector after upgrading to Connector Release 3.3.

This procedure applies to brownfield deployments that use an E1000 network adapter. Retaining the existing MAC address when you install the VMXNET3 adapter prevents the Connector from requiring retokening.

Before you begin

- Upgrade the Connector to Release 3.3.
- Ensure that you can access the VMware ESXi host or vCenter that manages the Connector virtual machine (VM).
- Ensure that you have permission to create snapshots, power off the VM, and modify its virtual hardware.

Procedure

Step 1 Access the VMware ESXi host or vCenter that manages the Connector VM.

Step 2 Take a snapshot of the VM.

The snapshot provides a recovery point if the adapter replacement is unsuccessful.

Step 3 Power off the VM.

Step 4 Open the VM settings by clicking **Edit Settings**.

Step 5 Expand the existing network adapter settings and record its MAC address.

Important
If the MAC address is within the VMware reserved MAC address range, you cannot reuse it. Deploy a new Connector instead of continuing with this procedure.

Step 6 Remove the existing E1000 network adapter.

Step 7 Add a network adapter and select **VMXNET3** as the adapter type.

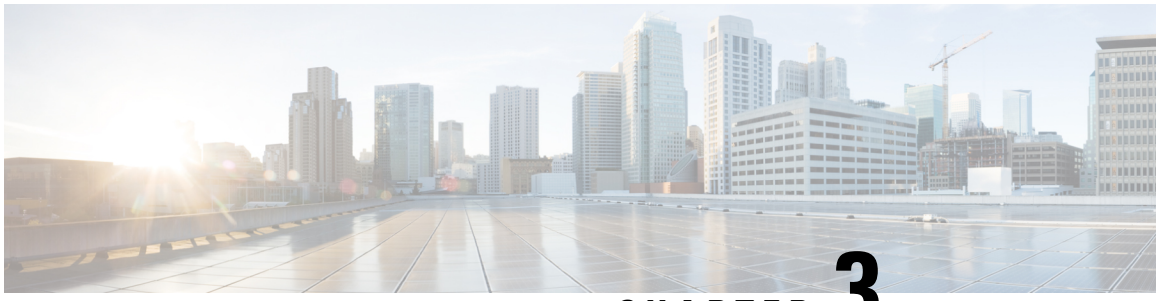
Step 8 Select **Manual MAC assignment** and enter the MAC address that you recorded.

Step 9 Save the VM settings and power on the VM.

The Connector VM uses the VMXNET3 network adapter with its existing MAC address. The adapter change improves network performance without requiring the Connector to be retokened.

What to do next

Verify that the Connector starts successfully and can communicate over the network.



CHAPTER 3

Scale Specifications

- Cisco Spaces: Connector 3 use case and capacity details, on page 9

Cisco Spaces: Connector 3 use case and capacity details

This table outlines the three types of Cisco Spaces: Connector types, their use cases, and key scaling KPIs such as AP counts, client or tag counts, and outgoing message rates.

These are the key factors affecting scale:

- Connector Message Rate Limitations:
  - The connector works only within outgoing message rate limits.
  - If the message rate limits are reached, the connector's ability to scale becomes restricted.
  - We recommend deploying within these limits to maintain system stability.
- CPU Type Requirement: The type of CPU on the connector’s host server affects scale and performance.
- Network Delays: High network latency reduces connector performance.
- Device Filtering Requirement for IoT Service: To fully utilize the connector for IoT services, you must enable device filtering. For more information, see the Cisco Spaces: IoT Service Configuration Guide (Wireless).

Scale specifications are tested under these environments: VMware ESXi platform with an Intel® Xeon® Platinum 8362 CPU and an **E1000** network adapter.

Table 3: Scale specification for Connector 3.2

Connector Type	Use Case	AP Count	Client/Tag Count	Connector Outgoing Message Rate
Standard	Location (NMSP) Only	9K	Client Count: 90K (Wireless Clients, RFIDs & Rogues)	21K NMSP msgs/sec

Connector Type	Use Case	AP Count	Client/Tag Count	Connector Outgoing Message Rate
Advanced 1	Location (NMSP) Only	15K	Client Count: 150K	38K NMSP msgs/sec
	IoT Only	3K	BLE Tag Count: 25K	150K BLE msgs/sec
	Location (NMSP) + IoT	12K(2K APs is doing IoT and location; 10K APs is doing only location)	Client Count: 100K BLE Tag Count: 20K	17K NMSP msgs/sec 120K BLE msgs/sec
Advanced 2	Location (NMSP) Only	25K	Client Count: 250K (Wireless Clients, RFIDs & Rogues)	44K NMSP msgs/sec
	IoT Only	4.5K	BLE Tag Count: 40K	220K BLE msgs/sec
	Location (NMSP) + IoT	18K (3K APs doing both IoT and Location; 15K APs is doing location only)	Client Count: 150K BLE Tag Count: 30K	38K NMSP msgs/sec 170K BLE msgs/sec

Scale specifications are tested under these environments: VMware ESXi 8.0 Update 3 platform with an Intel® Xeon® Platinum 8581V CPU and a **VMXNET3** network adapter.

Important To migrate from E1000 to VMXNET3 adapter support, upgrade your connector to 3.3 July 2026 release.

Note If you are upgrading from Cisco Spaces: Connector Release 3.2 February 2026, migrate from the **E1000** network adapter to the **VMXNET3** network adapter. For migration instructions, refer to Switch from E1000 Adapter to VMXNET3 Adapter.

The recommended network bandwidth for Cisco Spaces Cloud is 1 gigabit per second (Gbps).

Table 4: Recommendations

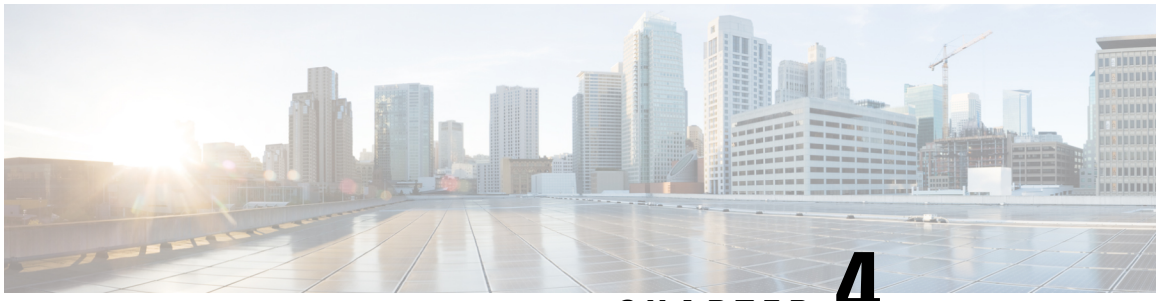
Cisco Spaces: Connector Link	Measured/Estimation	Recommended (with 30% headroom)
• egress to Cisco Spaces Cloud	~624 Mbps	1 Gbps
• ingress from Access Points • egress to Cisco Spaces Cloud	~816 Mbps	1 Gbps NIC (near saturation)

Table 5: Scale specification for Connector 3.3

Connector Type	Use Case	AP Count	Client/Tag Count	Connector Outgoing Message Rate
Advanced 1	Location (NMSP)	15K	Client Count: 150K (Wireless Clients, RFIDs & Rogues)	38K NMSP msgs/sec
	Location (NMSP) + IoT	10K	Client Count: 100K(Wireless Clients, RFIDs & Rogues) BLE Tag Count: 100K	17K NMSP msgs/sec 600K BLE Scan Rate/sec
Advanced 2	Location (NMSP)	25K	Client Count: 250K (Wireless Clients, RFIDs & Rogues)	44K NMSP msgs/sec
	Location (NMSP) + IoT	18K	Client Count: 150K (Wireless Clients, RFIDs & Rogues) BLE Tag Count: 150K	38K NMSP msgs/sec 900K BLE Scan Rate/sec

Table 6: Scale specification for Connector 3.3 with Gateway Service and Sensor Connect

Connector Type	Use Case	AP Count	BLE Tag Count	Connector Outgoing Message Rate
Advanced 2	Gateway	26K	200K	1.3M BLE scan rate/sec



CHAPTER 4

Open Ports (Wireless)

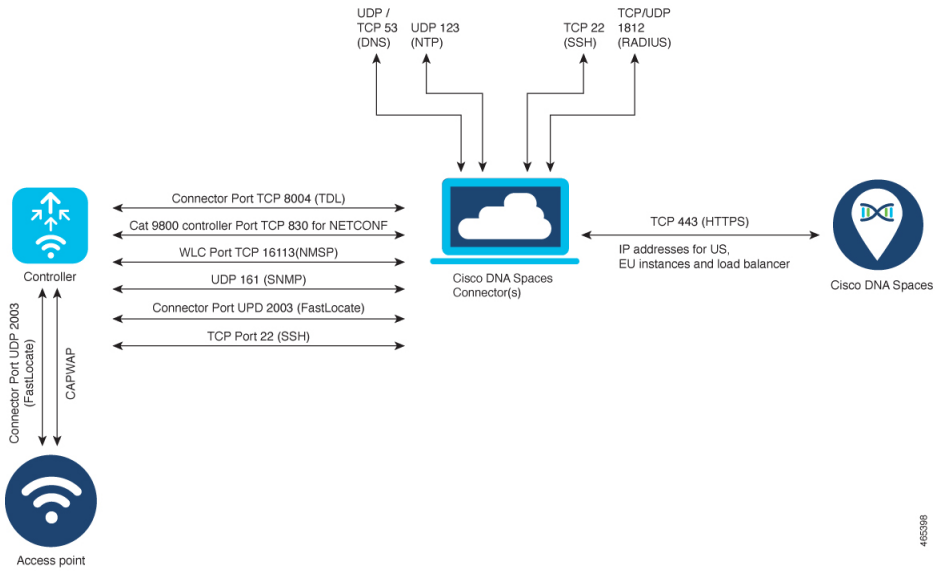
- Information About Open Ports (Wireless), on page 13
- OpenRoaming Firewall Rules, on page 14

Information About Open Ports (Wireless)

This chapter lists the connector ports that need to be open for the proper functioning of various services or protocols.

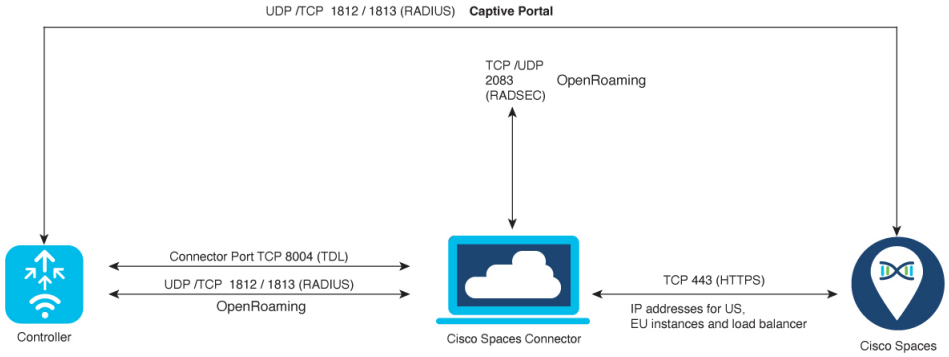
These ports need to be opened to allow for the basic functionality of Cisco Spaces.

Figure 1: Basic Functionality



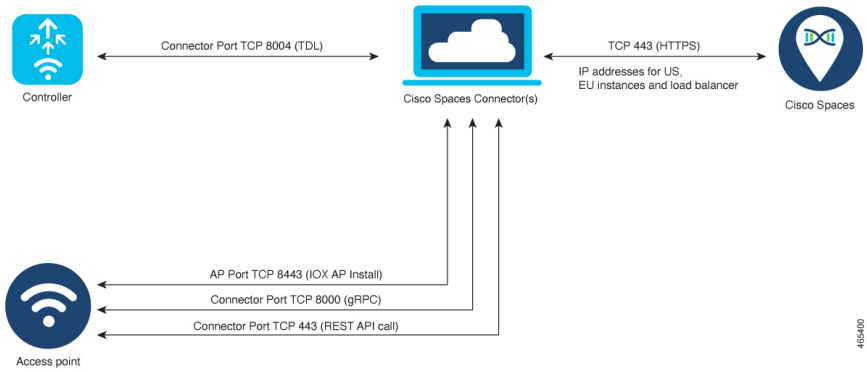
In addition to basic functionality, additional ports need to be opened for other additional functionality like guest onboarding and IoT Services.

Figure 2: Guest Onboarding



These ports need to be opened for configuring IoT Services (wireless). To configure IoT Services (wired), refer to Open Ports (Wired).

Figure 3: IoT Services

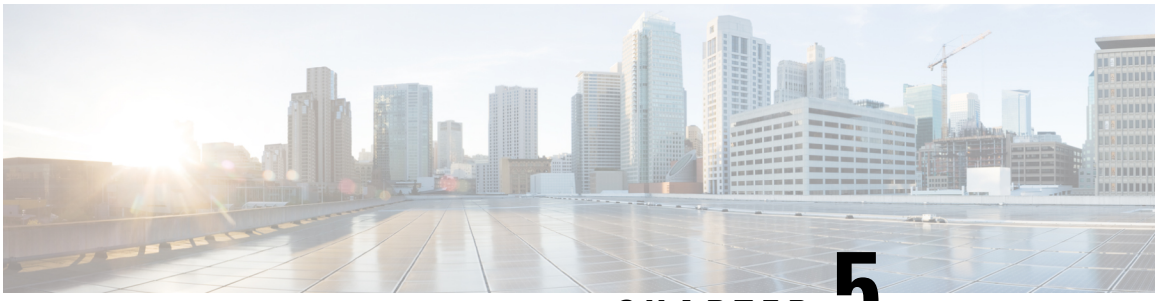


OpenRoaming Firewall Rules

Table 7: OpenRoaming Firewall Rules

Source IP Address	Destination IP Address	Direction	Transport	Source Port	Destination Port	Protocol	Further information
Cisco AireOS Wireless Controller IP address	Connector	Unidirectional	UDP and TCP	Any	1812, 1813	Remote Authentication Dial-In User Service (RADIUS)	Communication between Connector and Cisco AireOS Wireless Controller for OpenRoaming client's RADIUS messages.

Source IP Address	Destination IP Address	Direction	Transport	Source Port	Destination Port	Protocol	Further information
Connector	Any	Unidirectional	TCP	Any	2083	RADIUS over TLS (RADSEC)	Communication between Connector and OpenRoaming Identity Providers
Connector	Any	Unidirectional	TCP	Any	443		HTTPS for CSR signing - OpenRoaming Membership



CHAPTER 5

Open Ports (Wired)

- Open ports for IoT service (wired), on page 17

Open ports for IoT service (wired)

This section lists the connector ports that must be open for the proper functioning of each service or protocol.

Figure 4: Open Ports for IoT Service (Wired) with the IoT Gateway

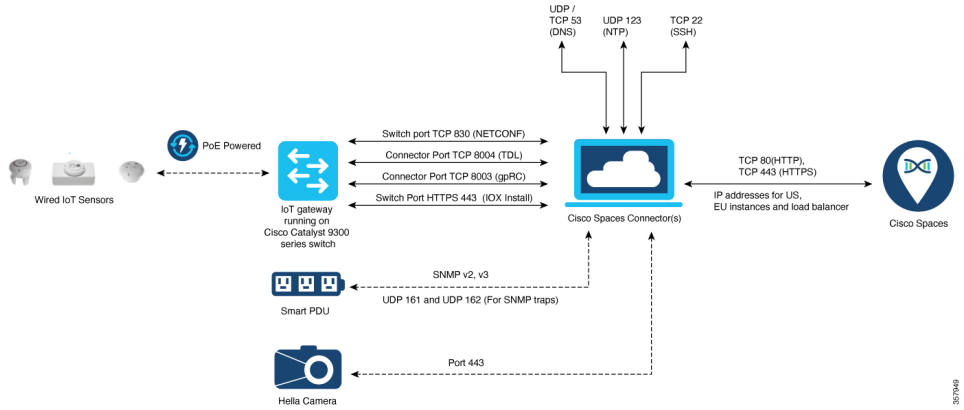
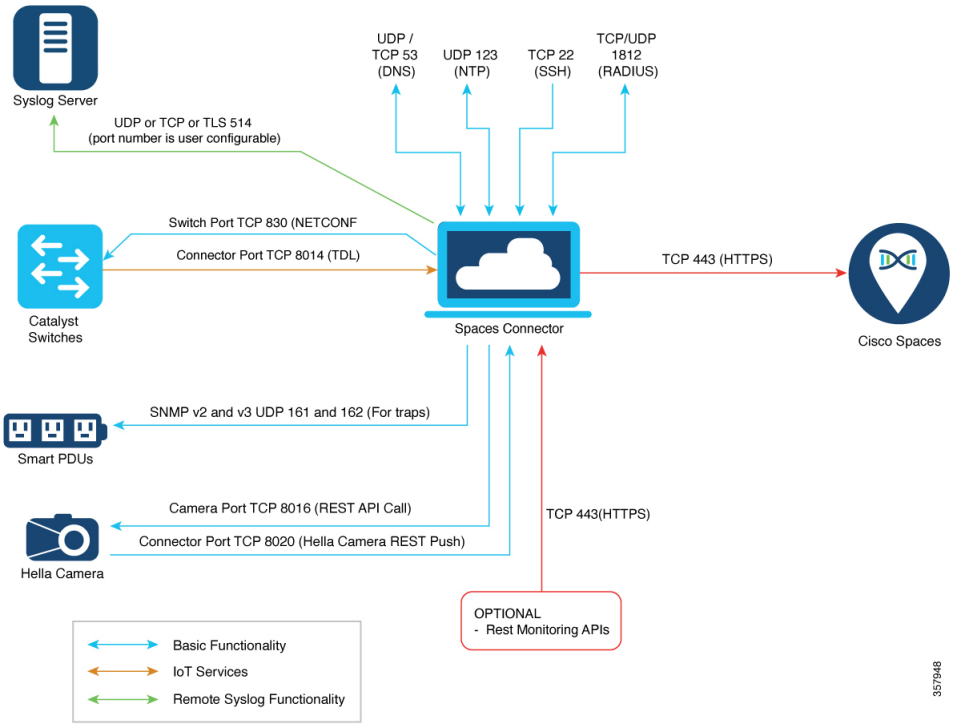
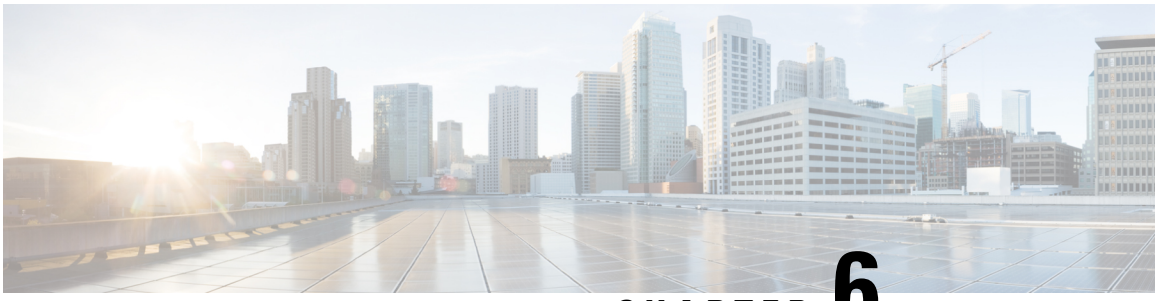


Figure 5: Open Ports for IoT Service (Wired) without the IoT Gateway





CHAPTER 6

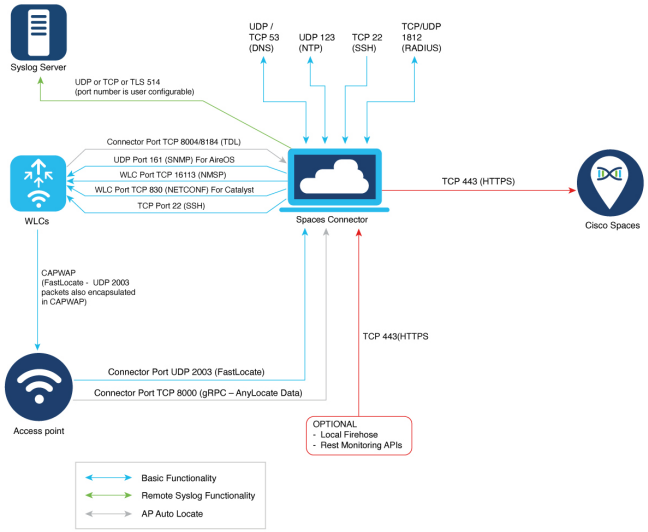
Open Ports (Location Service)

- Open Ports for Location Service, on page 19

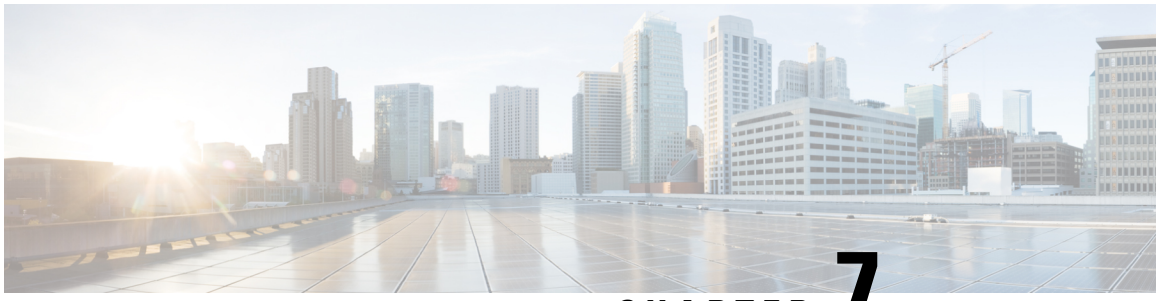
Open Ports for Location Service

This section lists the connector ports that must be open for the proper functioning of location service.

Figure 6: Open Ports for Location Service



Test the connectivity between the connector and the wireless controller. See Configure and Test Connectivity between the Connector 3 and AireOS controller or Configure and Test the Connectivity between a Connector 3 and a Catalyst 9800 controller.



CHAPTER 7

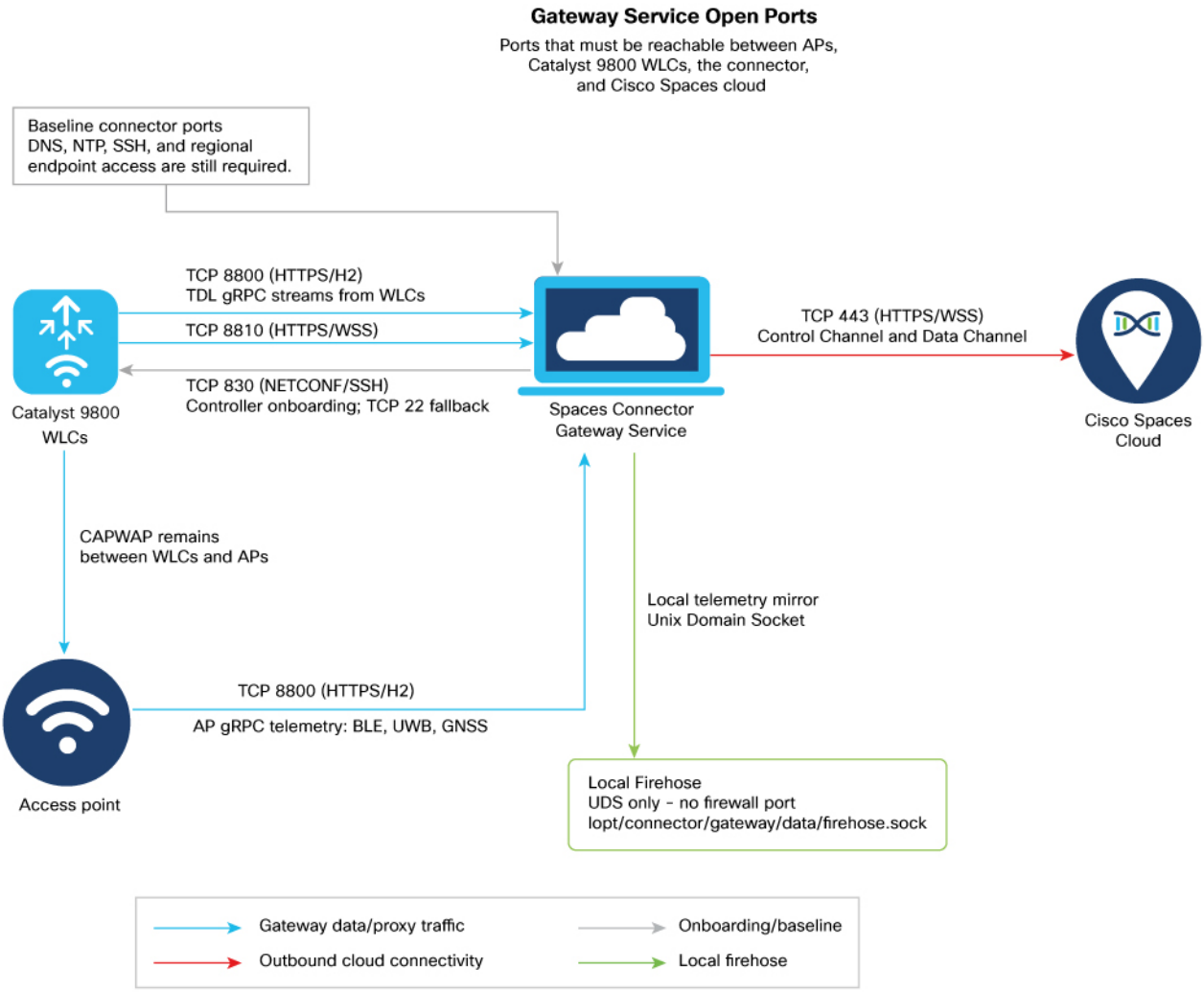
Open Ports (Gateway Service)

• Open Ports for Gateway Service and Sensor Connect, on page 21

Open Ports for Gateway Service and Sensor Connect

This chapter lists the connector ports required for the proper functioning of Gateway Service. Additional ports must be opened, in addition to the basic Cisco Spaces connector ports.

Figure 7: Open Ports for Gateway Service



To verify IP address setup details for different regions, refer to Prerequisites for Configuring Connector 3.

Table 8: Gateway Service Firewall Rules

Source IP Address	Destination IP Address	Direction	Transport	Source Port	Destination Port	Protocol	Further Information
Access Point IP address	Connector	Unidirectional	TCP	Any	8800	HTTPS/HTTP/2	gRPC telemetry streams for BLE, UWB, and GNSS data.
Catalyst 9800 Wireless Controller IP address	Connector	Unidirectional	TCP	Any	8800	HTTPS/HTTP/2	TDL gRPC telemetry streams from Catalyst 9800 wireless controllers.

Source IP Address	Destination IP Address	Direction	Transport	Source Port	Destination Port	Protocol	Further Information
Catalyst 9800 Wireless Controller IP address	Connector	Unidirectional	TCP	Any	8810	HTTPS/HTTP/2	WLC proxy connections for control channel.
Connector	Catalyst 9800 wireless controller IP address	Unidirectional	TCP	Any	830	NETCONF/SSH	Controller onboarding and configuration. If NETCONF uses the default SSH port, open TCP port 22 instead.
Connector	Cisco Spaces cloud	Unidirectional	TCP	Any	443	HTTPS/WSS	Control channel and data channel connectivity to Cisco Spaces.

Region	Control/Data Endpoint
US (legacy)	`connector.dnaspaces.io`
EU (legacy)	`connector.dnaspaces.eu`
US	`connect.ciscospaces.io`
EU	`connect.ciscospaces.eu`
Singapore	`connect.ciscospaces.sg`

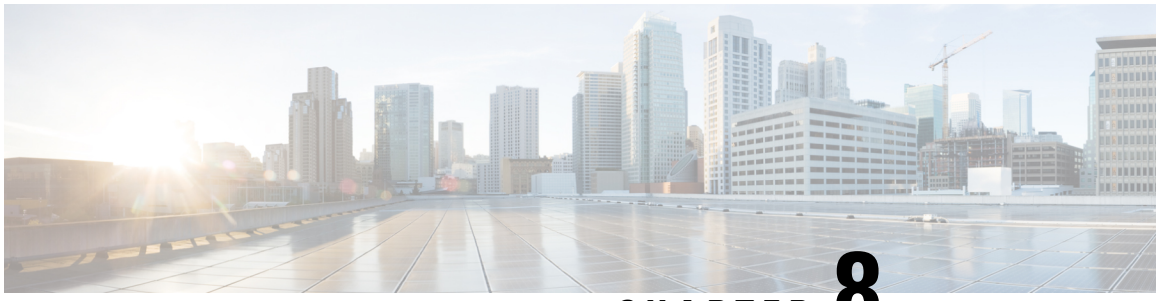
The Local Firehose uses a Unix Domain Socket and does not require firewall rules. The socket path is `/opt/connector/gateway/data/firehose.sock`.



PART II

Configuration

- Initial Setup, on page 27
- Cisco Spaces: Connector AMI, on page 43
- Cisco Spaces: Connector Hyper-V, on page 53
- Cisco Spaces: Connector Azure VMware, on page 71
- Cisco Spaces: Connector OVA , on page 83
- Cisco Spaces: Connector VM on Nutanix Environment, on page 101
- Cisco Spaces: Connector VM on Azure Environment, on page 107
- Cisco Spaces: Connector VM on OpenShift Platform, on page 115
- Connector on Cisco Spaces , on page 123
- Connector GUI, on page 137
- Proxy, on page 139
- High Availability, on page 145
- Cisco Spaces: FIPS Support in Connector, on page 155



CHAPTER 8

Initial Setup

- Initial Setup of Cisco Spaces: Connector, on page 27
- Activating Connector 3 on Cisco Spaces, on page 28
- Upgrading the Connector from Cisco Spaces Dashboard, on page 35
- Upgrading the Connector Using CLI, on page 39
- Upgrading the Connector Using Downloaded Connector Package, on page 40

Initial Setup of Cisco Spaces: Connector

To get the Cisco Spaces: Connector up and running, perform these steps:

1. Install connector 3 in your local deployment network. See Deploying the Connector 3 OVA (Single Interface), on page 83
2. On the Cisco Spaces dashboard, create a Cisco Spaces: Connector and generate a token for connector. See Activating Connector 3 on Cisco Spaces, on page 28
3. Configure this token on the deployed Cisco Spaces: Connector. This establishes a connection between Cisco Spaces and the deployed Cisco Spaces: Connector. The equivalent connector 3 (based on the token) on the Cisco Spaces now turns active. See Activating Connector 3 on Cisco Spaces, on page 28
4. Add the services based on your required workflow on Cisco Spaces.

Table 9: Enabling Services

Service	Link
Service manager service	Enabled by default.
IoT service (wireless)	For information, see Configure IoT Service (Wireless), on page 204.
IoT service (wired)	For information, see Configure IoT Service (Wireless), on page 204.
Hotspot service	For information, see Configure Hotspot Service, on page 233.
Local firehose service	For information, see Configure Hotspot Service, on page 233.

Activating Connector 3 on Cisco Spaces

This section provides information about how to activate a deployed connector on your Cisco Spaces account.

Using the following procedure, you generate a token for a deployed connector that you want to add to your Cisco Spaces account. Note that you need a separate token for each deployed connector. Each token is specific to a connector and hence enables Cisco Spaces to identify and connect to connector.

Cisco Spaces supports multiple connectors, and you can associate each connector with one or multiple wireless controllers.

Note A Cisco Spaces: Connector instance can communicate with only one Cisco Spaces account at a time.

Before you begin

Download and deploy the Cisco Spaces: Connector OVA. See Deploying the Connector 3 OVA (Single Interface), on page 83

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left navigation pane, choose **Setup > Wireless Networks**.

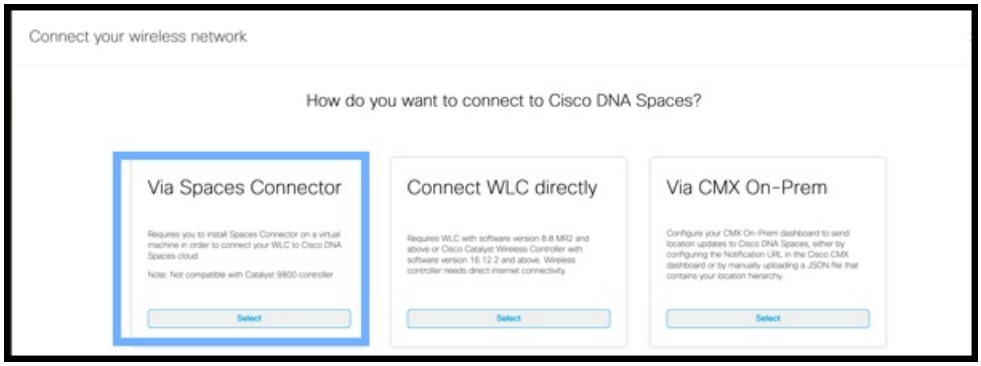
Step 3 In the **Get your wireless network connected with Cisco DNA Spaces** area, click **Add New**.

Step 4 In the **Cisco AireOS Controller/Catalyst 9800 Wireless Controller** area, click **Select.**

Figure 8: Choose Cisco AireOS Controller/Catalyst 9800 Wireless Controller

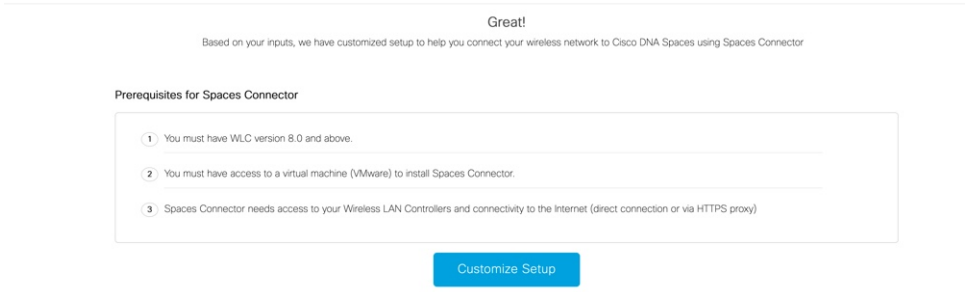
Step 5 In the **Via Spaces Connector** area, click **Select**.

Figure 9: Via Spaces Connector



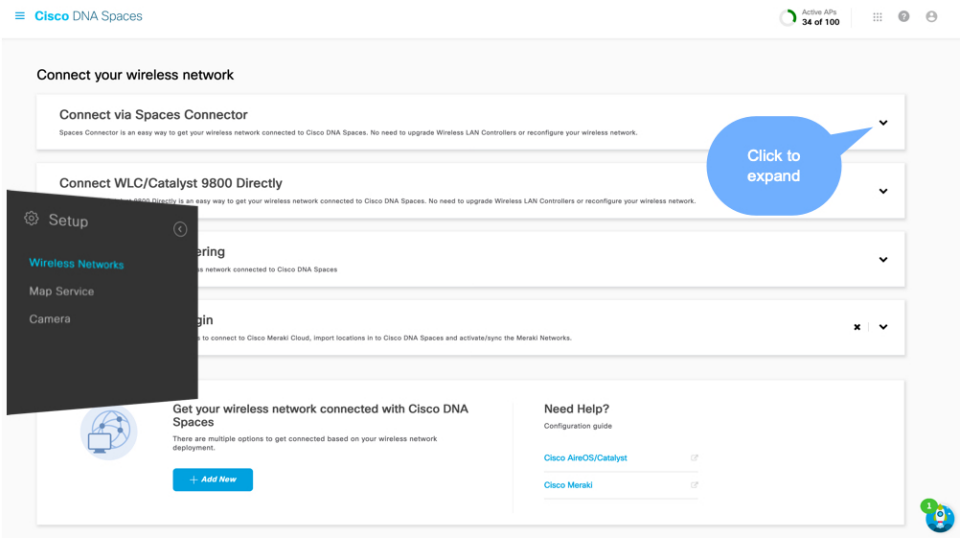
Step 6 In the **Prerequisites for Spaces Connector** dialog box, click **Continue Setup**.

Figure 10: Read Prerequisites for Spaces Connector



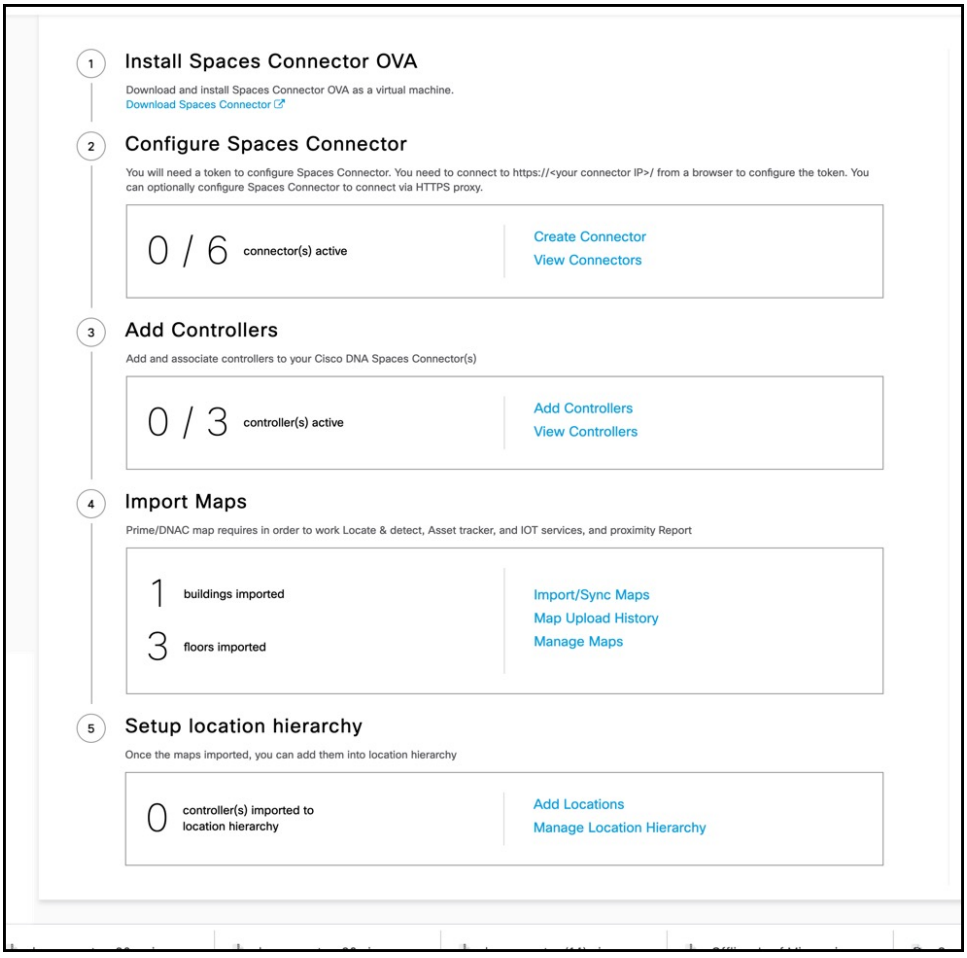
Step 7 Expand the **Connect via Spaces Connector** area using the respective drop-down arrow.

Figure 11: Expand Connect via Spaces Connector



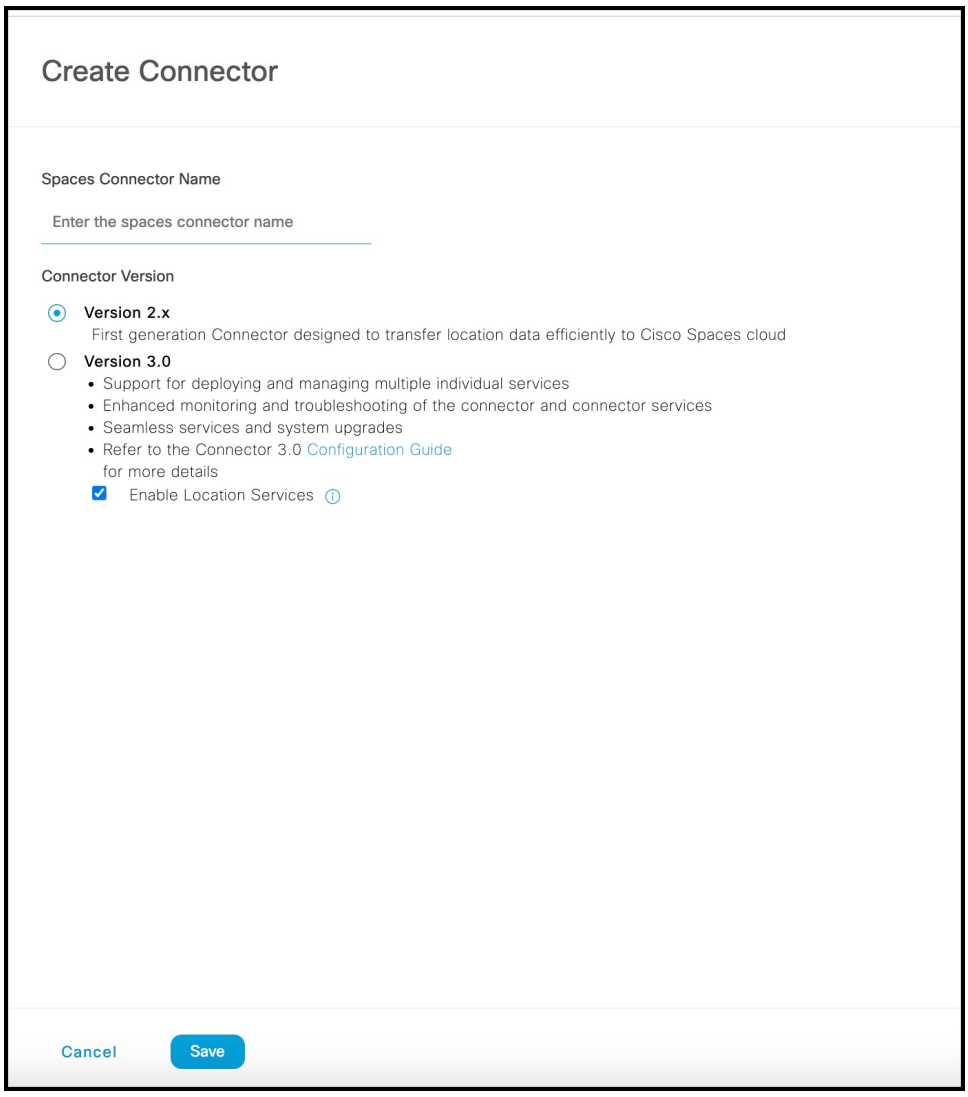
Step 8 In the displayed list of steps, in the **Configure Spaces Connector** area, click **Create Connector**.

Figure 12: Connect via Spaces Connector > Create Connector



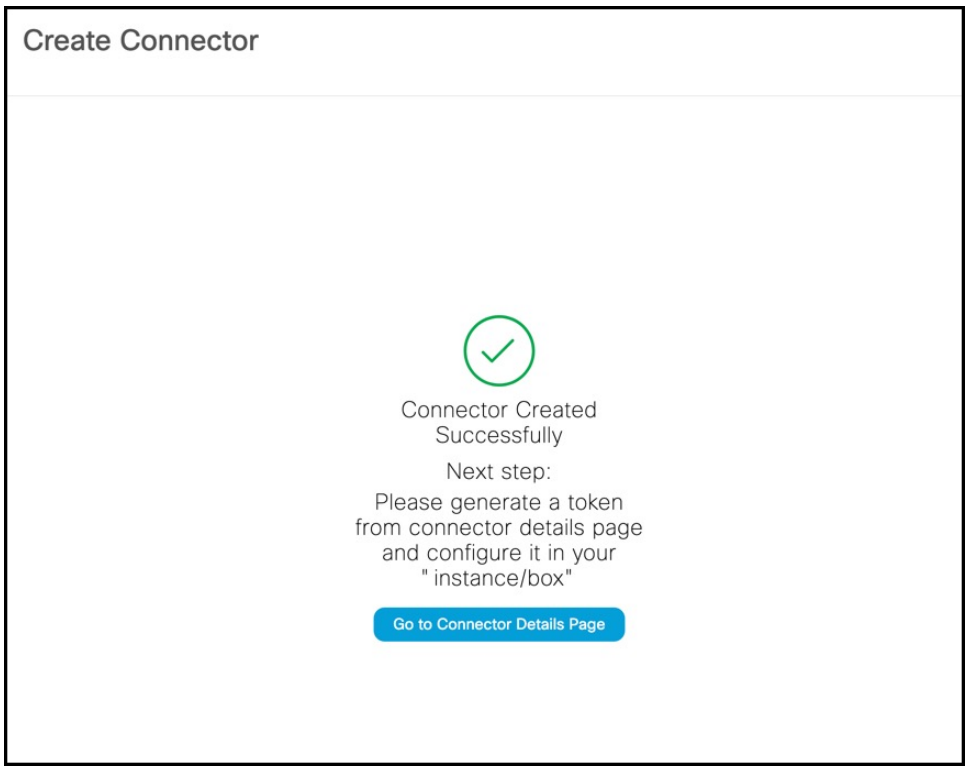
Step 9 In the **Create connector** window that is displayed, enter a name for connector, and click **Version 3.0 (beta).** as the **Connector Version**, and click **Save**.

Figure 13: Name and Version of Connector



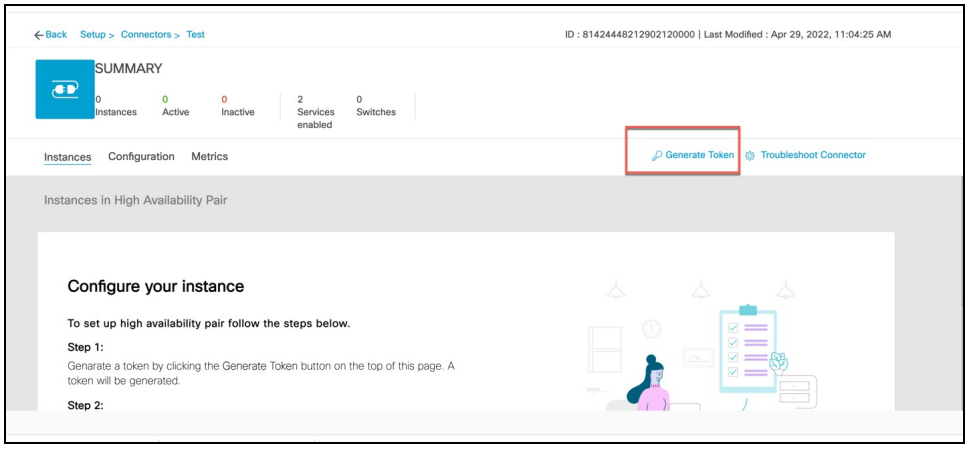
Connector is successfully created. Click **Go to Connector Details** Page.

Figure 14: Connector Created Successfully



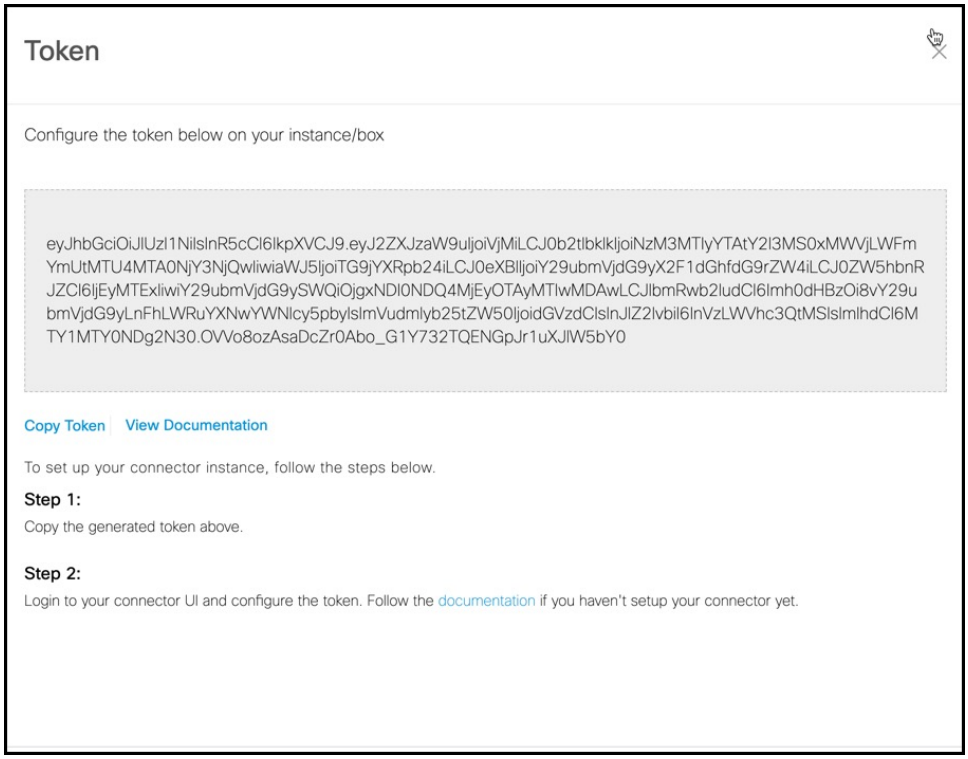
Step 10 In the connector details window, you can see a summary of the configurations for this connector. Click **Generate Token.**

Figure 15: Generate Token



Step 11 In the **Token** window that is displayed, click **Copy Token**.

Figure 16: Copy Token

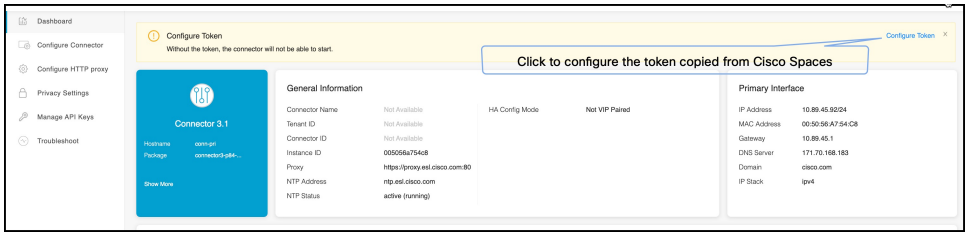


Step 12 Open the connector GUI.

Step 13 (Optional) If your network is behind a proxy, configure the GUI with the proxy. See Configure a Proxy , on page 139

Step 14 In the **Configure Token** area that is displayed, click **Configure Token**.

Figure 17: Configure Token



Step 15 In the window that is displayed, in the **Token** text, field enter the token copied from Cisco Spaces and click **Configure**.

Warning
During this step, if you face a connectivity issue between Cisco Spaces: Connector and Cisco Spaces dashboard, the Connector could hang without an error. You can still access the Connector through SSH. You may also be unable to log in the Connector GUI after this issue.

Step 16 Add the following services as required:

- Configuring IoT Services

- Configuring Hotspot Services

Upgrading the Connector from Cisco Spaces Dashboard

Use the connector's GUI to upgrade connector.

Note

- This is referred to as **system inline upgrade** or **system upgrade** using the connector GUI.
- Upgrade is not supported on AMI connector instances.

Log in to the connector GUI, check for new upgrades and the summary of changes, and initiate the upgrade. Note that you must ensure that the connector's Service manager service is updated before you start the connector upgrade. You can upgrade the Service manager service from the connector GUI. The following procedure describes how to first upgrade the Service manager service and then upgrade connector itself from the connector GUI.

Procedure

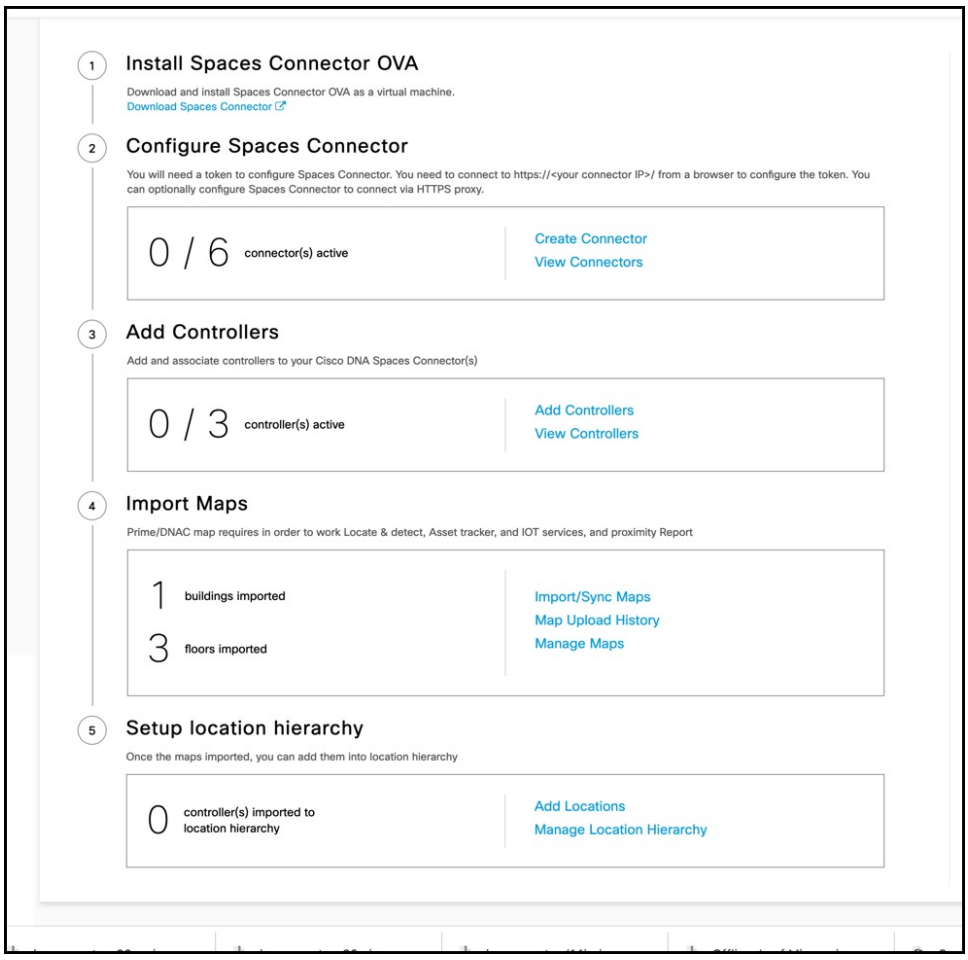
Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 In the Cisco Spaces dashboard, choose **Setup** > **Wireless Networks**.

Step 3 From the **2. Configure Spaces Connector** area, click **View Connectors**

Figure 18: View Connectors



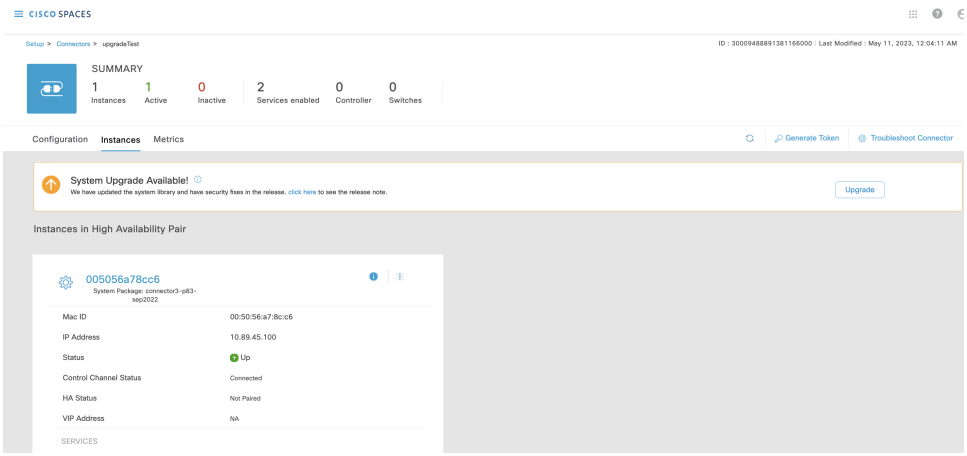
Step 4 From the list of connectors that are displayed, click the connector of your choice.

Step 5 From the **Configuration** tab of the specific connector, ensure that the Service manager service is upgraded. If not upgraded, under the **Actions** column, check for any available **Upgrade** option.

Step 6 Click the **Instances** tab, and choose the instances you want to upgrade.

Step 7 In the **System Upgrade Available** area, and click **Upgrade**.

Figure 19: Upgrade



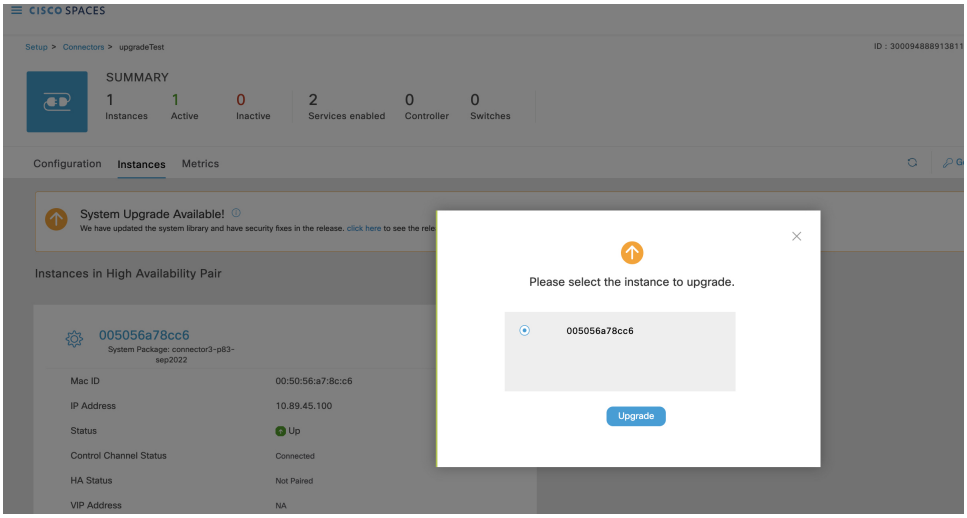
Note

For connector Release 3, the system inline upgrade may not succeed in a low latency network. You can upgrade the connector manually. Downloading the connector OVA from cisco.com and using the **connectoros upgrade <package-name>** command from the connector CLI.

For connector Release 3.1, the upgrade option is available from the Cisco Spaces dashboard as the timeout period has been increased to accommodate low latency networks.

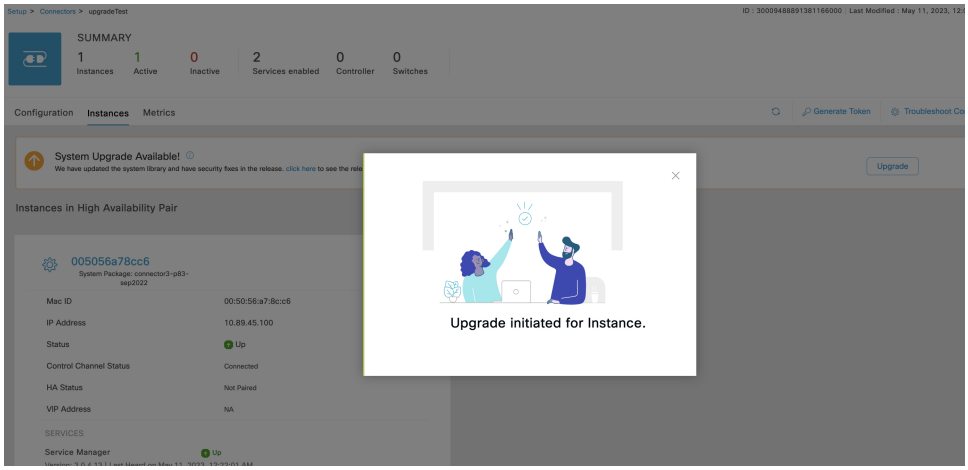
Step 8 From the popup displayed, select the instance you want to upgrade.

Figure 20: Select instance



An **Upgrade Initiated for instance** message is displayed.

Figure 21: Upgrade Initiated for Instance



Step 9 Observe the status of the installation by clicking the three-dot icon of an instance. From the menu displayed, choose **Configuration History**.

Figure 22: Configuration History

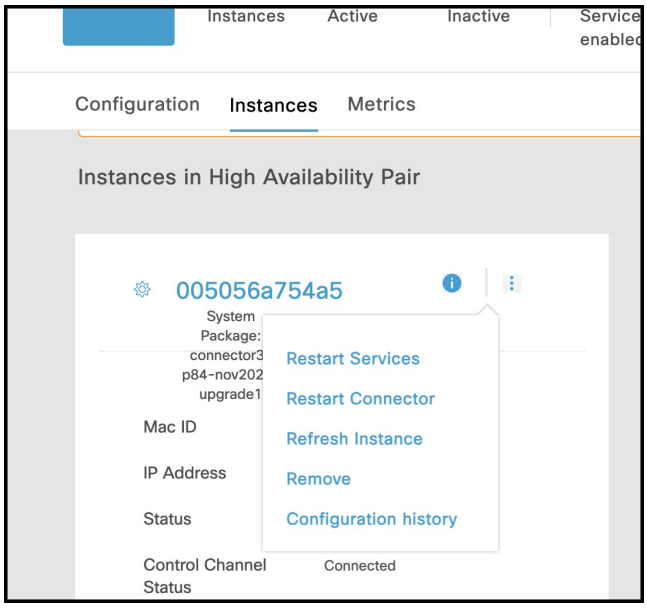
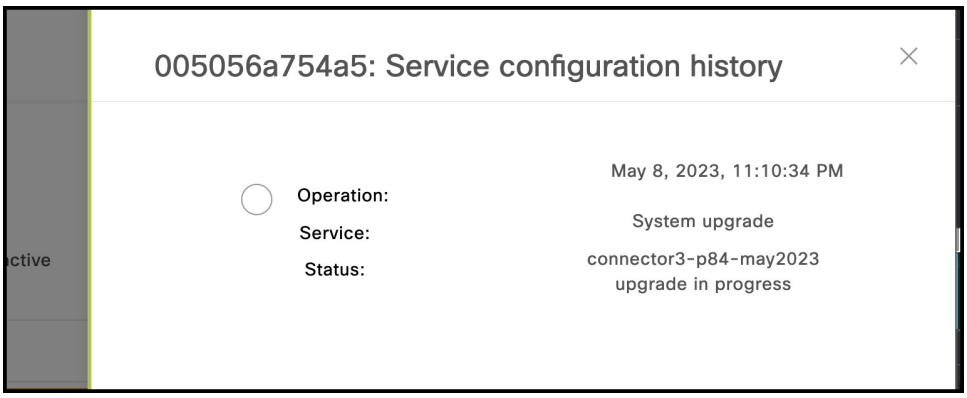


Figure 23: Configuration History



Upgrading the Connector Using CLI

Use the connector's CLI to upgrade connector.

Note

- This is referred to as **system inline upgrade** or **system upgrade** using the connector CLI.
- System upgrade is not supported for AMI connector instances.

Log in to the connector CLI, check for new upgrades and the summary of changes, and initiate the upgrade. Note that you must ensure that the connector's Service manager service service is updated before you start the connector command line upgrade. You can upgrade the Service manager service from the connector GUI. then upgrade connector itself from the connector CLI.

Before you begin

Ensure that the Service manager service is upgraded from the connector GUI.

Procedure

Step 1 Log in to the connector CLI.

Step 2 Check the availability of upgrades, and view a summary of the changes that are part of this upgrade package. Run the **connectorctl systemupgrade list** command.

Step 3 Initiate the upgrade of connector packages. Run the **connectorctl systemupgrade install** command:

```
[spacesadmin@connector03 ~]$ connectorctl systemupgrade install

Executing command:systemupgrade
Command execution status :Success

System upgrade operation is queued. Use tail -f
```

```
/opt/spaces-connector/runtime/logs/service-manager/system-upgrade/system-upgrade. log to see upgrade
 progress
```

Step 4 Observe the status of the upgrade. Do one of the following:

- To populate the CLI with regular updates of the upgrade, run the **tail -f /opt/spaces-connector/runtime/logs/service-manager/system-upgrade/system-upgrade.log** command.
- To view the status of the upgrade at any point in time, run the **connectorctl systemupgrade status** command:

```
 [spacesadmin@connector ~]$ connectorctl systemupgrade status
Executing conmand:systemupgrade
Command execution status: Success

System upgrade is in progress for package:connector3-p84-jan2023-upgrade2 at:Jan-10-2023 05:31:33.
 Details:Downloading image.

[spacesadmin@connector ~]$ connectorctl systemupgrade status
Executing command: systemupgrade
Command execution status: Success

Successfully upgraded system to package: connector3-p84-jan2023-upgrade2 at :Jan-1
 0-2023 04:34:04
```

Occasionally, you may see the following error while running the **connectorctl systemupgrade status** command. Ignore this output and wait for a few minutes before running the **connectorctl systemupgrade status** command again:

```
[spacesadmin@connector ~]$ connectorct1 systemupgrade status
Traceback (most recent call last>:
    File "/opt/spaces-connector/static/service-agent/core/src/cli/cli.py'.line10,in<module>
       from core.src.log.log_task import Loglask

File"/opt/spaces-connector/static/service-agent/core/src/cli/../../../core/src/log/log_task-py".line16,in<module>

        from -utils import pathconstant, constant, utilities
   File
"/opt/spaces-connector/static/service-agent/core/src/cli/../../../core/src/utils/utilities-py',line31,in<module>

        import psutil
ModuleNotFoundError: No module named ›psutil'
```

Upgrading the Connector Using Downloaded Connector Package

You can upgrade the connector manually. Downloading the connector OVA from cisco.com and using the **connectoros upgrade <package-name>** command from the connector CLI.

Procedure

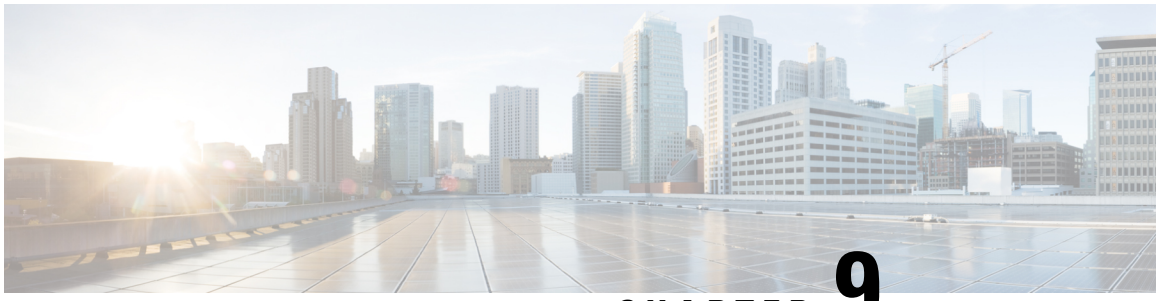
Step 1 Log in to the connector CLI.

Step 2 Run the **connectoros upgrade** command to upgrade the installed connector.

The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectoros upgrade
cisco-spaces-connector3-p84-jul2024.connectorpkg
```

For more information on this command, see **connectoros upgrade**.



CHAPTER 9

Cisco Spaces: Connector AMI

- Launch Connector 3 as an EC2 Instance from AMI , on page 43

Launch Connector 3 as an EC2 Instance from AMI

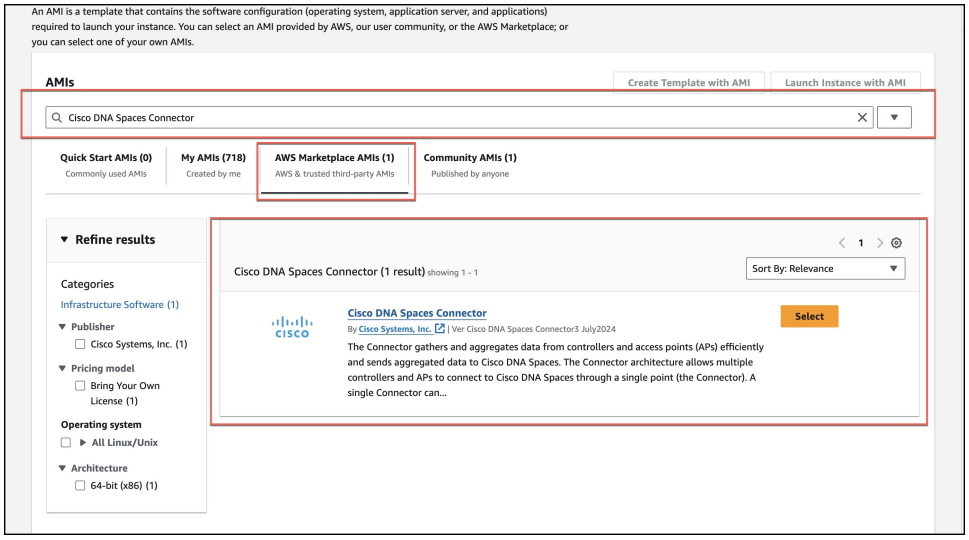
This chapter provides information about how to launch a connector 3 as an EC2 instance from Amazon Machine Images (AMI), configure the connector 3 instance, and finally obtain a URL to log in to the connector connector and CLI.

Procedure

Step 1 Log in to your Amazon Web Services account and navigate to the **EC2 Dashboard**. In the left-navigation pane, choose **Images > AMI Catalog**.

Step 2 In the **AMIs** search area, click **AWS MarketPlace AMIs** and enter `DNA Spaces Connector`. Press **Enter**.

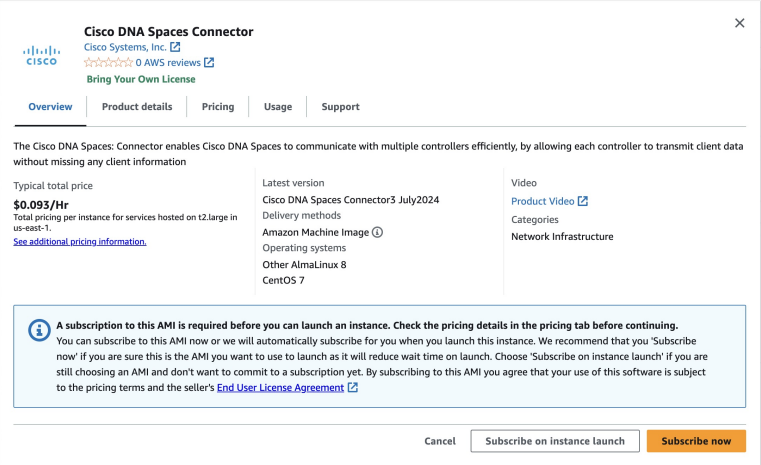
Figure 24: Configuration



Step 3 Click the displayed image and click **Select**.

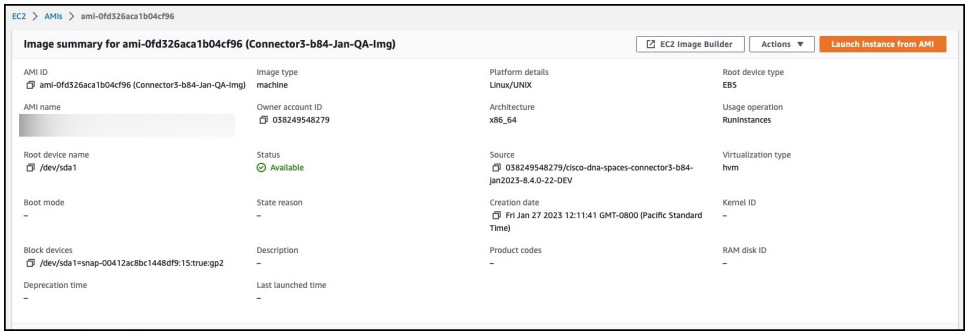
Step 4 In the **Cisco DNA Spaces Connector** window displayed, click **Continue**.

Figure 25: AWS MarketPlace AMIs



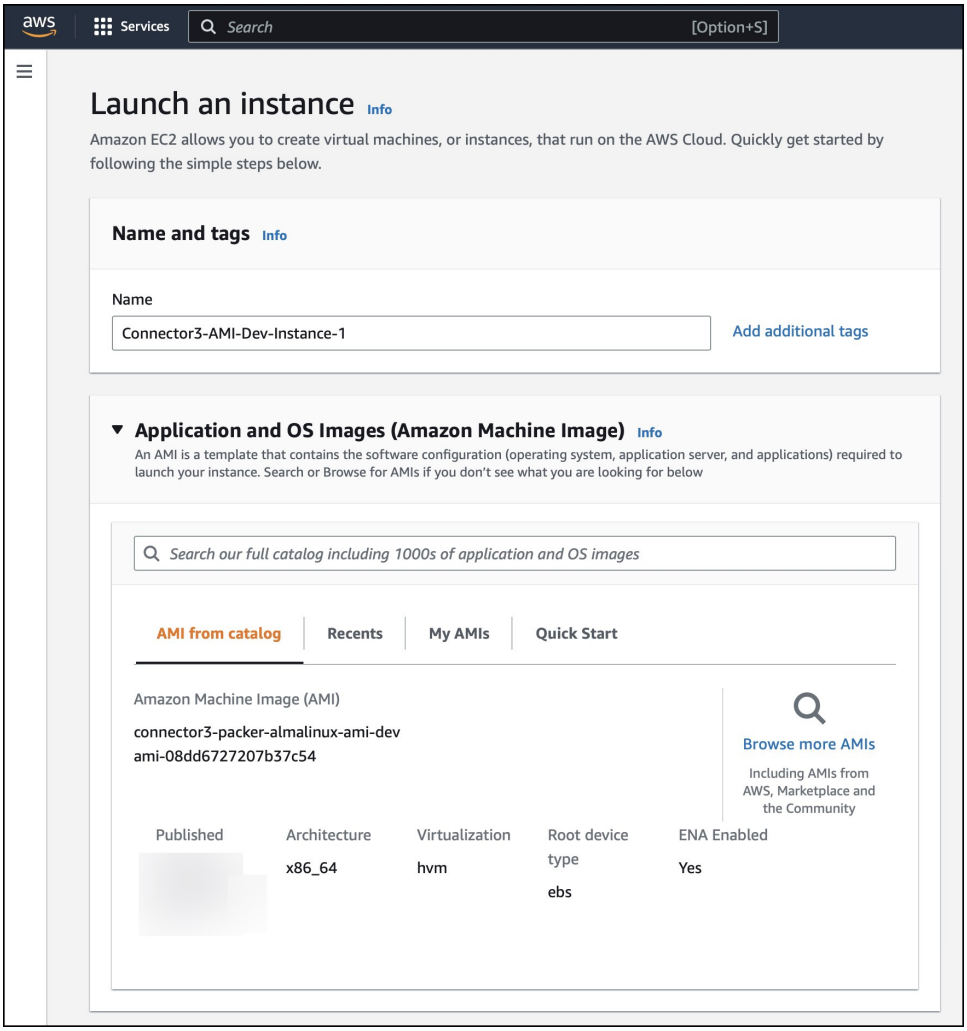
Step 5 In the **Image Summary** window displayed, click **Launch Instance from AMI**

Figure 26: Launch Instance from AMI



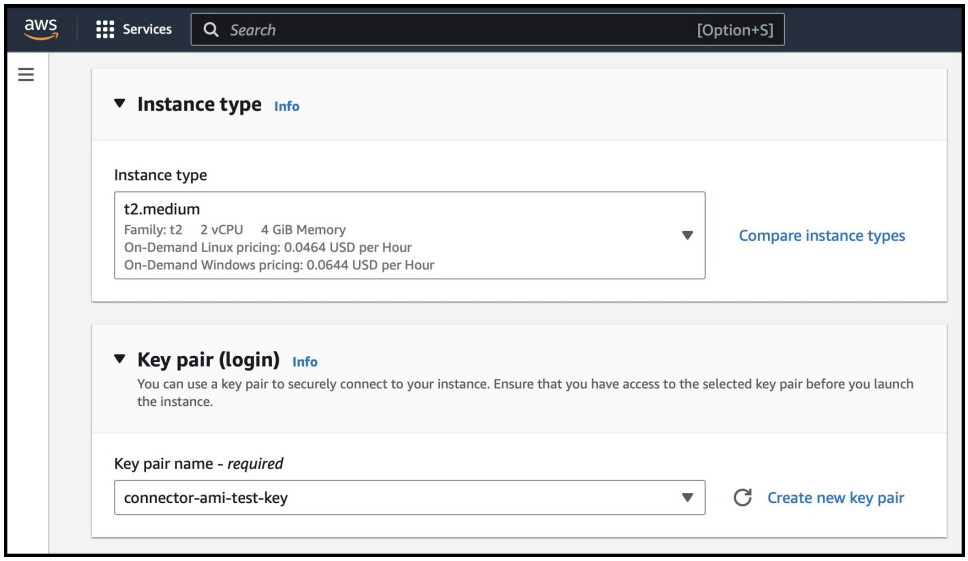
Step 6 In the **Launch an Instance** window displayed, enter an instance name, and add any additional labels for your instance by clicking the **Add Additional tags** button.

Figure 27: Launch Instance from AMI



Step 7 Choose any EC2 instance that has a minimum of 2 vCPU and 4GB Memory. Click **Next: Configure Instance Details**. **t2.medium** corresponds to a standard window with 2vCPUs and 4-GB memory and is the recommended setting.

Figure 28: Configure Instance Details



Note
You can have a more advanced configuration by choosing an option with higher vCPU and memory, by choosing an instance type with one of the following configurations. If an exact match is unavailable, you can choose a configuration with the next-available vCPU or memory:

- 4 vCPUs and 8-GB memory (referred to in this document as **Advanced1**)
- 8 vCPUs and 16-GB memory (referred to in this document as **Advanced2**)

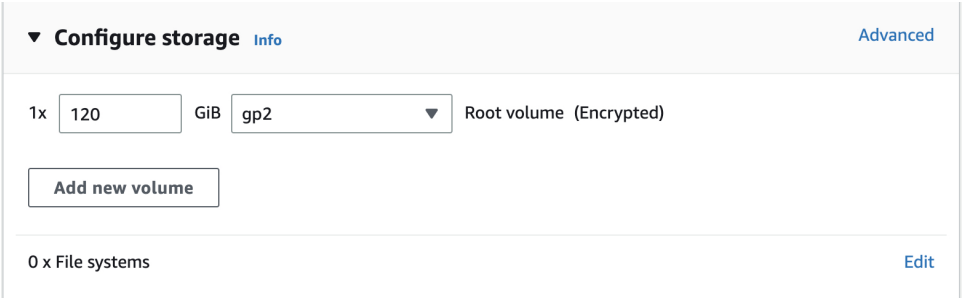
Step 8 Choose a **Network** and a **Subnet**. Click **Next: Add Storage**.

Figure 29: Add Storage



Step 9 Enter the value of **Size(GB)** as 120. Click **Next: Configure Security Group**.

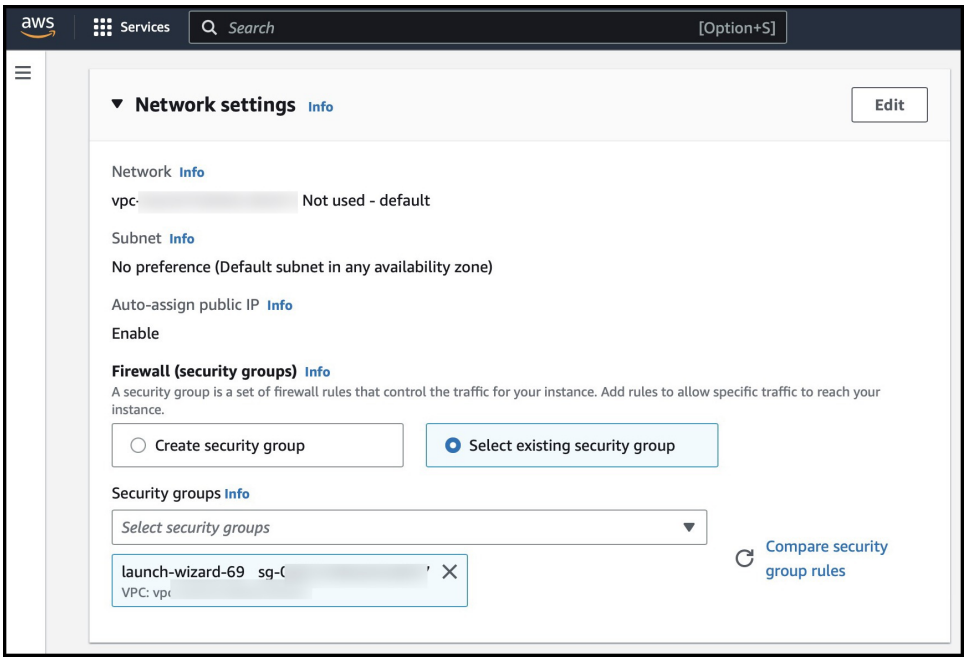
Figure 30: Configure Storage



Step 10 Configure a security group by following these steps:

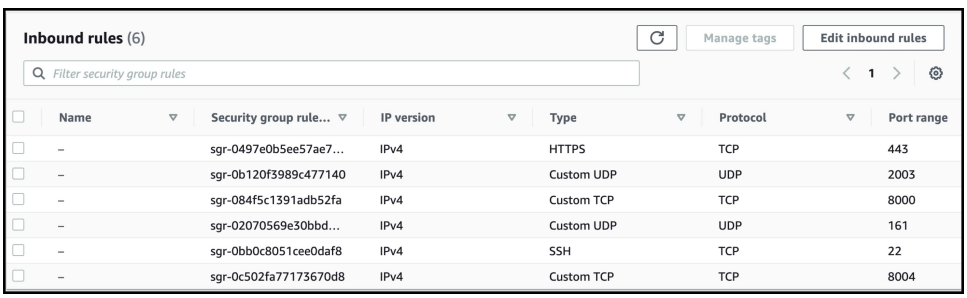
a) Create a new security group or modify an existing one by clicking the respective radio button.

Figure 31: Configure Security Group



b) Configure rules permitting inbound traffic to specific ports, as shown in the following image. You can allow inbound traffic to these ports for all IP addresses or choose to restrict them for specific IP addresses.

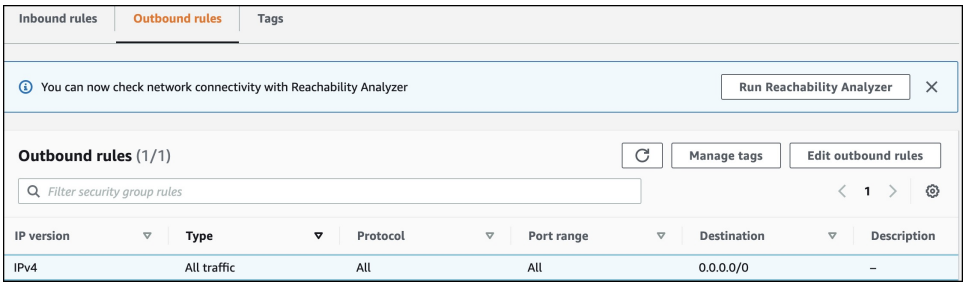
Figure 32: Configure These Inbound Rules Permitting Traffic to Specific Ports



Note
Using an inbound rule, you can also specify the network subnet range that can access this instance (For example, through SSH).

c) Configure the outbound rule shown in the following image.

Figure 33: Configure This Outbound Rule



Note
For various connector services to work, you must open specific ports. See the respective **Information About Open Ports** section of the connector service for more information.

Step 11 In the displayed **Select an existing key pair or create a new key pair** dialog box, do either of the following:

- Choose **Create a new key pair** from the drop-down list. Provide a **Key pair name** and click **Download Key Pair** to download it. Then click **Launch Instance** to launch the instance.
- Choose **Choose an existing key pair** from the drop-down list. Select the previously downloaded key pair from the **Select a key Pair** drop-down list. Then click **Launch Instance** to launch the instance.

Figure 34: Create a New Key Pair

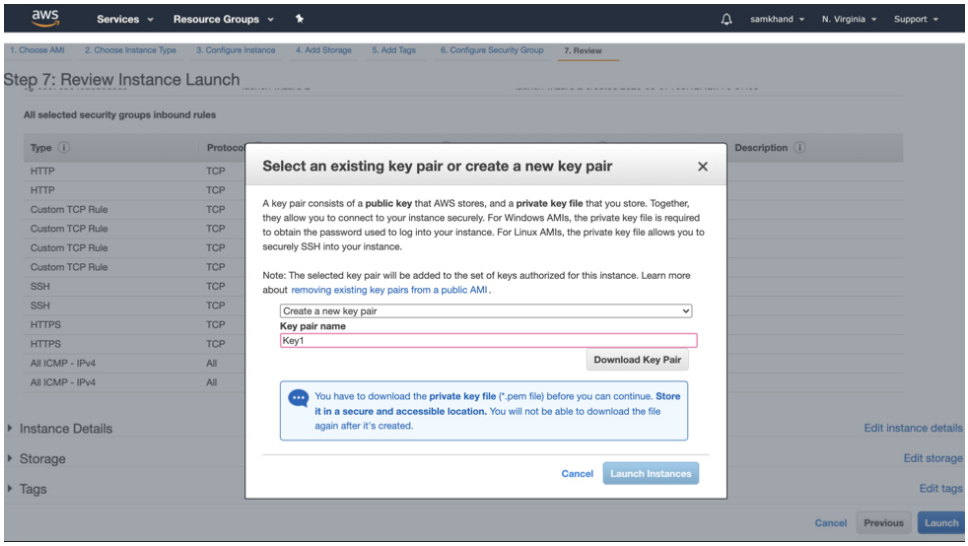
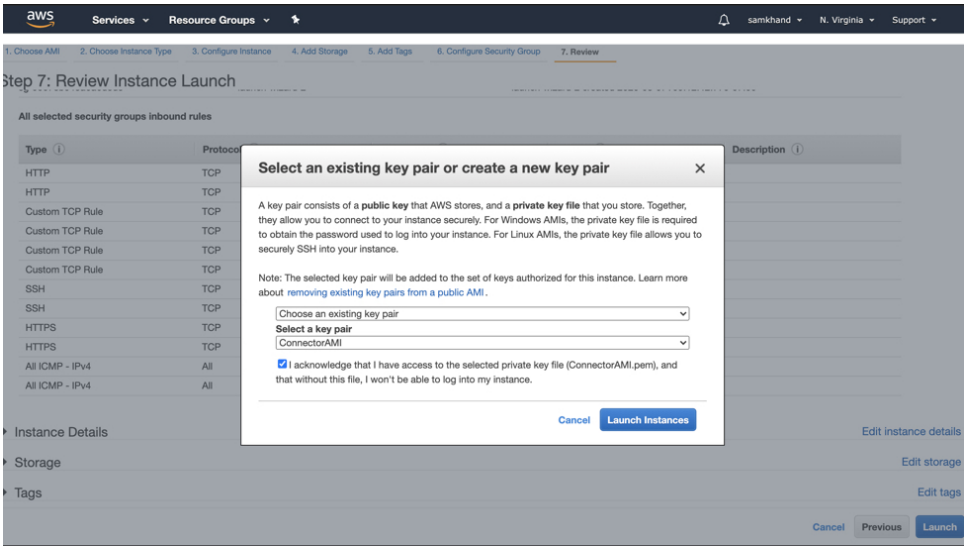


Figure 35: Choose an Existing Key Pair

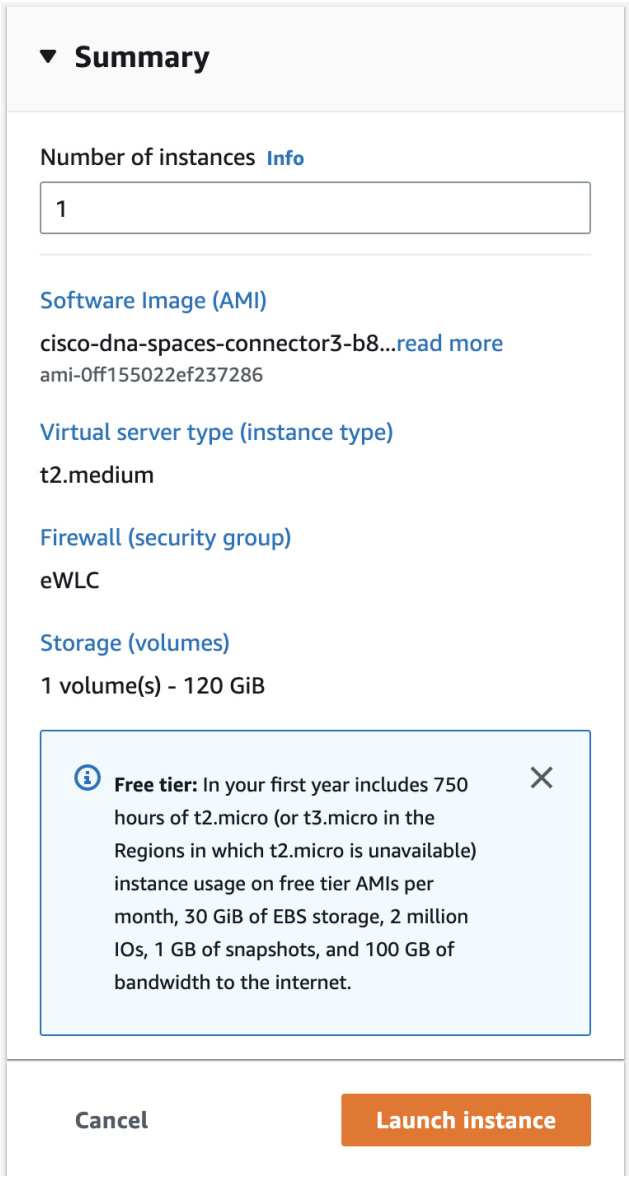


Step 12 After you have downloaded the key pair (.pem) file to your system, navigate to the file location. Configure appropriate permissions for the .PEM file using the **chmod** command.

```
chmod 400 /path/to/MyAccessKey1.pem
```

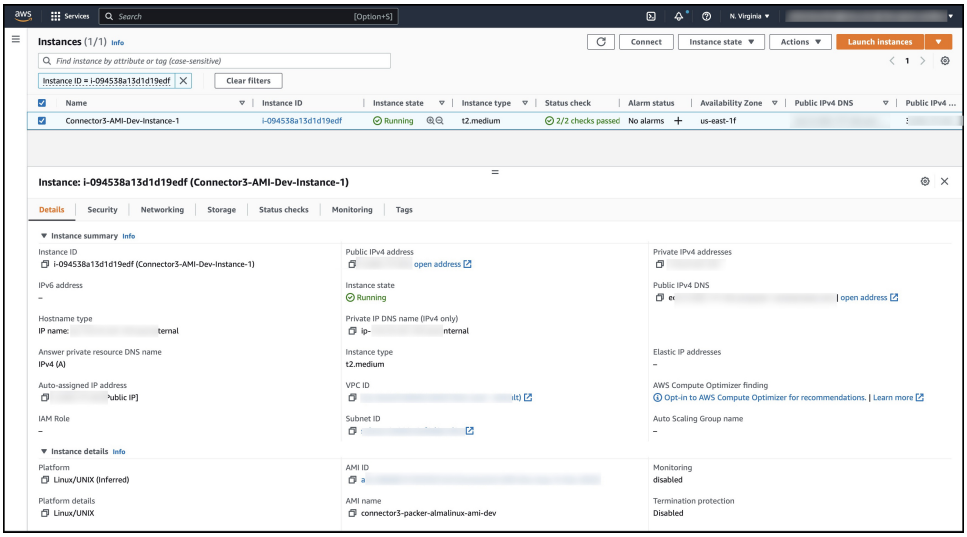
Step 13 Review the instance and click **Launch**.

Figure 36: Review Instance and Launch



Step 14 On the EC2 dashboard, wait for the instance to finish launching and the status to change to **Running**. Alternatively, you can see the running instances on the **Instances** page. Click the instance to obtain the IPv4 address of the instance.

Figure 37: Obtain IPv4 Address of Instance



Step 15 Perform initial setup to configure a hostname, and change passwords for **spacesadmin** and **root** users.

a) Log in to the connector using the **ssh -i** command and the following parameters:

- The .PEM key pair downloaded in step 11
- ec2-user
- The IPv4 address obtained in step 14

```
ssh -i /path/to/key/MyAccessKey1.pem ec2-user@IPv4-address
```

b) Change passwords for **spacesadmin** and **root** users. Avoid a BAD PASSWORD prompt by complying with the following password requirements:

- Length is more than 14 characters.
- Includes at least one uppercase letter.
- Includes at least one lowercase letter.
- Includes at least one special character.

The following is a sample output of the command:

```
Welcome to Cisco Spaces Connector Setup
Changing password for user spacesadmin.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
Password changed successfully
Setting rbash...
Restarting docker...
Changing shell for root.
Shell changed.
Changing shell for spaces.

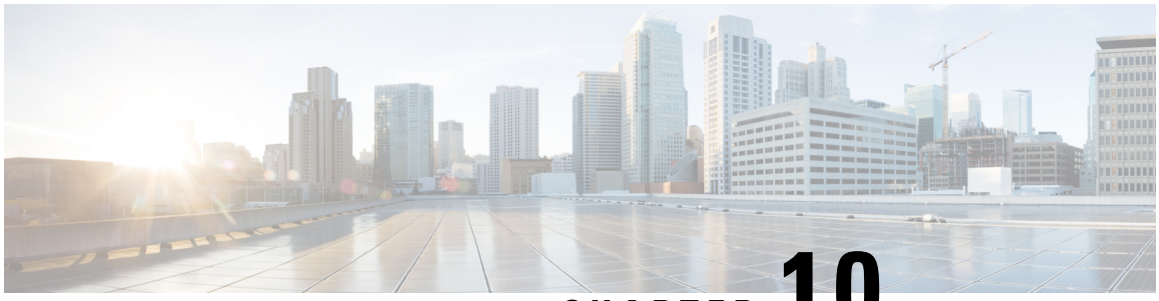

Remove default users...
Relabeled /etc/sudoers from unconfined_u:object_r:user_tmp_t:s0 to unconfined_u:object_r:etc_t:s0
```

```
Cisco Spaces Connector UI:
https://XX.XXX.XX.XXX
Username log in: spacesadmin
The install is complete, a reboot will occur in 10 seconds...
```

Once the installation is complete, a reboot occurs within 10 seconds. Note down the public IP address before reboot.

Step 16 Log in to the connector and configure the connector further. Do one of the following using the public IPv4 address from the previous step (step 15):

- Log in to the connector GUI using the browser window and the address https://*public-ipv4-address*
- Log in to the connector CLI using the SSH command and the username **spacesadmin**. Use the command **ssh spacesadmin@***public-ipv4-address*. When prompted, use the password configured for the **spacesadmin** user.



CHAPTER 10

Cisco Spaces: Connector Hyper-V

The chapter shows you how to install a connector as a Hyper-V instance. To do this, you must perform two tasks. The first task is to create a virtual switch and the second is to download and deploy Hyper-V image as a connector:

- Creating a Virtual Switch, on page 53
- Downloading and Deploying HYPER-V, on page 60

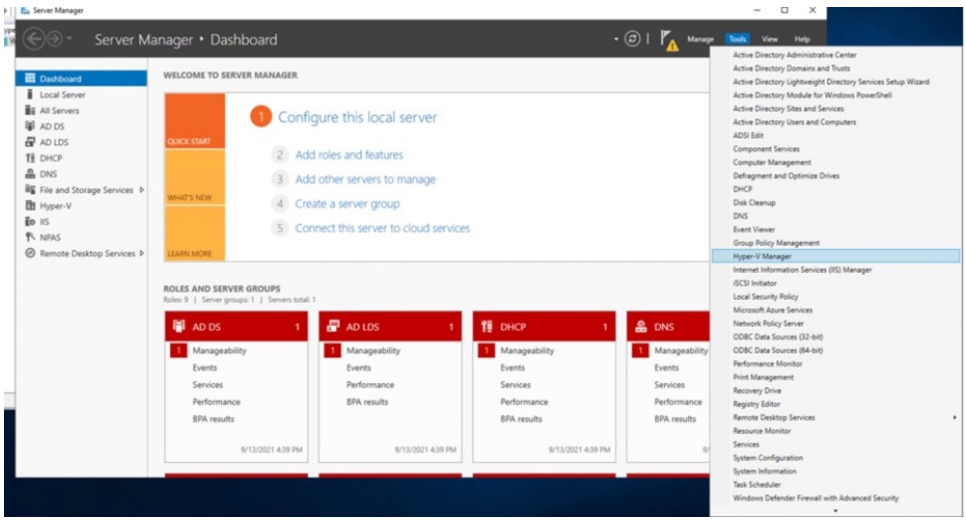
Creating a Virtual Switch

This task shows you how to install a Hyper-V manager. The task also shows you how to use the Hyper-V manager to installs a virtual switch.

Procedure

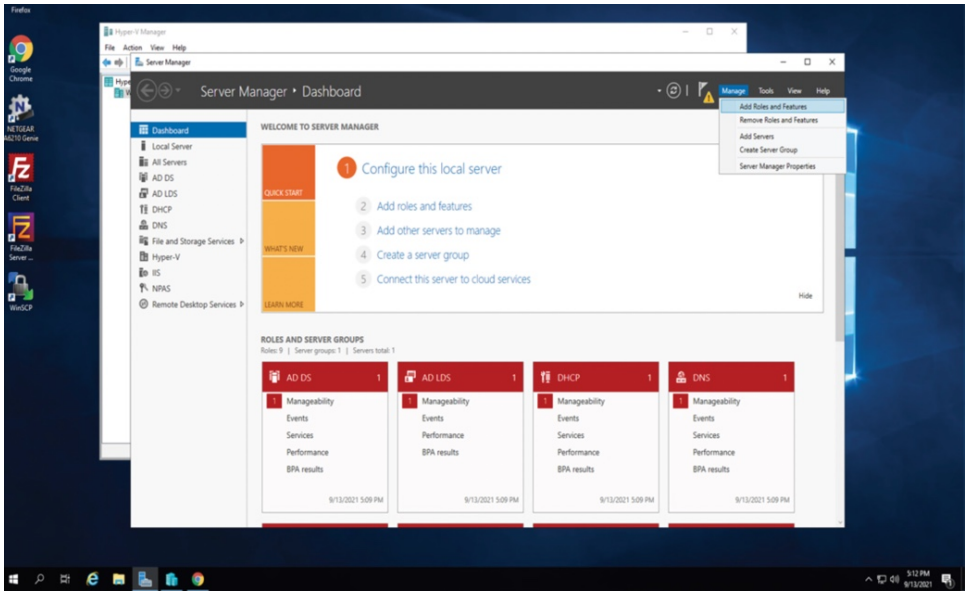
Step 1 Navigate to **Windows > Server Manager**.

Figure 38: Windows > Server Manager



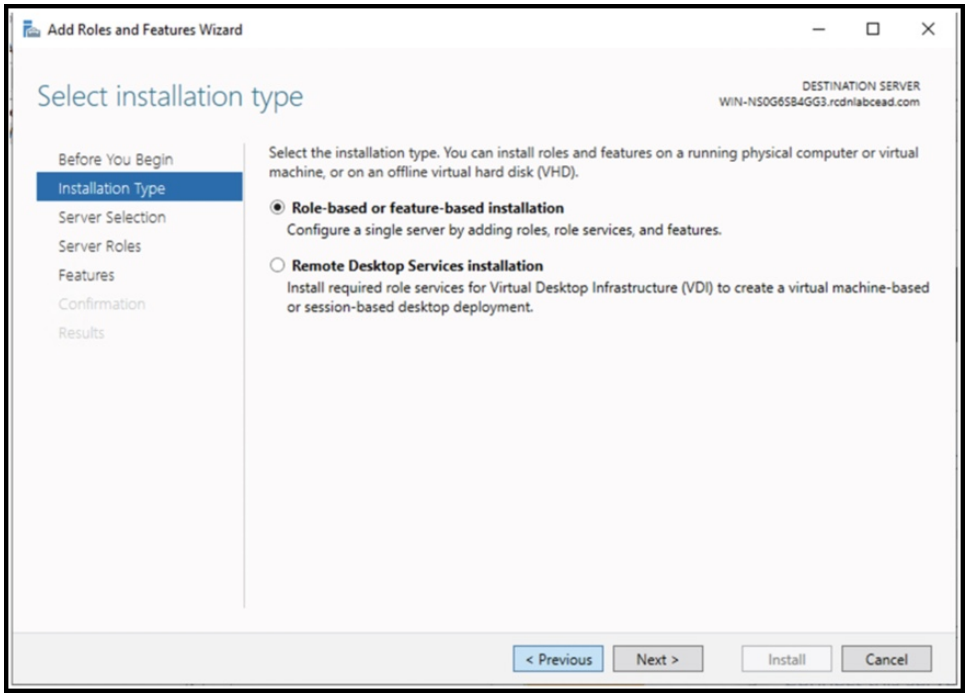
Step 2 Choose **Manage > Add Roles and Features**.

Figure 39: Manage > Add Roles and Features



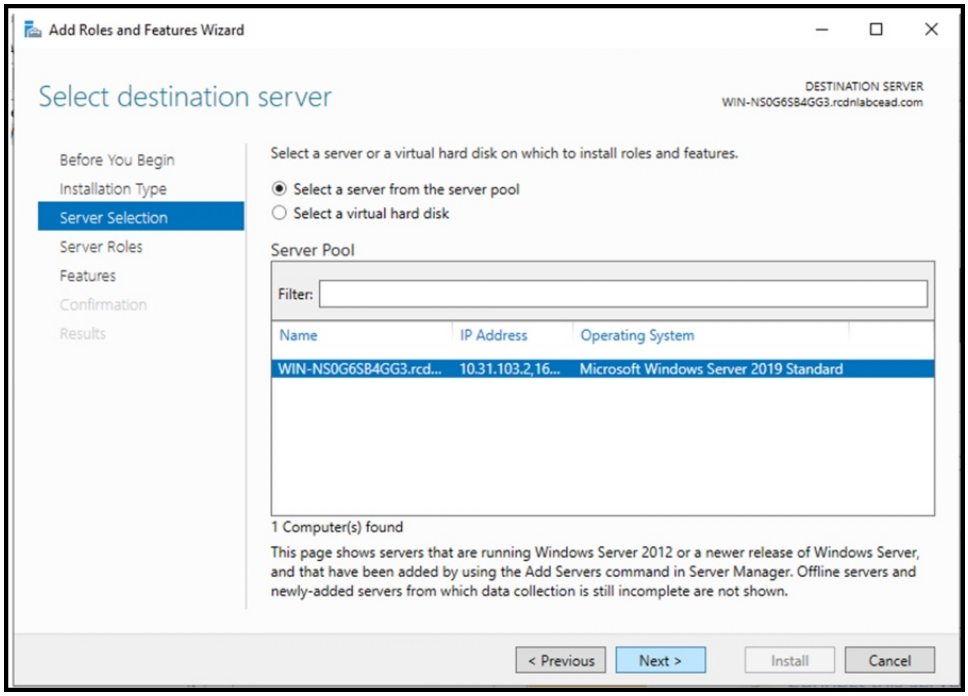
Step 3 Click the **Role-based or feature-based installation** radio button.

Figure 40: Role-based or Feature-Based Installation



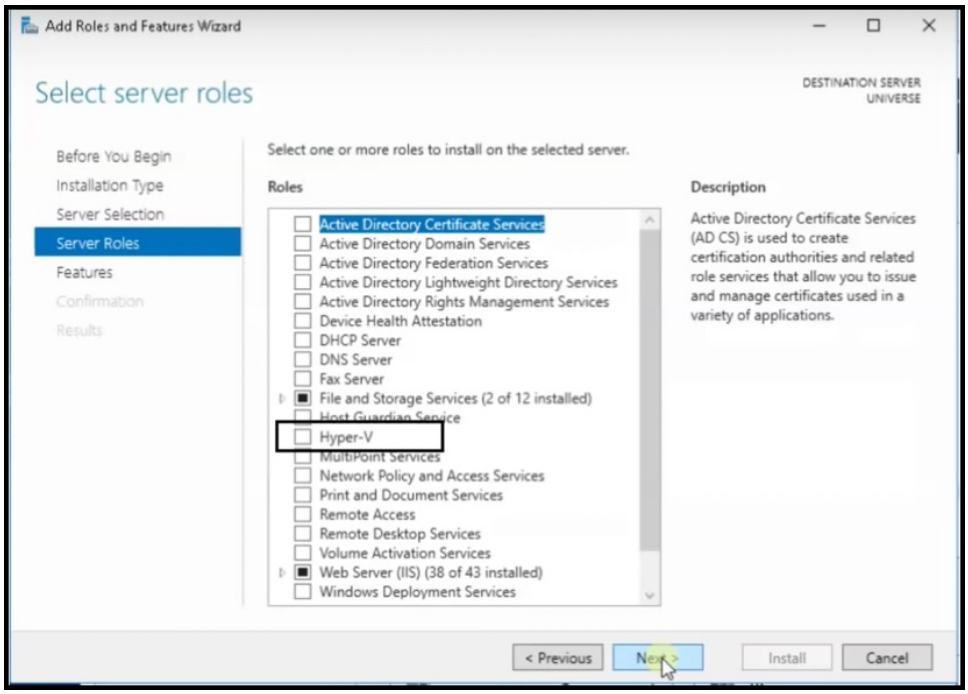
Step 4 Click the **Select a server from the server pool** radio button.

Figure 41: Select a Server From the Server Pool



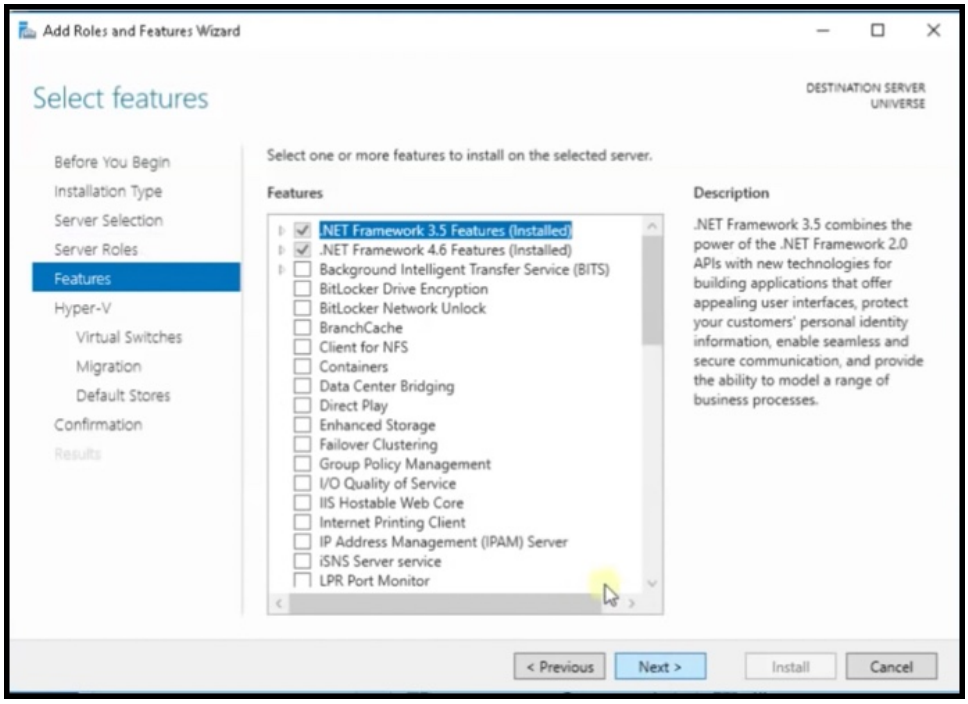
Step 5 In the **Select server roles** window, check the Hyper-V checkbox, and click **Next**.

Figure 42: Select Server Roles



Step 6 In the **Select features** window, check the **.NET Framework** checkbox, and click **Next**.

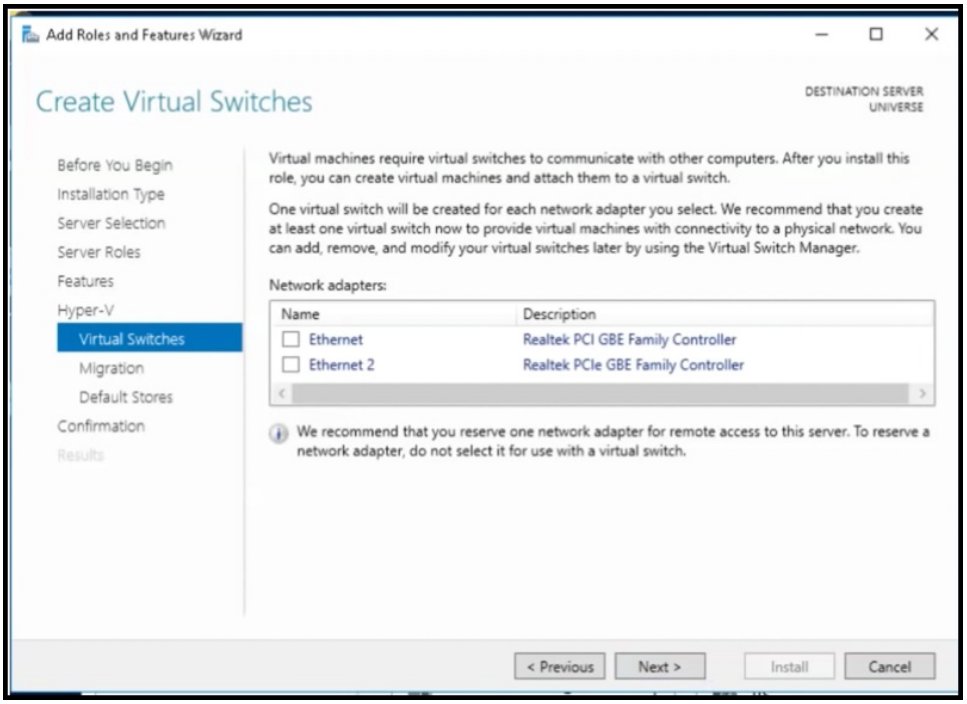
Figure 43: Select Features



Step 7 In the **Hyper-V** window, do the following:

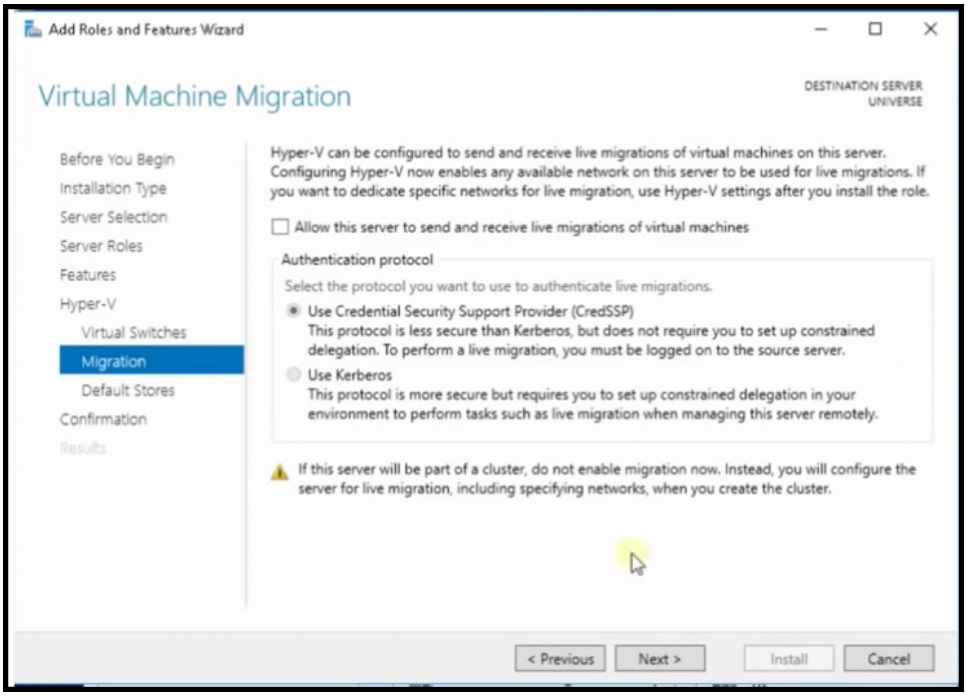
a) In the **Virtual Switches** window, click **Next**.

Figure 44: Virtual Switches



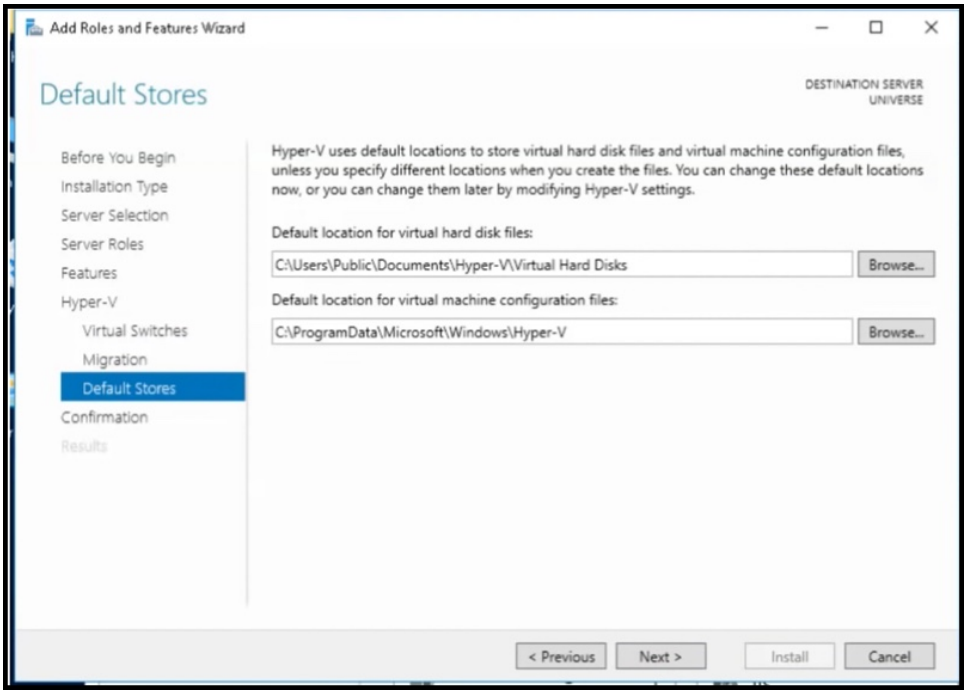
b) In the **Migration** window, click **Use Credential Security Support Provider** (CredSSP) radio button, and click **Next**.

Figure 45: Use Credential Security Support Provider



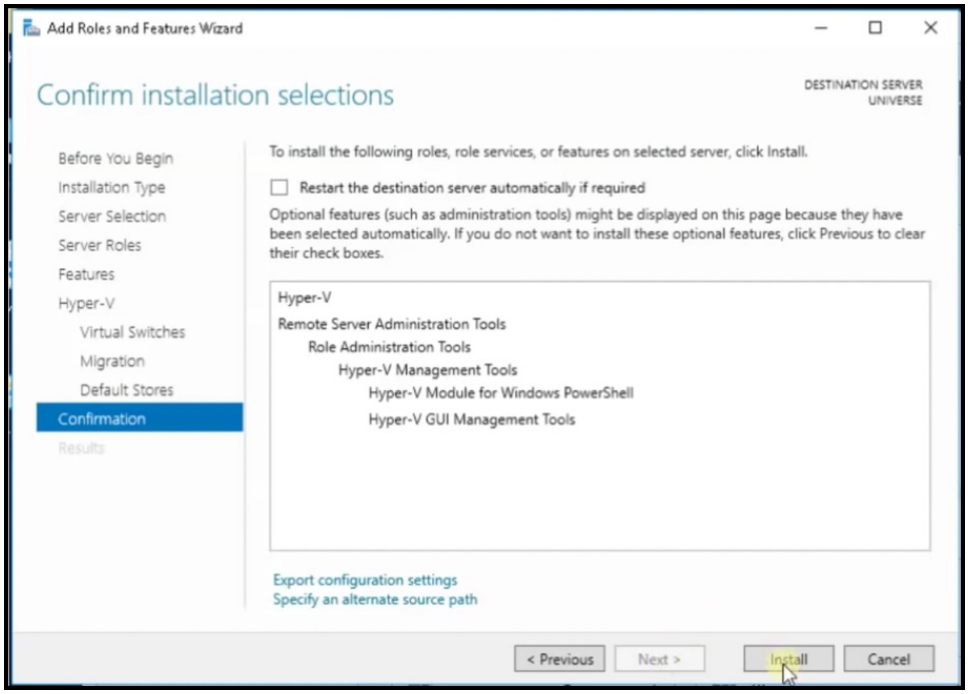
c) In the **Default Stores** window, select the location to install files or retain the default locations, and click **Next**.

Figure 46: Default Stores



Step 8 Confirm the installation settings for Hyper-V and click **Install**.

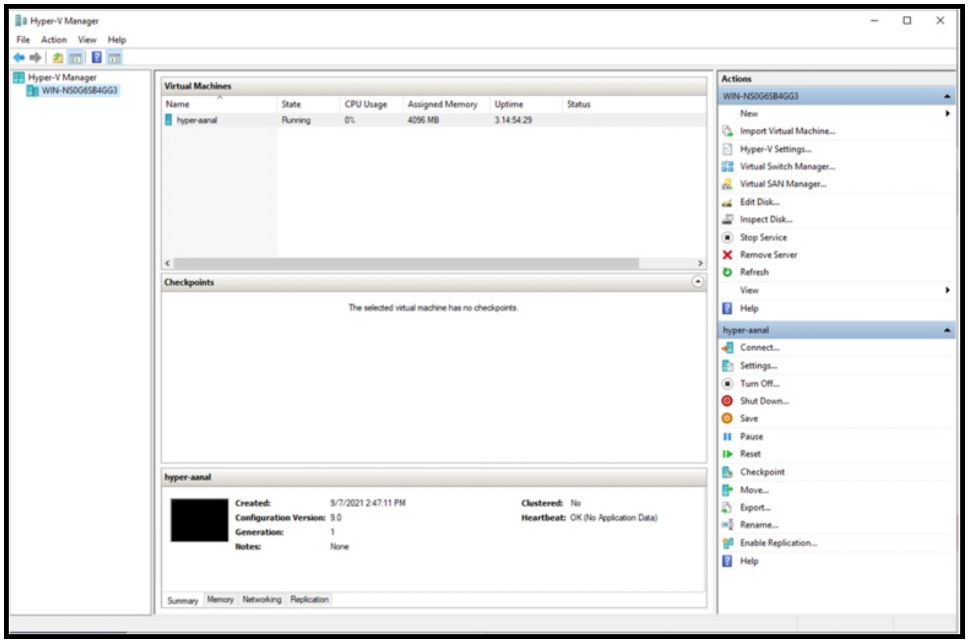
Figure 47: Confirm the Installation Settings



Step 9 Open **Hyper-V Manager**.

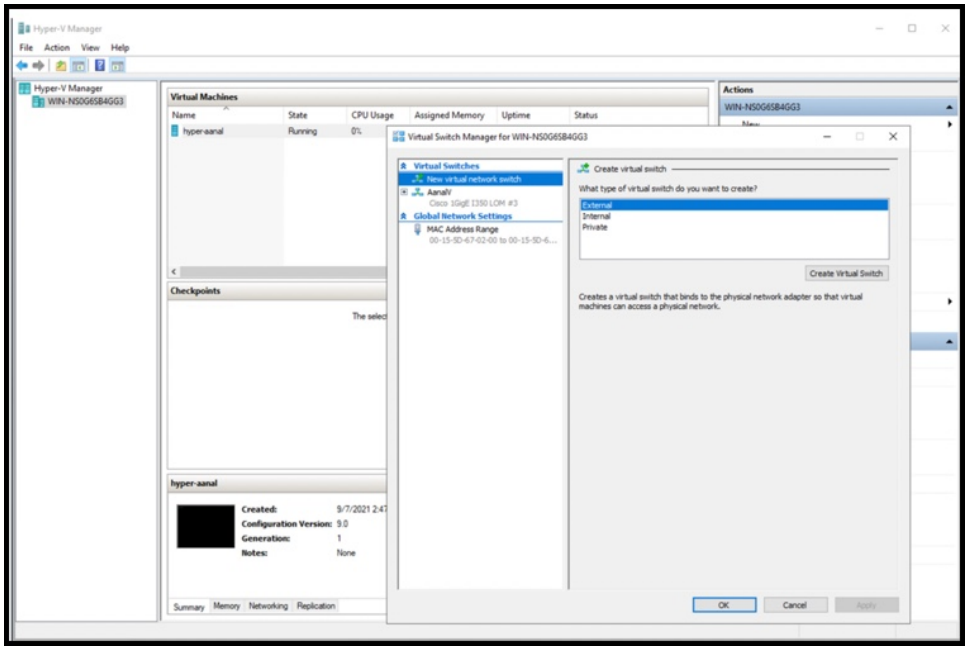
Step 10 In Hyper-V Manager, choose **Actions > Virtual Switch Manager**.

Figure 48: Actions > Virtual Switch Manager



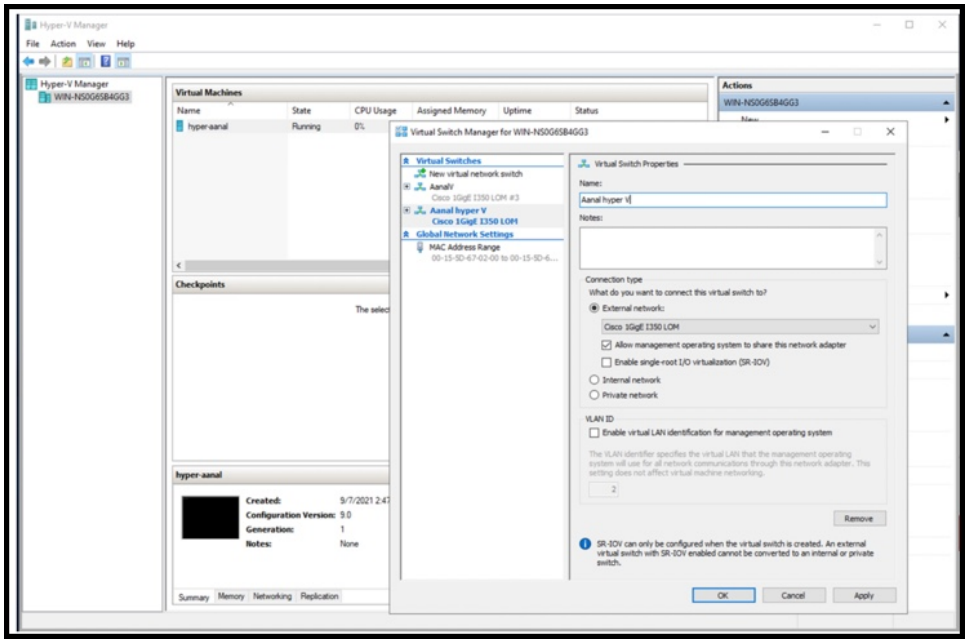
Step 11 In the **Virtual Switch Manager for** window, click **New virtual network switch**. In the **Create virtual switch** window, click **External** and then click **Create Virtual Switch**.

Figure 49: Create Virtual Switch



Step 12 In the **Virtual Switch Properties** window, provide a **Name** for the switch. From the **Connection Type** area, click the **External Network** radio button, and choose a network, and then click **Apply**.

Figure 50: Virtual Switch Properties



Downloading and Deploying HYPER-V

Before you begin

Create a vSwitch on HYPER-V. connector connects to this vSwitch. See Creating a Virtual Switch, on page 53

Procedure

Step 1 Download connector .hyperv (HYPERV) image from Cisco.com.



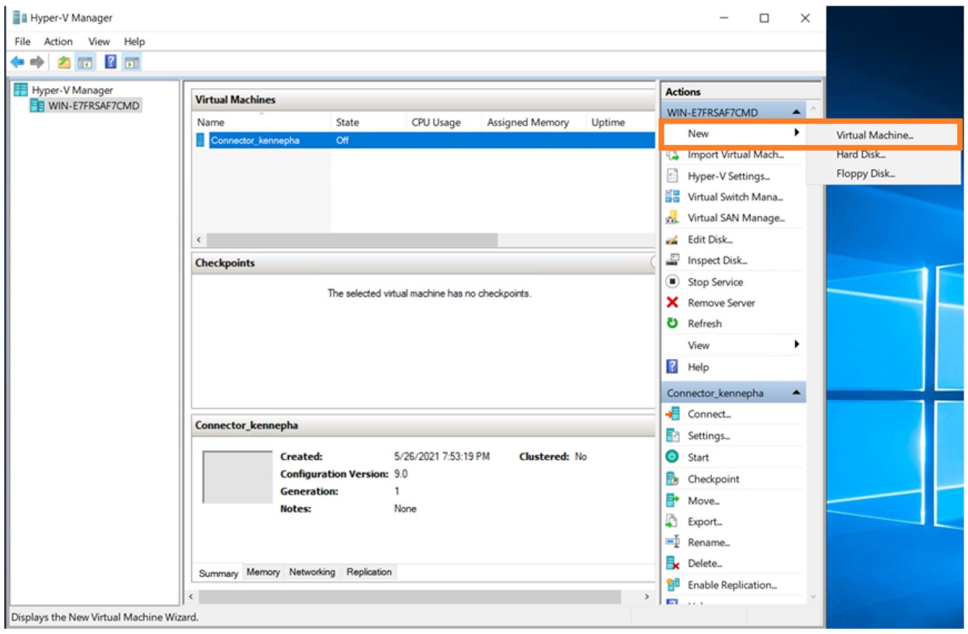
Step 2 Untar the HYPER-V to obtain a .vhdx (VHDX) file. You can use this to deploy a HYPER-V connector instance. Store the VHDX file in a folder location where you plan to create the HYPER-V instance.

For example, **`. tar -xvzf cisco-spaces-connector3-i84-jul2024.hyperv or ->. tar -xvzf <cisco-spaces-connector file name>.hyperv`**

Step 3 Open **Hyper-V Manager**.

Step 4 Right-click the vSwitch created, and choose **New** > **Virtual machine**.

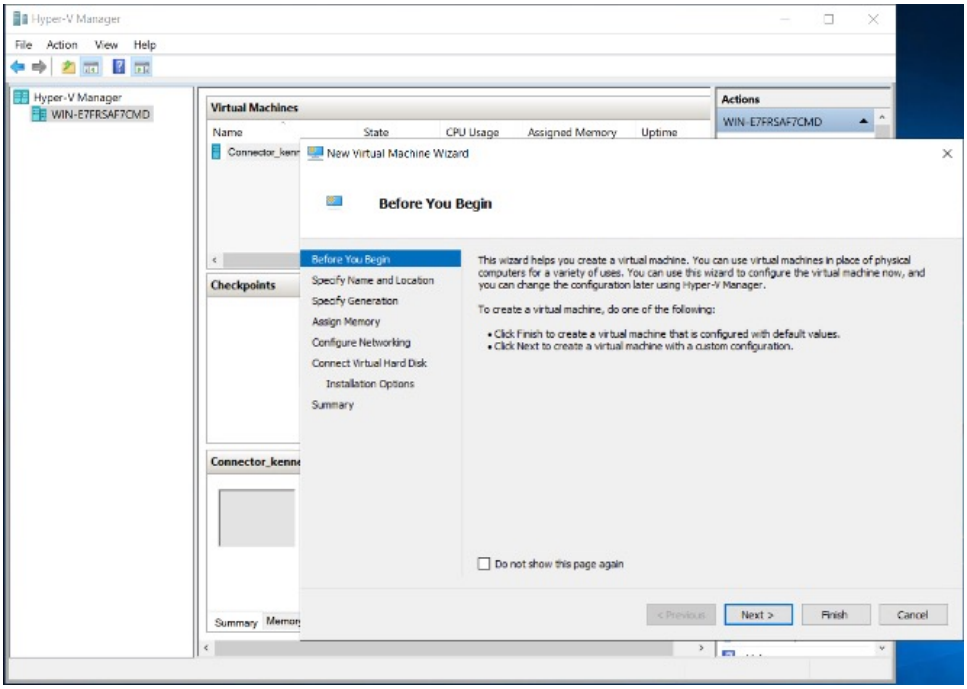
Figure 51: Create New Virtual Machine



Note
Do not use the **Import Virtual Machine** or **New** > **Hard Disk** options.

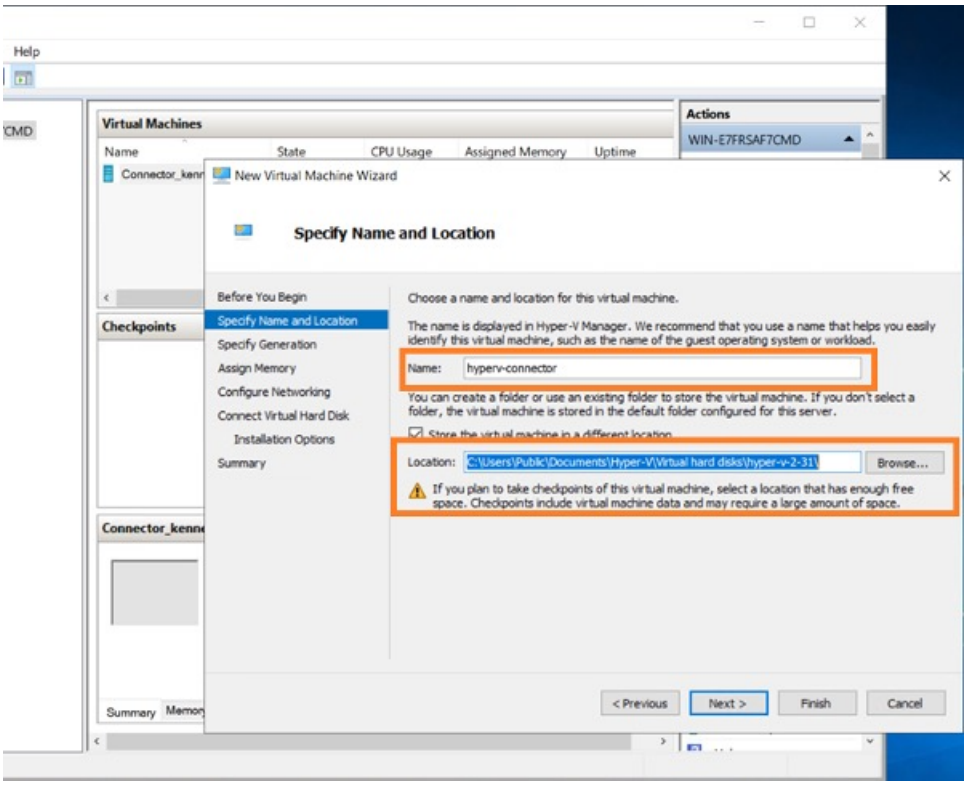
Step 5 Click **Next** to begin HYPER-V deployment.

Figure 52: Click Next to Begin Deployment



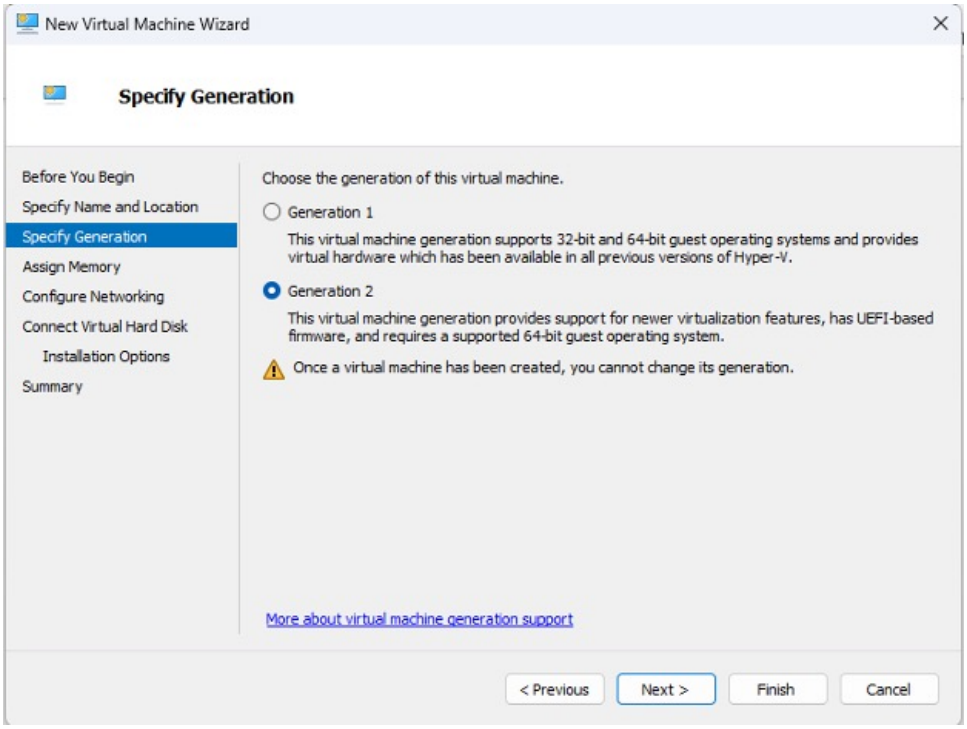
Step 6 Provide the **Name** of the connector and select the location to create the virtual machine.

Figure 53: Name of Connector



Step 7 In the **Specify Generation** window, choose **Generation 2** VM.

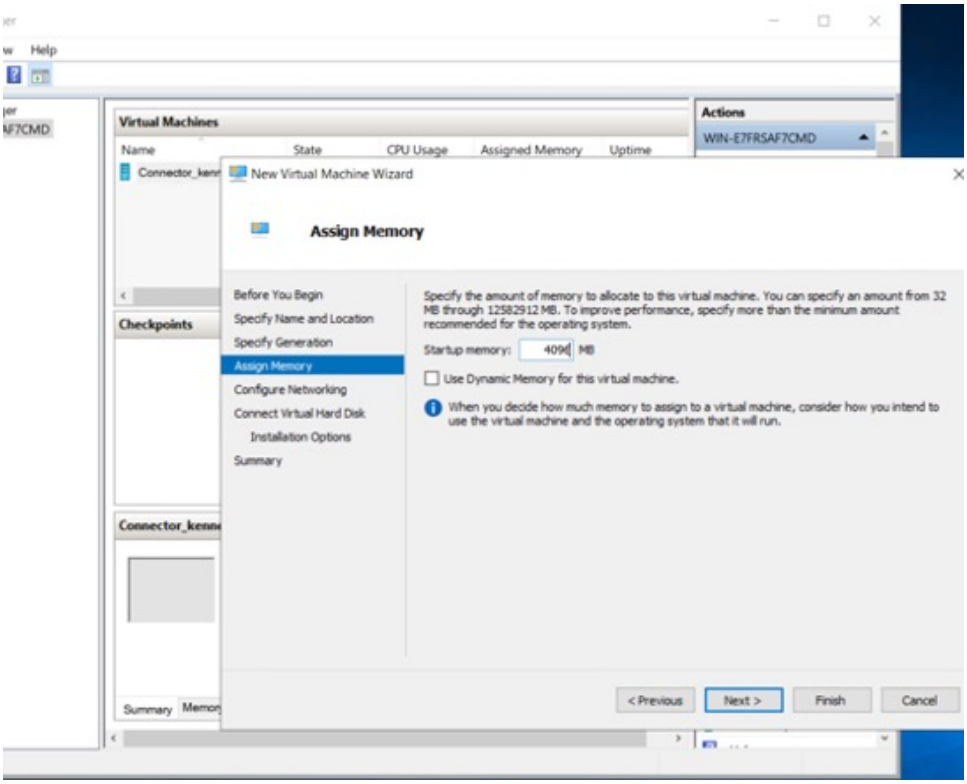
Figure 54: Specify Generation



Step 8 In the **Assign Memory** window, specify 4096 MB (4GB) of memory for the virtual machine instance.

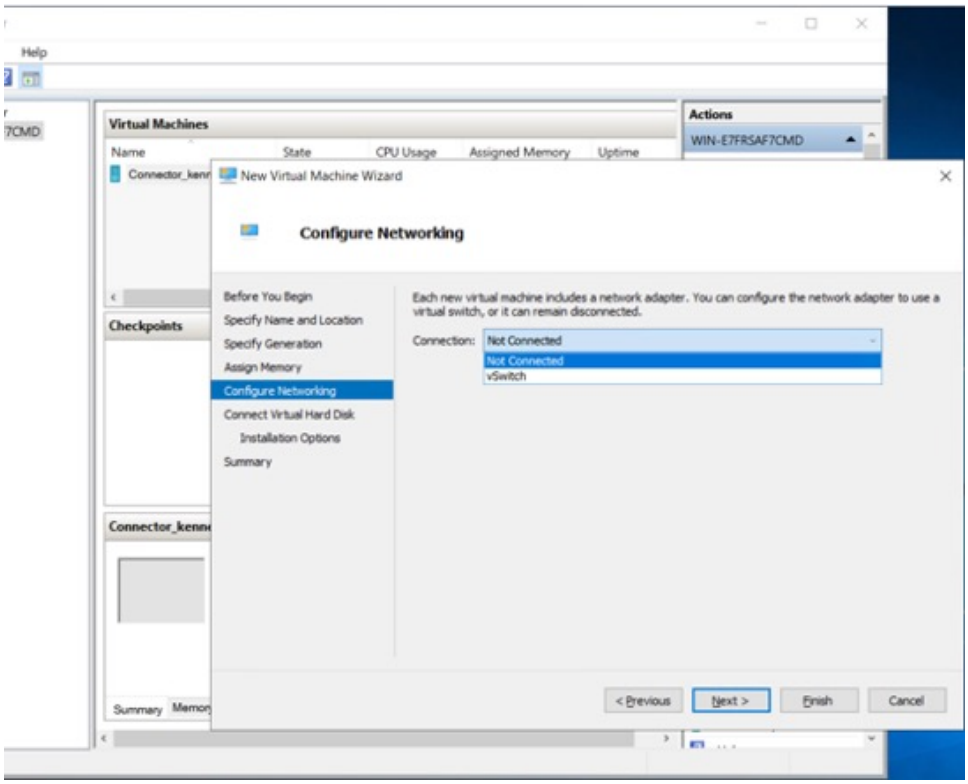
Note
4096 MB (4GB) of memory is equivalent to the standard configuration of HYPER-V.

Figure 55: Assign Memory



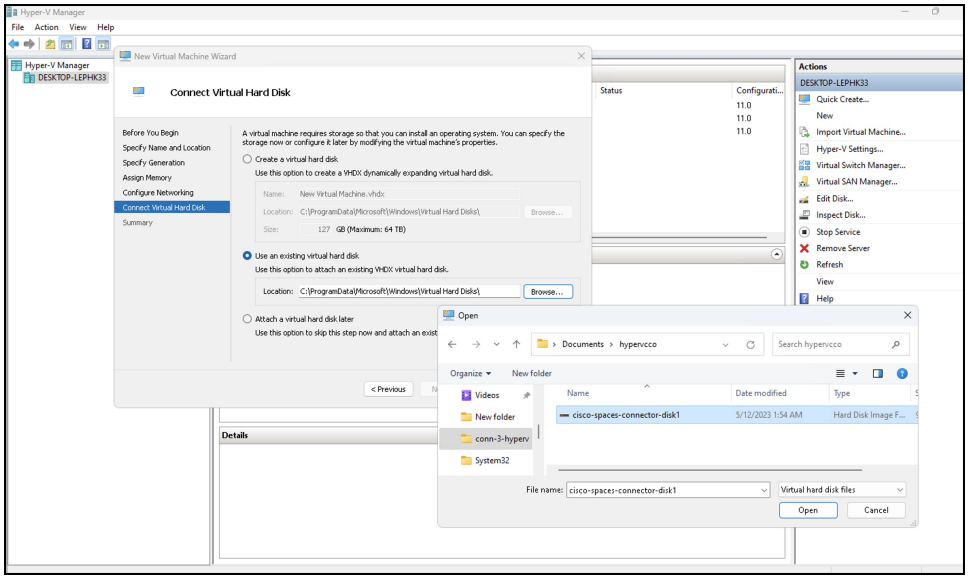
Step 9 In the **Configure Networking** window, select the vSwitch that you created as a prerequisite.

Figure 56: Configure Networking



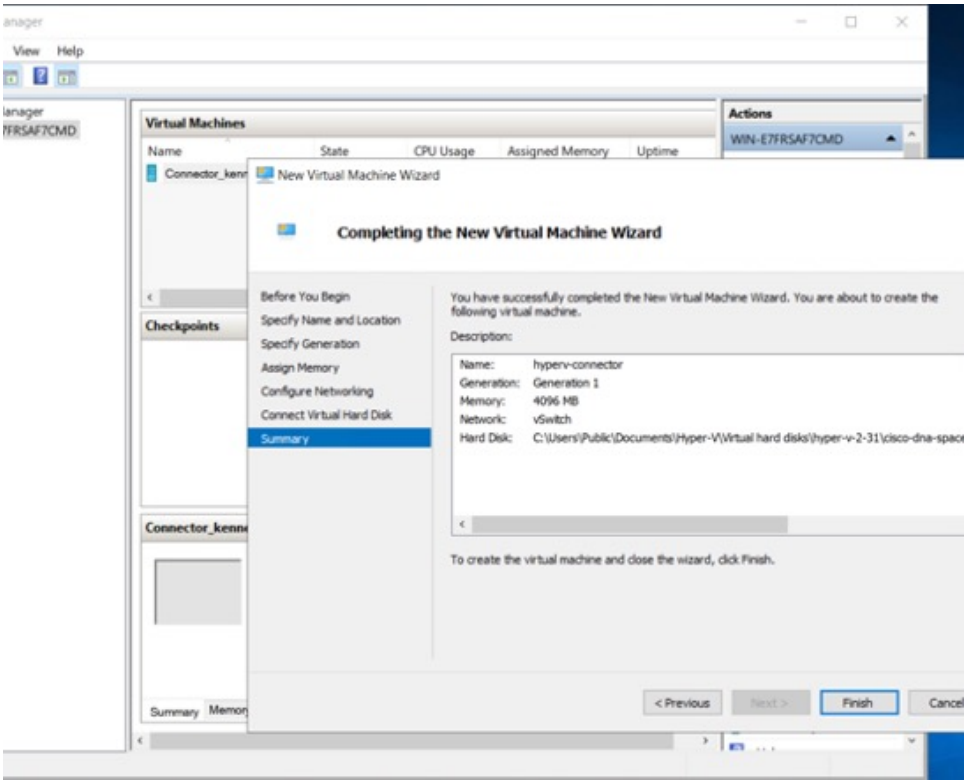
Step 10 In the **Connect Virtual Hard Disk** window, select the **Use an existing hard disk** option, and select the folder location where the VHDX file has been stored (Step 1).

Figure 57: Connect Virtual Hard Disk



Step 11 In the **Completing the New Machine Wizard** window, a final summary is displayed. Review this summary and click **Finish**.

Figure 58: Completing the New Machine Wizard

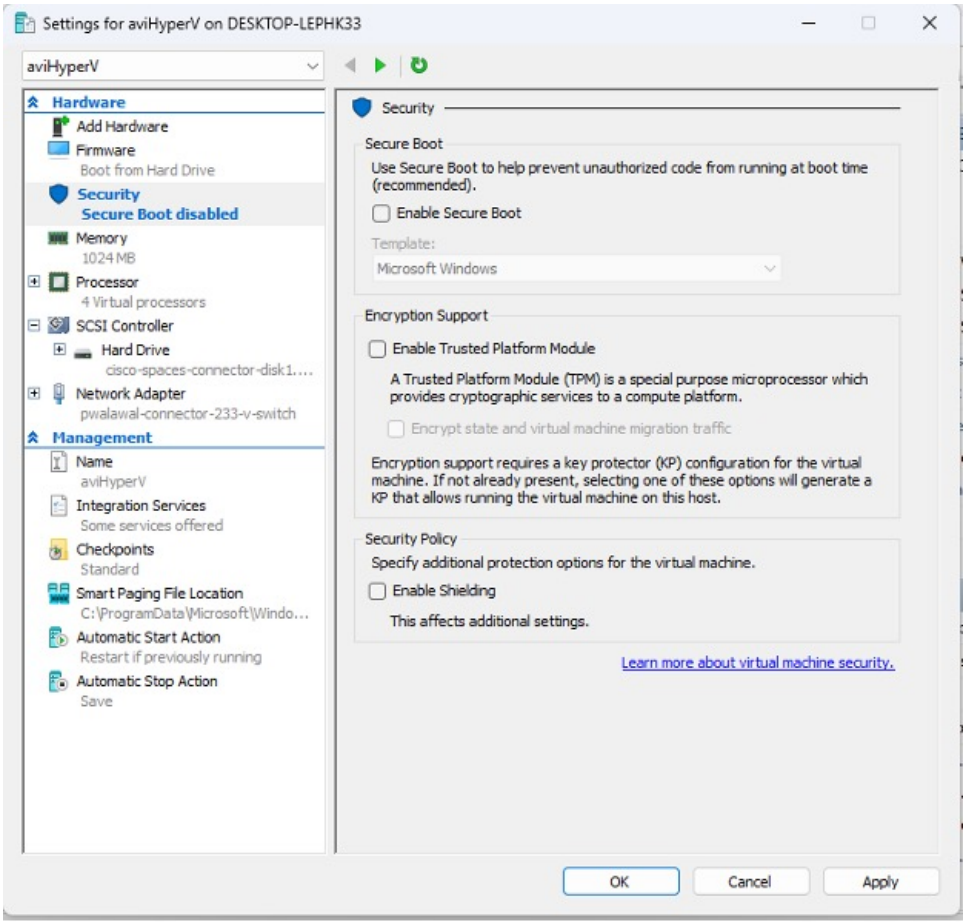


A HYPER-V instance is created.

Step 12 Select the HYPER-V instance created, and click **Settings**.

a) Navigate to **Security** and ensure you **uncheck** the **Enable Secure Boot** check box and leave the secure boot feature disabled.

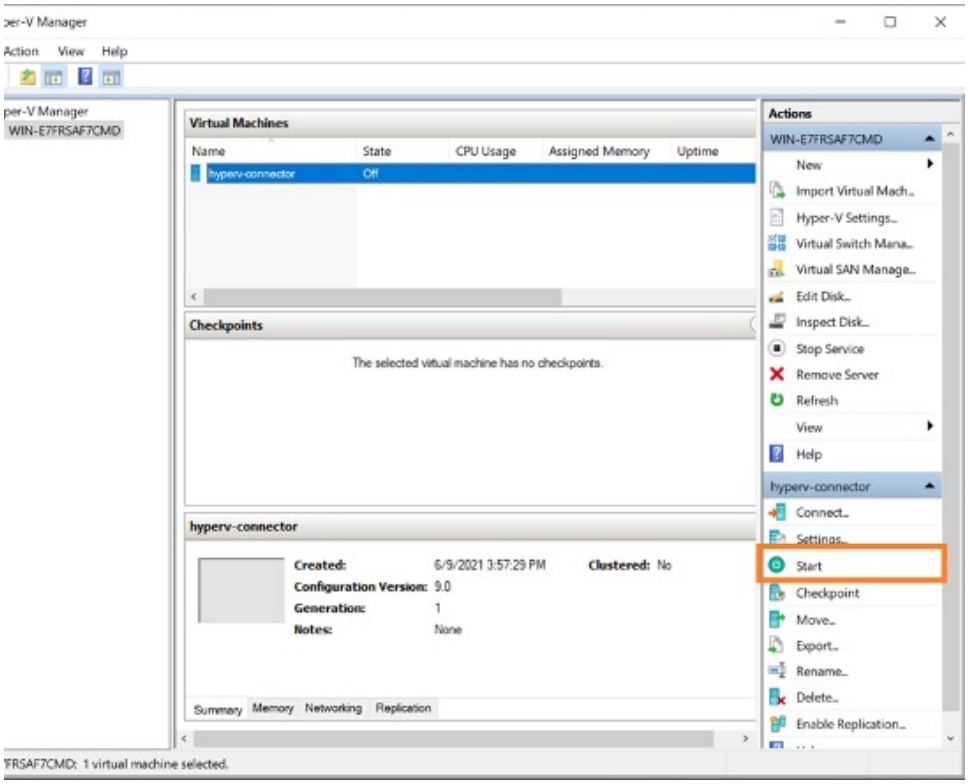
Figure 59: Enable Secure Boot



b) Navigate to **Processor** and ensure that CPU count is set to 2 vCPUs to match **Standard** connector deployment.

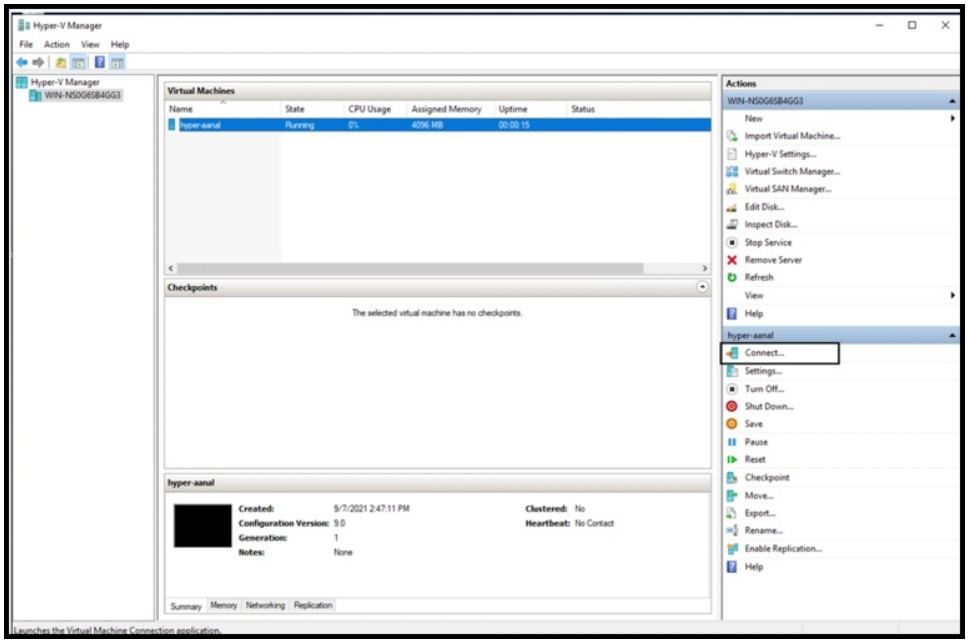
Step 13 Select the HYPER-V instance created, and click **Start**.

Figure 60: Select The Hyper-V Instance



Step 14 Select the HYPER-V instance created, and click **Connect** to open the HYPER-V console.

Figure 61: Select The Hyper-V Instance



The virtual machine terminal is opened.

Step 15 Log in to the terminal and enter the default username **root** and default password **root**.

Step 16 Configure the host name for the connector.

Step 17 Choose an network interface to configure as PRIMARY.

Figure 62: Configuring the Primary Interface: IPv4

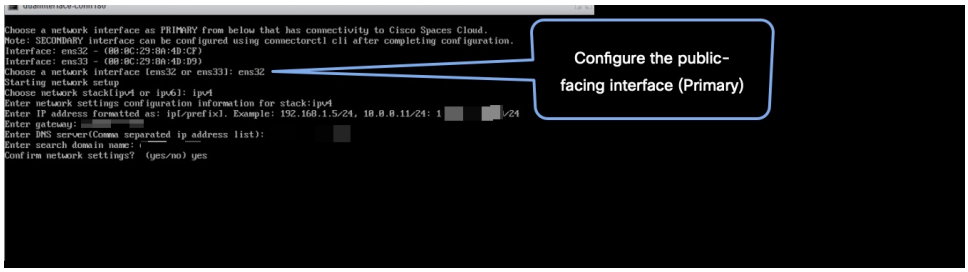
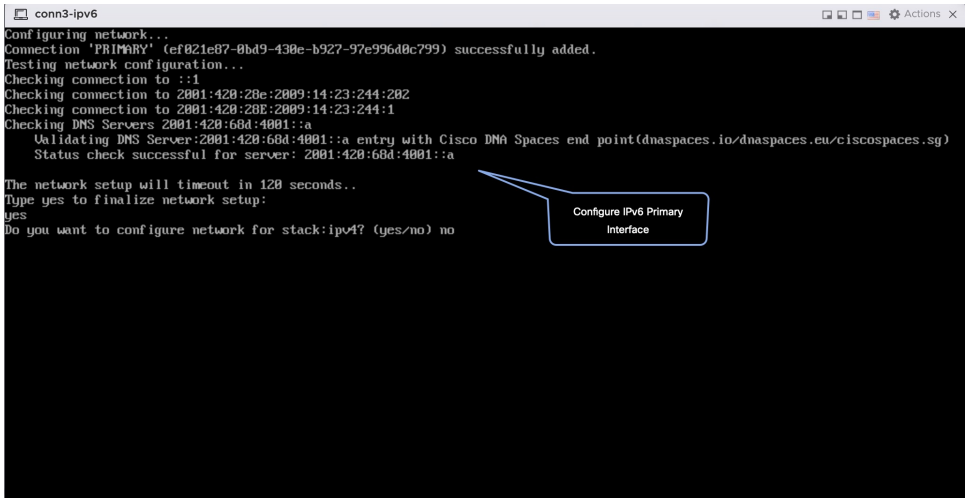


Figure 63: Configuring the Primary Interface: IPv6



Step 18 Do one of the following, and then configure the network settings for the PRIMARY interface. Specify parameters such as IP address, hostname, and so on.

- Configure the IPv6 stack.
- Configure the IPv4 stack.

You can add multiple DNS servers as a comma separated list in this step. After the task is complete and the Cisco Spaces: Connector is deployed, you can login to the connector CLI, and run the **connectorctl network config** command to add more DNS servers or edit the existing list.

Step 19 Confirm the setup.

Note
Because this configuration window times out in 120 seconds, ensure that you provide the input on time to avoid reconfiguration.

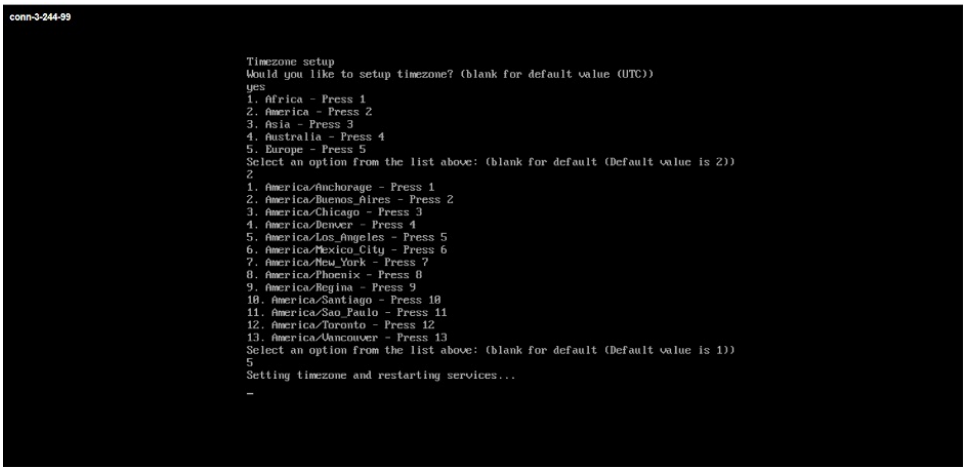
Step 20 Reset the password for the **spacesadmin** user.

Note

The **spacesadmin** password can only include special characters such as '@' or '!' and must not contain any spaces.

Step 21 Enter the time zone.

Figure 64: Time Zone



Step 22 Enter the Network Time Protocol (NTP) server name to synchronize the system time with that of NTP server, or leave it blank if you do not want to configure an NTP server.

Figure 65: Configure NTP

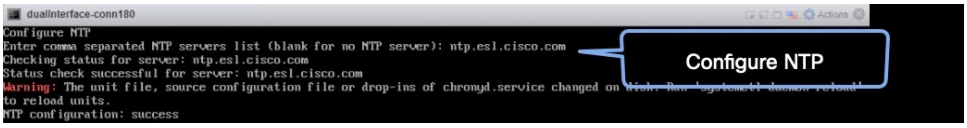
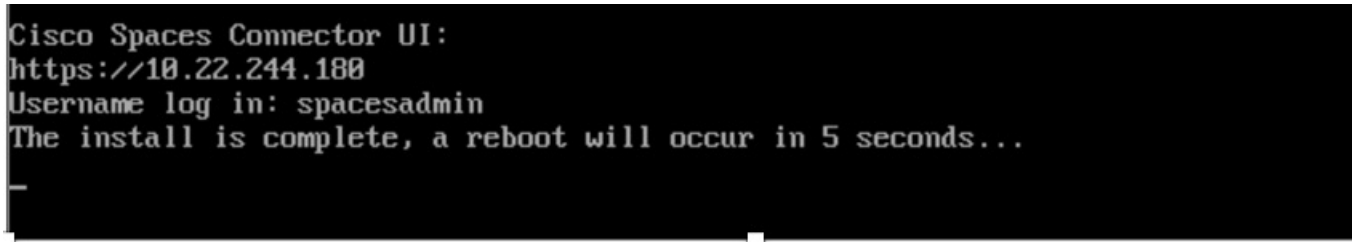


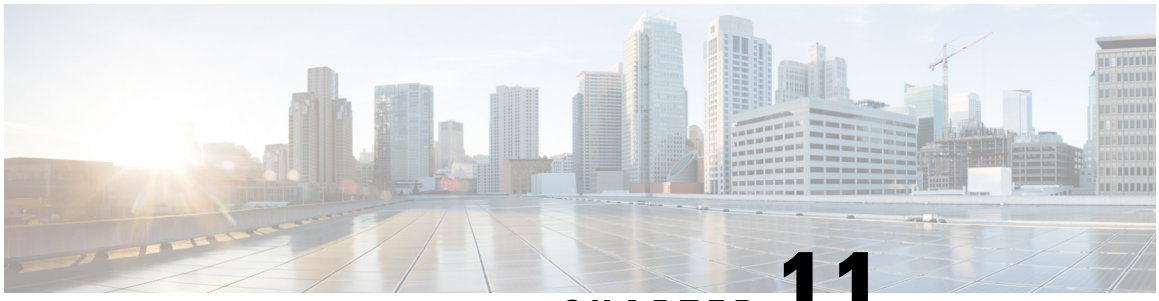
Figure 66: Configure NTP



Step 23 Note the URL (https://connector-ip) before the automatic reboot. You can use this URL later to open the connector GUI.

Figure 67: ConnectorGUI





CHAPTER 11

Cisco Spaces: Connector Azure VMware

• Cisco Spaces: Connector Azure VMware, on page 71

Cisco Spaces: Connector Azure VMware

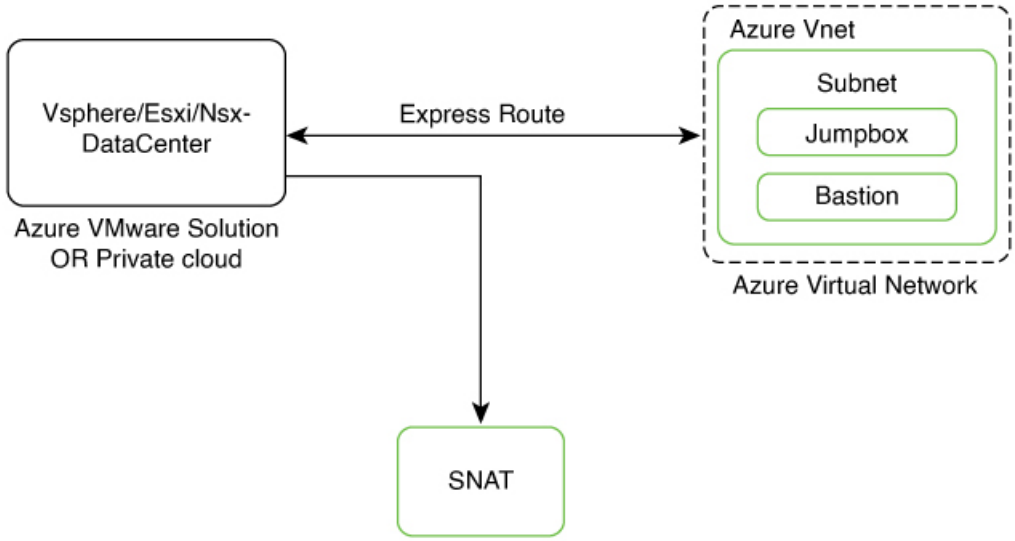
Note This Azure VMware solution is not supported and we recommend following the Cisco Spaces: Connector VM on Azure Environment guide.

The chapter shows you how to install a connector on Azure VMware. To do this, you must understand the various components of this solution.

- The **Azure VMware Solution (AVS) or Private Cloud** is a service offered by Microsoft Azure in collaboration with VMware. It enables organizations to run and manage VMware workloads natively on Azure infrastructure. You can host services such as Cisco Spaces: Connector or wireless controllers.
- **Azure Virtual Network (VNet)** is a building block in Microsoft Azure that enables you to securely connect and isolate Azure resources. It provides a way to create private, isolated, and highly available networks in the Azure cloud. You can deploy some of these services on this VNet:
  - **Azure Bastion** is a service provided by Microsoft Azure for secure and seamless Remote Desktop Protocol (RDP) and Secure Shell (SSH) access to virtual machines (VMs) in the Azure cloud. It acts as a secure gateway, eliminating the need to expose VMs on the Private Cloud to the public internet, and reducing the attack surface. With Azure Bastion, you can connect to your VMs directly from the Azure portal using a web browser, without the need for a public IP address or a VPN connection.
  - **Jumpbox (or Jump Server):** Jumpbox, or jump server, is a security measure used in networking environments. It's a system that sits between an internal network and external networks (such as the internet) and is a single point of entry for administrators. Instead of allowing administrators to connect directly to critical systems such as connector on the Private Cloud, they connect first to the jumpbox, which acts as a gateway to access other systems. This adds an additional layer of security and control over who can access sensitive systems.
- **Source Network Address Translation (SNAT):** SNAT refers to a type of network address translation that translates the source IP address of outgoing traffic. SNAT is commonly used in scenarios where

multiple private IP addresses from a local network need to access resources on the internet or another network.

Figure 68: Various Components to InstallConnector onAzure VMware



To deploy a connector on Azure VMware, you have to do the following:

1. Creating an Azure VMware solution (or Private Cloud), on page 72 and deploying the connector OVA on it.
2. Creating an Azure Virtual Network, on page 76. You can then allow administrators and users to access the connector through this VNet.

Creating an Azure VMware solution (or Private Cloud)

This chapter provides information about how to download and deploy the Cisco Spaces: Connector and obtain the URL for the connector GUI.

Before you begin

- Identify the subscription you plan to use for the Azure VMware solution.
- Identify the Size Hosts. This requires you to raise a case with Azure customer support.
- Identify the address range and subnet for the private cloud. All your VMware resources including connector are hosted in this IP range.

SUMMARY STEPS

1. Log in to portal.azure.com.
2. Create a **Resource**.
3. Choose the **Azure VMware Solution** service.
4. In the **Create a private cloud** window that appears, fill the required details.

5. Configure a segment for the private cloud.
6. Specify the DHCP range to be used for this segment.
7. Specify a DNS from the left-navigation pane or while installing the connector later.
   - You can use a public DNS while deploying the connector.
   - You can configure an internal DNS from the left-navigation pane.
8. Provide internet connectivity using SNAT. From the left-navigation pane, click **Internet Connectivity > Connect using SNAT**. This enables outbound internet access for this private cloud.
9. Find the credentials of this private cloud. From the private-cloud left pane, click **VMware credentials**. You can observe the credentials of various components of the private cloud. Make a note of these credentials for later use.

DETAILED STEPS

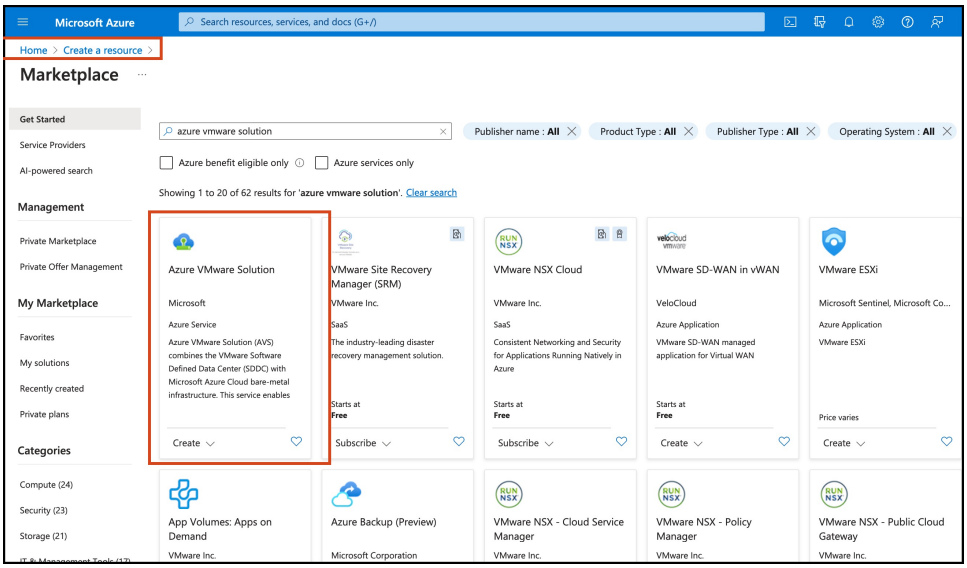
Procedure

Step 1 Log in to portal.azure.com.

Step 2 Create a **Resource**.

From the left-navigation pane, click **Create a Resource**.

Figure 69: Create a Resource



Step 3 Choose the **Azure VMware Solution** service.

a) In the **Search services and marketplace** field, search for an **Azure VMware solution.**
b) From the displayed search results, click **Create** and choose the **Azure VMware solution.**

Step 4 In the **Create a private cloud** window that appears, fill the required details.

a) Choose a subscription.
b) Choose a resource group or create a new one.
c) Choose the location of the service.

d) Choose the size of the host.
e) Choose the host location.
f) Choose the number of hosts. The minimum number of hosts is three.
g) Enter the address block. This IP address block is used to deploy various services such as connector, and these services are accessible via a browser from the Azure Virtual Network.

The Azure VMware solution (or private cloud) is created.

Figure 70: Create a private cloud

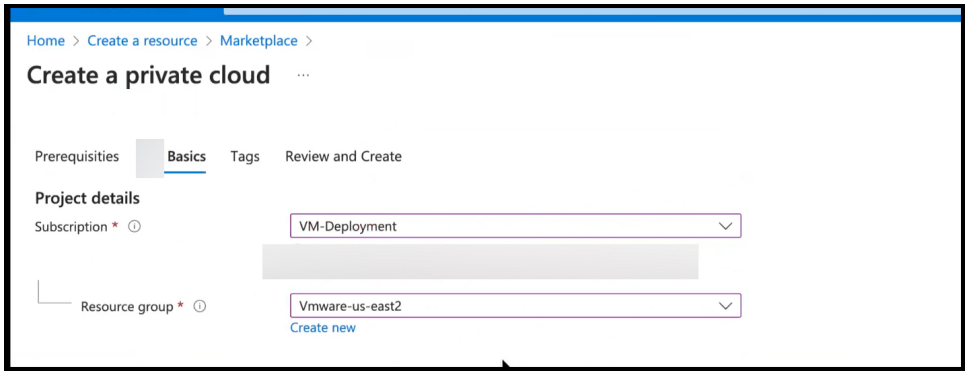
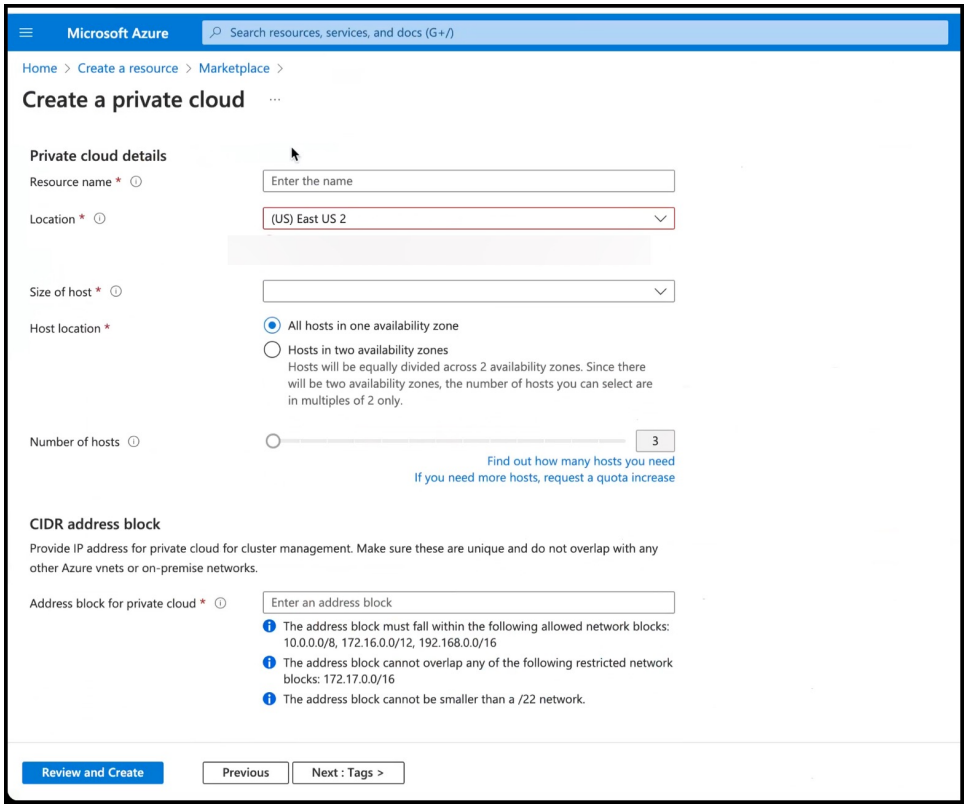


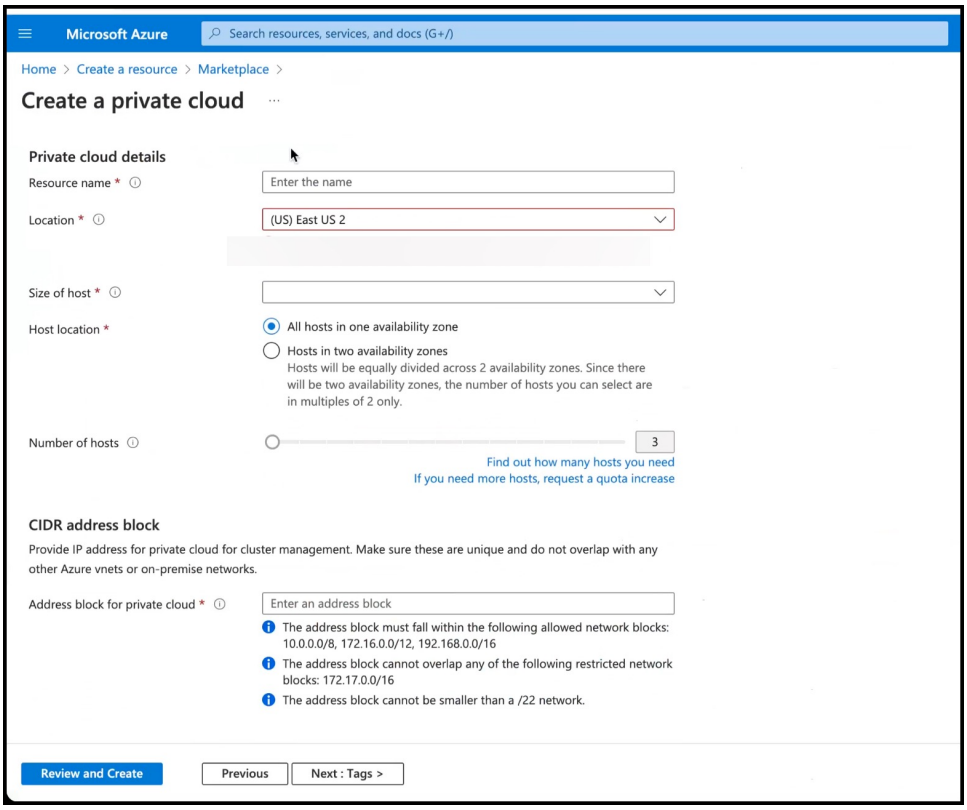
Figure 71: Create a private cloud



Step 5 Configure a segment for the private cloud.

a) From the private-cloud left pane, click **Segments**. You can see that a default segment has already been created and allocated with addresses from the address range specified by you earlier. You can use this existing segment or create a new one.

Figure 72: Create a Segment



Step 6 Specify the DHCP range to be used for this segment.

a) From the private-cloud left pane, click **DHCP**.
b) Select the **DHCP type** as **SERVER**.
c) Enter the **Server Name** as the segment chosen earlier for this private cloud.
d) Enter the **Server IP address** as the segment address range selected earlier.

Step 7 Specify a DNS from the left-navigation pane or while installing the connector later.

- You can use a public DNS while deploying the connector.
- You can configure an internal DNS from the left-navigation pane.

Step 8 Provide internet connectivity using SNAT. From the left-navigation pane, click **Internet Connectivity > Connect using SNAT**. This enables outbound internet access for this private cloud.

Step 9 Find the credentials of this private cloud. From the private-cloud left pane, click **VMware credentials**. You can observe the credentials of various components of the private cloud. Make a note of these credentials for later use.

Figure 73: Various Components to InstallConnector onAzure VMware

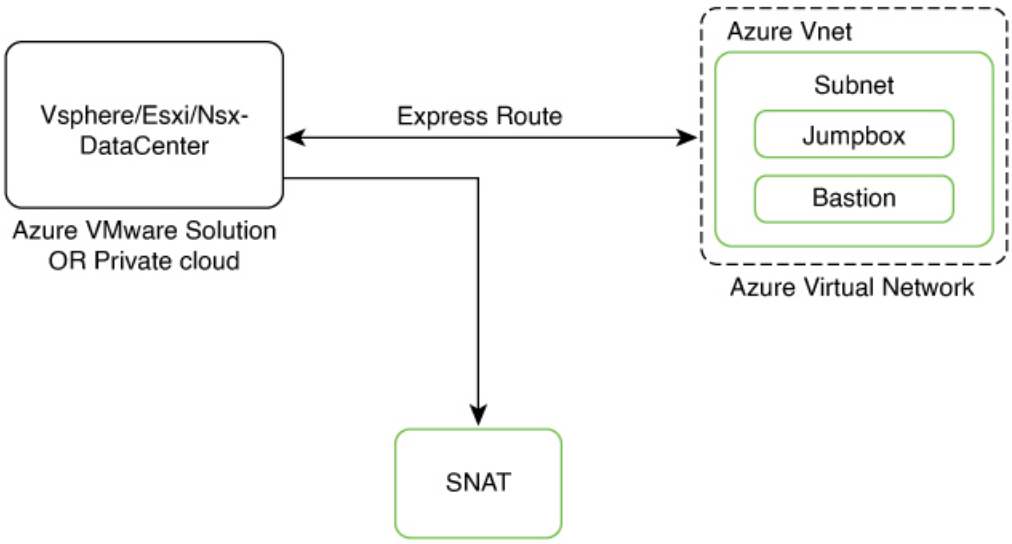
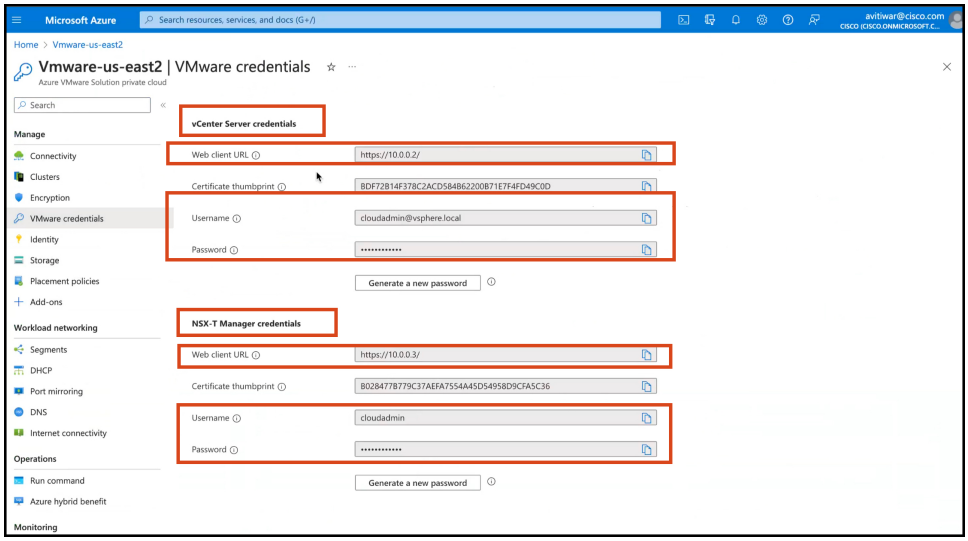


Figure 74: VMware Credentials



Note
Note that ESXi also inherits the vSphere credentials.

Creating an Azure Virtual Network

Before you begin

Create a Azure VMware solution (or Private Cloud) and configure it with SNAT.

Procedure

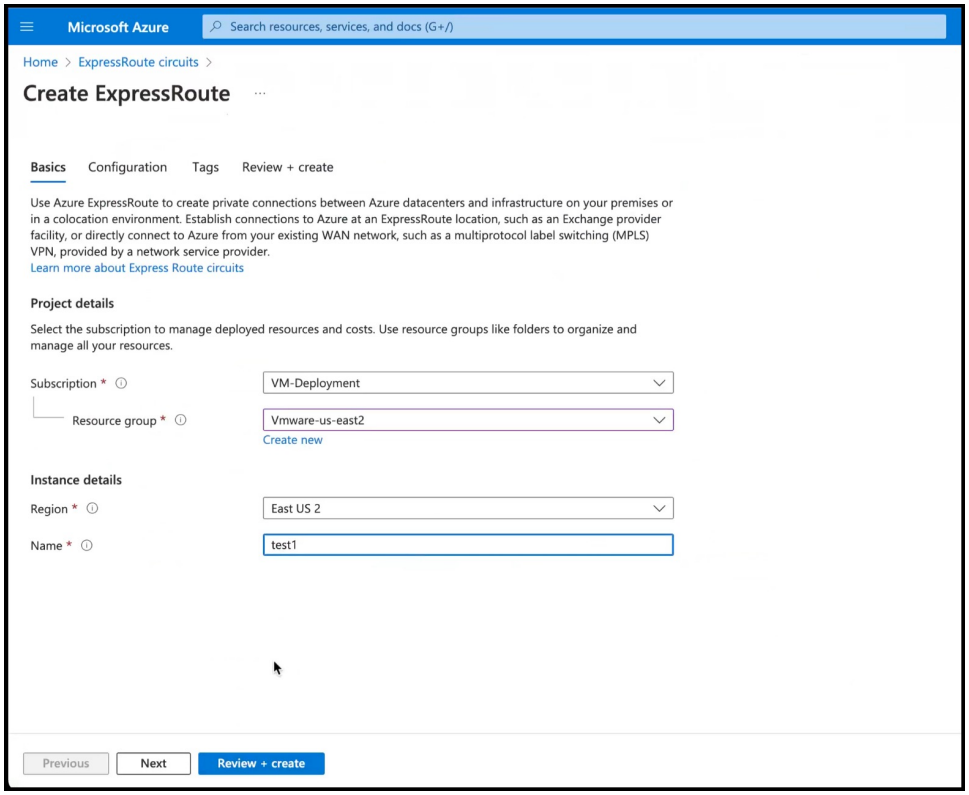
Step 1 Create an **ExpressRoute**.

a) From the Microsoft Azure Home Page, click **ExpressRoute circuits**.

b) From the **ExpressRoute circuits** page that is displayed, click **Create**.

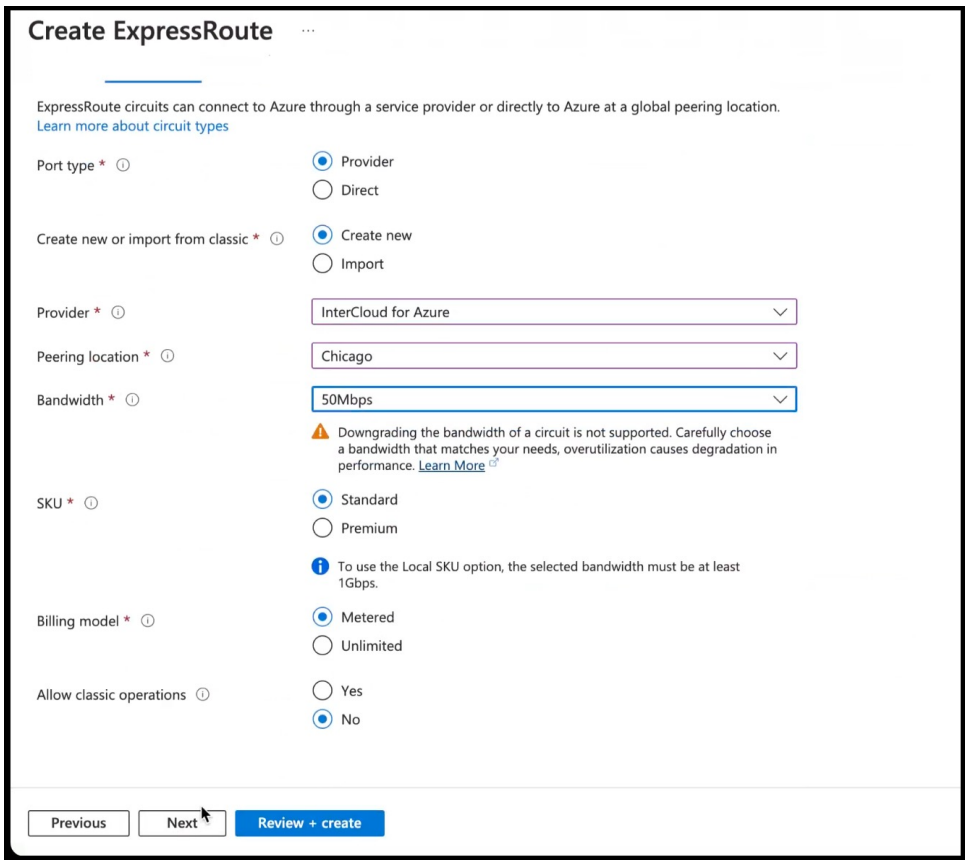
c) From the **Create ExpressRoute** page that is displayed, enter the details of the **Basic** tab. Click **Next**.

Figure 75: Basics Tab



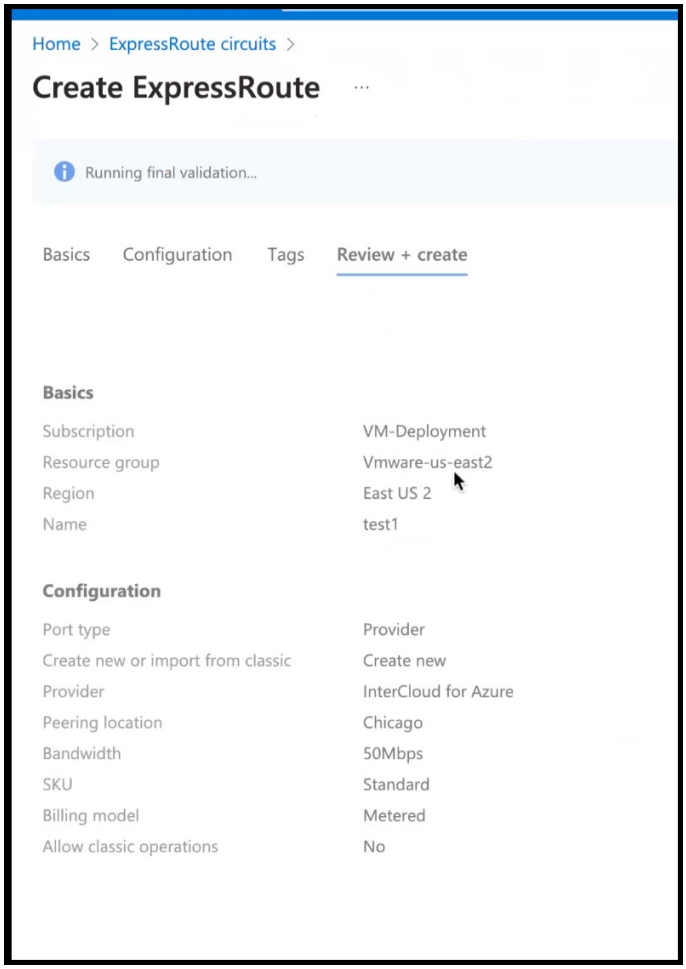
d) Click the **Configuration** tab. Fill in details such as **Provider**.

Figure 76: Configuration Tab



e) Click the **Review + Create** tab, and review the changes you have made. Click **Create** to create the ExpressRoute.

Figure 77: Review + Create



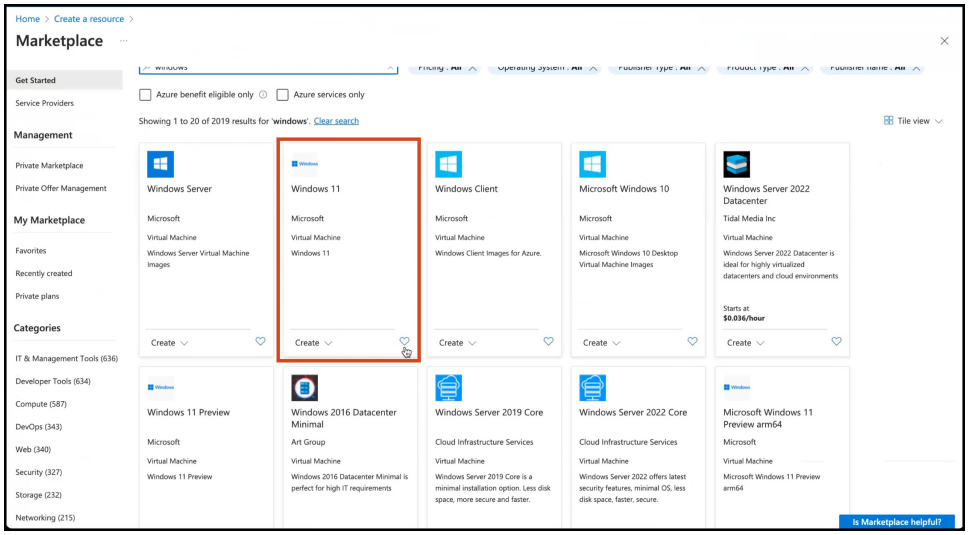
Step 2 From the created **Virtual Network**, do the following.

a) Create a Gateway subnet and provide an IP address.
b) Create a Bastion and provide an IP address.
c) Create an AzureBastion subnet and provide an IP address.

Step 3 Deploy a Windows Machine as a virtual machine. You can use this as a Jumpbox to access vSphere or NSXT-Manager.

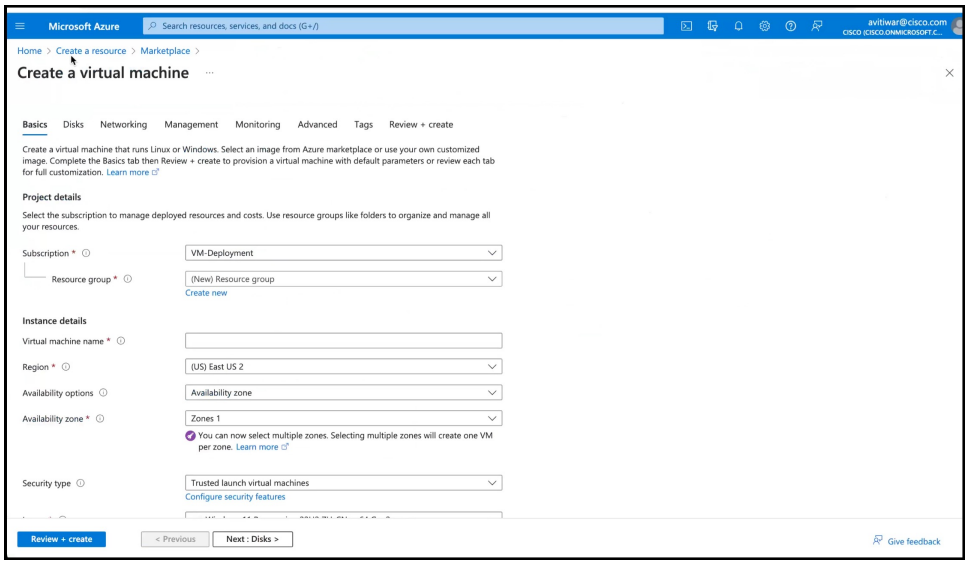
a) From the left-navigation pane, click **Create a Resource**
b) Search for an operating system of choice. For example, Windows 11, click **Create** and choose the version of choice.

Figure 78: Windows 11 virtual machine



c) In the **Create a virtual machine** window, enter the relevant details

Figure 79: Create a Virtual Machine



A jumpbox of your preferred operating system is deployed. Use this to access your services.

Step 4 You can login to the vSphere service. Use the credentials retreived when creating the private cloud, from the **VMware Credentials** > **vCenter Server credentials** section.

- Launch the Jumpbox, and use a browser to access the service.
- Since Bastion is deployed on the virtual network, you can use SSH or remote desktop protocol (RDP) to access the service.

Figure 80: VMware Credentials

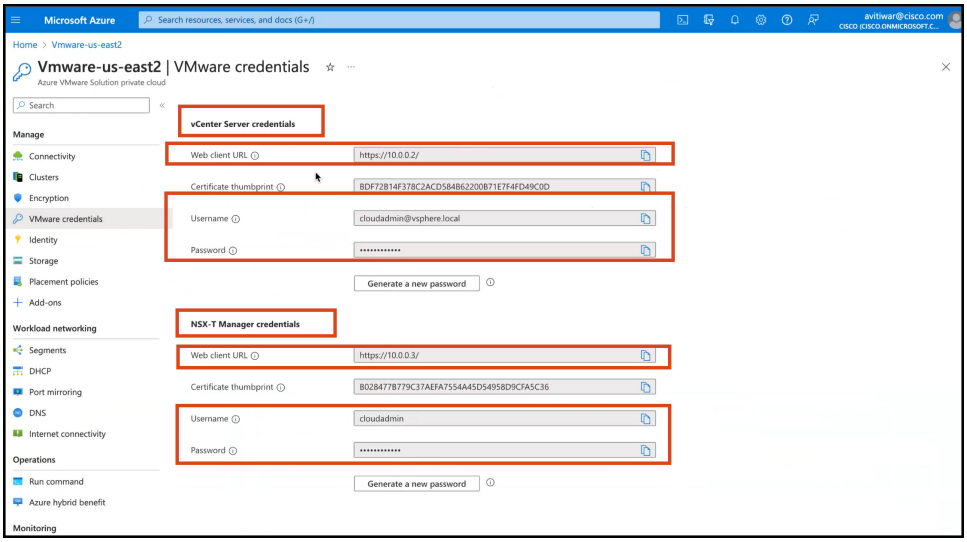
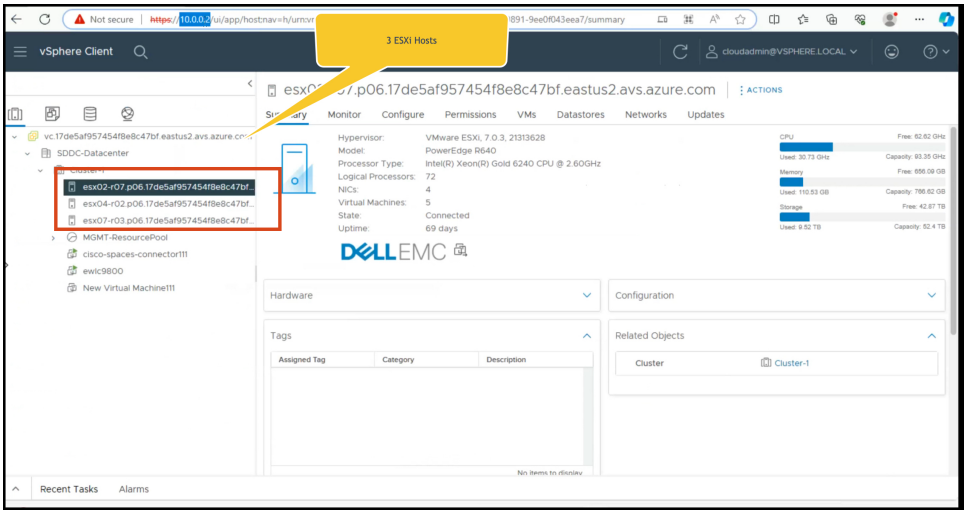


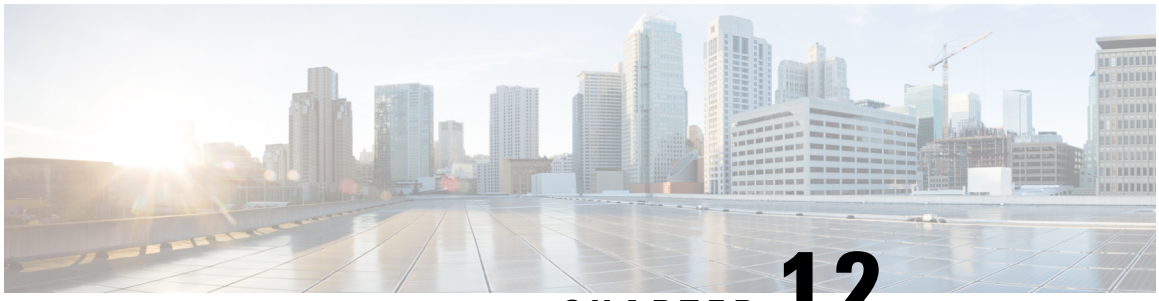
Figure 81: VMware Credentials



Note
ESXi inherits the vSphere credentials.

You can notice that there are at least three ESXi hosts available by default.

Step 5 Deploy the OVA on one of the hosted ESXi. See Deploying the Connector 3 OVA (Single Interface), on page 83



CHAPTER 12

Cisco Spaces: Connector OVA

- Deploying the Connector 3 OVA (Single Interface), on page 83
- Deploying the Cisco Spaces: Connector OVA (Dual Interface), on page 91
- Using Snapshots for Backup , on page 99

Deploying the Connector 3 OVA (Single Interface)

This chapter provides information about how to download and deploy the Cisco Spaces: Connector 3 and obtain the URL for the connector GUI.

Before you begin

Ensure you have the minimum configuration required for installing connector OVA:

- 2 vCPU
- 4-GB RAM
- 120-GB hard disk

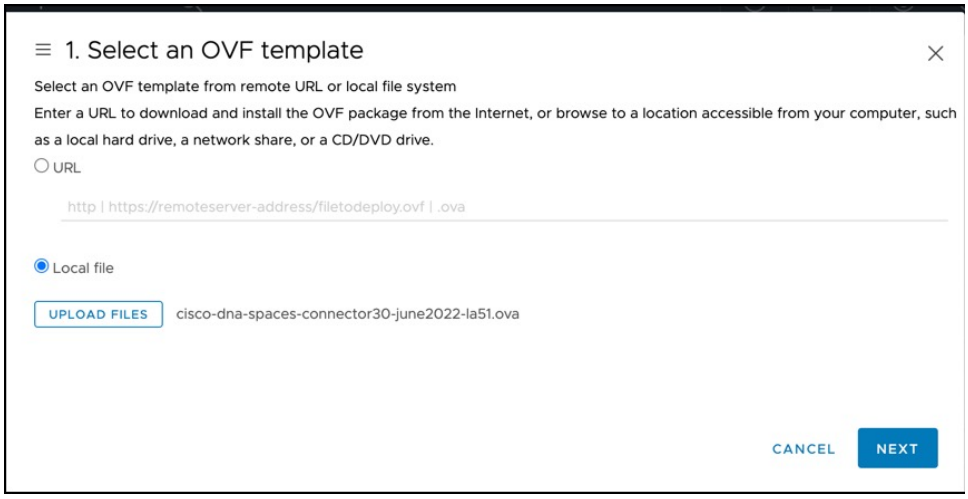
Procedure

Step 1 Download connector OVA to your local system.

Step 2 Create a virtual machine (VM) in the ESXi server and deploy the downloaded Cisco Spaces: Connector OVA.

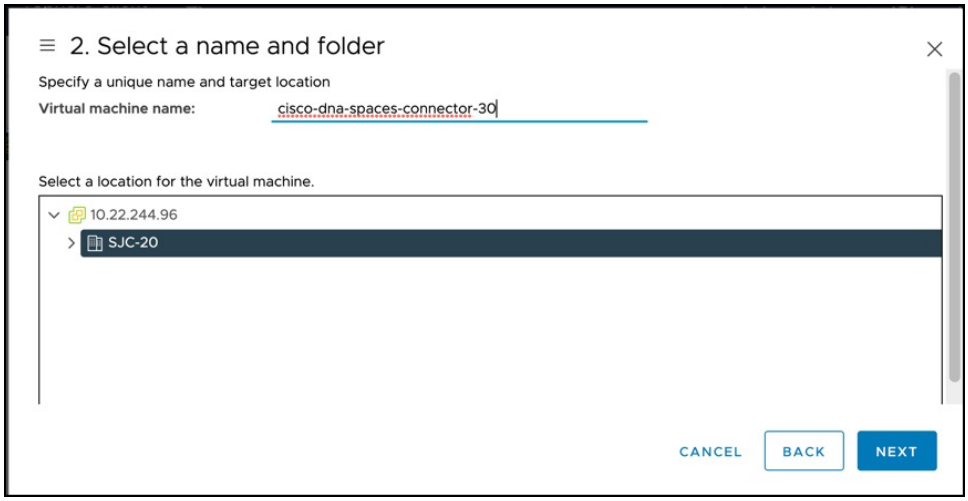
Step 3 In the **1. Select an OVF template** window, click **UPLOAD FILES**, and select the corresponding connector OVA files or drag and drop the downloaded file, and click **Next**.

Figure 82: 1. Select an OVF template



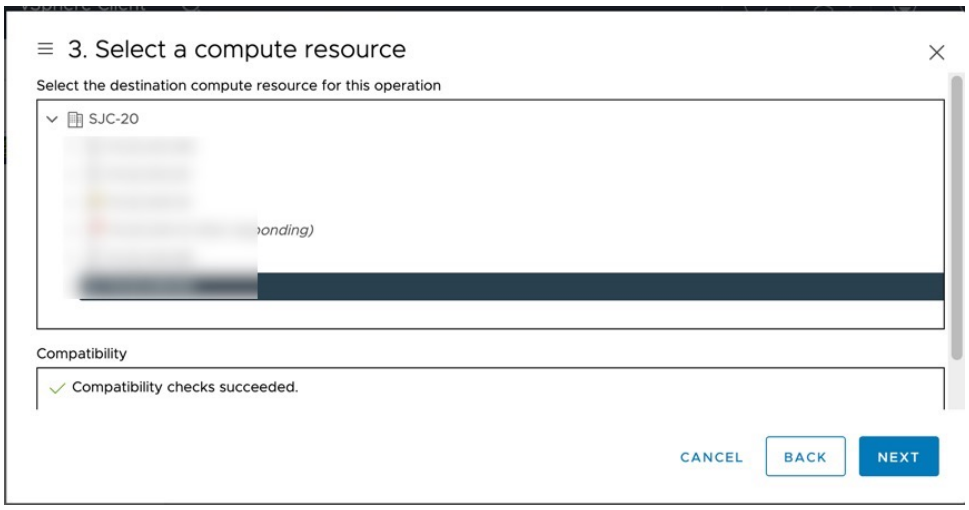
Step 4 In the **2. Select a name and folder** window, enter a name for the VM, and choose a location for the VM, and click **Next**.

Figure 83: 2. Select a Name and Folder



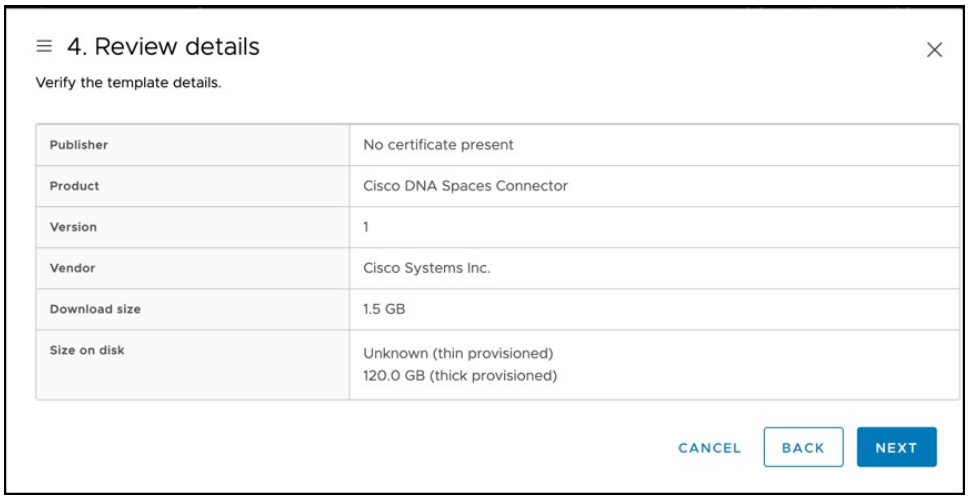
Step 5 In the **3. Select a compute resource** window, select a destination compute resource, and click **Next**.

Figure 84: 3. Select a Compute Resource



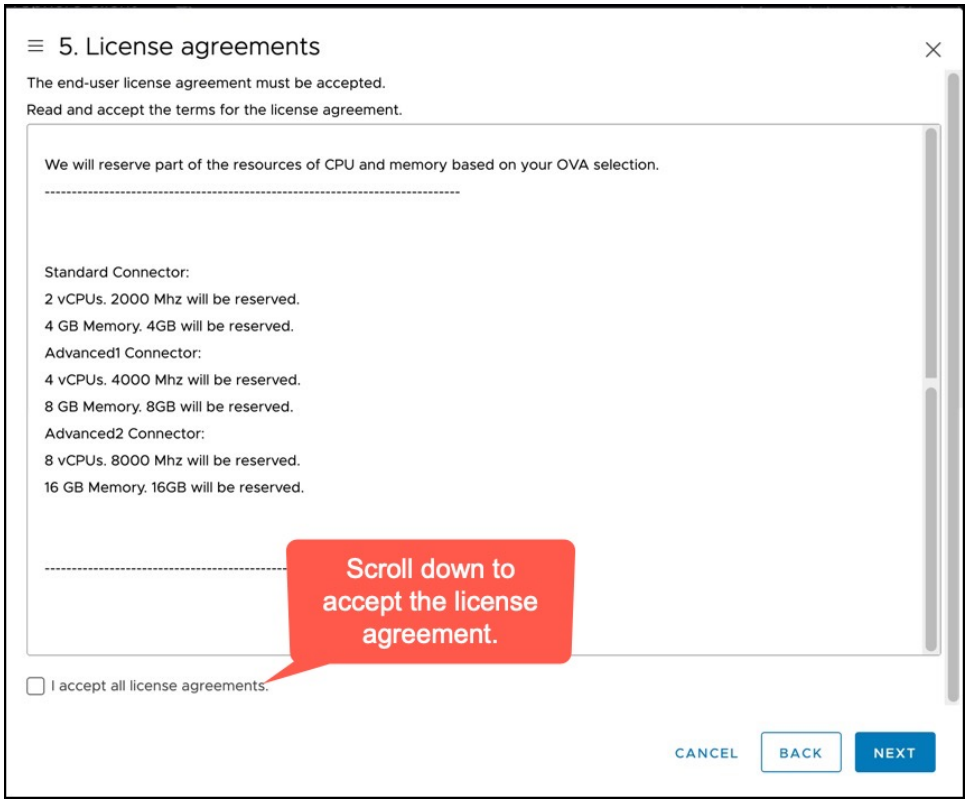
Step 6 In the **4. Review details** window, read and verify the template details, and click **Next**.

Figure 85: 4. Review Details



Step 7 In the **5. License agreements** window, read the license agreement that is displayed and scroll to the end. Check **I accept all license agreements** and then click **Next**.

Figure 86: 5. License Agreements

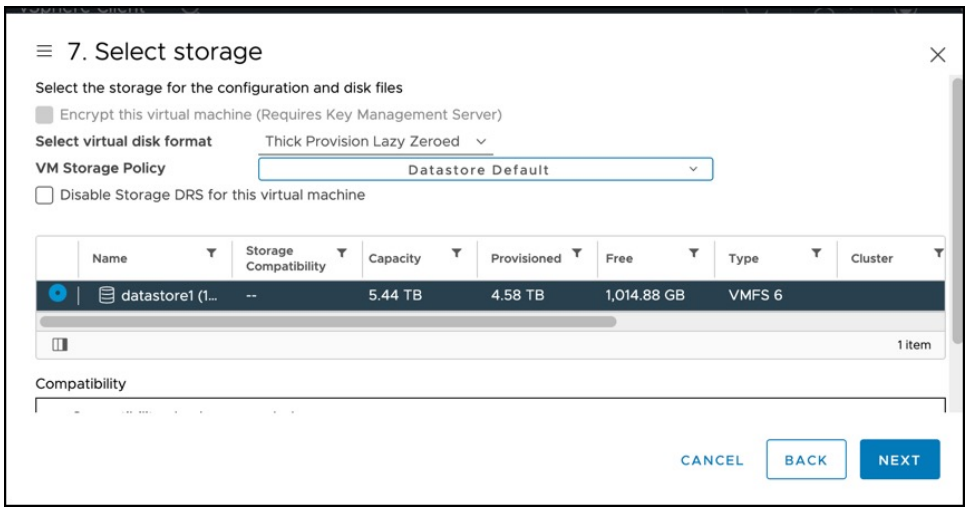


Step 8 In the **6. Configuration** window, choose one of the following, and click **Next**.

- **Standard**
- **Advanced1**
- **Advanced2**

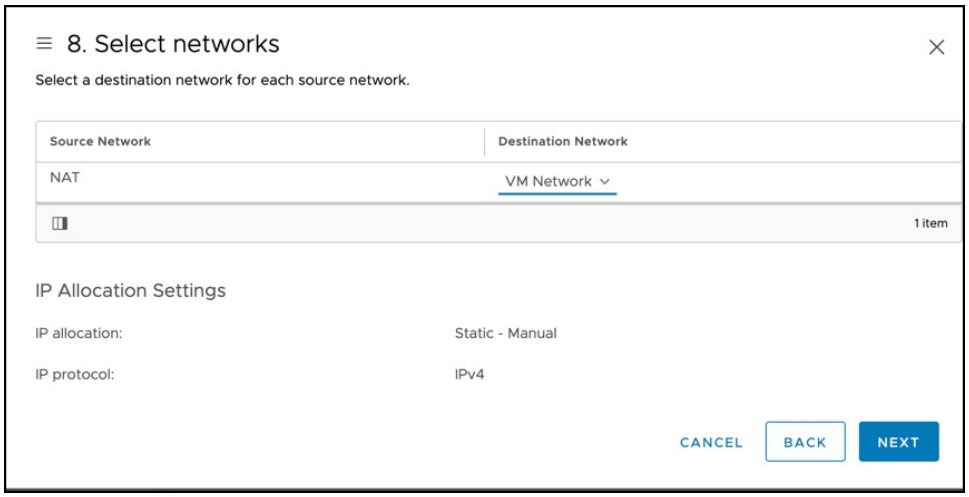
Step 9 In the **7. Select storage** window, choose the standard storage configuration, and click **Next**.

Figure 87: 7. Select storage



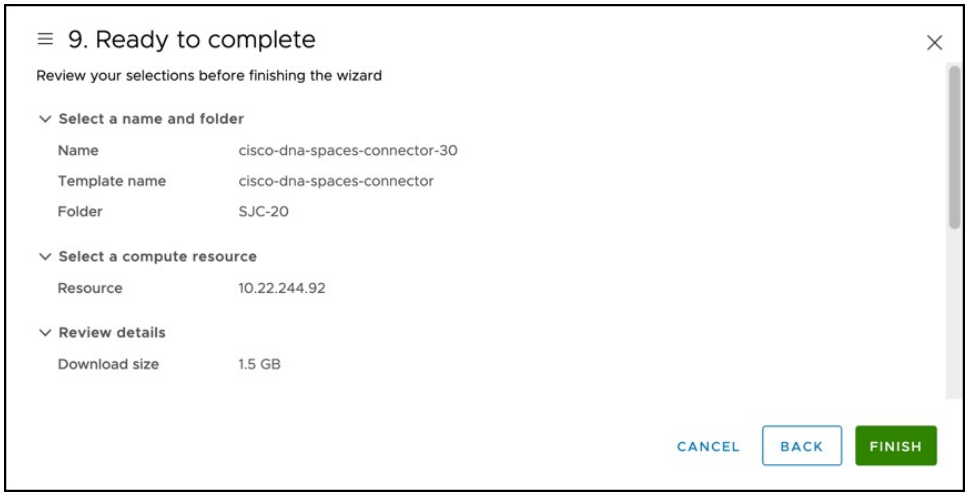
Step 10 In the **8. Select networks** window, choose a destination network, and click **Next**.

Figure 88: 8. Select Networks



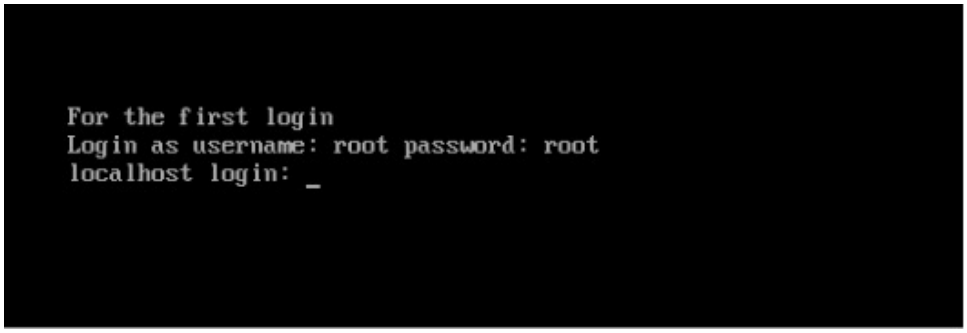
Step 11 In the **9. Ready to complete** window, review the configurations and click **Finish**.

Figure 89: 9. Ready to Complete



Step 12 Power on your VM and log in to the terminal and enter the default username **root** and default password **root**.

Figure 90: First Login Credentials root/root



Step 13 Choose an network interface to configure as PRIMARY.

Figure 91: Configuring the Primary Interface: IPv4

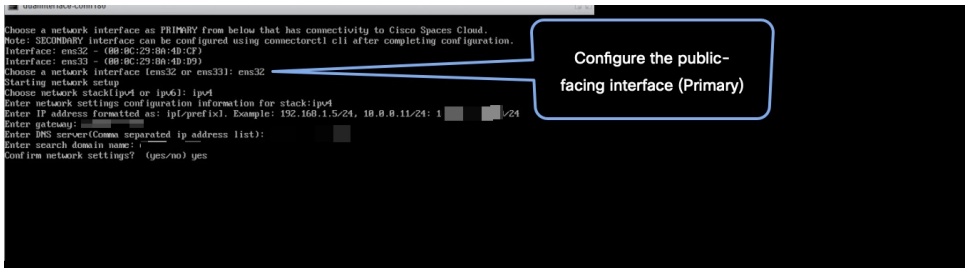
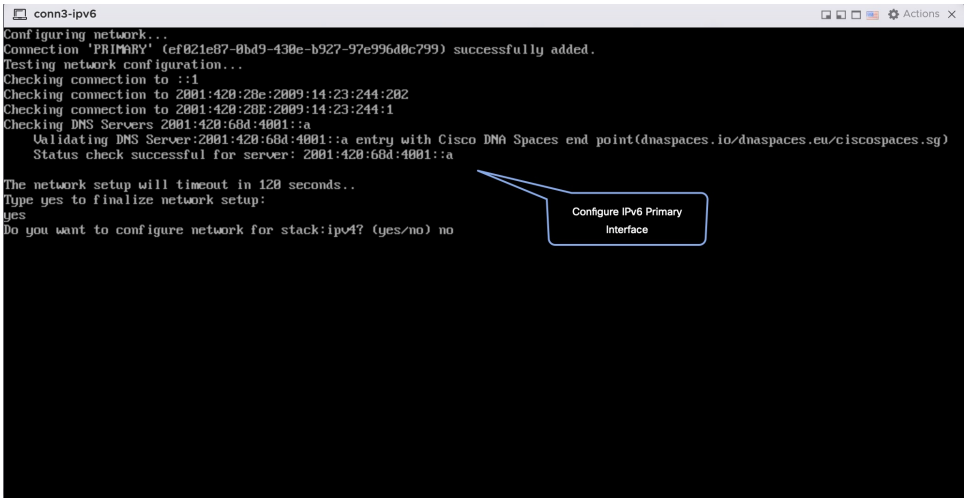


Figure 92: Configuring the Primary Interface: IPv6



Step 14 Do one of the following, and then configure the network settings for the PRIMARY interface. Specify parameters such as IP address, hostname, and so on.

- Configure the IPv6 stack.
- Configure the IPv4 stack.

You can add multiple DNS servers as a comma separated list in this step. After the task is complete and the Cisco Spaces: Connector is deployed, you can login to the connector CLI, and run the **connectorctl network config** command to add more DNS servers or edit the existing list.

Step 15 Confirm the setup.

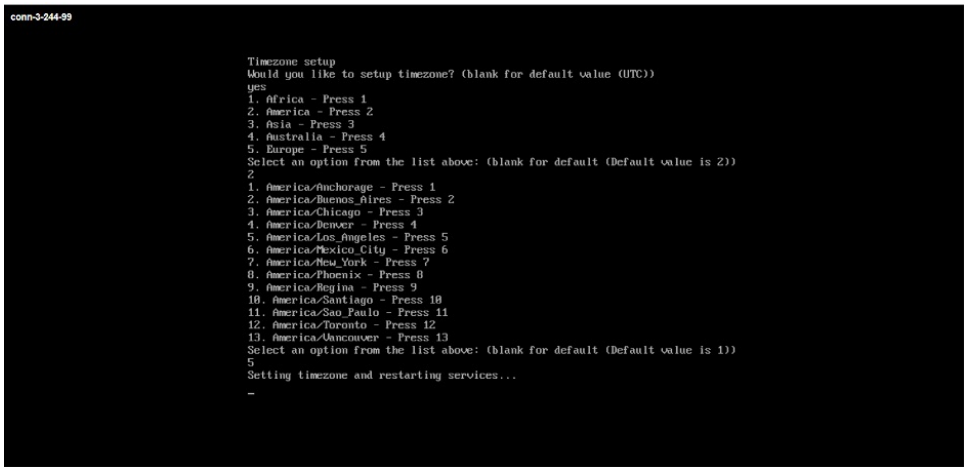
Note
Because this configuration window times out in 120 seconds, ensure that you provide the input on time to avoid reconfiguration.

Step 16 Reset the password for the **spacesadmin** user.

Note
The **spacesadmin** password can only include special characters such as '@' or '!' and must not contain any spaces.

Step 17 Enter the time zone.

Figure 93: Time Zone



Step 18 Enter the Network Time Protocol (NTP) server name to synchronize the system time with that of NTP server, or leave it blank if you do not want to configure an NTP server.

Figure 94: Configure NTP

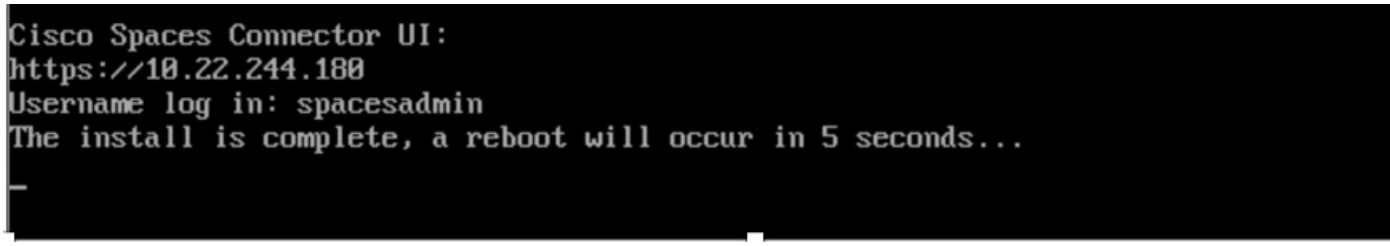


Figure 95: Configure NTP



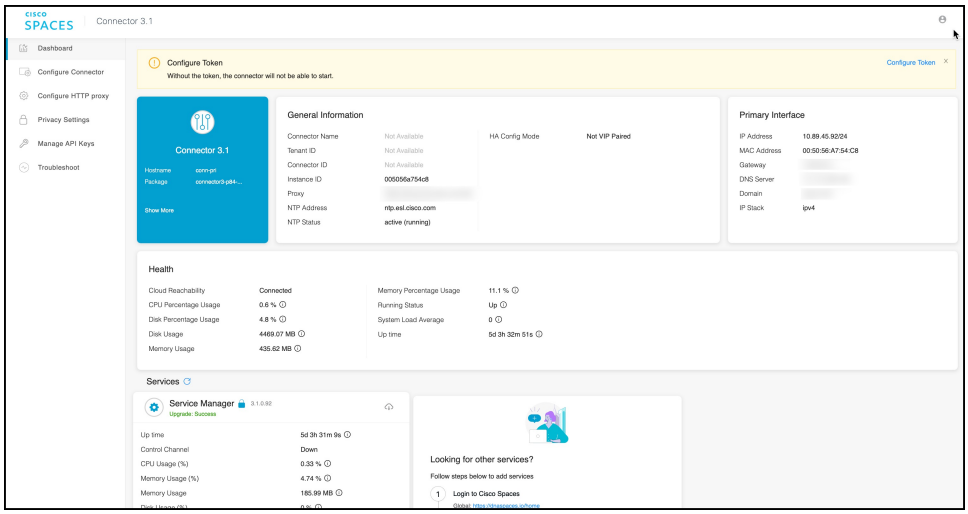
Step 19 Note the URL (https://connector-ip) before the automatic reboot. You can use this URL later to open the connector GUI.

Figure 96: ConnectorGUI



Step 20 In a browser window, enter the noted URL and press Enter to open the connector GUI. Log in as a **spacesadmin** user.

Figure 97: Connector GUI



Note
The root user is disabled and is used only for advanced troubleshooting by the Cisco Support team.

What to do next

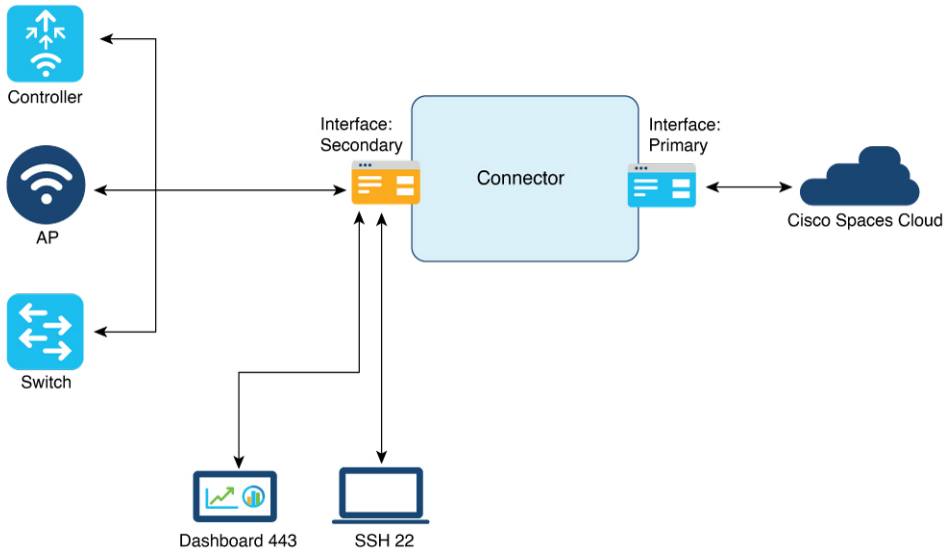
You can now Configure this Connector on Cisco Spaces.

Deploying the Cisco Spaces: Connector OVA (Dual Interface)

If you need to connect the connector to two separate customer networks in network deployments, you can use a dual-interface deployment. We recommend this deployment in scenarios where you manage devices on private or internal networks. To set up this deployment, you must use two interfaces:

- PRIMARY interface: Used to transmit traffic to Cisco Spaces.
- SECONDARY interface: Used by connector to interact with devices such as wireless controller, access points, or switches, over a private or internal network. You can also allow SSH and GUI (443) access to connector on this interface with additional configurations (disabled by default). Ensure that the connector is part of subnet routes to access it.

Figure 98: Dual Interface Deployment



Note We recommend that you connect the wireless controller to a private network as it enables the connector to establish SSH connections with the wireless controller.

Before you begin

Ensure that the Cisco Unified Computing System (Cisco UCS) device where you install the Open Virtualization Appliance (OVA) is connected to two separate networks. In this network configuration, the Cisco UCS device is configured with two physical network interface cards (NICs). Each NIC is connected to a switch. In this way, the Cisco UCS device is connected to two networks.

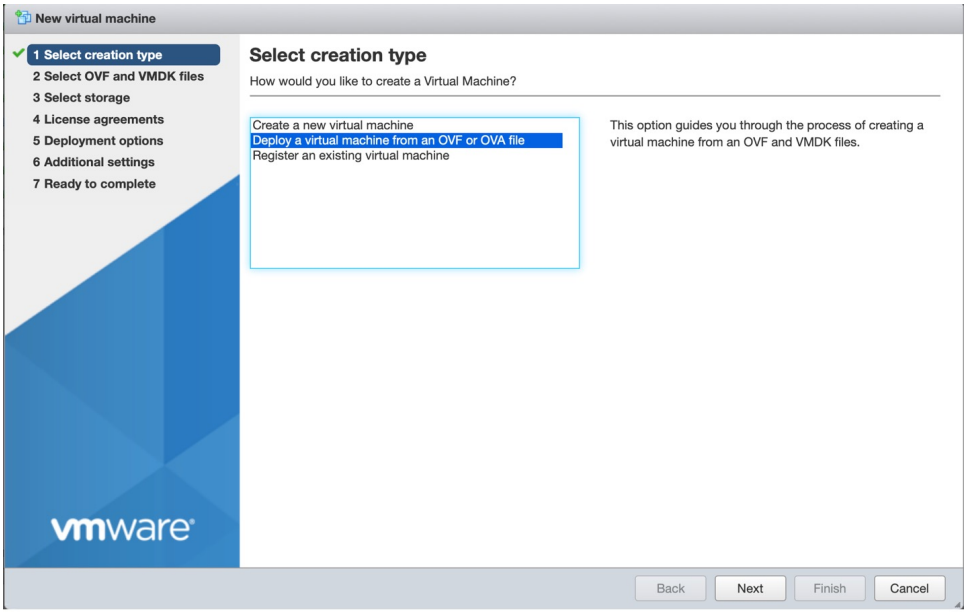
Procedure

Step 1 Download connector 3 from Cisco.com.

Step 2 Create a virtual machine in the ESXi server and deploy the downloaded Cisco Spaces: Connector OVA.

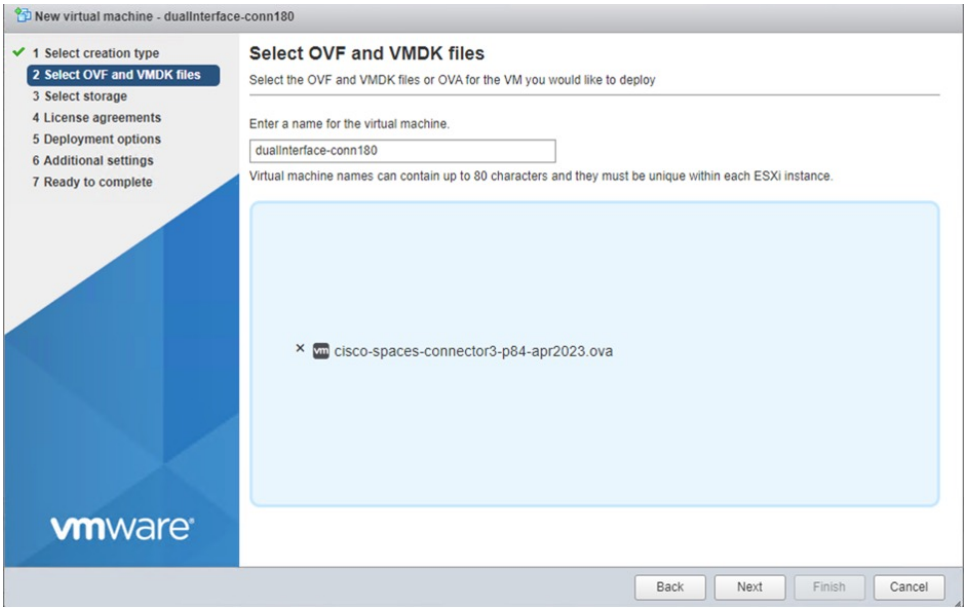
Step 3 In the **Select creation type** window, choose **Deploy a virtual machine from an OVF or OVA file**, and click **Next**.

Figure 99: Select Creation Type



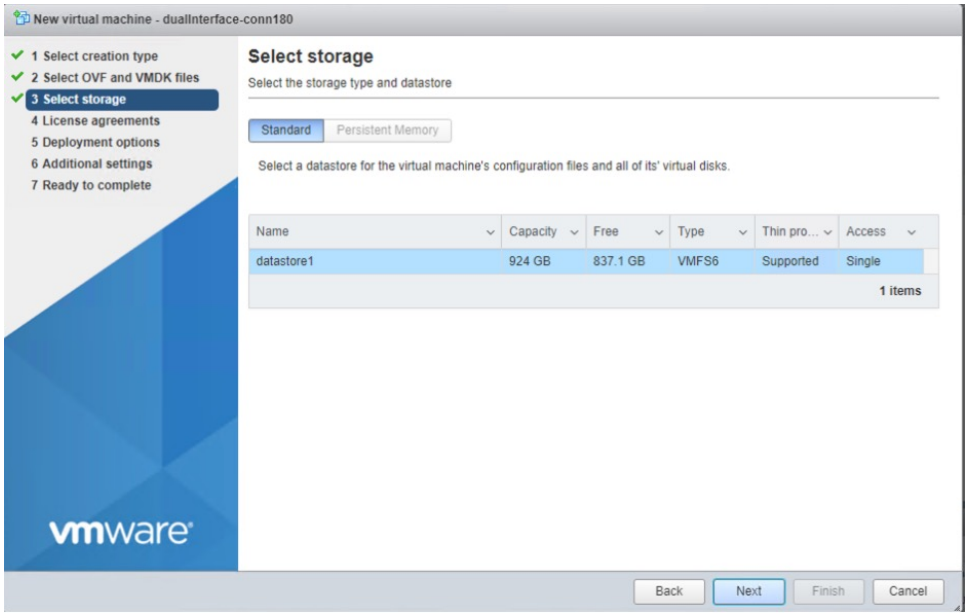
Step 4 In the **Select OVF and VMDK files** window, enter a name for the virtual machine. Click the blue area to either select files from the computer or drag and drop files. Click **Next**.

Figure 100: Select OVF and VMDK files



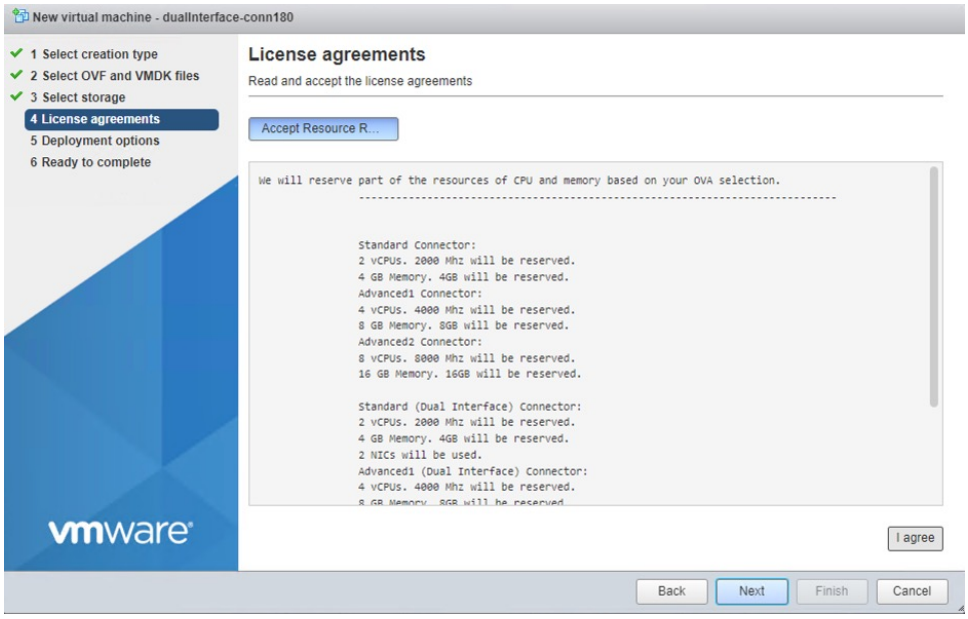
Step 5 In the **Select storage** window, the **Standard** storage configuration is displayed. Click **Next**.

Figure 101: Select Storage



Step 6 In the **License agreements** window, read the license agreement that is displayed and scroll to the end. Click **I Agree** and then click **Next**.

Figure 102: License agreements

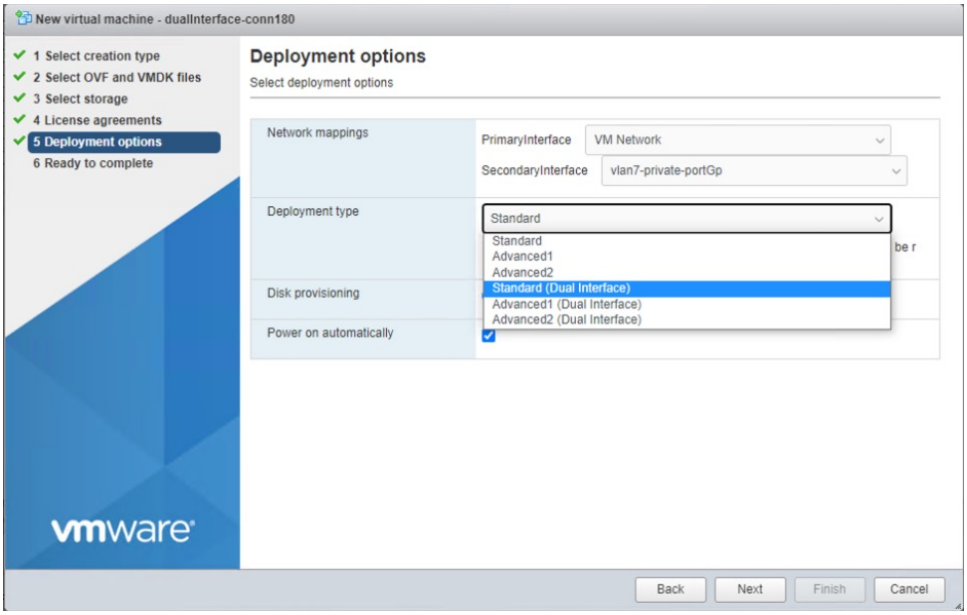


Step 7 In the **Deployment options** window, do the following:

a) In the **PrimaryInterface** field, enter the name of the external-facing interface.
b) In the **SecondaryInterface** field, enter the name of the private-facing interface.
c) From the **Deployment type** drop-down list, choose one of the following deployment types.

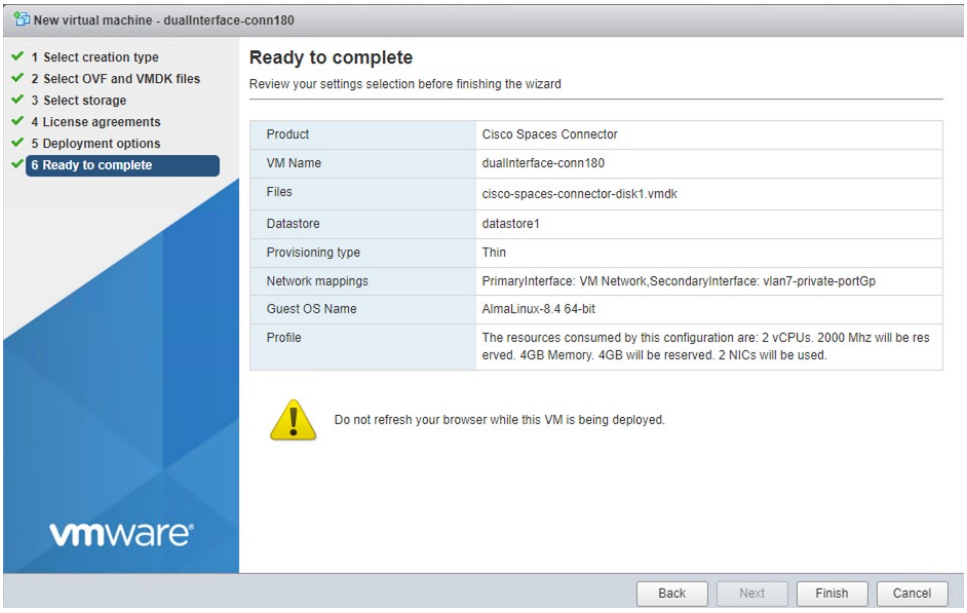
- **Standard (Dual Interface)**
- **Advanced1 (Dual Interface)**
- **Advanced2 (Dual Interface)**

Figure 103: Deployment options



Step 8 Review the configurations and click **Finish**.

Figure 104: Ready to complete



Step 9 Log in to the terminal and enter the default username **root** and default password **root**.

Step 10 Configure the host name for the connector.

Step 11 Choose an network interface to configure as PRIMARY.

Figure 105: Configuring the Primary Interface: IPv4

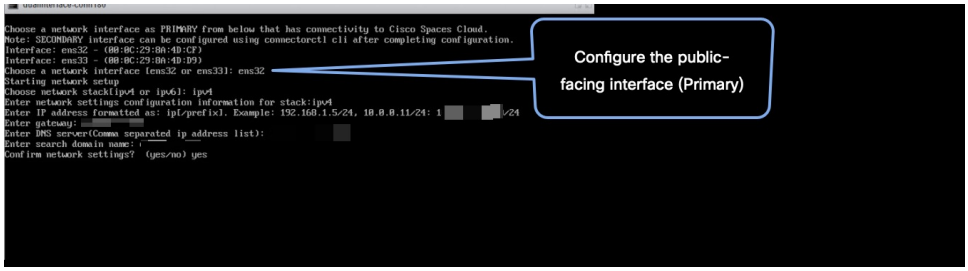
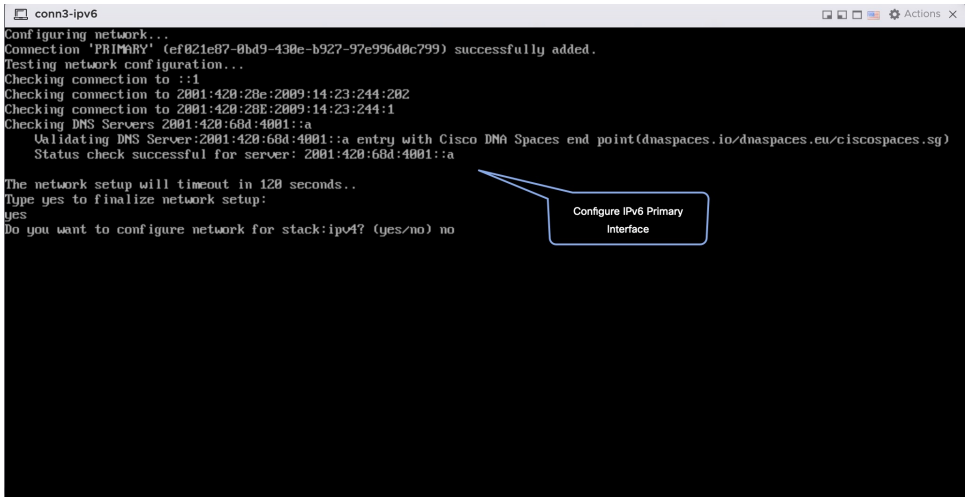


Figure 106: Configuring the Primary Interface: IPv6



Step 12 Do one of the following, and then configure the network settings for the PRIMARY interface. Specify parameters such as IP address, hostname, and so on.

- Configure the IPv6 stack.
- Configure the IPv4 stack.

You can add multiple DNS servers as a comma separated list in this step. After the task is complete and the Cisco Spaces: Connector is deployed, you can login to the connector CLI, and run the **connectorctl network config** command to add more DNS servers or edit the existing list.

Step 13 Reset the password for the **spacesadmin** user.

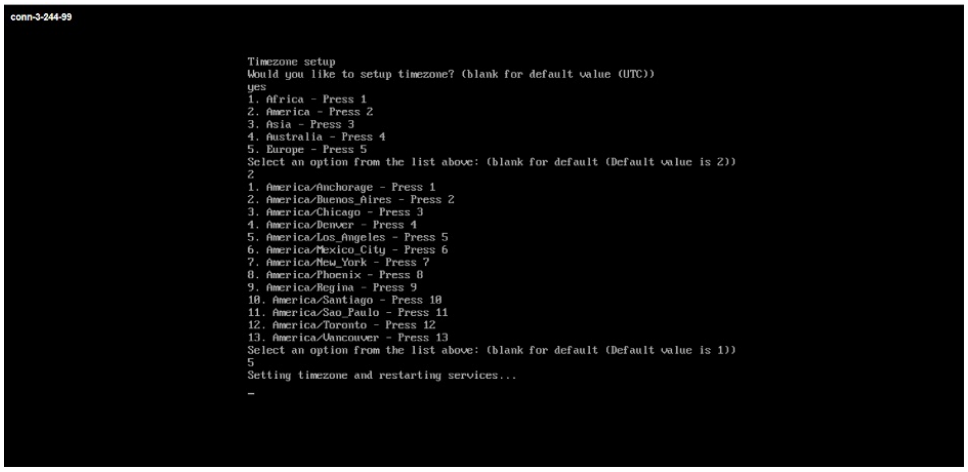
Note
The **spacesadmin** password can only include special characters such as '@' or '!' and must not contain any spaces.

Step 14 Confirm the setup.

Note
Because this configuration window times out in 120 seconds, ensure that you provide the input on time to avoid reconfiguration.

Step 15 Enter the time zone.

Figure 107: Time Zone



Step 16 Enter the Network Time Protocol (NTP) server name to synchronize the system time with that of NTP server, or leave it blank if you do not want to configure an NTP server.

Figure 108: Configure NTP

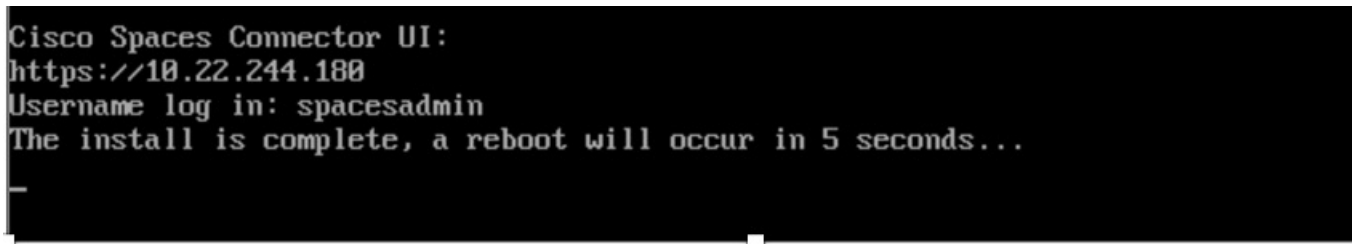


Figure 109: Configure NTP



Step 17 Note the URL (https://connector-ip) before the automatic reboot. You can use this URL later to open the connector GUI.

Figure 110: ConnectorGUI



Step 18 Wait for the completion of the reboot, and login as a **spacesadmin** user.

Step 19 Configure the secondary interface using the **connectorctl network config** command

```
[spacesadmin@connector ~]$ connectorctl network config  -p ipv4 -i 10.7.0.11/24 -g 10.7.0.1 -o
cisco.com -d 172.70.168.183 -n SECONDARY
Executing command:network
Command execution status:Success
------------------------
```

```
Connection SECONDARY (5e970417-13b4-4ad8-af12-d125ce407c49) successfully added.
Network setup completed with given configuration.
Secondary interface - Added routes.
Secondary interface - Configured firewall zone.
System reboot will happen in 10 seconds. Do not execute any other command...
```

Step 20 Verify the network Settings of external-facing network using the **connectorctl network show** command.

```
[spacesadmin@connector ~]$ connectorctl network show
 Executing command:network
Command execution status:Success
----------------------
=================Network Config=================
Hostname    - connector-p84-april1


Interface   - PRIMARY
---------------------------------

Network configuration for stack:ipv4
Ip Address  - 10.22.244.180/24
Mac Address - 00:0C:29:EE:24:8A
Gateway     - 10.22.244.1
Dns         - 172.70.168.183
Domain      - cisco.com


Interface   - SECONDARY
---------------------------------

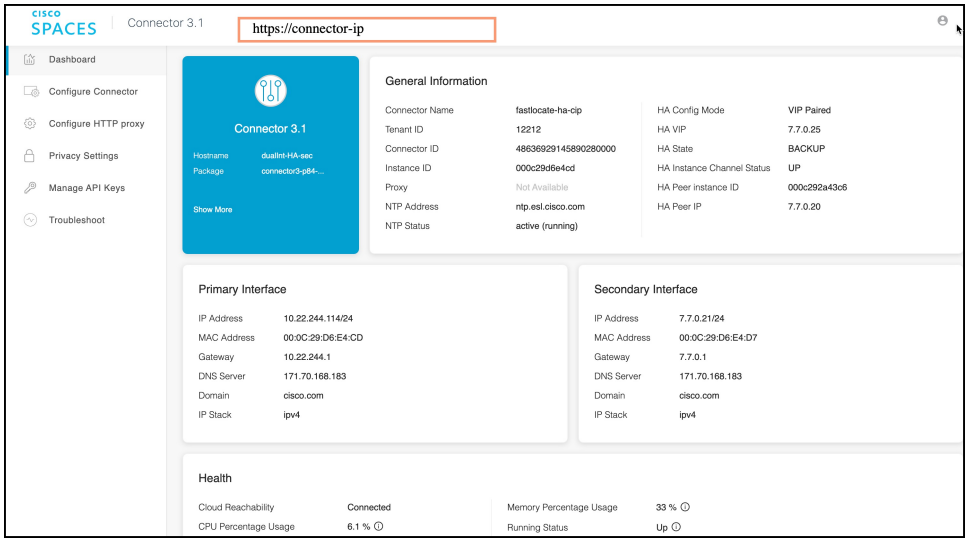
Network configuration for stack:ipv4
Ip Address  - 7.7.0.11/24
Mac Address - 00:0C:29:EE:24:94
Gateway     - 7.7.0.1
Dns         - 172.70.168.183
Domain      - cisco.com

=================end================
```

You can use the **connectorctl network show -n PRIMARY** and **connectorctl network show -n SECONDARY** to see information specific to these interfaces.

Step 21 In a browser window, navigate to the noted URL to open the connector GUI. Log in as a **spacesadmin** user.

Figure 111: ConnectorGUI



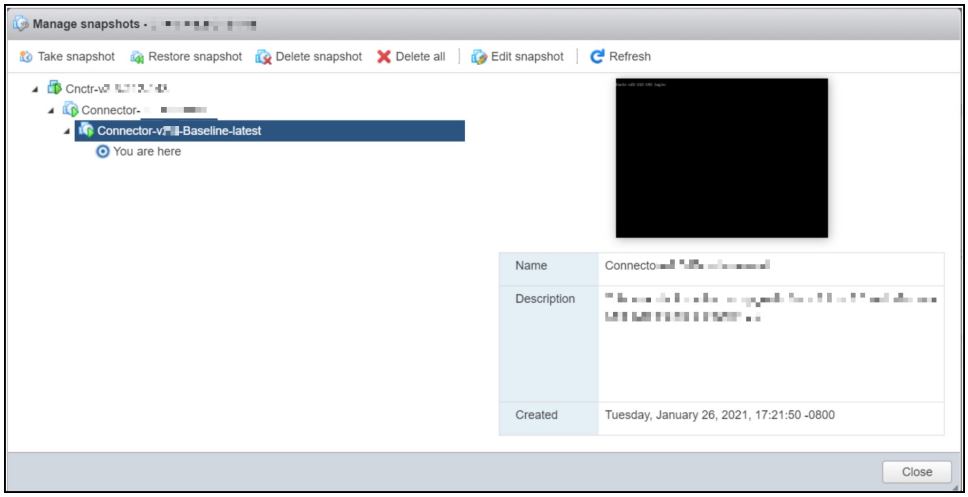
Note
The root user is disabled and is used only for advanced troubleshooting by the Cisco Support team.

Using Snapshots for Backup

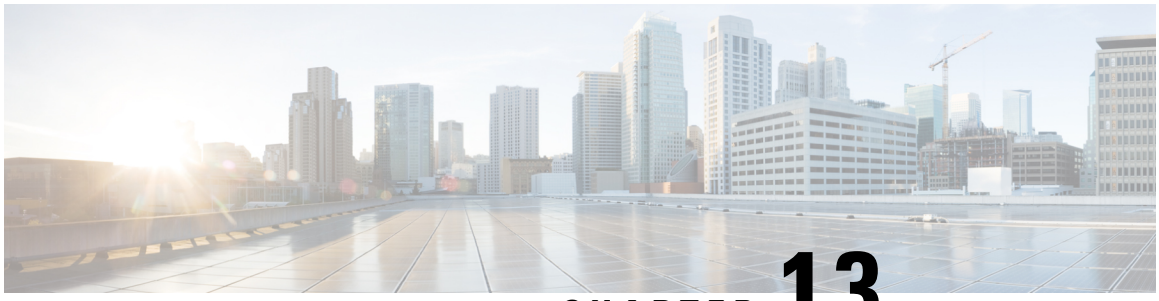
You can use the snapshot of a deployed connector OVA for backing up your connector. Ensure that the following prerequisites in place:

- connector is deployed.
- All the services are started.
- connector is added to Cisco Spaces.

Figure 112: Backing Up Using a Snapshot



Note Proxies are not carried over during a snapshot restore. You have to reconfigure proxies.



CHAPTER 13

Cisco Spaces: Connector VM on Nutanix Environment

- Cisco Spaces: Connector Nutanix cloud platform support, on page 101
- Set Up Connector VM on Nutanix Environment, on page 101

Cisco Spaces: Connector Nutanix cloud platform support

The Cisco Spaces: Connector 3.2 June, 2025 release supports the Nutanix cloud platform. To deploy the Nutanix Connector virtual machine (VM), you need the necessary Nutanix setup, which includes Nutanix-supported hardware and PRISM software.

Set Up Connector VM on Nutanix Environment

Follow the steps to setup a connector VM.

Before you begin

Ensure that Nutanix Prism is setup before you configure the connector virtual machine (VM).

Procedure

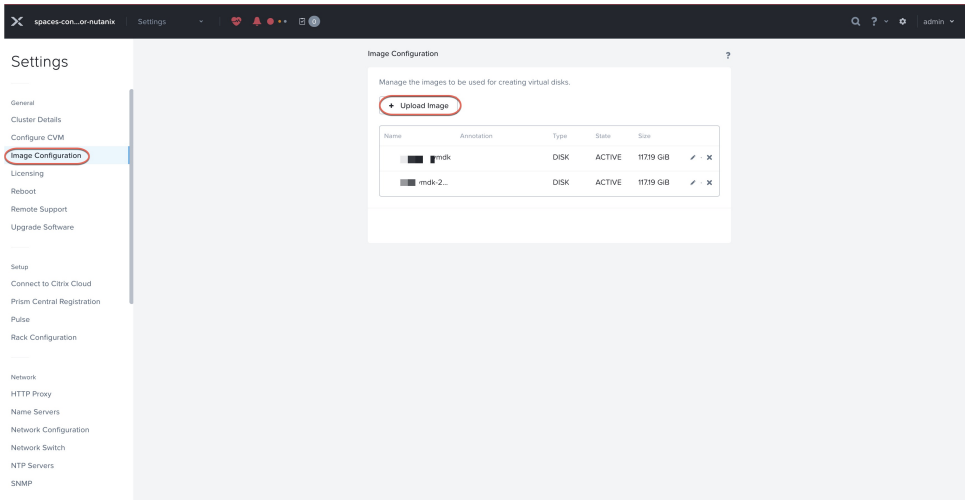
Step 1 Download the June 2025 OVA file available on the Software Download page.

Step 2 Unzip the downloaded OVA to extract the Virtual Machine Disk (VMDK) file.

Step 3 Log in into **Nutanix Prism** > **Settings** > **Image Configuration** tab.

The **Image Configuration** dialog box is displayed.

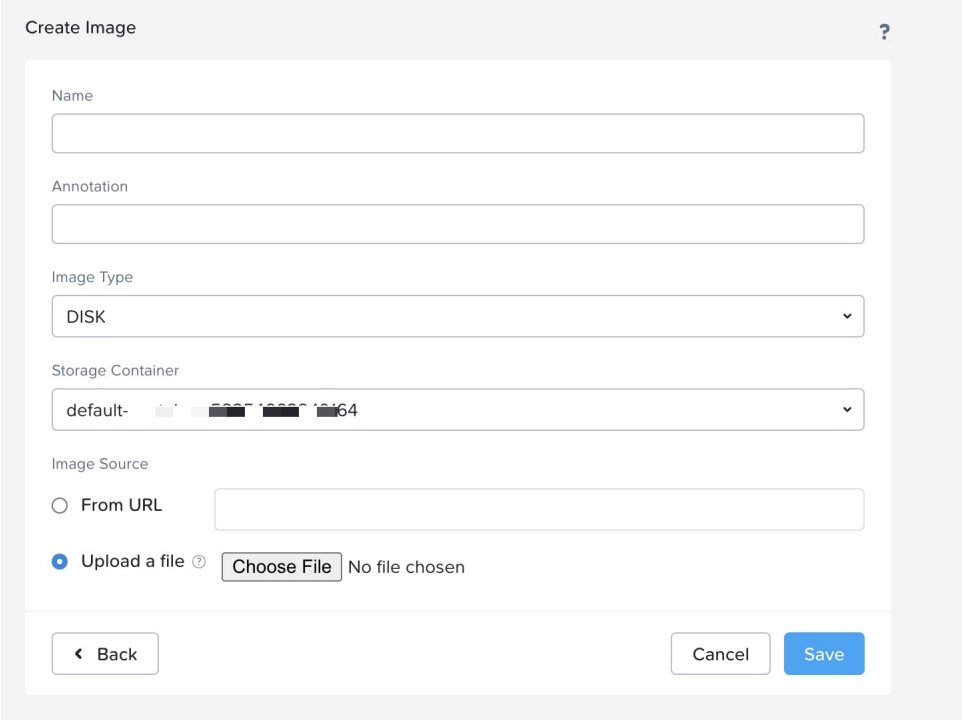
Figure 113: Image Configuration



Step 4 Click **Upload Image**.

The **Create Image** dialog box is displayed.

Figure 114: Create Image



Step 5 In the **Create Image** dialog box, enter the details.

a) **Name**: Enter name for the image.

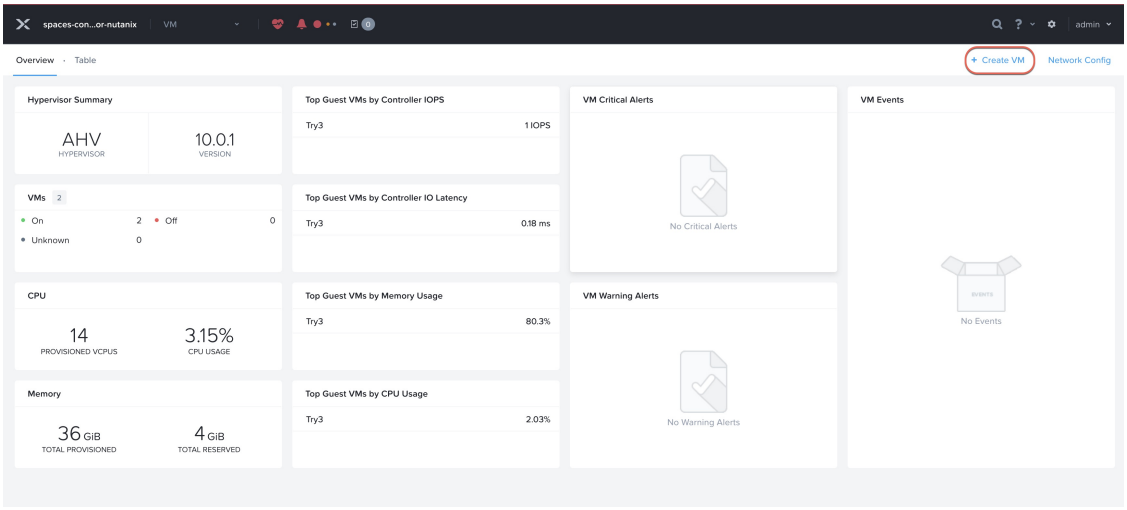
b) **Annotation**: Enter annotation for the image.

c) **Image Type**: From the drop-down list, select image type as **DISK**.

d) **Storage Container**: From the drop-down list, select the preferred storage container.
e) **Image Source**: Select **Upload a file** and click **Choose File** to add the extracted VMDK File.
f) Click **Save**.

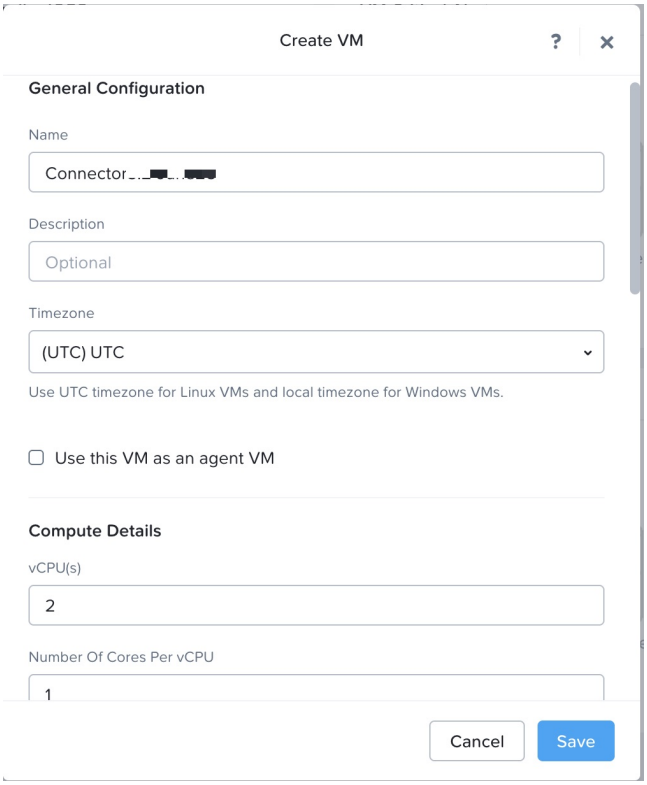
Step 6 Navigate to the **VM** > **Overview** tab and click **Create VM**.

Figure 115: VM Overview



The **Create VM** dialog box is displayed.

Figure 116: Create VM



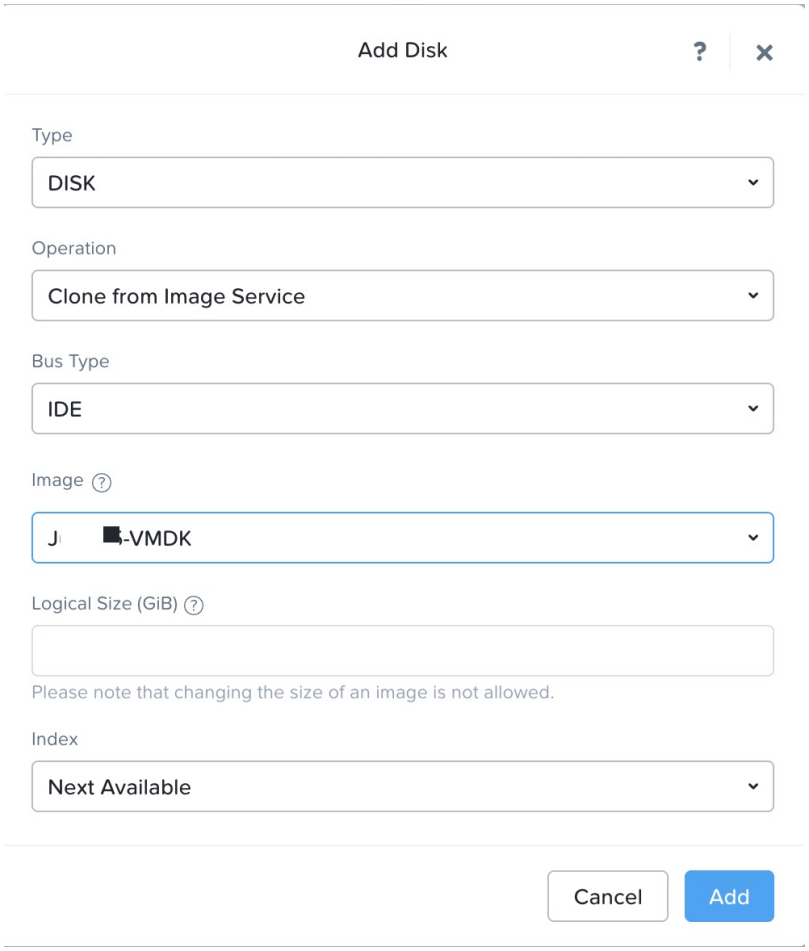
a) In the **Create VM** dialog box, under **General Configuration** section, enter the details.

- **Name**: Enter the name for the new **VM**.
- (Optional) **Description**: Enter the description for the new VM.
- **Timezone**: From the drop-down list, select the timezone.
- (Optional) **Use this VM as an agent VM**: Check this option to use the VM as an agent.
- **Compute Details**: Enter the **vCPU(s)**, **No of Cores per vCPU** details and **Memory**.
- **Boot Configuration Mode**: From the drop-down list, select **UEFI** as the boot configuration mode.

b) Click **Add Disk**.

The **Add Disk** dialog box is displayed.

Figure 117: Add Disk



c) In the **Add Disk** dialog box, enter the details.

- **Type**: From the drop-down list, select **DISK** as type.
- **Operation**: From the drop-down list, select **Clone from Image Service** as operation.

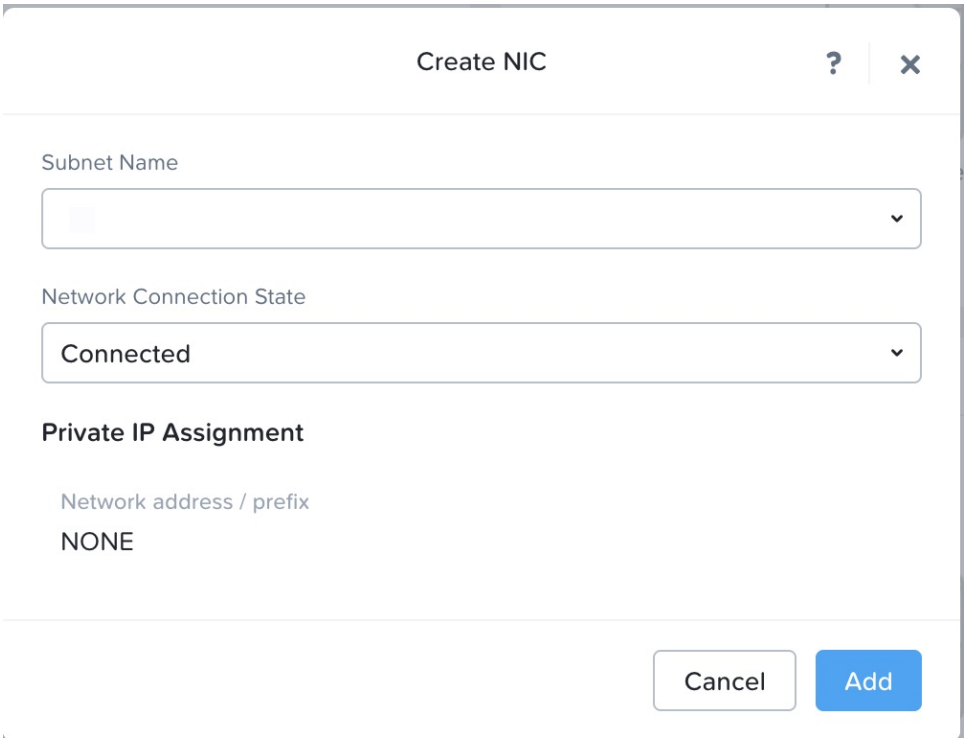
- **Bus Type**: From the drop-down list, select, **IDE** as the bus type.
- **Image**: From the drop-down list, select the **VMDK** image uploaded previously.
- **Logical Size (GiB)**: Displays the **VMDK** file size. You cannot edit the image logical size value.
- **Index**: From the drop-down list, select **Next Available** as index option.

d) Click **Add**.

Step 7 Click **Add New NIC**.

The **Create NIC** dialog box is displayed.

Figure 118: Create NIC



a) In the **Create NIC** dialog box, enter the details.

- **Subnet Name**: From the drop-down list, select the appropriate subnet.
- **Network Connection State**: From the drop-down list, select **Connected** as the network connection state.
- **Private IP Assignment**: Enter **Network address / prefix** as required.

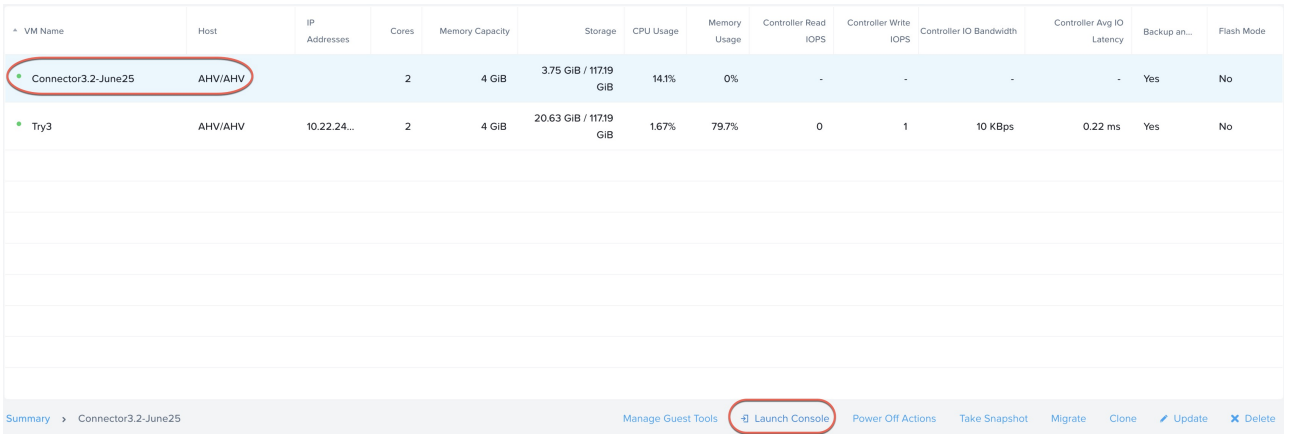
b) Click **Add**.

Step 8 Click **Save** to save and store the new VM configuration.

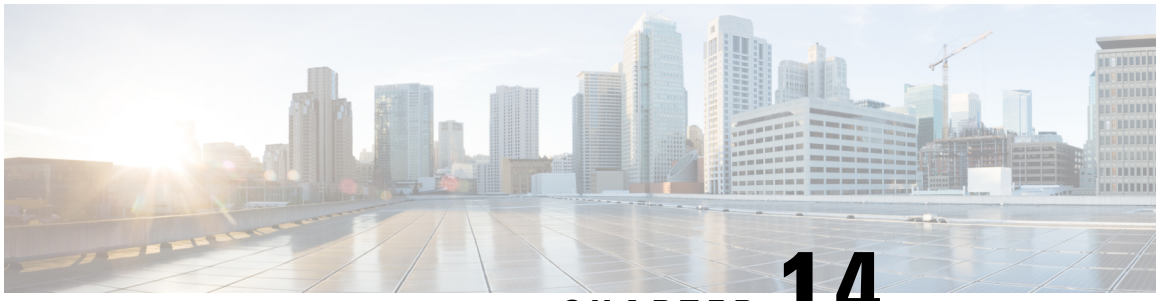
What to do next

To power on the new VM, navigate to **VM** > **Table tab** > **Power on the created VM**. After the VM is powered on, select the VM and click **Launch Console**.

Figure 119: Launch Console



Note For more information, see step 12 onwards in the Deploying the Connector 3 OVA (Single Interface).



CHAPTER 14

Cisco Spaces: Connector VM on Azure Environment

- Cisco Spaces: Connector Azure cloud platform support, on page 107
- Set Up Connector VM on Azure Environment, on page 107

Cisco Spaces: Connector Azure cloud platform support

The Cisco Spaces: Connector 3.2 June, 2025 release supports the Azure cloud platform. To deploy the Azure Connector virtual machine (VM), you need the necessary access to Azure cloud platform.

Set Up Connector VM on Azure Environment

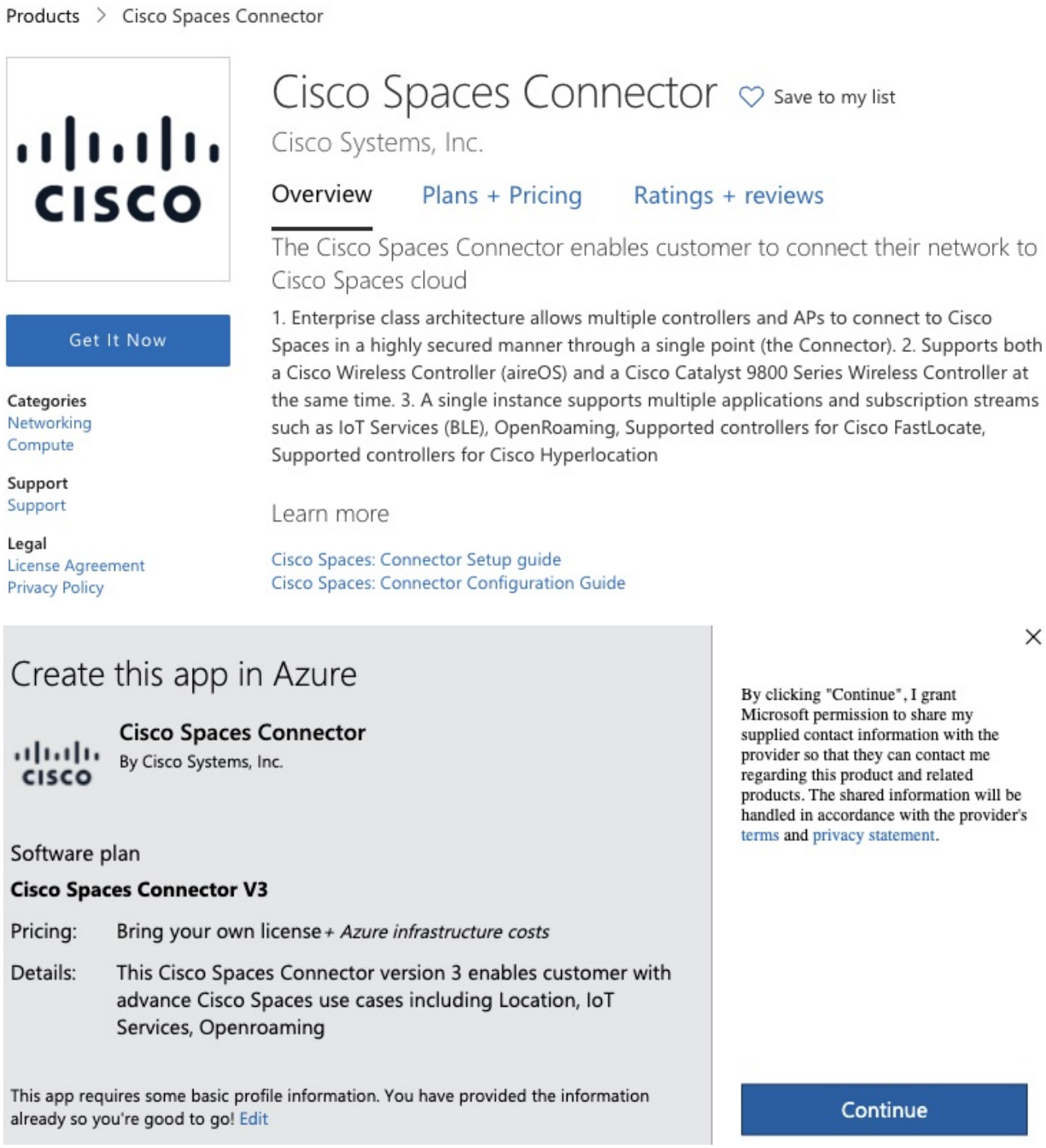
Follow the steps to setup a connector VM.

Procedure

Step 1 Log in to your Azure portal.

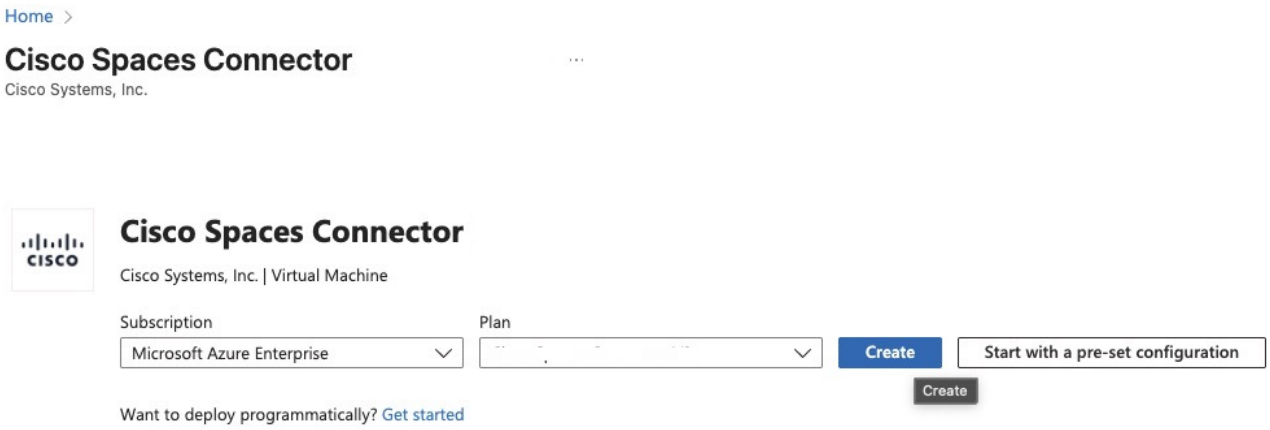
Step 2 In the Azure Marketplace, search for the latest connector release and click **Get It Now** > **Continue**.

Figure 120: Cisco Spaces Connector



The Cisco Spaces: Connector preview page is displayed.

Figure 121: Cisco Spaces: Connector

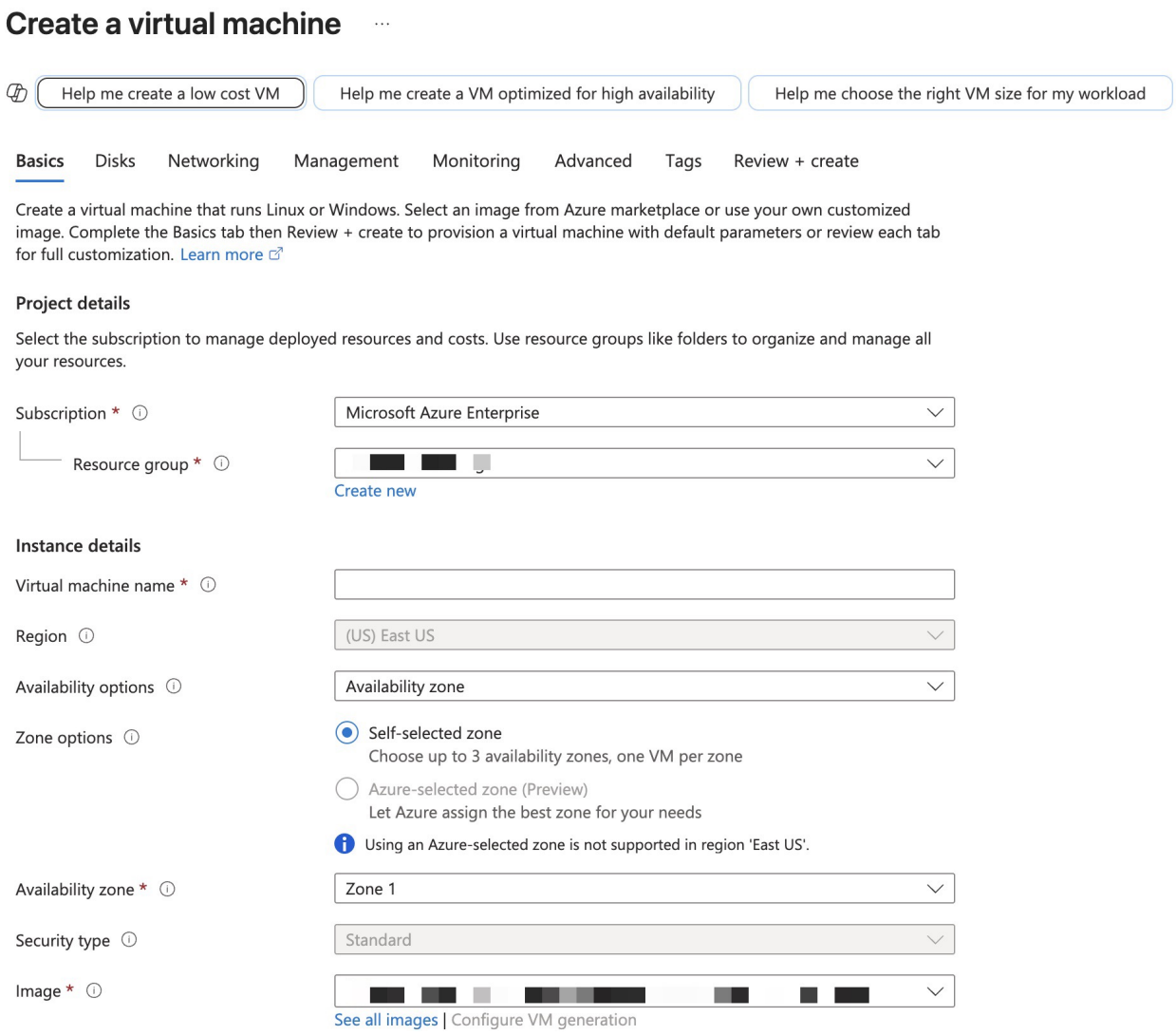


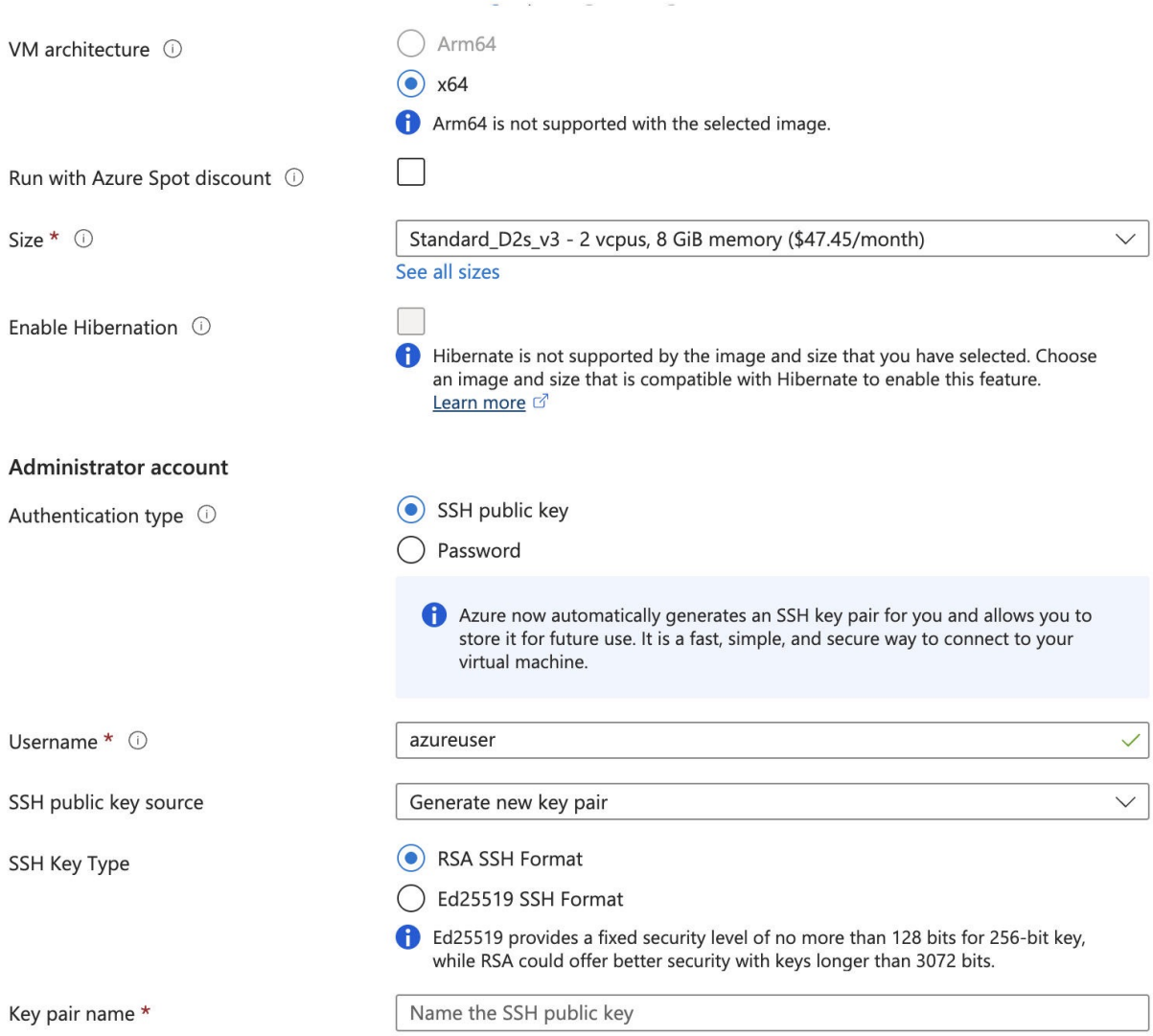
Step 3 In the Cisco Spaces: Connector preview page, perform these steps:

a) From the **Subscription** drop-down list, select **Microsoft Azure Enterprise** as the type.
b) From the **Plan** drop-down list, select **Cisco Spaces Connector V3** as your subscription plan.
c) Click **Create**.

Step 4 In the **Create a virtual machine** window, under the **Basics** tab, perform these steps:

Figure 122: Create a virtual machine Basics tab





a) From the **Resource group** drop-down list, under the **Subscription** section within the **Project details** area, select your resource group.

   Create a new one if required.

b) In the **Virtual machine name** field, under the section **Instance details**, enter a name for the connector VM.

c) From the **Size** drop-down list, select the recommended size. These are the recommended sizes:

   - Standard_D2ds_v4 - 2 vcpus, 8 GiB memory.
   - Standard_D4ds_v4 - 4 vcpus, 16 GiB memory.
   - Standard_D8ds_v4 - 8 vcpus, 32 GiB memory.

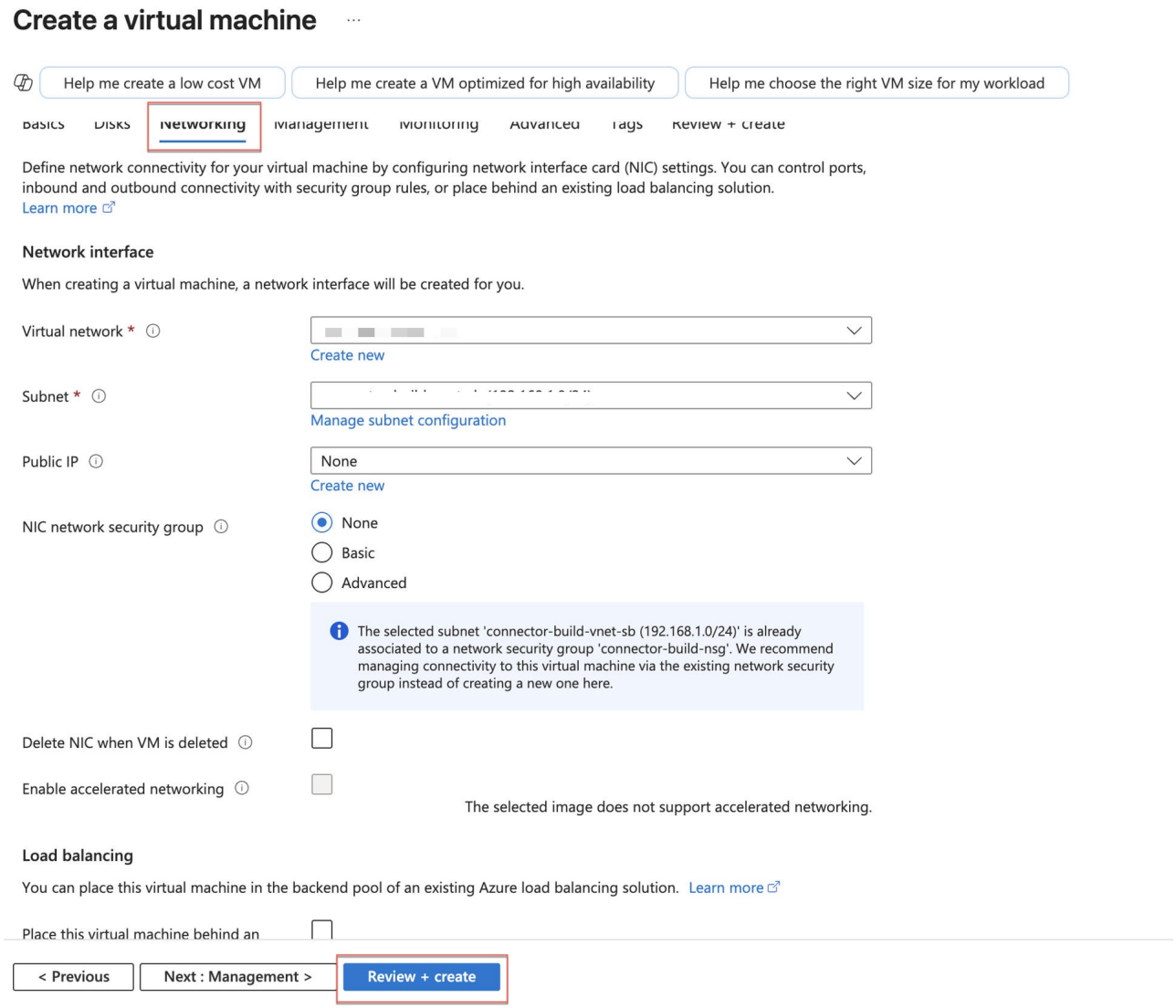
d) From the **Authentication type**, under the section **Administrator account**, select the **SSH public key** radio button.

Note

The **Username** is **azureuser** by default. The **Key pair name** is populated by itself. You can keep the same or replace with your own.

Step 5 In the **Create a virtual machine** window, perform these steps under the **Networking** tab:

Figure 123: Create a virtual machine Networking tab



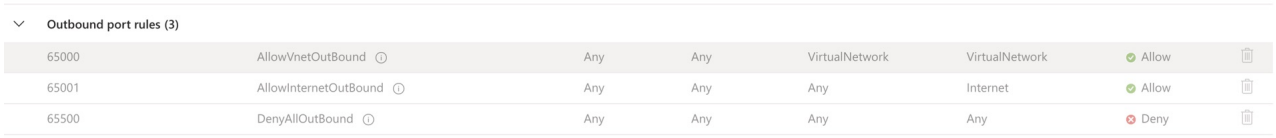
a) From the **Virtual network** drop-down list, under the section **Network interface**, select your network group.
b) From the **Subnet** drop-down list, select your specific subnet.
c) Configure and allow the required inbound rules.

Figure 124: Configure required inboud rules



d) Configure and allow the required outbound rules.

Figure 125: Configure required outboud rules



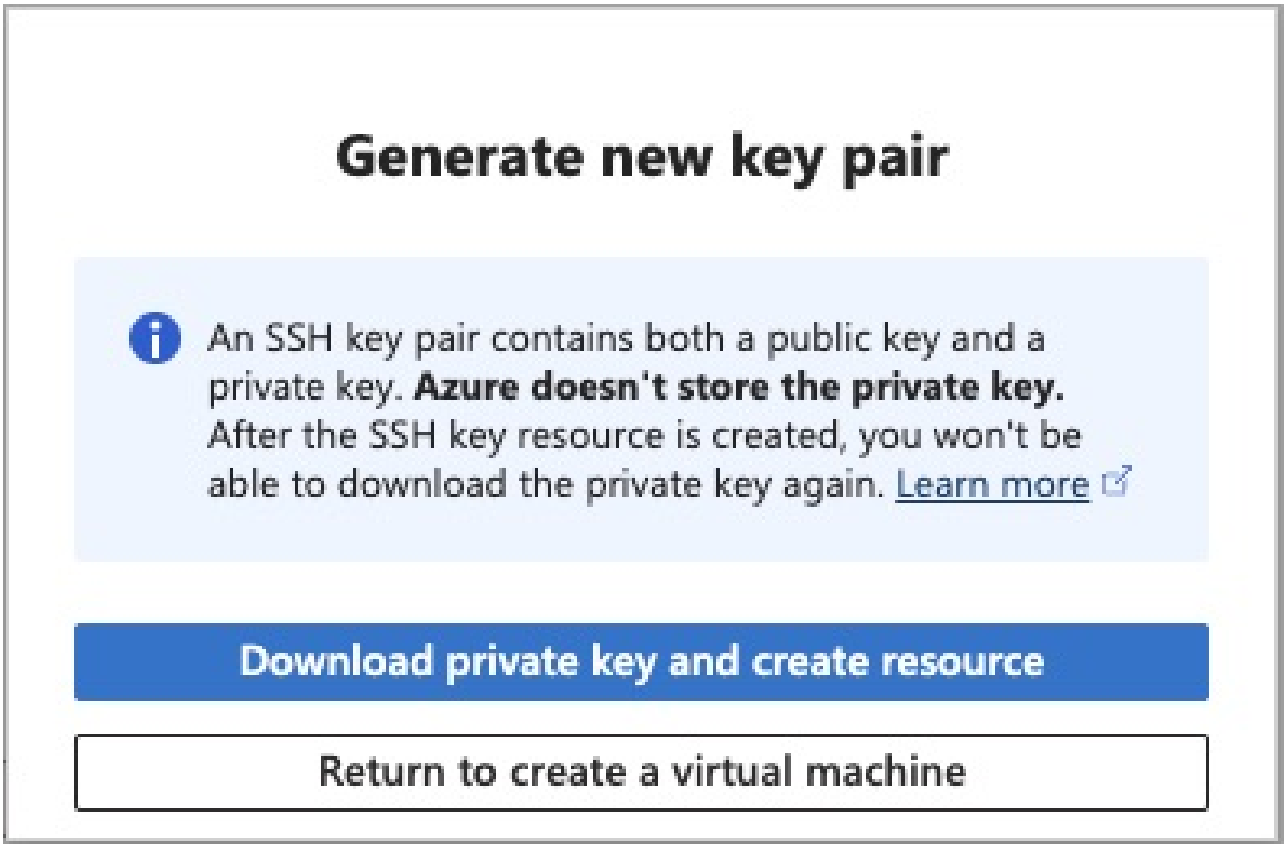
e) Click **Review + create**.

Step 6 To change the permission of the downloaded key, run the command:

Example:

```
chmod 600 <keyname
```

Figure 126: Generate new key pair



Note
We recommend you to download this key pair to your system, navigate to the file location and configure appropriate permissions using the **chmod** command.

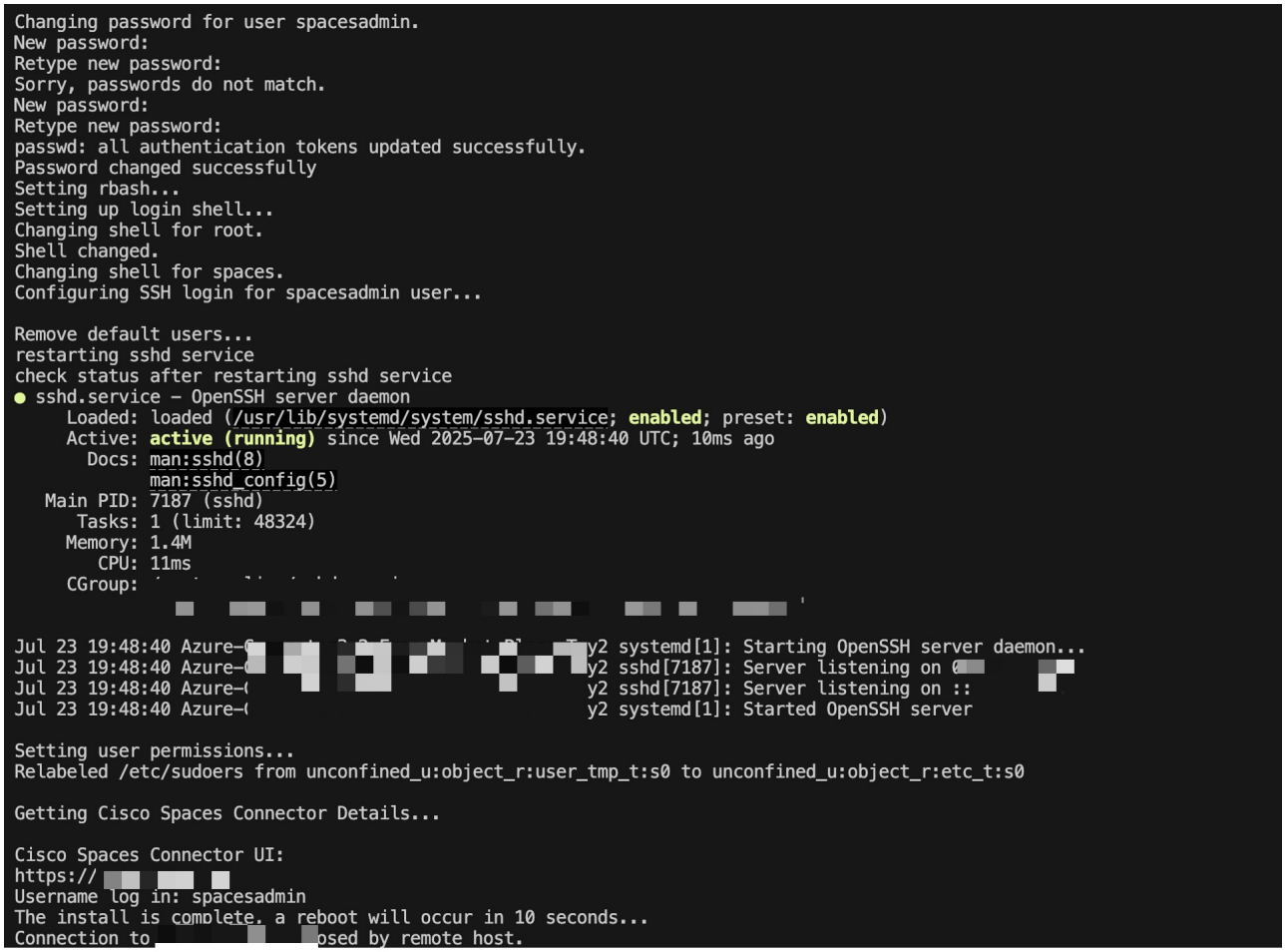
Step 7 SSH to the connector as **azureuser** using the public key.

Example:

```
ssh -i <public_key> azureuser@<ip_address>
```

Step 8 Enter the password for the Cisco Spaces administrator (spacesadmin).

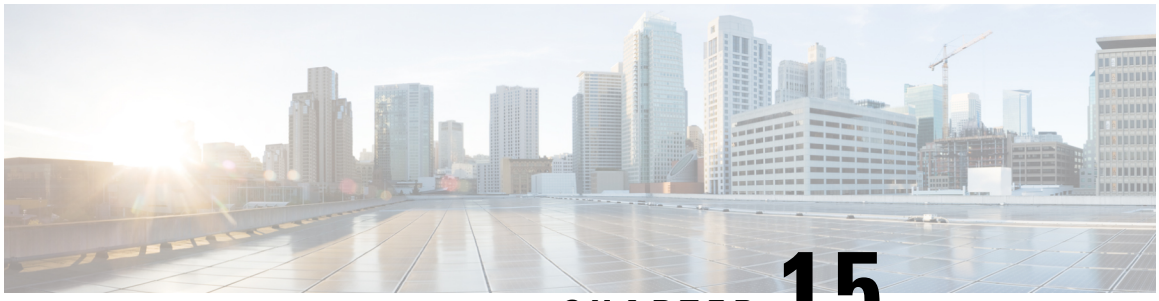
Figure 127: Cisco Spaces administrator



The Cisco Spaces: Connector reboots and the connector UI is displayed.

What to do next

For more information on how to generate a token for your deployed azure connector, follow the steps here: Connector on Cisco Spaces.



CHAPTER 15

Cisco Spaces: Connector VM on OpenShift Platform

- Connector VM on OpenShift Platform, on page 115
- Deploy a Connector VM on the OpenShift Platform, on page 115

Connector VM on OpenShift Platform

A Cisco Spaces: Connector is deployed as a virtual machine to establish communication between the on-premises network infrastructure and Cisco Spaces. On OpenShift, the Connector VM is created from a QCOW2 image uploaded as a bootable volume. The VM must have sufficient storage, an appropriate instance type, and the required network interface to communicate with the required network services and Cisco Spaces.

This task describes how to upload the QCOW2 image, provision the Connector VM on OpenShift, and complete the Day-0 setup.

Deploy a Connector VM on the OpenShift Platform

Use this procedure to create a bootable volume, provision a Connector VM, and complete the initial Connector configuration.

The Connector VM requires a bootable volume with a minimum disk size of 120 GB and an appropriate network interface.

Before you begin

Ensure that the OpenShift Platform is installed and operating properly.

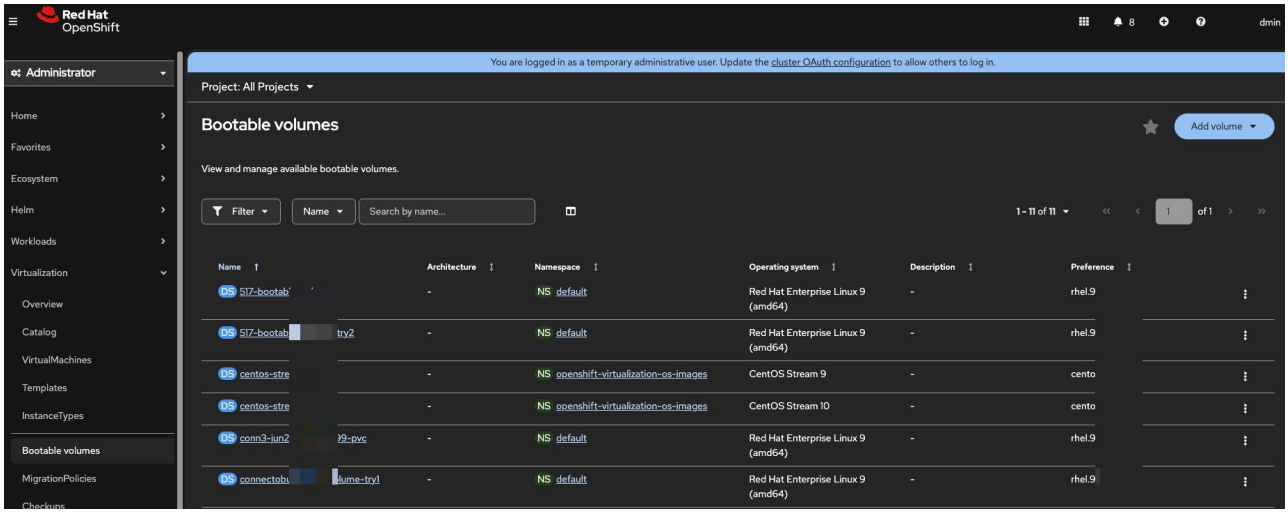
Obtain the QCOW2 image for the Connector VM.

To Bring up a Connector VM on OpenShift Platform follow the below steps:

Procedure

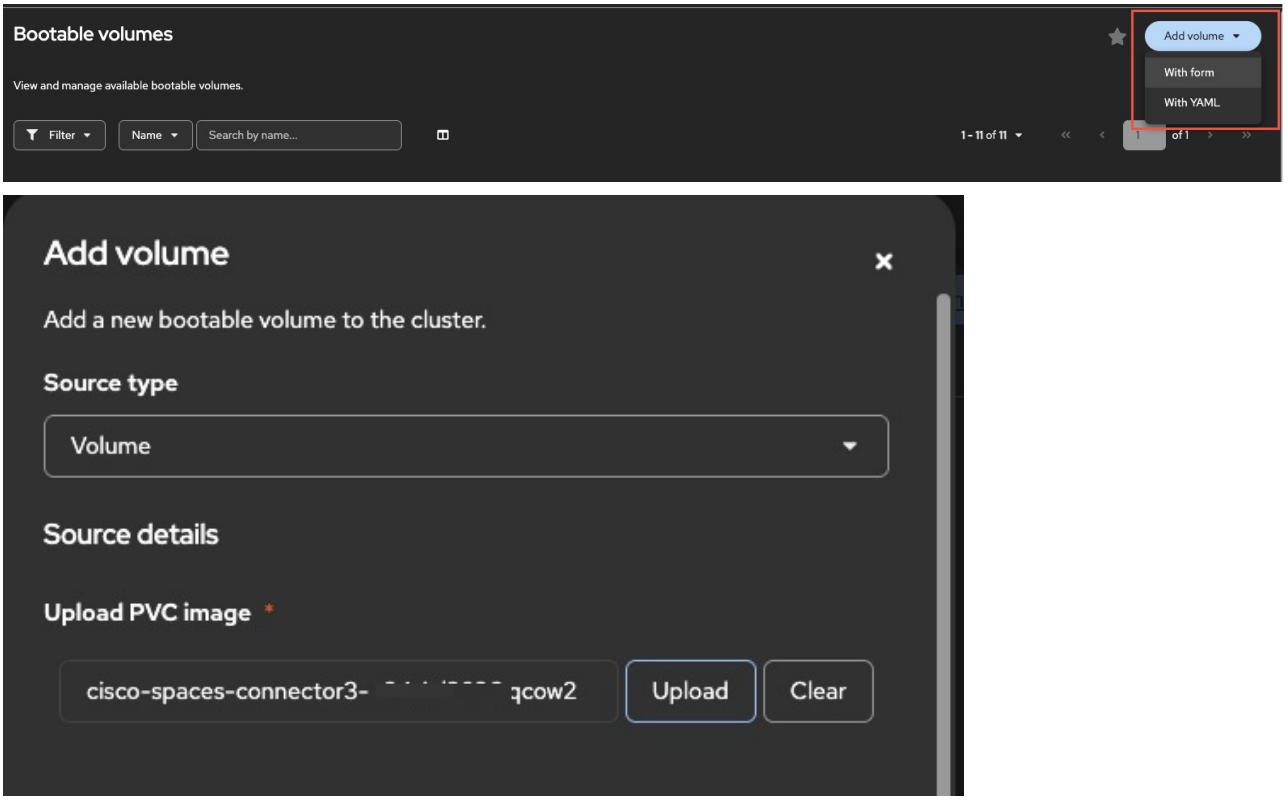
Step 1 In the OpenShift console navigation, choose **Virtualization** > **Bootable volumes**.

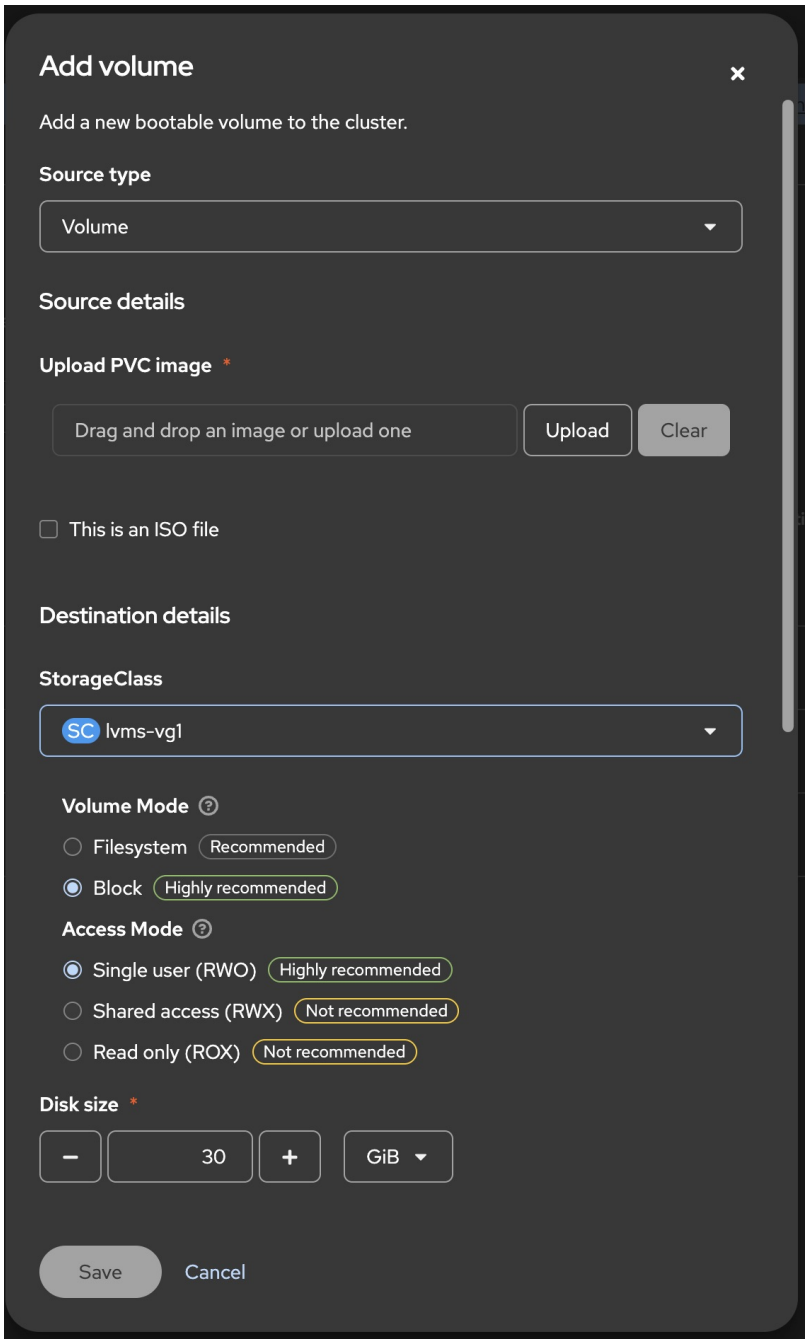
Figure 128: Bootable volumes



Step 2 Click **Add Volume** > **With form**, and then provide the required parameters.

Figure 129: Add volume





Step 3 Configure the bootable volume.

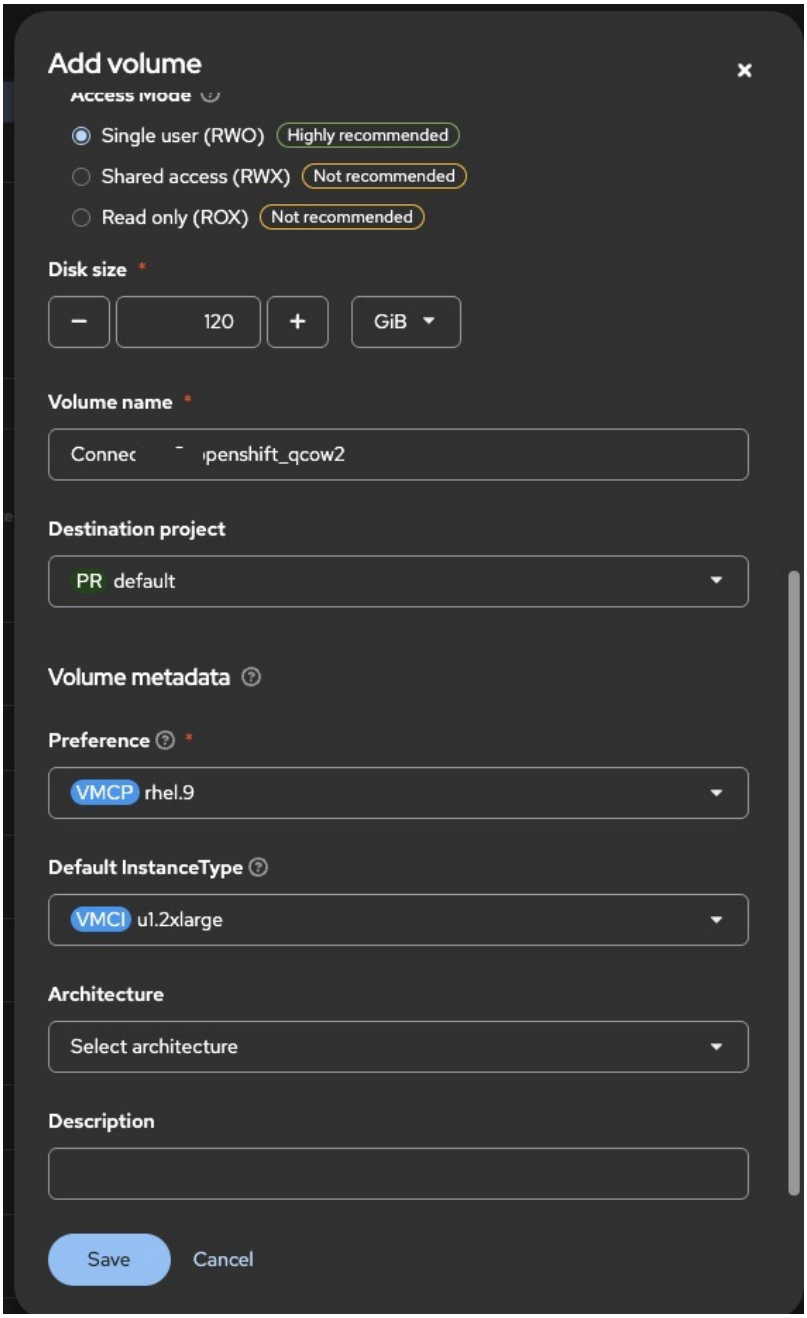
a. Upload the Connector QCOW2 image.

b. Set the disk size to 120 GB.

c. Enter a name for the volume.

d. Select an appropriate Red Hat Enterprise Linux (RHEL) operating system as the preference type.

Figure 130: Add Volume

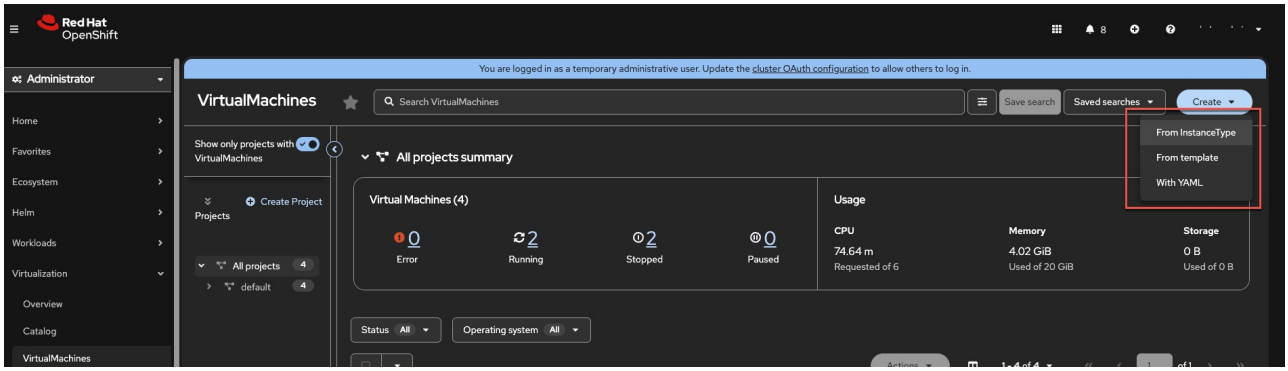


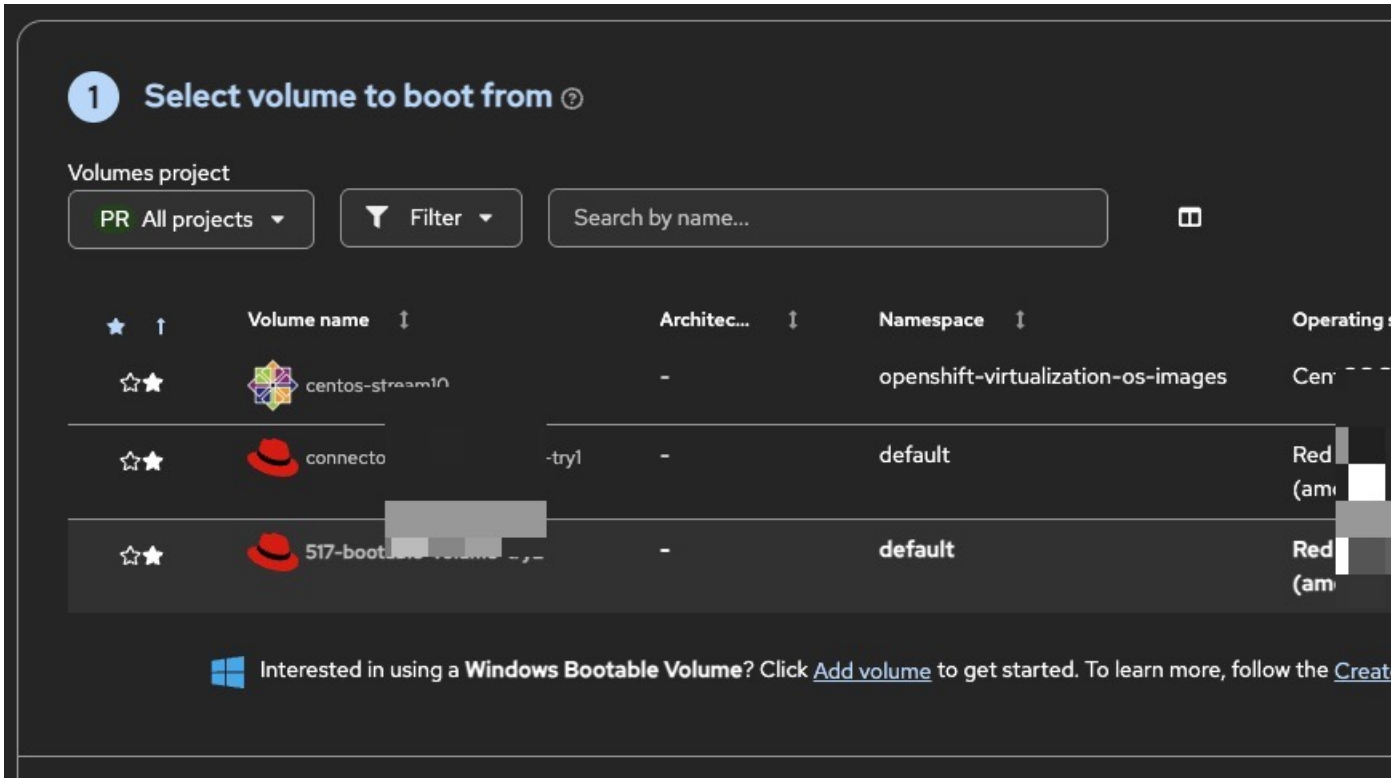
Step 4 Create the bootable volume.

The uploaded QCOW2 image is available as a bootable volume.

Step 5 In the console navigation, choose **Virtualization** > **Virtual Machines** > **Create from InstanceType**.

Figure 131: Create InstanceType

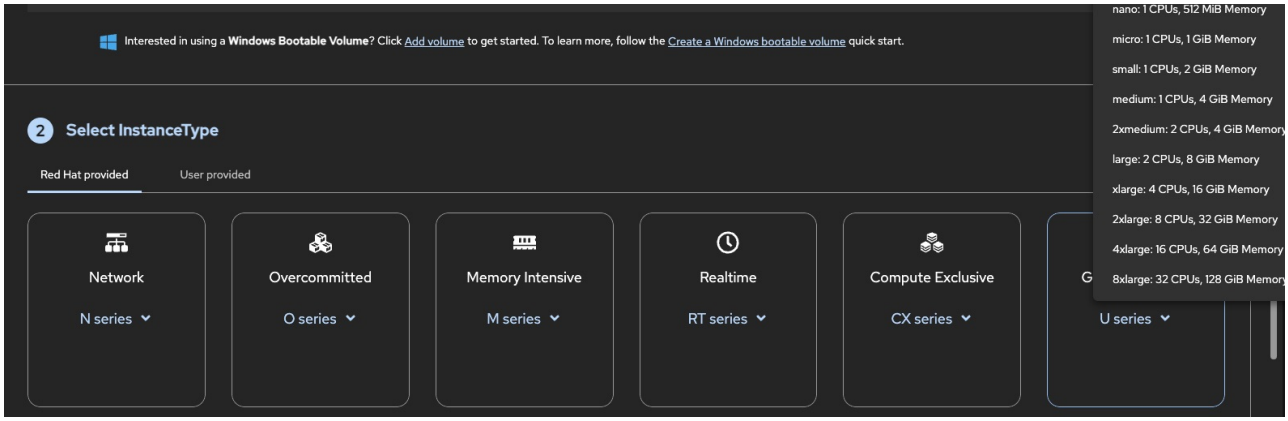




Step 6 Select an instance type based on the Connector model that you are deploying. The options are:

- Standard Connector: 2xMedium
- Advanced1 Connector: xlarge
- Advanced2 Connector: 2xlarge

Figure 132: Select InstanceType



Step 7 Select the bootable volume that you created in the previous steps.

Step 8 Edit the virtual machine details and enter a name for the Connector VM.

Step 9 Click **Customize VirtualMachine**.

Step 10 Click the **Network** tab and add the appropriate network interface card (NIC).

Step 11 Click **Create VirtualMachine**.

The Connector VM is provisioned on the OpenShift Platform.

Figure 133: Create VirtualMachine details

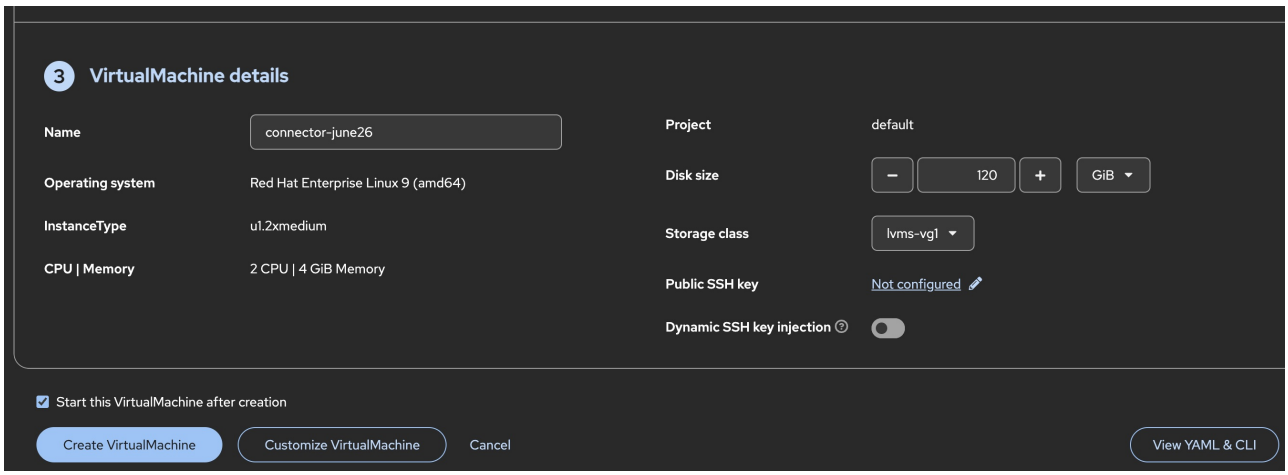
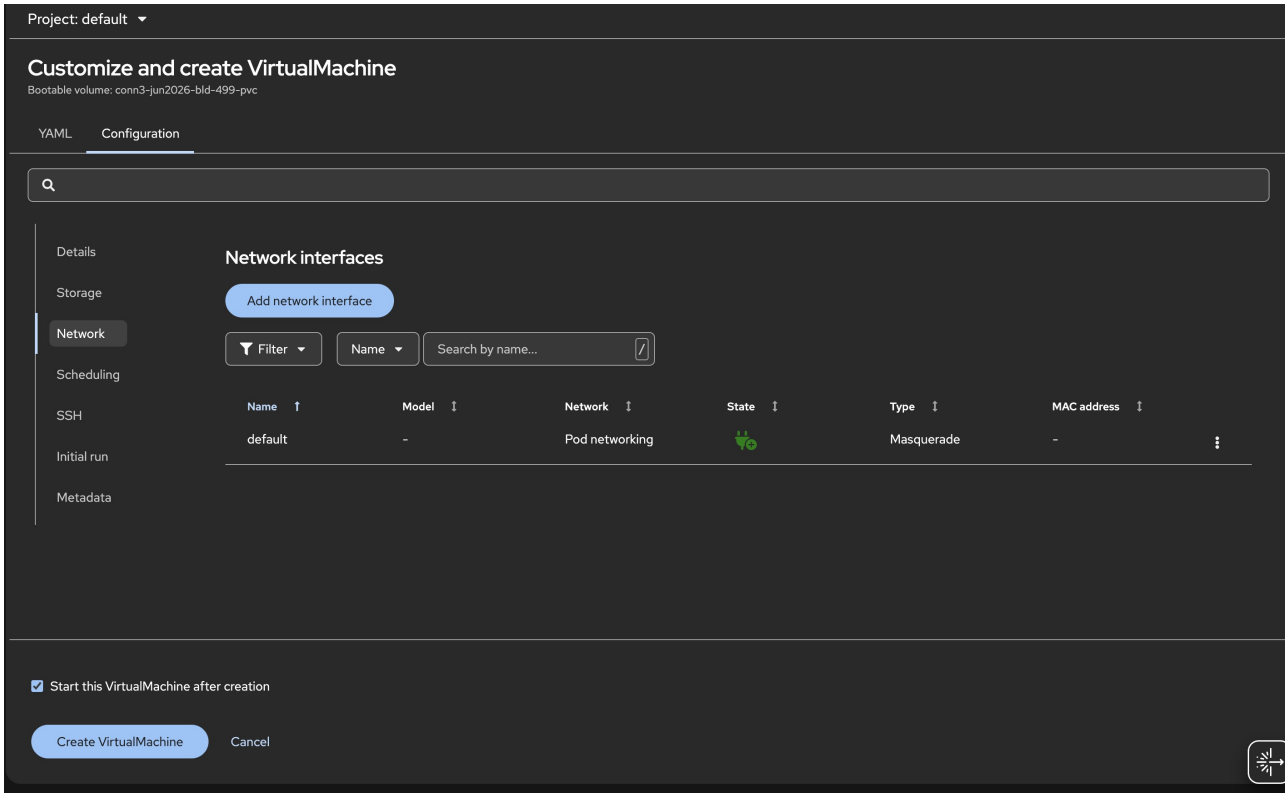


Figure 134: Customize and Create VirtualMachine

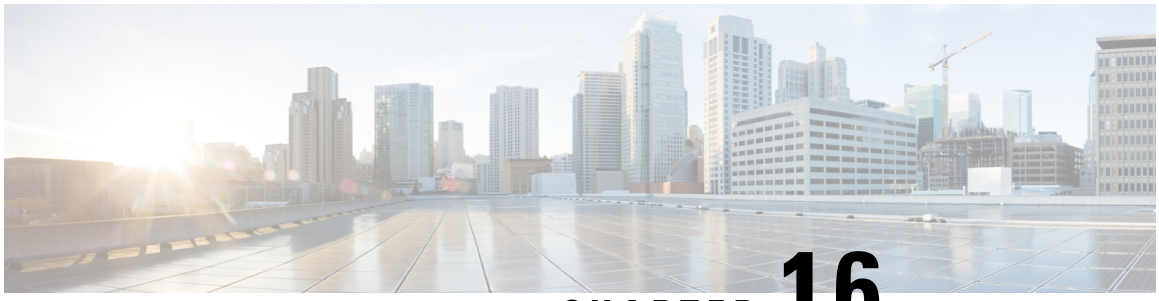


Step 12 Power on the Connector VM.

Step 13 Click the **Console** tab.

Step 14 Complete the Day-0 setup by following the Connector OVA configuration procedure in the Cisco Spaces: Connector 3 Configuration Guide.

The Connector VM is deployed on the OpenShift Platform and is ready for use after the Day-0 setup is complete.



CHAPTER 16

Connector on Cisco Spaces

- Activating Connector 3 on Cisco Spaces, on page 123
- Monitor the Status of Service Installation, on page 131
- Bulk Credential Update, on page 132
- Bulk Update Wireless Controller Credentials, on page 133
- Bulk Update Switch Credentials, on page 134

Activating Connector 3 on Cisco Spaces

This section provides information about how to activate a deployed connector on your Cisco Spaces account.

Using the following procedure, you generate a token for a deployed connector that you want to add to your Cisco Spaces account. Note that you need a separate token for each deployed connector. Each token is specific to a connector and hence enables Cisco Spaces to identify and connect to connector.

Cisco Spaces supports multiple connectors, and you can associate each connector with one or multiple wireless controllers.

Note A Cisco Spaces: Connector instance can communicate with only one Cisco Spaces account at a time.

Before you begin

Download and deploy the Cisco Spaces: Connector OVA. See Deploying the Connector 3 OVA (Single Interface), on page 83

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left navigation pane, choose **Setup** > **Wireless Networks**.

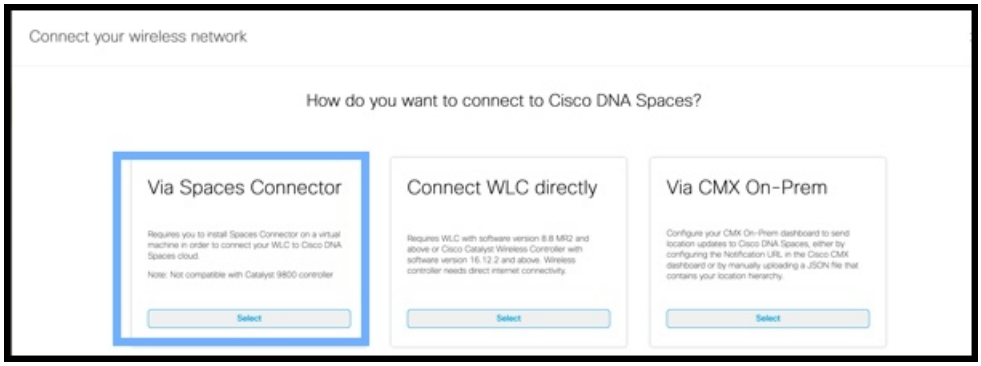
Step 3 In the **Get your wireless network connected with Cisco DNA Spaces** area, click **Add New**.

Step 4 In the **Cisco AireOS Controller/Catalyst 9800 Wireless Controller** area, click **Select.**

Figure 135: Choose Cisco AireOS Controller/Catalyst 9800 Wireless Controller

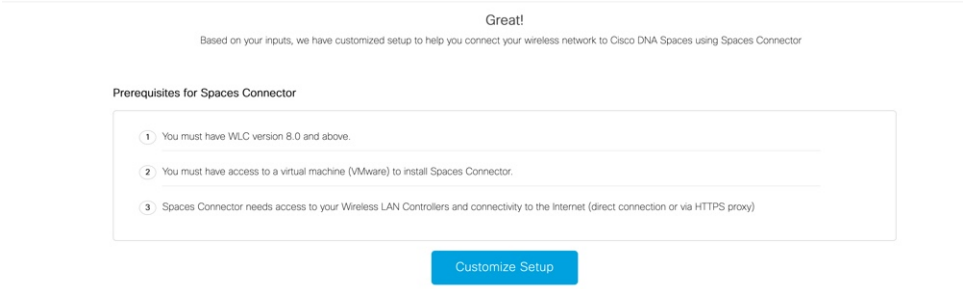
Step 5 In the **Via Spaces Connector** area, click **Select**.

Figure 136: Via Spaces Connector



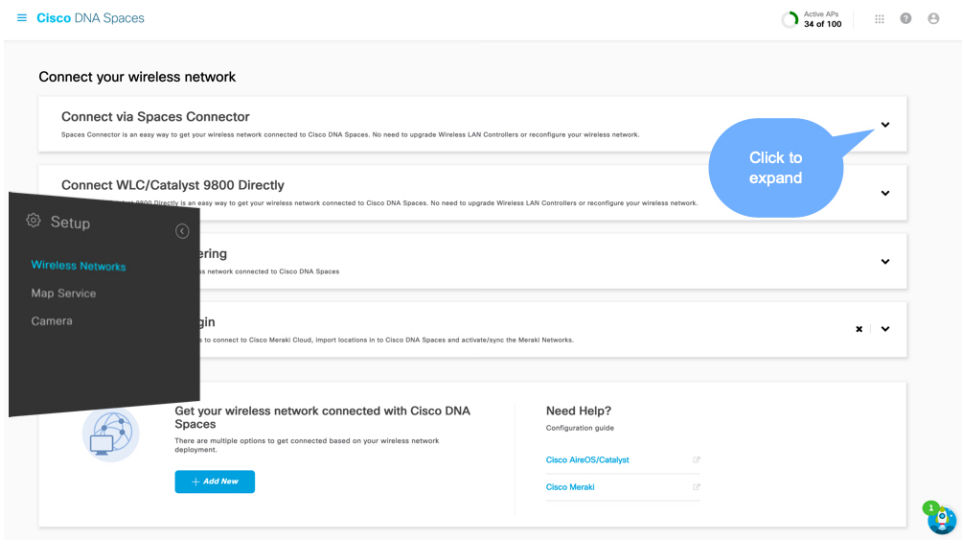
Step 6 In the **Prerequisites for Spaces Connector** dialog box, click **Continue Setup**.

Figure 137: Read Prerequisites for Spaces Connector



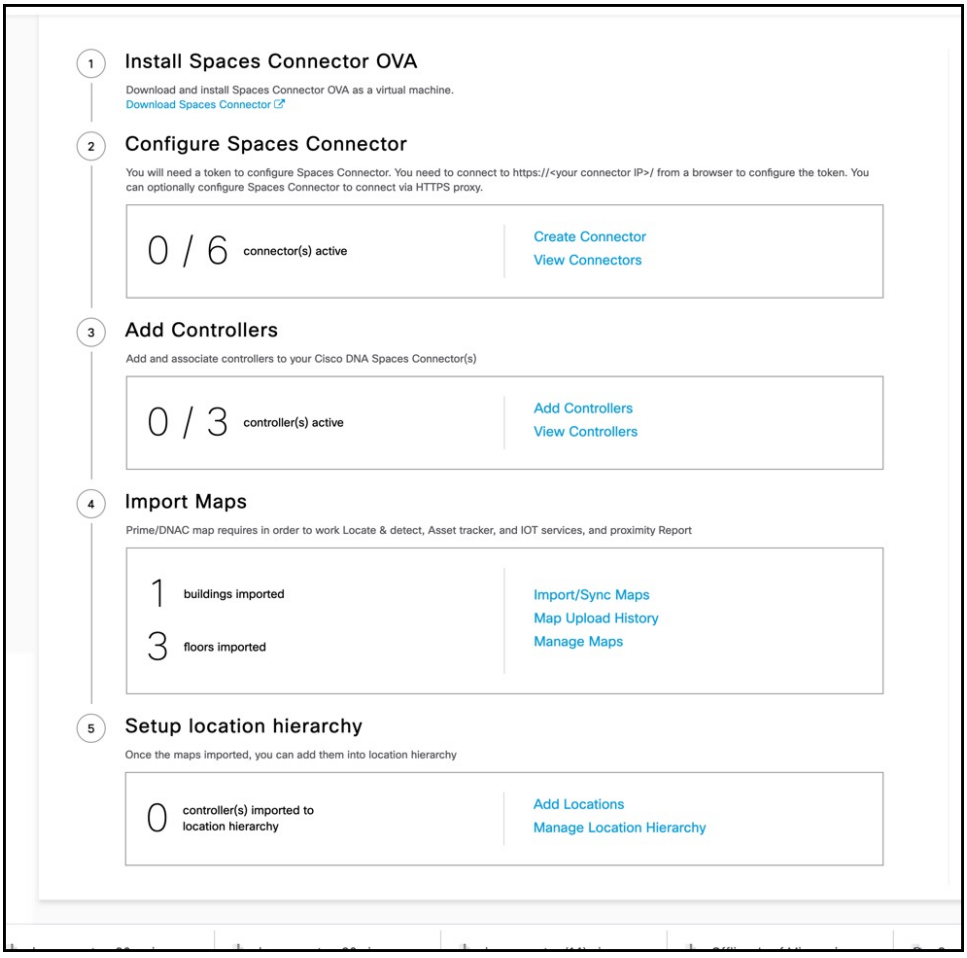
Step 7 Expand the **Connect via Spaces Connector** area using the respective drop-down arrow.

Figure 138: Expand Connect via Spaces Connector



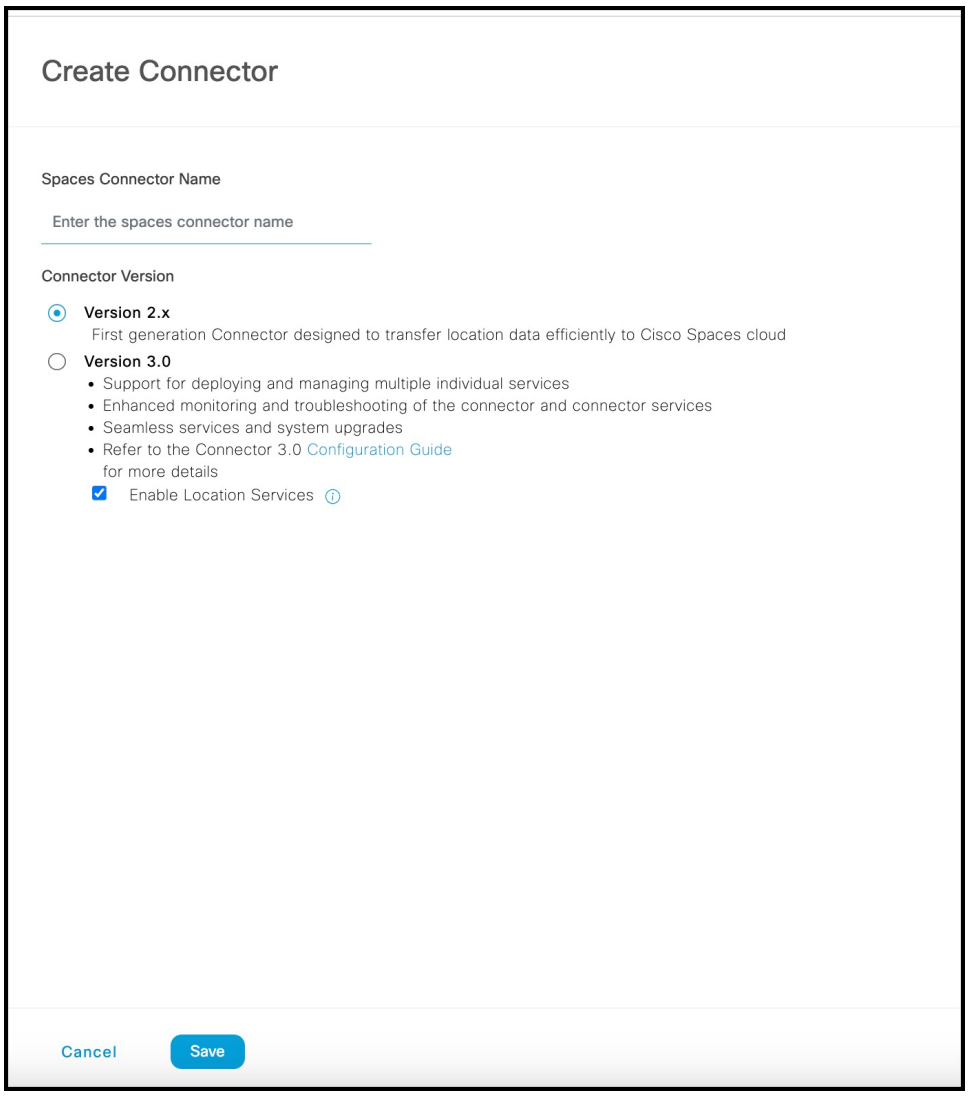
Step 8 In the displayed list of steps, in the **Configure Spaces Connector** area, click **Create Connector**.

Figure 139: Connect via Spaces Connector > Create Connector



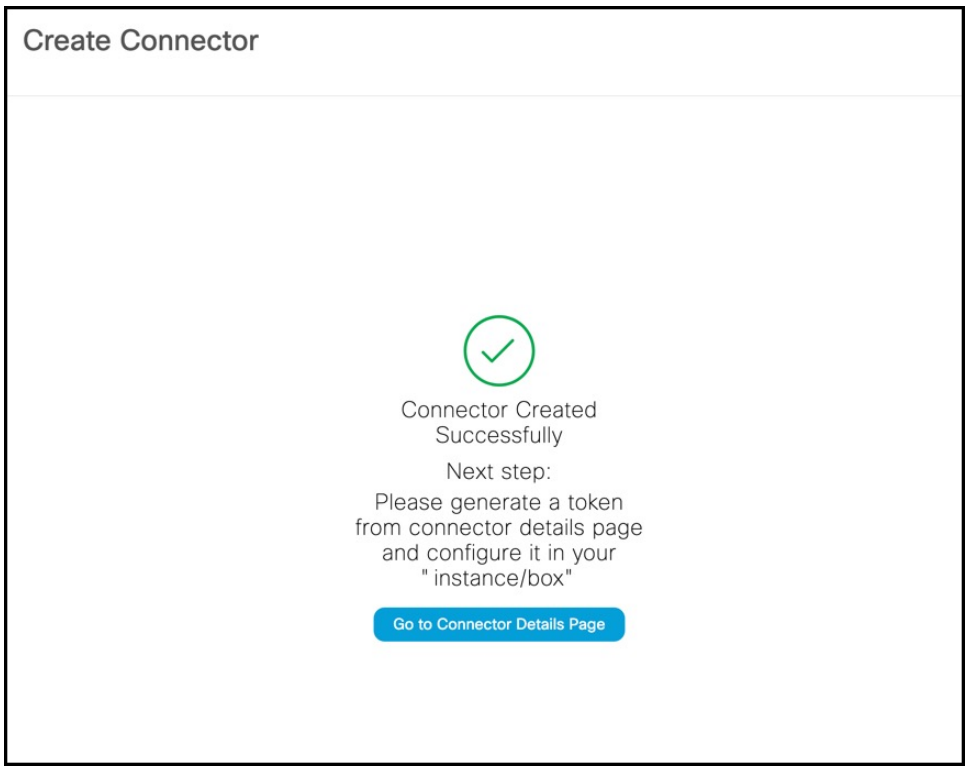
Step 9 In the **Create connector** window that is displayed, enter a name for connector, and click **Version 3.0 (beta).** as the **Connector Version**, and click **Save**.

Figure 140: Name and Version of Connector



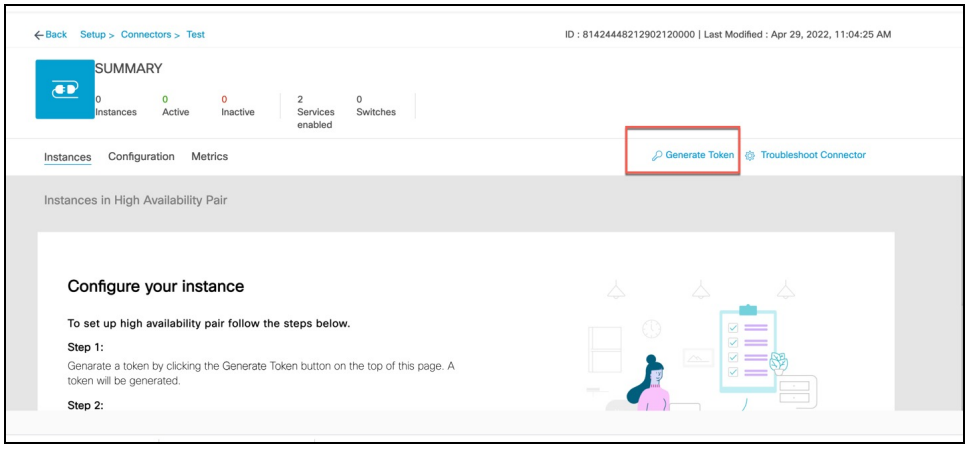
Connector is successfully created. Click **Go to Connector Details** Page.

Figure 141: Connector Created Successfully



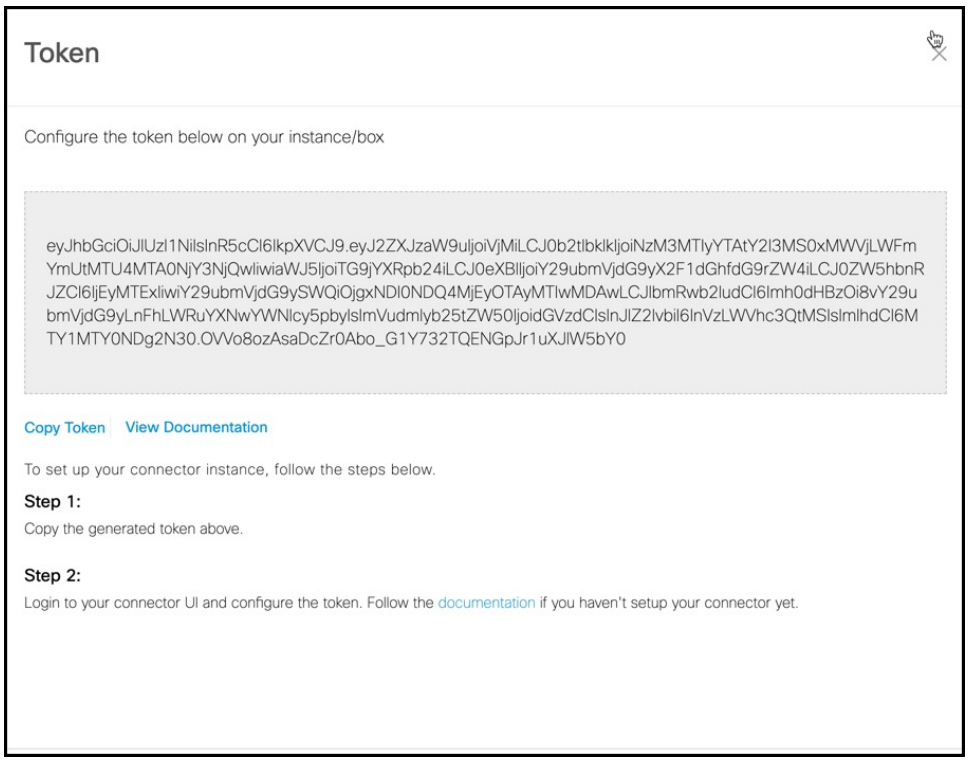
Step 10 In the connector details window, you can see a summary of the configurations for this connector. Click **Generate Token.**

Figure 142: Generate Token



Step 11 In the **Token** window that is displayed, click **Copy Token**.

Figure 143: Copy Token

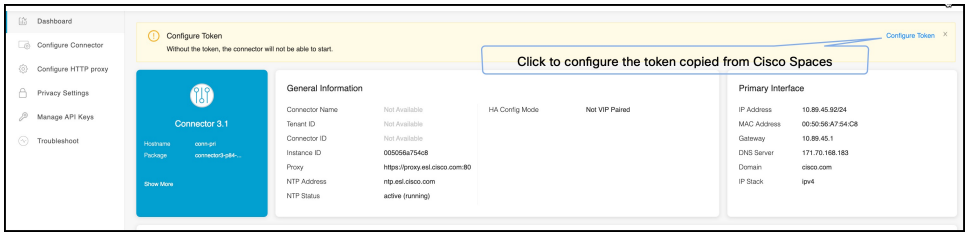


Step 12 Open the connector GUI.

Step 13 (Optional) If your network is behind a proxy, configure the GUI with the proxy. See Configure a Proxy , on page 139

Step 14 In the **Configure Token** area that is displayed, click **Configure Token**.

Figure 144: Configure Token



Step 15 In the window that is displayed, in the **Token** text, field enter the token copied from Cisco Spaces and click **Configure**.

Warning
During this step, if you face a connectivity issue between Cisco Spaces: Connector and Cisco Spaces dashboard, the Connector could hang without an error. You can still access the Connector through SSH. You may also be unable to log in the Connector GUI after this issue.

Step 16 Add the following services as required:

- Configuring IoT Services

- Configuring Hotspot Services

Monitor the Status of Service Installation

After you have initiated the installation of a service, you can monitor the status of the service installation in connector from the Cisco Spaces dashboard.

Procedure

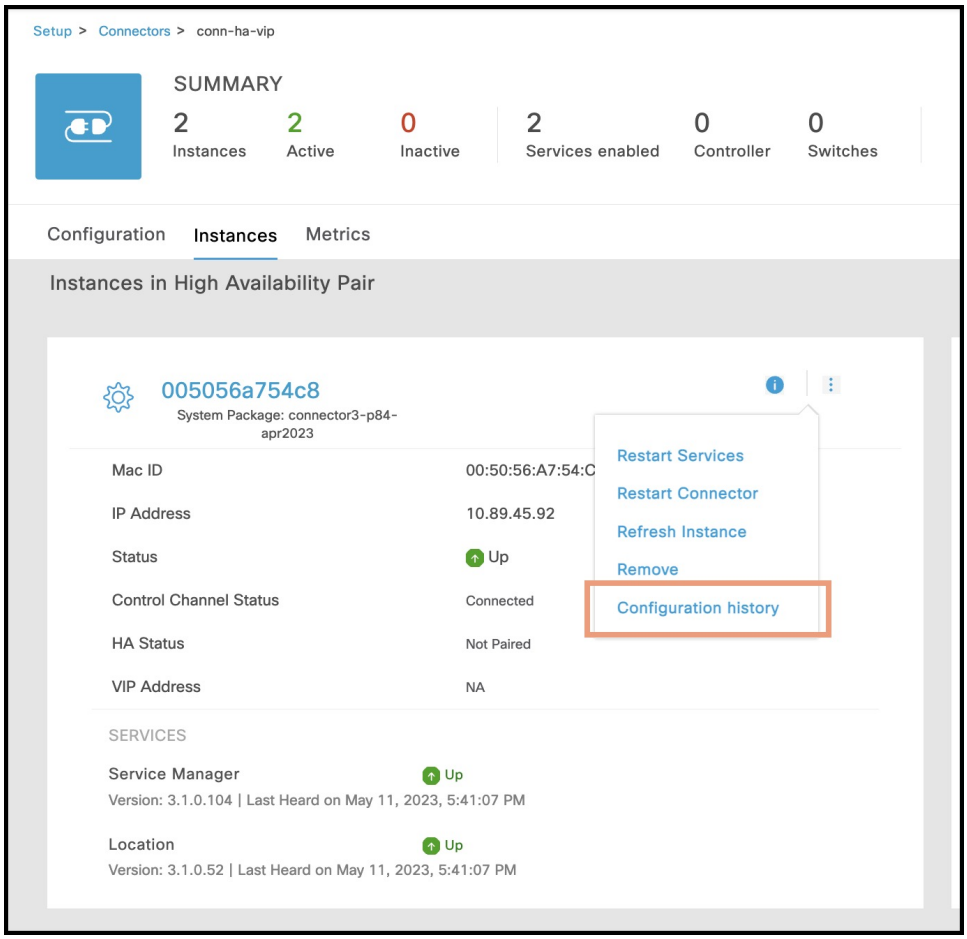
Step 1 From Cisco Spaces dashboard, choose **Setup > Wireless Networks.**

a) In the **Connect via Spaces Connector** area titled **Step 2 Configure Spaces Connector**, click **View Connectors**.

Step 2 From the **Connectors** window that is displayed, choose the connector of your choice.

Step 3 In the connector details window that is displayed, click the **Instances** tab.
You can click the i button and then **Configuration History** to monitor the status of the service installation here.

Figure 145: Monitoring the Status of Service installation



Bulk Credential Update

Bulk credential update allows administrators to update device passwords for multiple wireless controllers or switches associated with a Cisco Spaces connector. The update is performed from the connector details page and applies only to devices that match the selected username and password type.

Use this workflow when the Netconf or SSH password is rotated for wireless controllers or switches and the same credential must be updated in Cisco Spaces for the matching devices.

Note Cisco Spaces displays the matching devices after you select the password type and username. Devices that do not match the selected username are not updated.

Supported Credentials

Table 10: Supported Credentials

Device Type	Credential Type
Wireless Controllers	SSH and NETCONF
Switches	NETCONF

Bulk Update Wireless Controller Credentials

Use this task to update the Netconf or SSH password for one or more wireless controllers that are associated with a connector.

Before you begin

- Ensure that the connector is created and available in Cisco Spaces.
- Ensure that the wireless controllers or switches are associated with the selected connector.
- Ensure that you have permission to manage connector and device credentials.
- Keep the new device password available before you start the update.

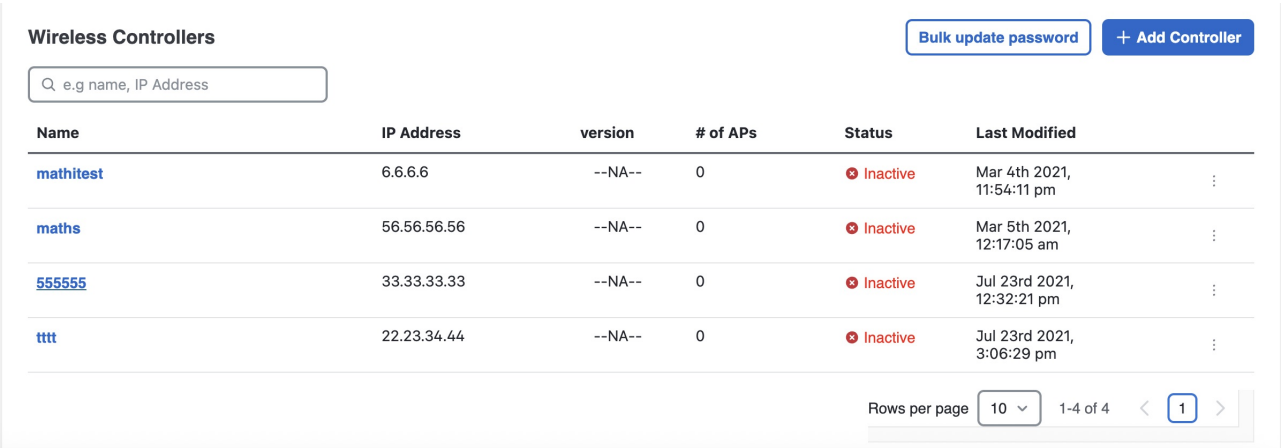
Procedure

Step 1 On the Cisco Spaces, choose **Setup** > **Wireless Networks** > **Connect via Spaces Connector** > **View Connectors**.

Step 2 In the **Connectors** table, click the connector name for which you want to update wireless controller credentials.

The connector details page is displayed.

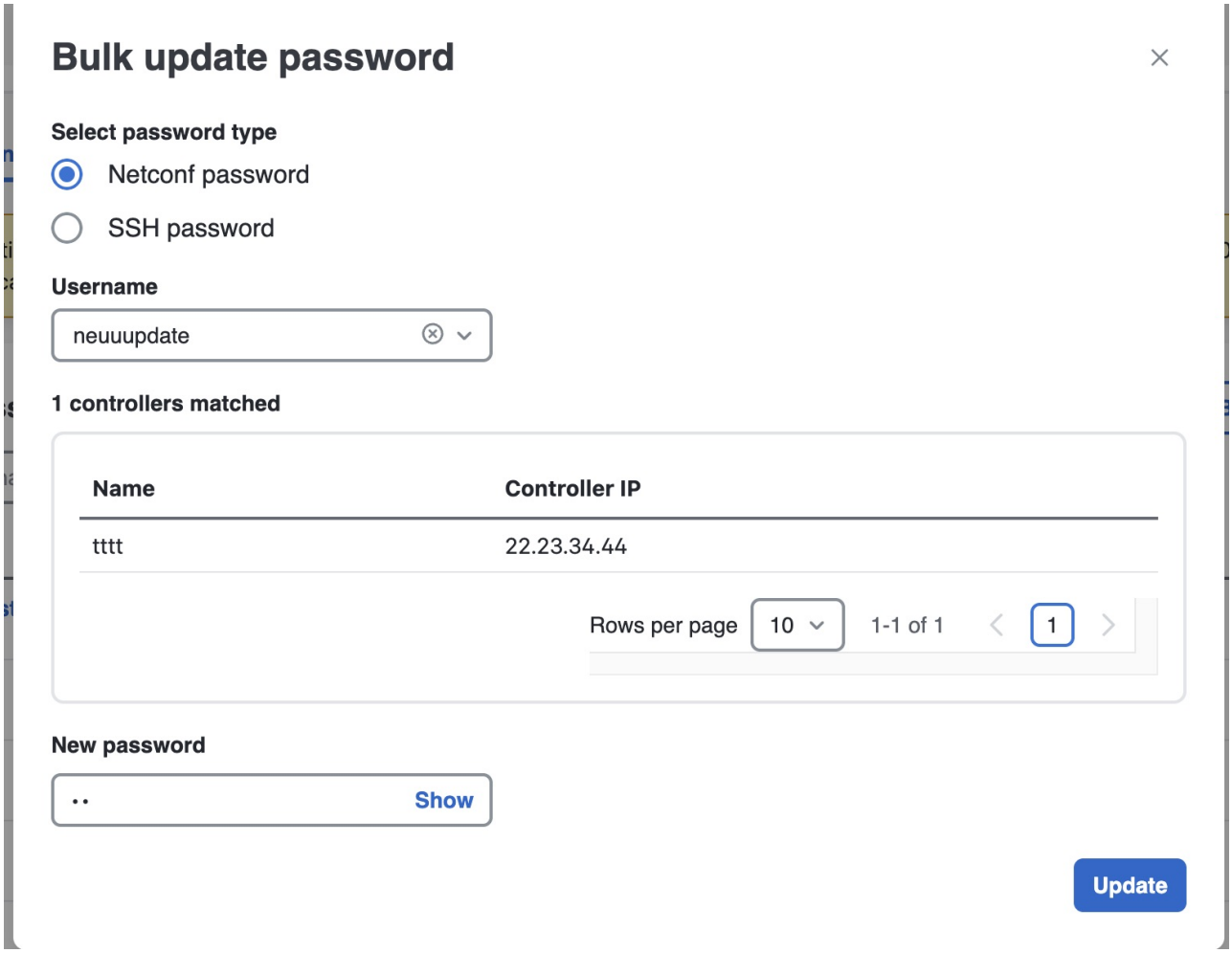
Figure 146: Wireless Controllers



Step 3 In the **Wireless Controllerss** section, click **Bulk update password**.
The **Bulk update password** window is displayed.

Step 4 Under **Select password type**, choose **Netconf password** or **SSH password**.

Figure 147: Wireless Controllers



Step 5 From the **Username** drop-down list, choose the username for which you want to update the password.
Cisco Spaces displays the wireless controllers that match the selected username.

Step 6 In **New password**, enter the new password. To view the entered password, click **Show**.

Step 7 Click **Update**.
After the update is complete, Cisco Spaces displays the **Password update completed** summary with the **Success** and **Failed** counts.

Step 8 Review the **Status** column for each wireless controller, and click **Done**.

Bulk Update Switch Credentials

Use this task to update the **Netconf** or **SSH** password for one or more switches that are associated with a connector.

Procedure

Step 1 On the Cisco Spaces, choose **Setup** > **Wireless Networks** > **Connect via Spaces Connector** > **View Connectors**.

Step 2 In the **Connectors** table, click the connector name for which you want to update wireless controller credentials.

The connector details page is displayed.

Step 3 In the **Switches** section, click **Bulk update password**.
The **Bulk update password** window is displayed.

Step 4 Under **Select password type**, choose **Netconf password** or **SSH password**.

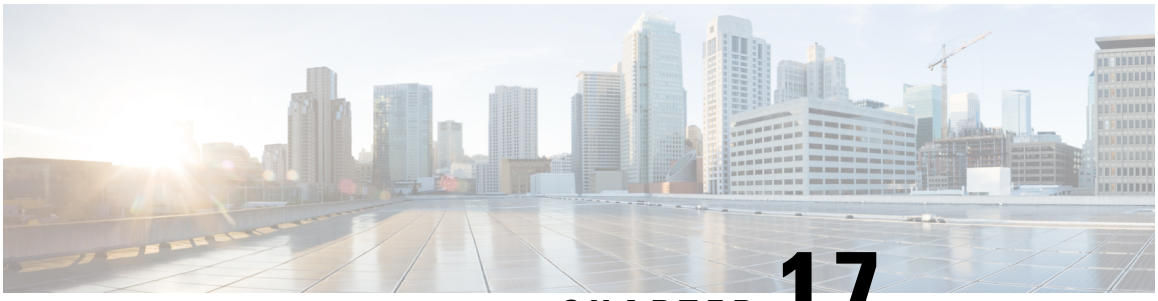
Step 5 From the **Username** drop-down list, choose the username for which you want to update the password.
Cisco Spaces displays the wireless controllers that match the selected username.

Step 6 In **New password**, enter the new password. To view the entered password, click **Show**.

Step 7 Click **Update**.
After the update is complete, Cisco Spaces displays the **Password update completed** summary with the **Success** and **Failed** counts.

Step 8 Review the **Status** column for each wireless controller, and click **Done**.

Note
If no switches match the selected username, choose another username or verify the switch credential configuration before running the update.



CHAPTER 17

Connector GUI

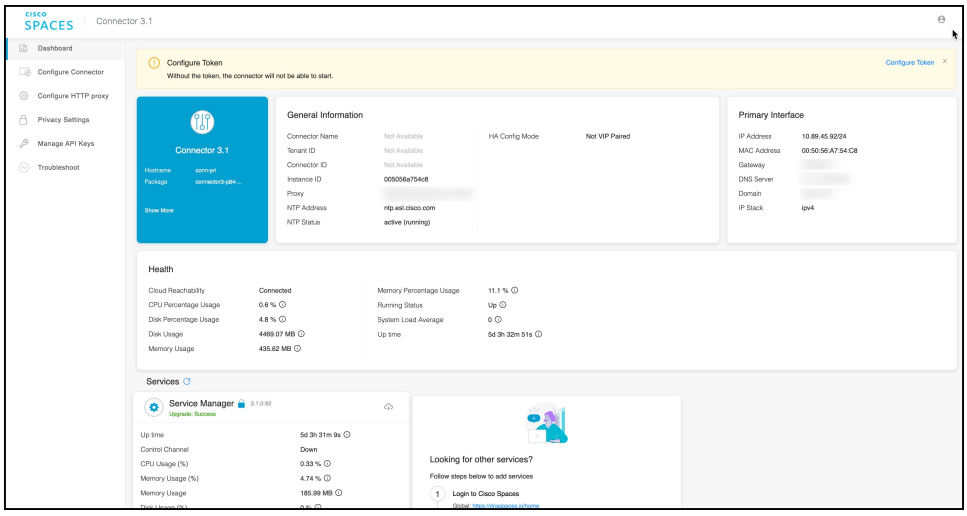
- Connector GUI, on page 137
- Configuring Privacy Settings, on page 138

Connector GUI

The connector GUI allows you to configure the following:

- Proxy
- Tokens retrieved from Cisco Spaces

Figure 148: Connector GUI



The dashboard is divided into areas that provide you with clear information about the following:

- Connector-specific configurations
- Status of connectivity to Cisco Spaces
- Status of services running on connector. Additional buttons here allow you to navigate away and view more detailed information about each service, such as relevant service configurations and status.

The following are the names of various areas on the dashboard, and a description of the information presented:

- General Information: This area has information about the configurations that are made on this connector, the tenant ID, and whether the token is configured.
- Health: This area has information about the health of connector, the connectivity to Cisco Spaces, and other metrics.
- Services: Separate areas are available for each service. See the respective service section for details of the information displayed here.

Configuring Privacy Settings

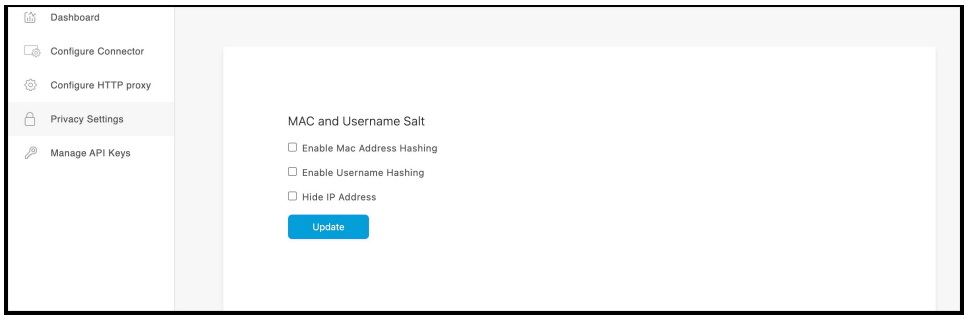
Connector provides a way to protect the Personal Identity Information (PII) of a user and maintain privacy. A hashing algorithm takes the user input (referred to as Salt) and masks the PII fields. When Cisco Spaces receives the data, the MAC addresses, IP addresses, or usernames are masked and the actual user information is protected.

Note This task is optional.

Procedure

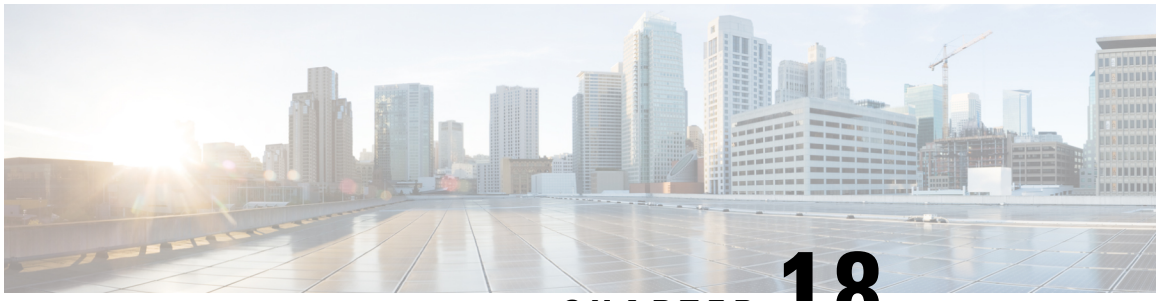
From the Connector GUI left-navigation pane, choose **Privacy Settings**, enter the fields you want to secure with hashing, and press **Submit.**

Figure 149: Configure Privacy Settings



After Cisco Spaces: Connector enables MAC address Salt, it reports pseudo MACs instead of physical device MACs in the data sent to Cisco Spaces cloud. On the Cloud application side, computed locations include the pseudo MACs and share them with applications such as Cisco Spaces: Detect and Locate and Cisco Catalyst Center.

Note
Mac salt cannot contain special characters other than **@** or **!**.



CHAPTER 18

Proxy

- Configure a Proxy , on page 139
- Configure a Transparent Proxy , on page 141

Configure a Proxy

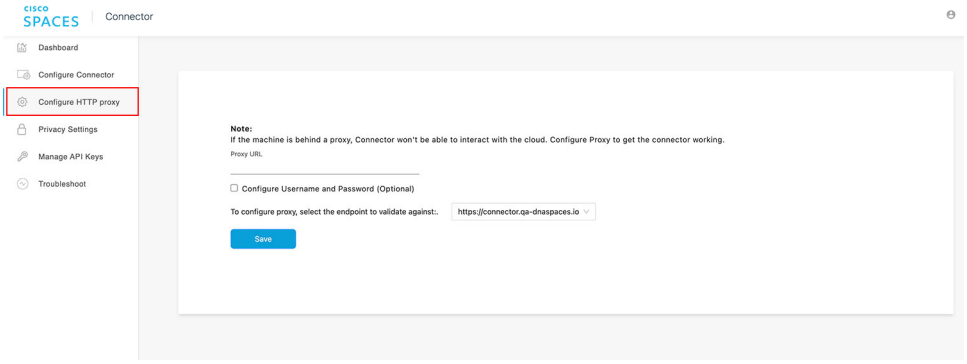
You can set up a proxy to connect the Connector to Cisco Spaces, if the infrastructure hosting the Connector is behind a proxy. Without this proxy configuration, the Connector is unable to communicate with Cisco Spaces

To configure proxy on the Connector, you must do the following:

Procedure

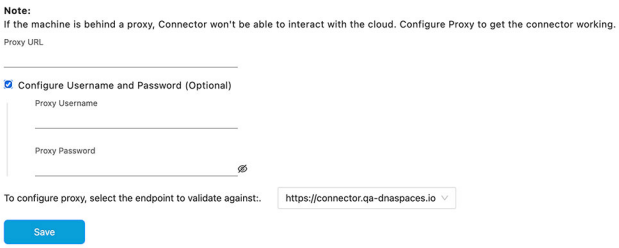
Step 1 In the Connector GUI left navigation pane, click **Configure HTTP Proxy**. Enter your proxy address in the dialog box that is displayed.

Figure 150: Setup Proxy



Note
Choose the endpoint based on your Cisco Spaces Account. For information on how to choose endpoints, see the Table 2: Enable Endpoints, on page 5.

Figure 151: Configure Basic Authentication for Proxy (Optional)



To configure the proxy's basic authentication credentials, click **Configure Username and Password**.

Step 2 You can troubleshoot any issues in proxy configuration. Click **Troubleshoot** and select the Cisco Spaces URL.

Figure 152: Troubleshoot Proxy Issues

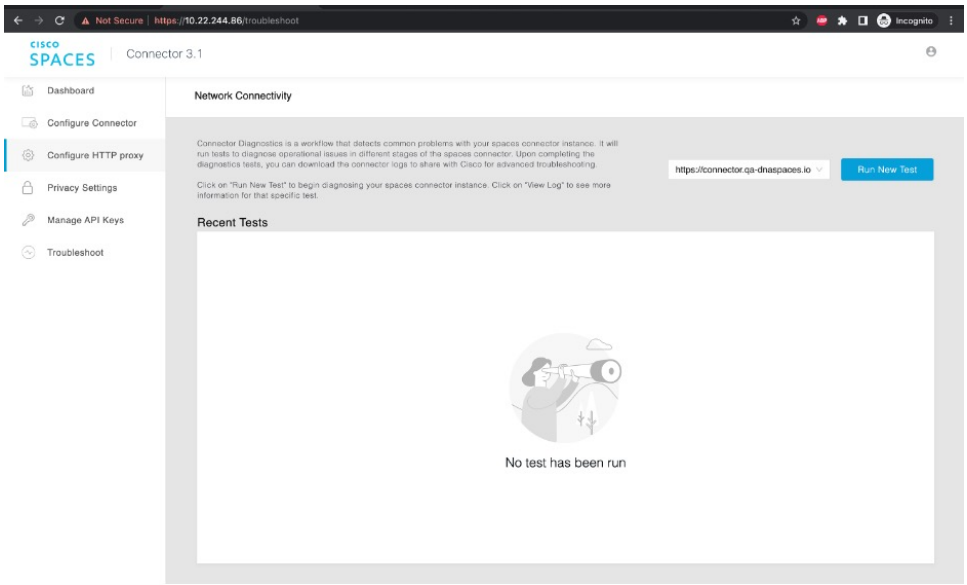
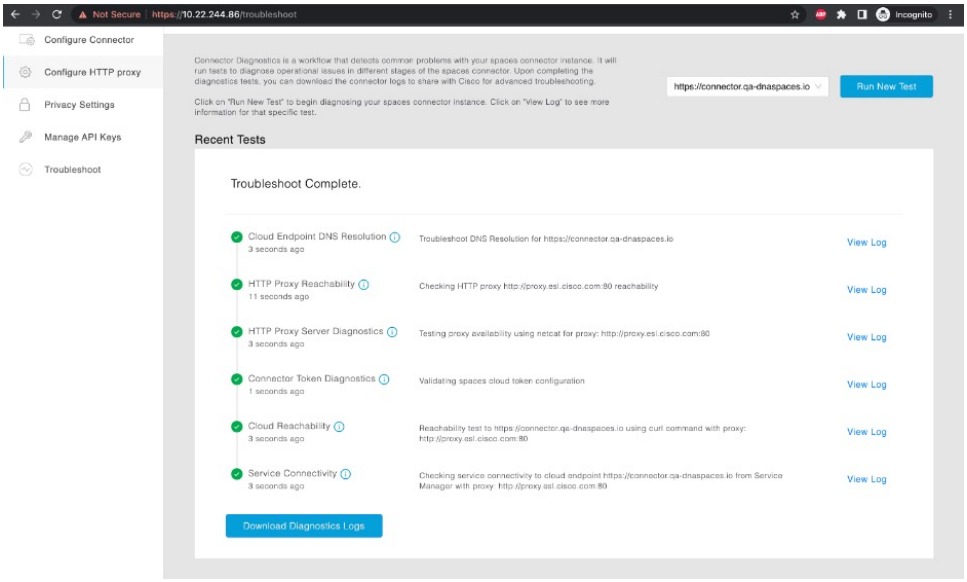


Figure 153: Sample Run Test Results



Configure a Transparent Proxy

To configure a transparent proxy on the Connector, you must do the following:

1. Copy the proxy server certificate and the proxy server certification authority (CA) bundle to the Connector.
2. From the Connector CLI, validate the proxy certificate.
3. From the Connector CLI, import proxy certificates.
4. From the Connector GUI, configure the proxy URL.

Procedure

Step 1 Copy the proxy certificate to the Connector using scp.

The following is a sample command.

```
scp proxy-ca-bundle.pem spacesadmin@[connector-ip]:/home/spacesadmin/
scp proxy-server-cert.pem spacesadmin@[connector-ip]:/home/spacesadmin/
```

Step 2 Log in to the Connector CLI, and validate the copied proxy certificate using the **connectorctl cert proxycert-validate** command.

The following is a sample output of the command:

```
[spacesadmin@may23~]$ connectorctl cert proxycert-validate
-c/home/spacesadmin/squid_proxy_cert_chain.pem
-s/home/spacesadmin/squid23-connector.cisco.com-139582.cer
Executing command: cert
```

```
Command execution status: Success
/home/spacesadmin/squid_proxy_cert_chain.pemand/home/spacesadmin/squid23-connector.cisco.com-139582.cer
exists
/home/spacesadmin/squid23-connector.cisco.com-139582.cer: OK
Validation of certificate is successful
```

For more information on this command, see connectorctl cert proxycert-validate.

Step 3 Import the proxy certification authority (CA) certificates along with other certificates using the **connectorctl cert proxycert-updateca-bundle** command.

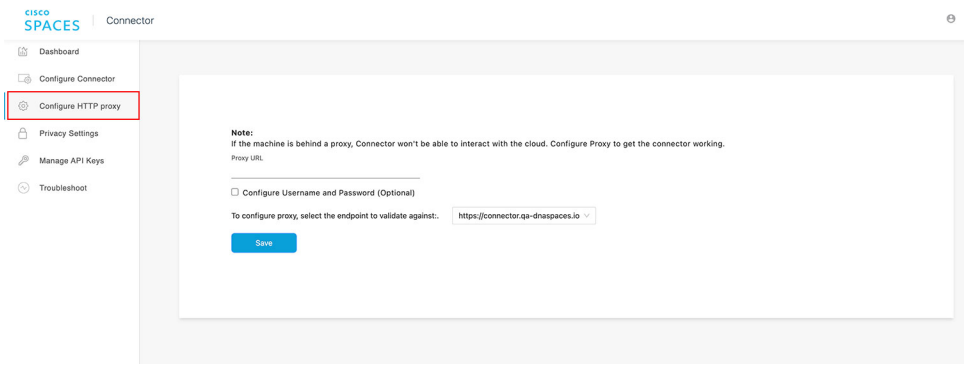
The following is a sample output of the command:

```
[spacesadmin@may23 ~]$ connectorctl
Executing command: cert
Command execution status: Success
cert proxycert-updateca-bundle-c/home/spacesadmin/squid_proxy_cert_chain.pem-s
/home/spacesadmin/squid23-connector.cisco.com-139582.cer
/home/spacesadmin/squid_proxy_cert_chain.pem and
/home/spacesadmin/squid23-connector.cisco.com-139582.cer exist
/home/spacesadmin/squid23-connector.cisco.com-139582.cer: OK
Proxy CA trust bundle updated successfully
System reboot will happen in 10 seconds. Do not execute any other command...
```

For more information on this command, see connectorctl cert proxycert-updateca-bundle.

Step 4 In the Connector GUI left navigation pane, click **Configure HTTP Proxy**. Enter your proxy address in the dialog box that is displayed.

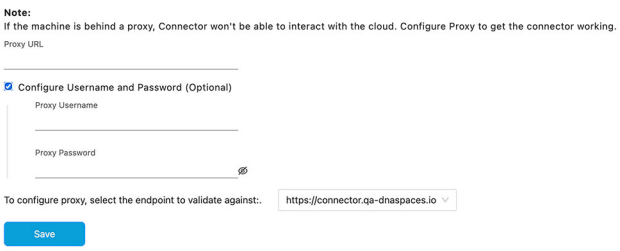
Figure 154: Setup Proxy



Note

Choose the endpoint based on your Cisco Spaces Account. For information on how to choose endpoints, see the Table 2: Enable Endpoints, on page 5.

Figure 155: Configure Basic Authentication for Proxy (Optional)



To configure the proxy's basic authentication credentials, click **Configure Username and Password**.

Step 5 You can troubleshoot any issues in proxy configuration. Click **Troubleshoot** and enter the Cisco Spaces URL.

Figure 156: Troubleshoot Proxy Issues

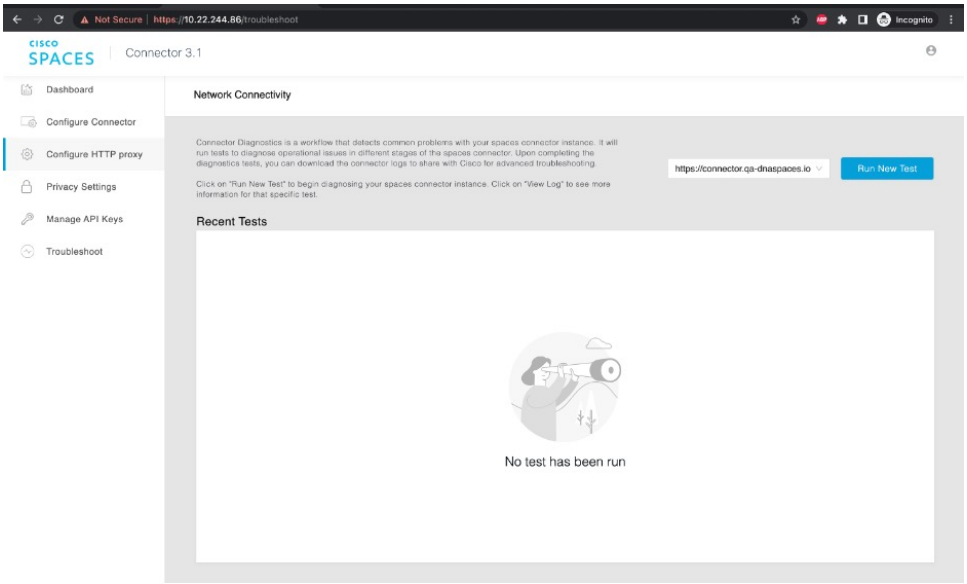
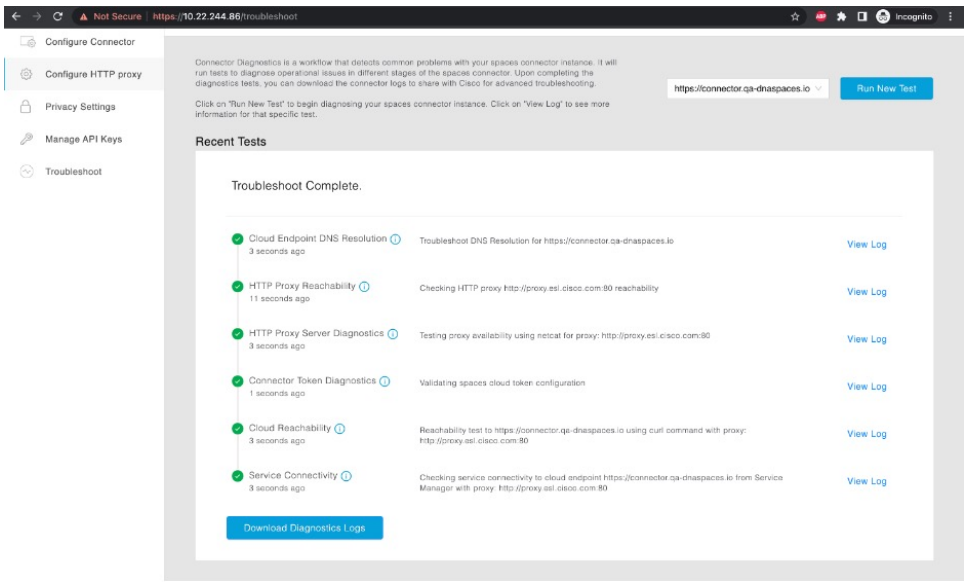
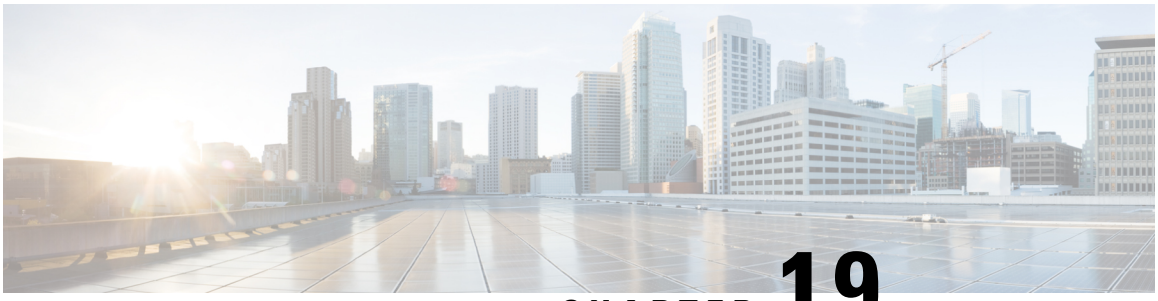


Figure 157: Sample Run Test Results





CHAPTER 19

High Availability

- Configuring Connectors as VIP Paired, on page 145
- Connector Active-Active, on page 150

Configuring Connectors as VIP Paired

Cisco Spaces: Connector high availability uses Virtual Router Redundancy Protocol (VRRP) protocol to determine the state of the instance in the high availability pair. When using VIP pairing with connector 3 and deploying firewalls between the connectors, it's crucial to enable the Virtual Router Redundancy Protocol (VRRP) IP protocol 112.

Ensure that both the source and destination IP addresses match the physical IPs of the connectors. Additionally, to enable proper VRRP functionality, ensure that both connectors reside within the same layer 2 or VLAN segment.

This task shows you how to configure two connectors and pair them with a virtual IP address (VIP).

Note

- Cisco Spaces: Connector in Amazon Web Services (AWS) and Azure Cloud does not support High Availability in the VIP Paired mode.

Before you begin

Install two different Cisco Spaces: Connectors. Configure each connector with a unique IP address.

Note Cisco Spaces: Connector High Availability utilizes HTTPS to assess the status of its peer. When implementing VIP pairing with connector 3 and deploying firewalls between the connectors, ensure that HTTPS TCP port 443 is permitted in both directions.

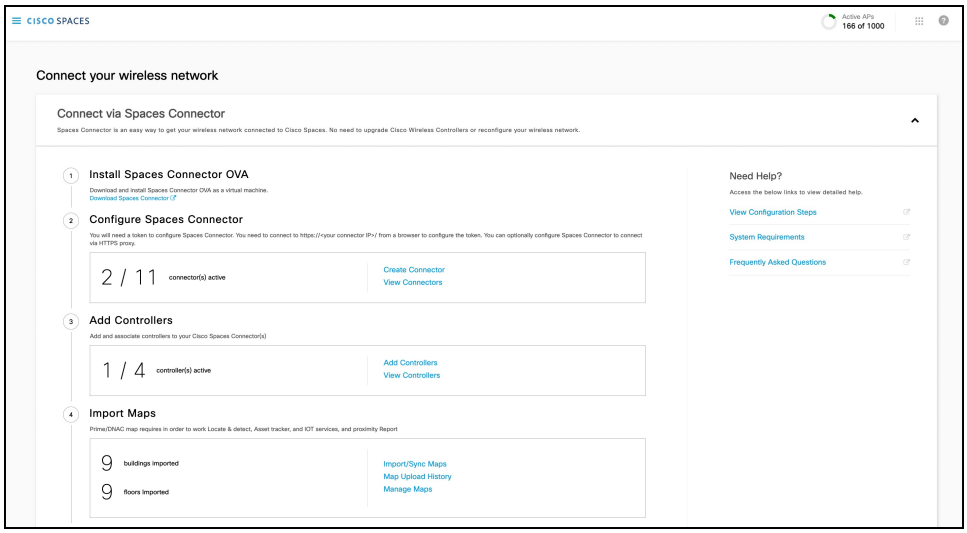
Table 11: HTTPS TCP port 443 connections

Source TCP ephemeral	Destination TCP 443
Connector IP 1	Connector IP 2
Connector IP 2	Connector IP 1

Procedure

Step 1 Login to **Cisco Spaces > Setup > Wireless Networks** and in the **Configure Spaces Connector** area, click **Create Connector**.

Figure 158: Create Connector

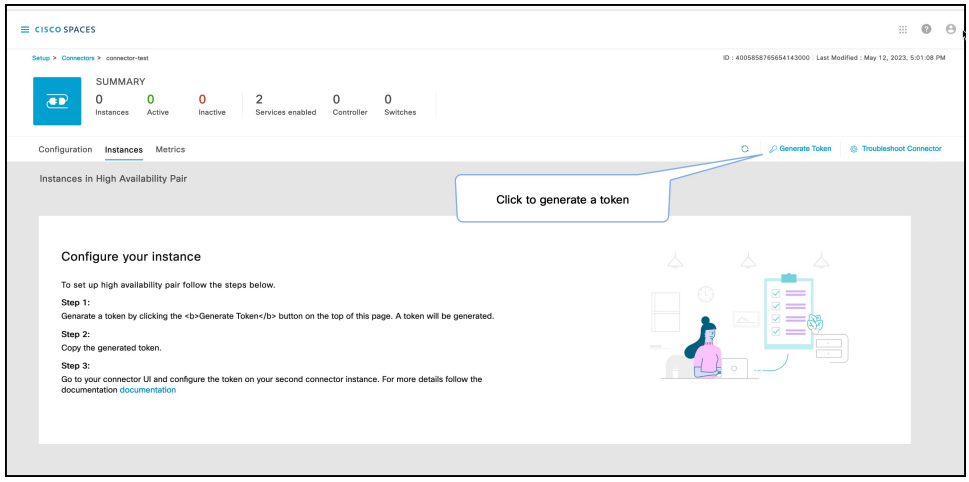


Step 2 Enter a name for the connector and choose the version.

A connector is created. Click **Go to the connector Details** page.

Step 3 In the connector details page, click **Generate Token** in the top-right corner.

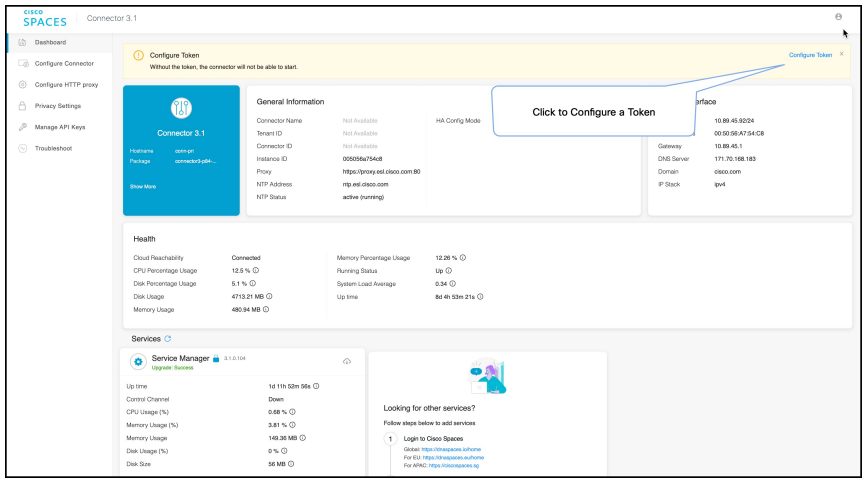
Figure 159: Generate Token



Copy the displayed token.

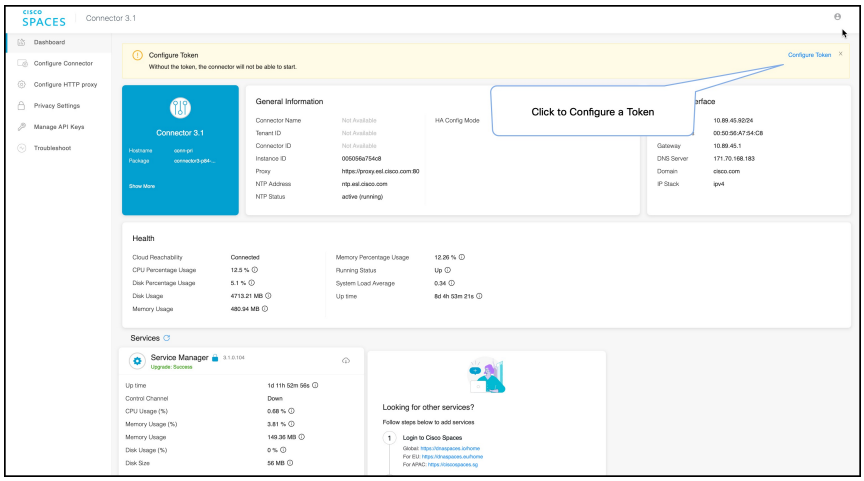
Step 4 Log in to the GUI of the first instance of connector and click **Configure Token** in the top-right corner to provision the first copied token there.

Figure 160: Configure a Token



Step 5 Log in to the GUI of the second instance of connector, and click **Configure Token** in the top-right corner to provision the second copied token there as well.

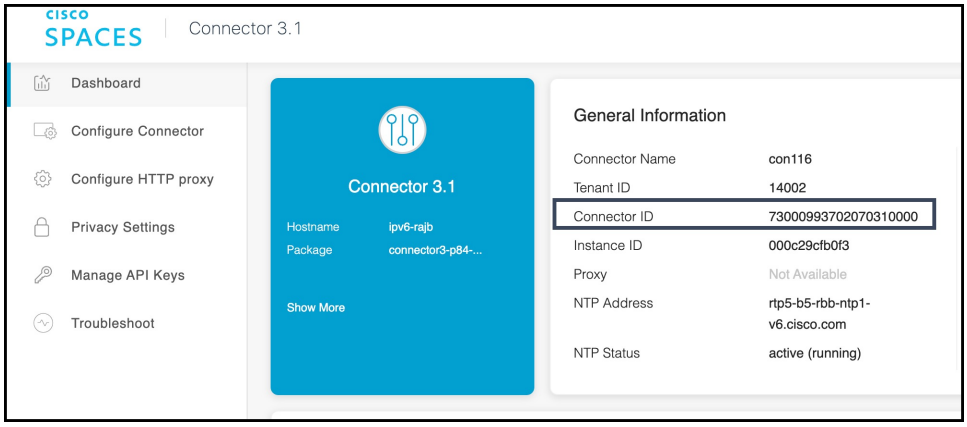
Figure 161: Configure a Token



Two tokens have been configured on two connector instances. You can observe that the connector ID on each instance of the connector is the same

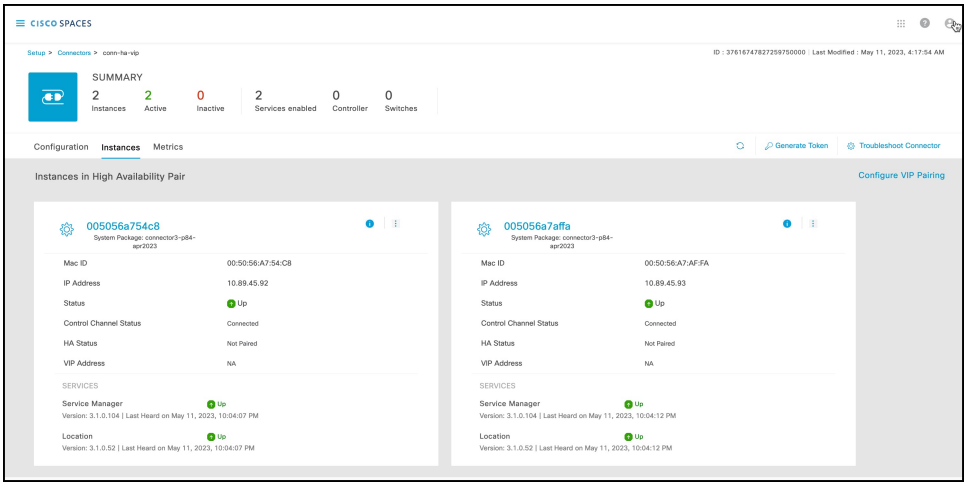
Step 6 On each instance of the connector, observe that the value of the connector ID is the same.

Figure 162: Observe connector ID



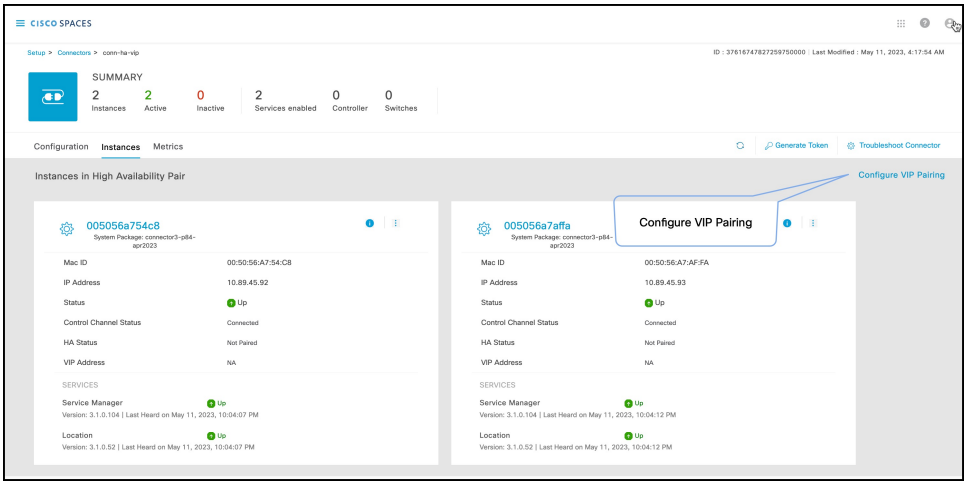
Step 7 On the Cisco Spaces dashboard, go back to the connector details page, and click the **Instances** tab. Here, you can see both the connectors that you configured. Observe that the connector IP addresses are reflected here.

Figure 163: Cisco Spaces dashboard

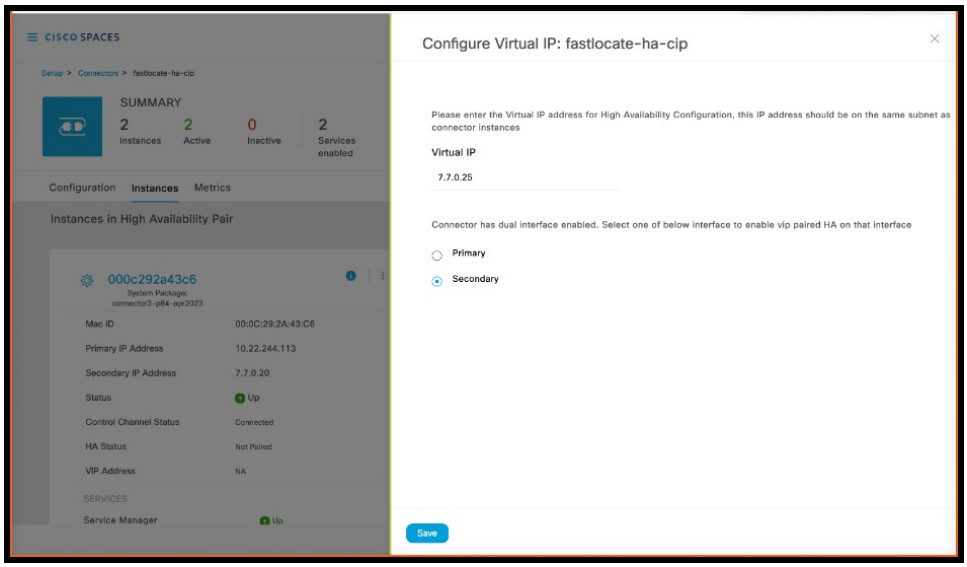


The two connectors are now configured as an active-active pair.

Step 8 To configure the two connector instances as VIP-Paired, click **Configure VIP Pairing** in the top-right corner.



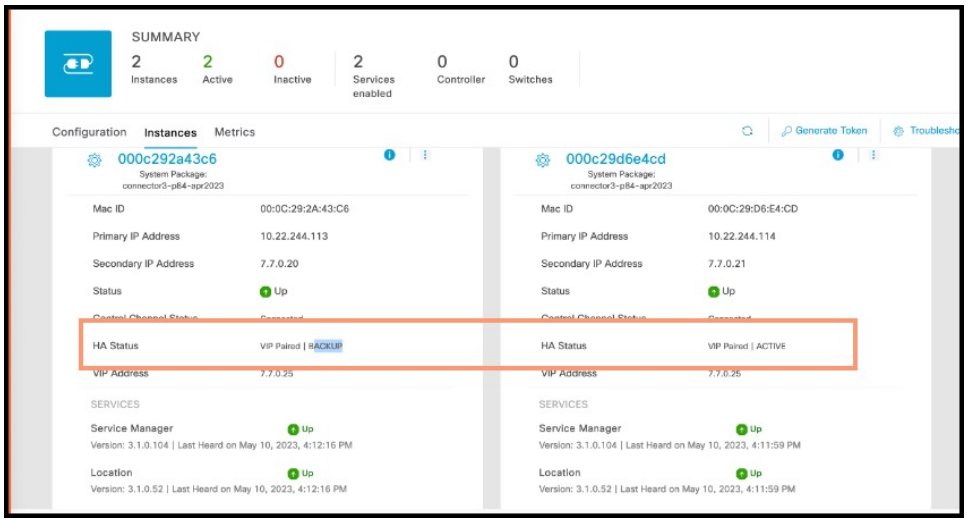
Step 9 In the **Configure Virtual IP** popup that is displayed, enter the Virtual IP address (VIP). If the connector has dual interface enabled, you have to chose which interface would be used VIP pairing.



Note

- Ensure that the VIP is in the same subnet as the connector IP address.
- If you have dual-interface connector, then VIP should be from the subnet of the secondary interface.

You can now see that the instances are configured as a VIP pair.



Connector Active-Active

You can pair two Cisco Spaces: Connectors in an active-active mode to enable the uninterrupted flow of data to Cisco Spaces.

1. You have to generate two tokens on Cisco Spaces and configure these token on two different connector instances. Each connector instance must have a unique IP address.
2. Both connectors receive configurations from Cisco Spaces .
3. The connectors can then connect to devices and send data back to Cisco Spaces.
4. Cisco Spaces then manages the redundant data.
5. If one connector is down, the other connector continues to send data.

Restrictions for Active-Active

- On the Cisco Spaces dashboard, there is no configuration required for two Connectors to be an active-active pair.
- Both Connectors connect to all Wireless Controllers and send traffic to Cisco Spaces. The traffic from Wireless Controllers to Cisco Spaces hence increases.
- To be an active-active Connector pair, two connectors must run OVA version 3.0 or higher.
- There is no failover support for Hyperlocation.

Note

- Cisco FastLocate is re-established after failover with a delay of three to four minutes.
- Reprovision services after a failover for active-active. For VIP-paired mode, re-provisioning is unnecessary.

- There is no support for monitoring the Connector active-active feature.
- You cannot run IoT Service high availability in Active - Active mode. To run IoT Service high availability, use VIP-paired mode.

Configuring Connectors in Active-Active

This task shows you how to configure two connectors as active-active.

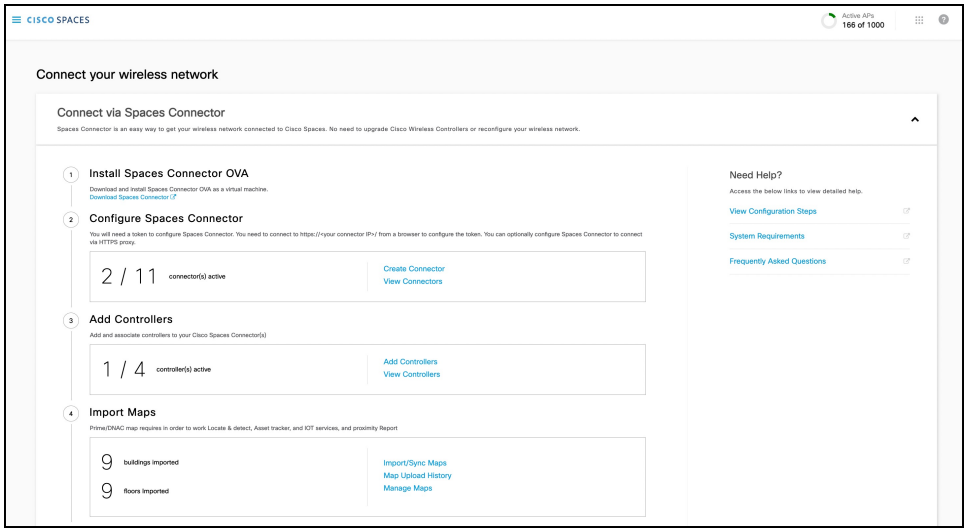
Before you begin

Install two different instances of Cisco Spaces: Connectors of OVA version 3.0 or higher. Configure each instance of connector with a unique IP address.

Procedure

Step 1 Login to **Cisco Spaces > Setup > Wireless Networks** and in the **Configure Spaces Connector** area, click **Create Connector**.

Figure 164: Create Connector

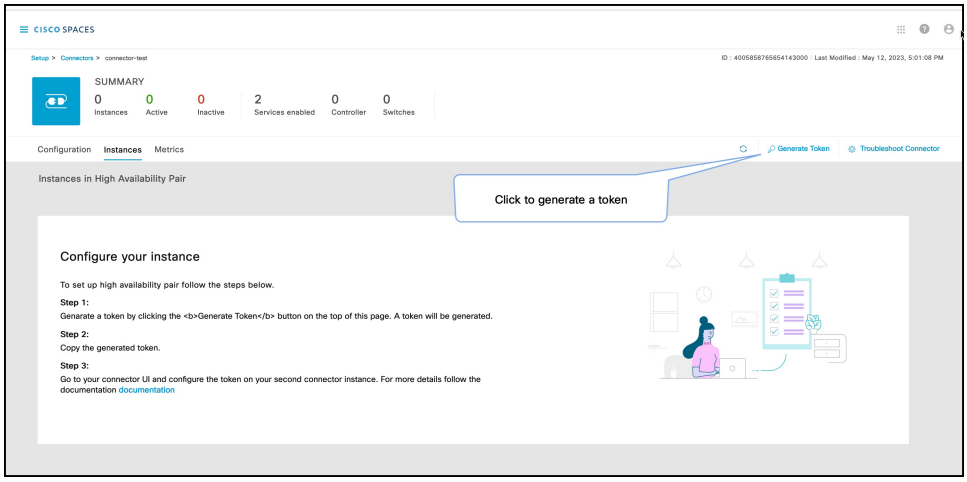


Step 2 Enter a name for the connector and choose the version.

A connector is created. Click **Go to the connector Details** page.

Step 3 In the connector details page, click **Generate Token** in the top-right corner.

Figure 165: Generate Token

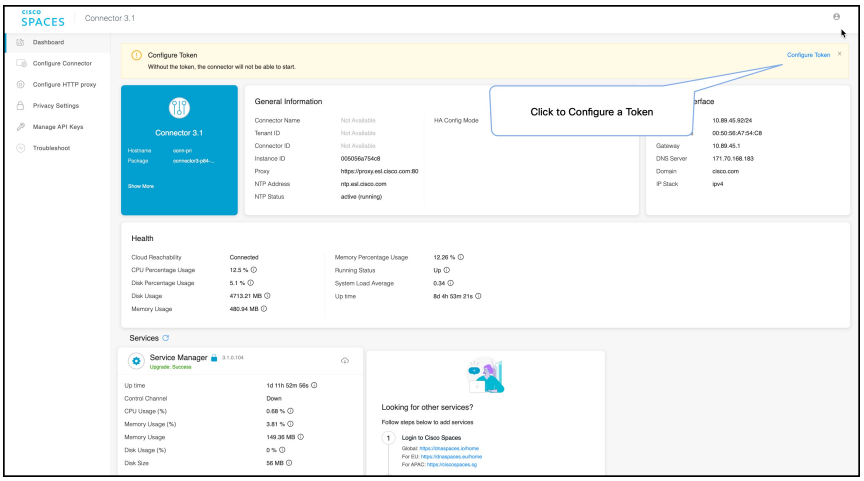


Copy the displayed token.

Step 4 Repeat Step 3 to generate and copy a second token.

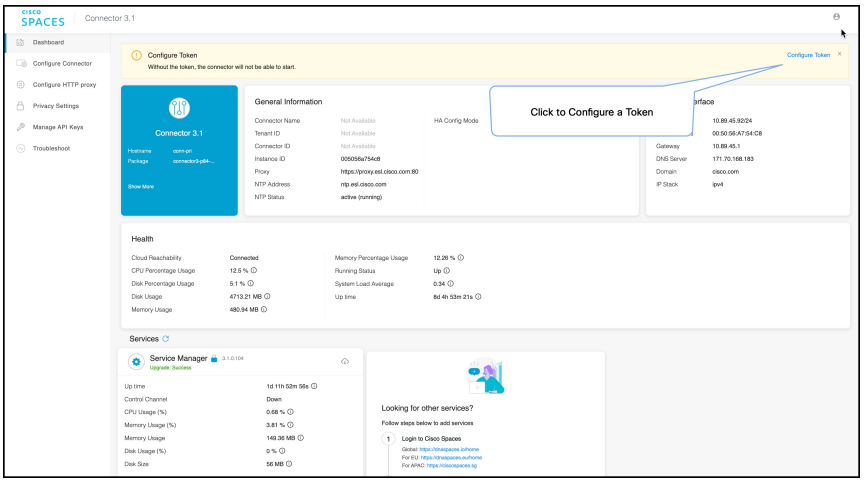
Step 5 Log in to the GUI of the first instance of connector and click **Configure Token** in the top-right corner to provision the first copied token there.

Figure 166: Configure a Token



Step 6 Log in to the GUI of the second instance of connector, and click **Configure Token** in the top-right corner to provision the second copied token there as well.

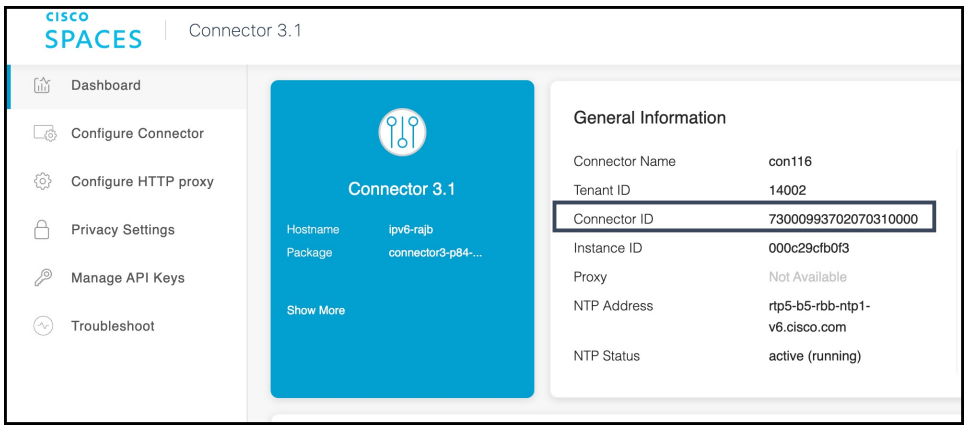
Figure 167: Configure a Token



Two tokens have been configured on two connector instances. You can observe that the connector ID on each instance of the connector is the same

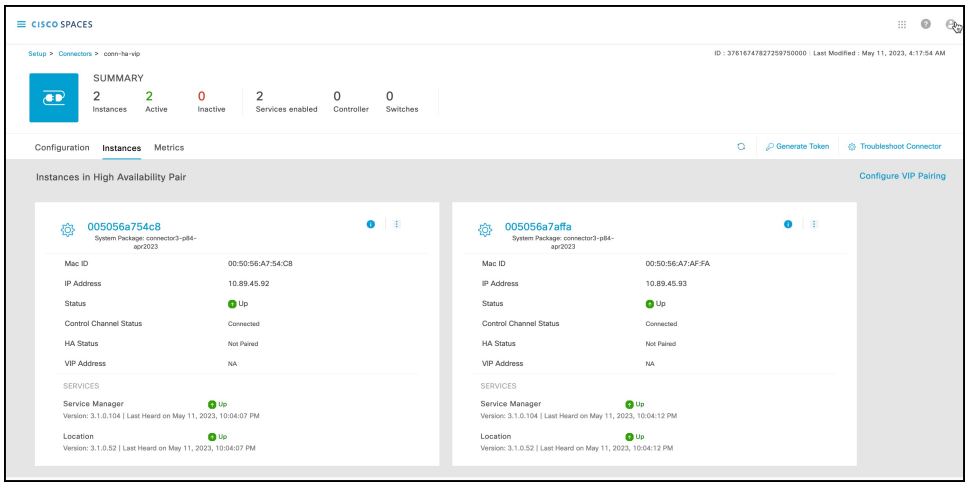
Step 7 On each instance of the connector, observe that the value of the connector ID is the same.

Figure 168: Observe connector ID

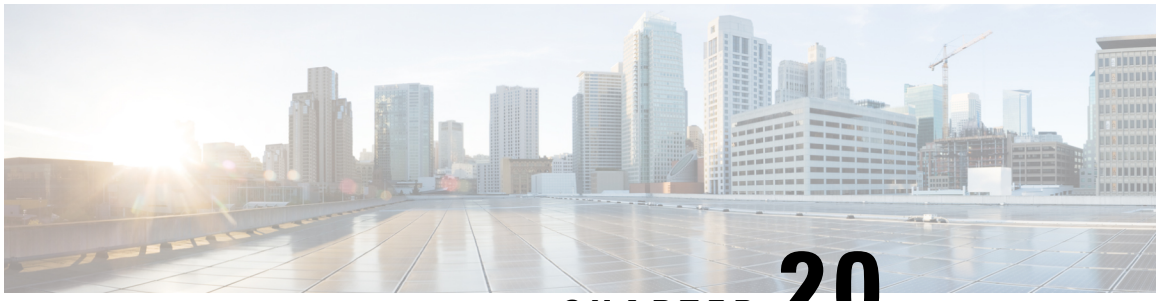


Step 8 On the Cisco Spaces dashboard, go back to the connector details page, and click the **Instances** tab. Here, you can see both the connectors that you configured. Observe that the connector IP addresses are reflected here.

Figure 169: Cisco Spaces dashboard



The two connectors are now configured as an active-active pair.



CHAPTER 20

Cisco Spaces: FIPS Support in Connector

- FIPS Overview, on page 155
- Protocol Requirements, on page 155
- FIPS Support, on page 156
- Enabling FIPS, on page 156
- Disabling FIPS, on page 157
- Adding a FIPS-Enabled Catalyst 9800 Controller to a FIPS-Enabled Connector, on page 159

FIPS Overview

Cisco Spaces Connector supports the Federal Information Processing Standard 140-3 (FIPS). FIPS is a standard published by the National Institute of Standards and Technology (NIST) and is used to accredit cryptographic modules in software components. The US Federal government requires that all IT products dealing with Sensitive, but Unclassified (SBU) information meet these standards. If your system needs to be FIPS compliant, you can enable FIPS. Once you do so, the system uses the cryptographic algorithms defined by the NIST for FIPS for all encrypted communication between its internal and external components.

Protocol Requirements

- Transport Layer Security (TLS) 1.2or higher.
- Advanced Encryption Standard (AES) 256.
- Secure Hash Algorithm (SHA) 128 or higher.
- One of the following:
    - Rivest, Shamir, and Adelman (RSA) 2048 or higher.
- Elliptic Curve Digital Signature Algorithm (ECDSA) with a National Institute of Standards and Technology (NIST) curve of P-256 or higher.

FIPS Support

From the Connector 3.2 Jan 2025 release, FIPS support is available. The supported Connector deployment platforms and services are as follows:

Table 12: FIPS supported Connector deployment platforms

Platforms	FIPS Support Availability (Yes/No)
OVA	Yes
Nutanix	Yes
Hyper-V	Yes
AWS	No
Azure	No

Table 13: FIPS supported Connector services

Services	FIPS Support Availability (Yes/No)
Location	Yes
IOT Wireless	Yes
Local Firehose	Yes
IOT Wired	No
Hotspot	No

Enabling FIPS

To enable FIPS on the Connector, you must do the following:

1. Log in to the Connector CLI, and enable FIPS on the connector.
2. From the Connector CLI, validate if the FIPS is enabled.

Procedure

Step 1 Log in to the Connector CLI, and enable FIPS on the connector using the **connectorctl fips enable** command.
The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl fips enable
Executing command:fips
```

```
Terminated
[spacesadmin@connector ~]$ Connection to 10.22.244.2 closed by remote host
```

For more information on this command, see the **connectorctl fips enable** command page.

Note
Enabling FIPS restarts the connector VM.

Step 2 Validate if the FIPS is enabled using the **connectorctl fips show** command.
The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl fips show
Executing command:fips
Command execution status:Success
----------------------
FIPS mode status:
FIPS mode is enabled.
verify FIPS mode is enabled at the operating system level:
crypto.fips_enabled = 1
OpenSSL version:
FIPS Toolkit Enabled
CiscoSSL 1.1.1y.7.3.377-fips
ssh runs in FIPS mode
x509v3-ecdsa-sha2-nistp256
x509v3-ecdsa-sha2-nistp384
x509v3-ecdsa-sha2-nistp521
x509v3-ssh-rsa
x509v3-rsa2048-sha256
ecdsa-sha2-nistp256
ecdsa-sha2-nistp256-cert-v01@openssh.com
ecdsa-sha2-nistp384
ecdsa-sha2-nistp384-cert-v01@openssh.com
ecdsa-sha2-nistp521
ecdsa-sha2-nistp521-cert-v01@openssh.com
ssh-rsa
ssh-rsa-cert-v01@openssh.com

[spacesadmin@connector ~]$
```

For more information on this command, see the **connectorctl fips show** command page.

Disabling FIPS

Note There is no option to disable FIPS. The only way is to roll back to the initial setup by resetting the connector using the **connectorctl reset** command.

To roll back to the initial setup, you must do the following:

1. Log in to the Connector CLI, and reset the connector.
2. From the Connector CLI, validate whether the roll back is successful.

Procedure

Step 1 Log in to the Connector CLI, and reset the connector using the **connectorctl reset** command.

The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl reset
Executing command:reset
WARNING: This command resets all connector configuration including http proxy and token and bring
system to initial state. You can't undo these changes.
Type yes to continue or any other letter to abort:yes
Terminated
[spacesadmin@connector ~]$ Connection to 10.22.244.45 closed by remote host.
Connection to 10.22.244.45 closed.
```

For more information on this command, see the **connectorctl reset** command page.

Note
Rolling back to the initial setup restarts the connector VM.

Step 2 Validate whether the roll back is successful using the **connectorctl fips show** command.

The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl fips show
Executing command:fips
Command execution status:Success
----------------------
FIPS mode status:
FIPS mode is disabled.
verify FIPS mode is enabled at the operating system level:
crypto.fips_enabled = 0
OpenSSL version:
FIPS Toolkit Enabled
CiscoSSL 1.1.1y.7.3.377-fips
ssh runs in FIPS mode
x509v3-ecdsa-sha2-nistp256
x509v3-ecdsa-sha2-nistp384
x509v3-ecdsa-sha2-nistp521
x509v3-ssh-rsa
x509v3-rsa2048-sha256
ecdsa-sha2-nistp256
ecdsa-sha2-nistp256-cert-v01@openssh.com
ecdsa-sha2-nistp384
ecdsa-sha2-nistp384-cert-v01@openssh.com
ecdsa-sha2-nistp521
ecdsa-sha2-nistp521-cert-v01@openssh.com
ssh-rsa
ssh-rsa-cert-v01@openssh.com
```

For more information on this command, see the **connectorctl fips show** command page.

Adding a FIPS-Enabled Catalyst 9800 Controller to a FIPS-Enabled Connector

Enabling FIPS

To enable FIPS on the Connector, you must do the following:

1. Log in to the Connector CLI, and enable FIPS on the connector.
2. From the Connector CLI, validate if the FIPS is enabled.

Procedure

Step 1 Log in to the Connector CLI, and enable FIPS on the connector using the **connectorctl fips enable** command.
The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl fips enable
Executing command:fips
Terminated
[spacesadmin@connector ~]$ Connection to 10.22.244.2 closed by remote host
```

For more information on this command, see the **connectorctl fips enable** command page.

Note
Enabling FIPS restarts the connector VM.

Step 2 Validate if the FIPS is enabled using the **connectorctl fips show** command.
The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl fips show
Executing command:fips
Command execution status:Success
----------------------
FIPS mode status:
FIPS mode is enabled.
verify FIPS mode is enabled at the operating system level:
crypto.fips_enabled = 1
OpenSSL version:
FIPS Toolkit Enabled
CiscoSSL 1.1.1y.7.3.377-fips
ssh runs in FIPS mode
x509v3-ecdsa-sha2-nistp256
x509v3-ecdsa-sha2-nistp384
x509v3-ecdsa-sha2-nistp521
x509v3-ssh-rsa
x509v3-rsa2048-sha256
ecdsa-sha2-nistp256
ecdsa-sha2-nistp256-cert-v01@openssh.com
ecdsa-sha2-nistp384
ecdsa-sha2-nistp384-cert-v01@openssh.com
ecdsa-sha2-nistp521
ecdsa-sha2-nistp521-cert-v01@openssh.com
```

```
ssh-rsa
ssh-rsa-cert-v01@openssh.com

[spacesadmin@connector ~]$
```

For more information on this command, see the **connectorctl fips show** command page.

Configuring FIPS Authorization Key in the Controller

Procedure

Step 1 Log in to the Catalyst 9800 controller CLI, and enter the global configuration mode using the **configure terminal** command.

The following is a sample output of the command:

```
Device# configure terminal
```

Step 2 Configuring FIPS authorization key on the controller using the **fips authorization-key** *hex* command.

The following is a sample output of the command:

```
Device(config)# fips authorization-key <hex>
```

Step 3 Return to the privileged **EXEC** mode using the **end** command.

The following is a sample output of the command:

```
Device(config)# end
```

Step 4 Save all configurations using the **write memory** command.

The following is a sample output of the command:

```
Device# write memory
```

Step 5 Boot the controller in FIPS mode using the **reload** command.

The following is a sample output of the command:

```
Device# reload
```

Obtaining a Certificate from the Connector

Procedure

Step 1 Log in to the Connector CLI, and obtain a certificate using the **connectorctl -s location keystore showcert -n fipsca** command.

The following is a sample output of the command:

```
[spacesadmin@connector ~]$ connectorctl -s location keystore showcert -n fipsca
Executing command:keystore
Command execution status:Success
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

Step 2 Copy the certificate and paste it into a text editor to add it to the controller trustpool. For information, see the Importing a CA Certificate to the Trustpool section.

Importing a CA Certificate into the Trustpool

Procedure

Step 1 Log in to the Catalyst 9800 controller CLI, and enter the global configuration mode using the **configure terminal** command.

The following is a sample output of the command:

```
Device# configure terminal
```

Step 2 Import a CA certificate into the Trustpool using the **crypto pki trustpool import terminal** command.

Paste the certificate and press **Enter**. For more information, see Obtaining a Certificate from the Connector.

The following is a sample output of the command:

```
Device(config)# crypto pki trustpool import terminal
% Enter PEM-formatted CA certificate.
% End with a blank line or "quit" on a line by itself.
line by itself.
-----BEGIN CERTIFICATE-----
....
....
-----END CERTIFICATE-----
quit
% PEM files import succeeded.
```

Step 3 Return to the privileged **EXEC** mode using the **end** command.

The following is a sample output of the command:

```
Device(config)# end
```

Step 4 Save all configurations using the **write memory** command.

The following is a sample output of the command:

```
Device# write memory
```

Reinitiating the NMSP Connection on the Controller

Perform one of the following actions to reinitiate the NMSP connection to the controller after importing the connector certificate into the controller:

Procedure

Step 1 Restart the location service from the Connector CLI or Cisco Spaces dashboard.

Step 2 Edit and save the controller credentials in the Cisco Spaces dashboard.

Note

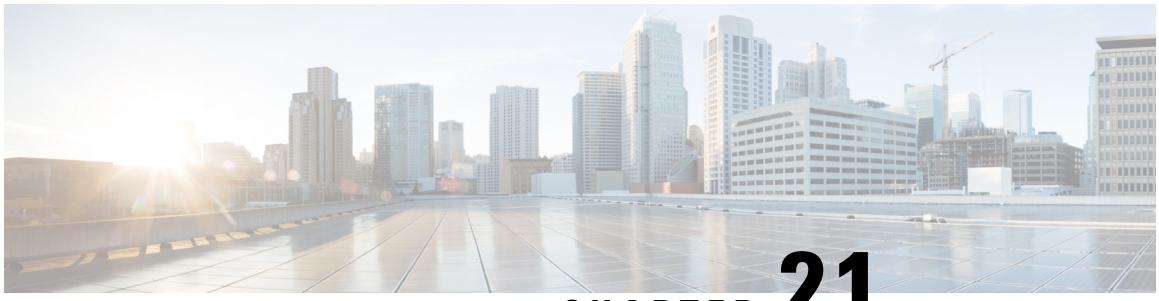
- This will retrigger the SYNC message from the cloud to connector for all the controllers that are added.
- The connector attempts to reestablish the connection between the connector and the controller.
- If the NMSP handshake fails after importing the fipsca certificate into the controller, re-import the certificate. To clean any trustpool certificates imported earlier, use the **crypto pki trustpool import clean** command.



PART III

Troubleshooting

- Troubleshooting Tools, on page 165
- Troubleshooting Scenarios, on page 169



CHAPTER 21

Troubleshooting Tools

- Enable Debug Logs, on page 165
- Resetting the Password, on page 166
- Monitor Service Metrics, on page 166

Enable Debug Logs

This task shows you how to enable debug logs for connector. The task also shows you how to upload these logs to Cisco Spaces, if necessary.

Note You can also enable debug log using the **connectorctl service restart** command.

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left navigation pane, choose **Setup > Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed.

Step 5 In the **SUMMARY** window that is displayed, click **Troubleshoot Connector**.

Step 6 In the **Troubleshoot Connector** window that is displayed, you can see that logs can be enabled by a service. Click the respective **Enable Debug Mode** of a service if not enabled already.

After being enabled, connector starts collecting debug logs for that service, and these logs are stored locally on connector.

Step 7 (Optional) To upload the logs to the Cisco Spaces dashboard, click **Upload Logs to Cloud.**

Resetting the Password

This task shows you how to recover your connector GUI password.

Procedure

Step 1 Log in to **Cisco Spaces.**

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left navigation pane of the Cisco Spaces dashboard, choose **Setup > Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed.

Step 5 In the **SUMMARY** window that is displayed, click **Troubleshoot Connector**.

Step 6 In the **Troubleshoot Connector** window that is displayed, click **Password Reset Key** .

Step 7 In the **Password Reset Key** window that is displayed, click **Copy The Key**.
Save the copied key on a notepad.

Step 8 Open the connector GUI, and click **Forgot Password**.

Step 9 In the **Password Reset Key** field, enter the key copied in theprevious step.

Step 10 In the **New Password** field, enter a new password.

Monitor Service Metrics

You can monitor the various metrics of the different services that are installed on connector from the Cisco Spaces dashboard.

Procedure

Step 1 From the Cisco Spaces dashboard, navigate to **Setup > Wireless Networks**.

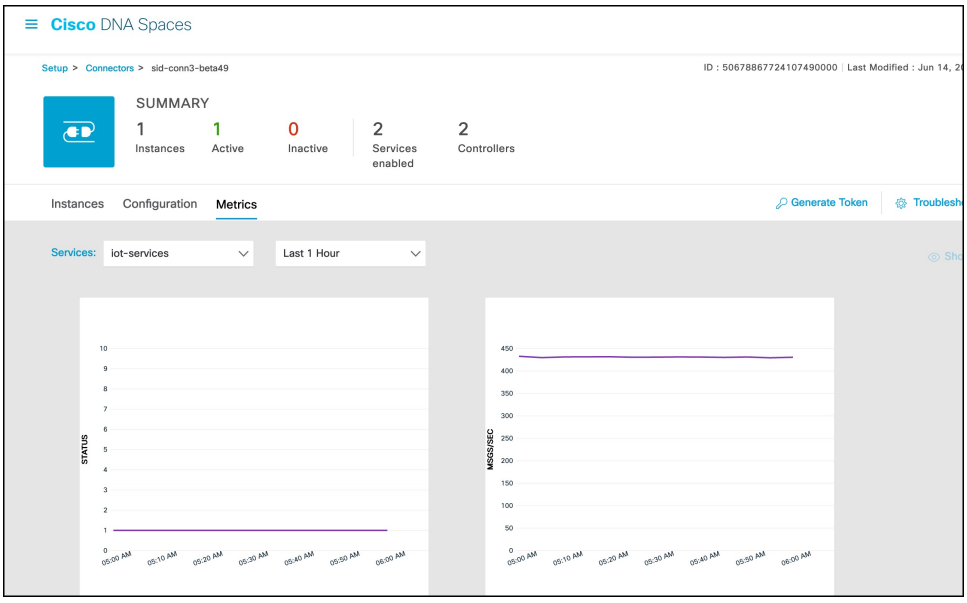
Step 2 In the **Connect via Spaces Connector** area titled **Step 2 Configure Spaces Connector**, click **View Connectors**.

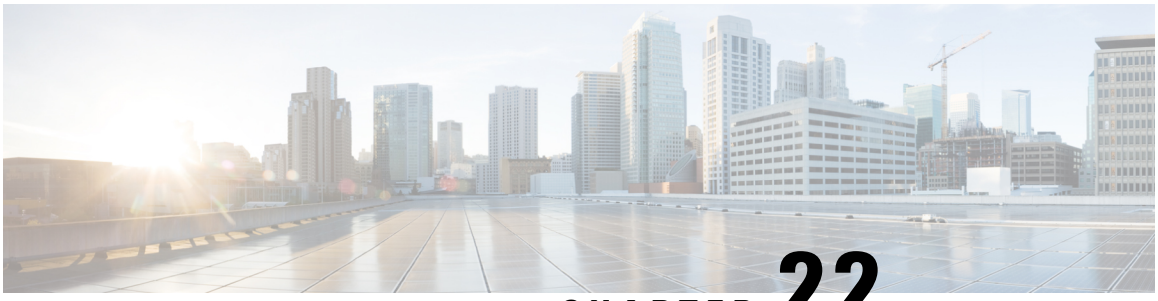
Step 3 In the **Connectors** window that opens up, click a connector of your choice.

Step 4 In the connector details window that is displayed, click the **Metrics** tab.

Step 5 From the **Services** drop-down list, choose a service that is installed on this connector to observe the metrics that are related to the service. You can also choose the period for which the metrics is collected.

Figure 170: Observing Service Metrics





CHAPTER 22

Troubleshooting Scenarios

- Connectivity Issues Between Connector and Cisco Spaces, on page 169
- Unresponsive Connector, or Failure of SSH to Connector, on page 173
- Instance is Corrupted or Deleted , on page 176
- Service Crash, or Restart Services , on page 176
- Upgrade has Failed, or How To Forcibly Push Configurations to Instances, on page 177
- Managing Weak Algorithms from SSHD Configurations, on page 178

Connectivity Issues Between Connector and Cisco Spaces

This task allows you to troubleshoot connectivity issues between your connector and Cisco Spaces. You can troubleshoot this connection both before and after the configuration of the connector token on Cisco Spaces.

Procedure

Step 1 Log in to the connector GUI.

Step 2 In the connector left navigation pane, click **Troubleshoot** and do one of the following:

- If you have configured the token for this connector in Cisco Spaces, the text field beside the **Run New Test** button is automatically populated with the Cisco Spaces URL.
- If you have not configured the token for this connector on Cisco Spaces, then from the **Run New Test** drop-down, choose from one of the Cisco Spaces region-dependent URLs.

Step 3 Click **Run New Test** to initiate troubleshooting the connectivity.

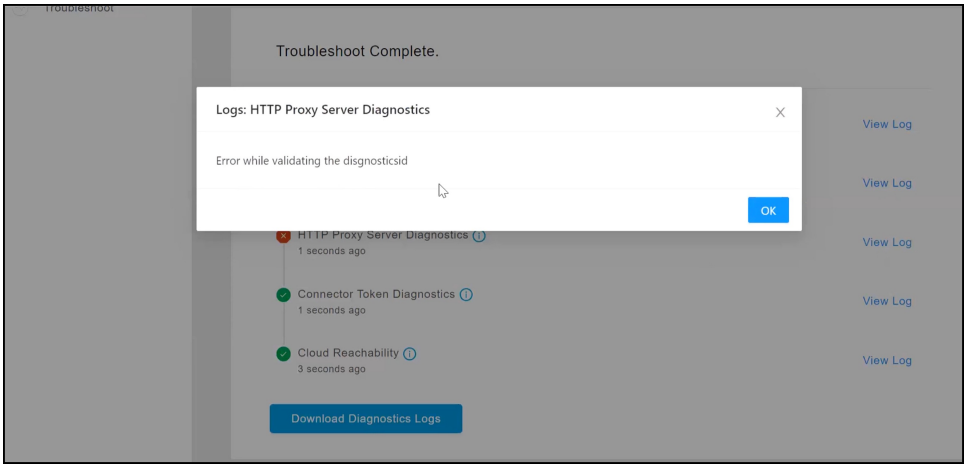
Step 4 Observe the running tests for the following:

ⓘ	Click to view further information about the test.



Click **View Logs** to view further information.

Figure 171: View Logs



Represents a successful test. Click ⓘ to view additional information about this successful test.

Figure 172: View Logs for a Successful Test in the DNA spaces site

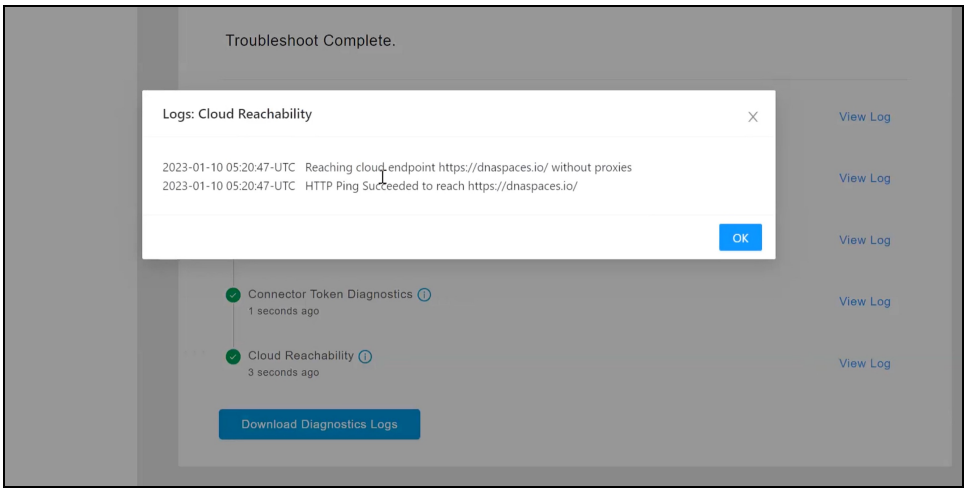
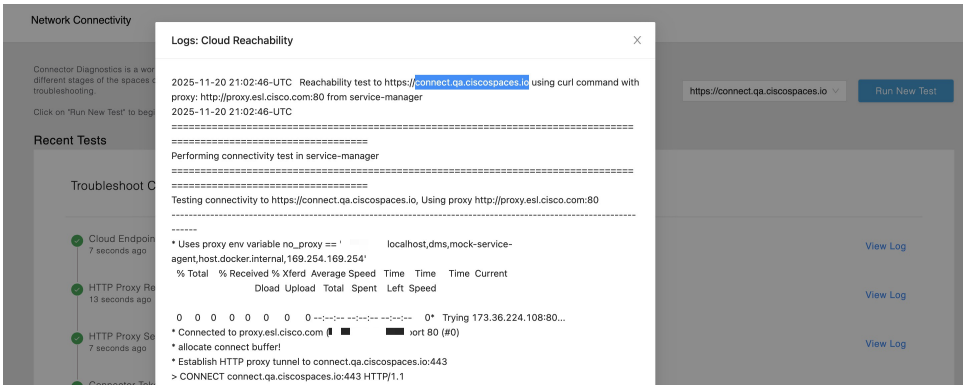
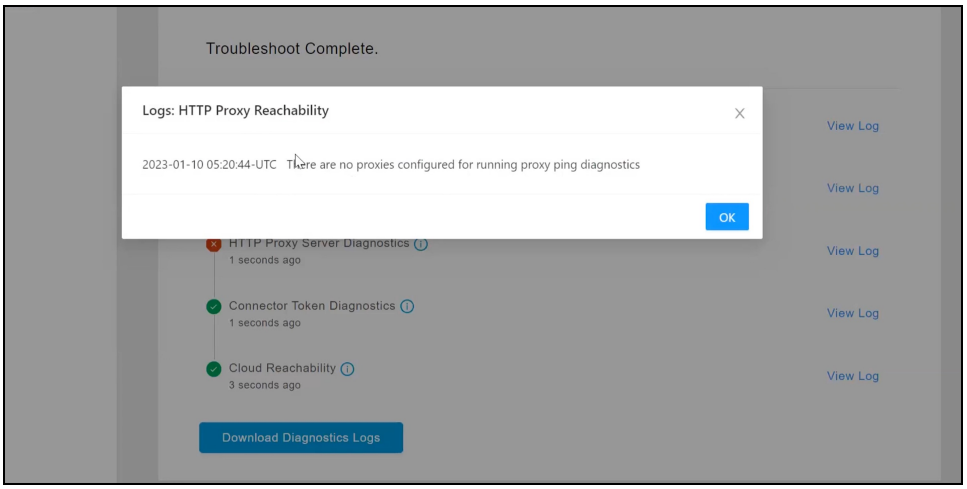


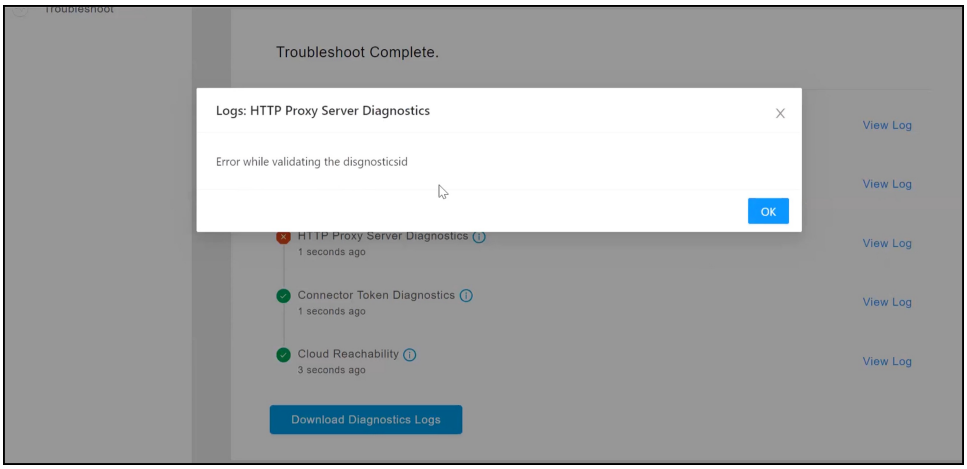
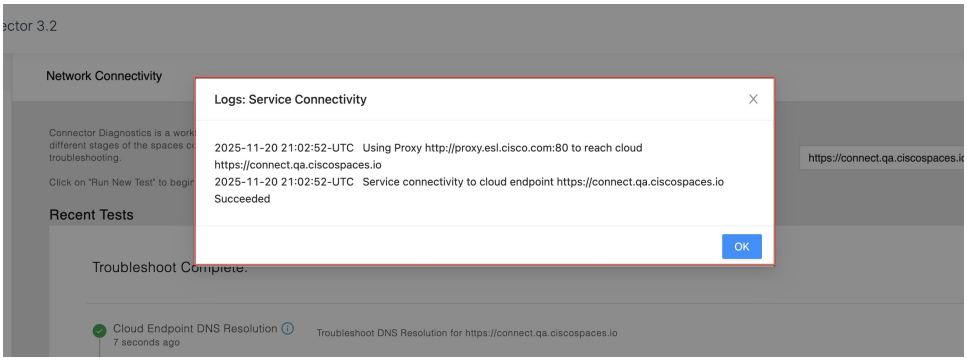
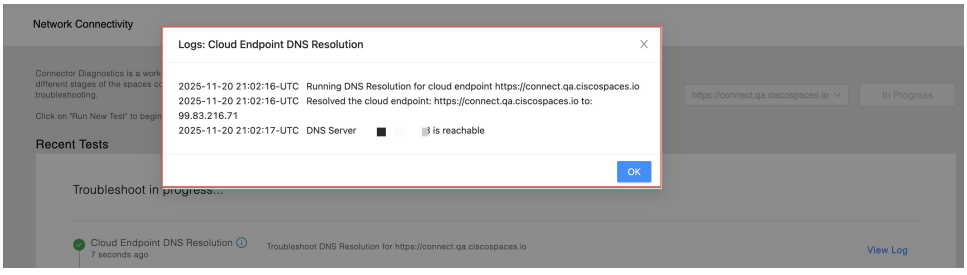
Figure 173: View Logs for a Successful Test in the Cisco spaces site



Represents a warning. Click to view additional information about this warning.

Figure 174: View Logs for a Warning



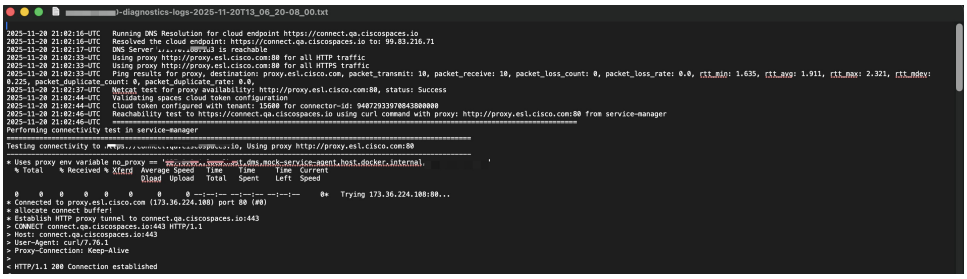
	Represents a failure in the diagnostic test. Click **View Logs** to see additional details. ***Figure 175: View Logs for a Successful Test*** 
	Represents a failure in the diagnostic test. Click **View Logs** to see additional details. ***Figure 176: View Logs for a Successful Service Connectivity*** 
	Represents a failure in the diagnostic test. Click **View Logs** to see additional details. ***Figure 177: View Logs for a Successful Cloud Endpoint DNS Resolution*** 

Step 5 Click **Download Diagnostic Logs** to download a text file with details of logs, including diagnostic information.

Figure 178: Download Diagnostic Logs in the DNA Spaces site



Figure 179: Download Diagnostic Logs in the Cisco Spaces site



What to do next

You can also use the connector CLI to troubleshoot connectivity issues between the connector and the Cisco Spaces dashboard. See the command **connectorctl troubleshooting connectivity** in the Cisco Spaces: Connector 3 Command Reference Guide.

Unresponsive Connector, or Failure of SSH to Connector

If a connector is unresponsive to SSH requests, reboot the device on which the connector OVA is installed. You can do this from the Cisco Spaces dashboard .

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left navigation pane, choose **Setup > Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed, and then click the **Instances** tab.

Step 5 In the **Actions** column, click the three-dot icon to open a list of options for the connector instance, and choose **Restart Connector**.

Figure 180: Restart Connector

Instance is Corrupted or Deleted

You may have to delete a connector instance for one of the following reasons:

- An instance is not required anymore.
- An instance is corrupted or invalid.

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 In the left navigation pane of the Cisco Spaces dashboard, choose **Setup** > **Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed and then click the **Instances** tab.

Step 5 In the **Actions** column, click the three-dot icon to open a list of options for the connector instance, and choose **Remove**.
To create a new instance, do the following.

a. In the Cisco Spaces dashboard, reissue a token.

b. Configure the new token on the installed connector.

See Activating Connector 3 on Cisco Spaces, on page 28.

Service Crash, or Restart Services

This task shows you how to restart a service on a connector when the service crashes or hangs.

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

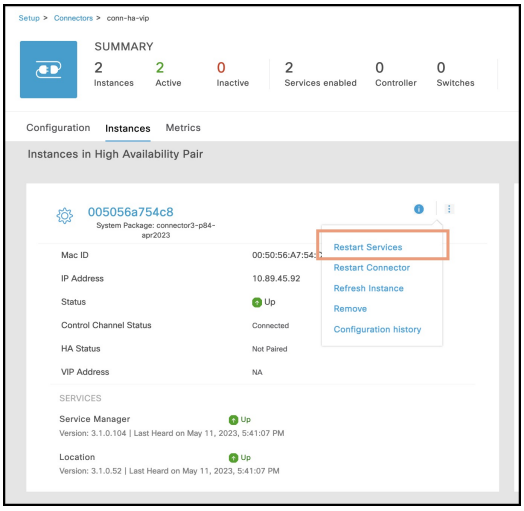
Step 2 From the left navigation pane of the Cisco Spaces dashboard, choose **Setup** > **Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed, and then click the **Instances** tab.

Step 5 In the **Actions** column, click the three-dot icon to open a list of options for the connector instance, and choose **Restart Services.**

Figure 181: Restart Services



Upgrade has Failed, or How To Forcibly Push Configurations to Instances

If a service upgrade fails and a connector instance does not receive Cisco Spaces configurations, you can forcibly push configurations to the instance using this procedure.

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 From the left-navigation pane, choose **Setup > Wireless Networks**.

Step 3 In the **2. Configure Spaces Connector** area, click **View Connectors**.

Step 4 Click a connector from the list of connectors that are displayed, and then click the **Instances** tab.

Step 5 In the **Actions** column, click the three-dot icon to open a list of options for the connector instance, and choose **Refresh Instance**.

Managing Weak Algorithms from SSHD Configurations

Weak SSHD Key Exchange (kex) Algorithms

Key exchange algorithms are used to securely exchange cryptographic keys between the client and the server over an insecure network. This ensures that the communication remains private and secure.

However, some SSHD Key Exchange algorithms are considered weak for many reasons.

Here are a few reasons:

- Older key exchange methods using smaller key sizes (768 bits).
- Some key exchange algorithms do not support perfect forward secrecy. This means that if a private key is compromised, past communications encrypted with that key could potentially be decrypted.

Disabling weak SSHD Key Exchange algorithms is essential to improve the security of your SSH server.

Disable Weak SSHD Key Exchange (kex) Algorithms

Procedure

Step 1 Display the list of Key Exchange algorithms using the **connectorctl sshd kex show** command. Observe that this list includes SSHD Key Exchange (kex) algorithms that may be considered weak (weak SSHD Key Exchange algorithms) for different reasons.

```
[spacesadmin@connector ~]$ connectorctl sshd kex show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported Key Exchange algorithms is:
kexalgorithms
ecdh-sha2-nistp256,
ecdh-sha2-nistp384,
ecdh-sha2-nistp521,
diffie-hellman-group14-sha1,
diffie-hellman-group14-sha256,
diffie-hellman-group16-sha512
```

Step 2 To remove support for weak SSHD Key Exchange algorithms from this device, use the **connectorctl sshd kex remove** command. Run the **connectorctl sshd kex show** command to verify that weak SSHD Key Exchange algorithms are removed from the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd kex remove -a
Executing command:sshd
Command execution status:Success
-----------------------
Removing all unsupported weak algorithms
Successfully removed -diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1 key exchange
algorithm(s)
```

```
[spacesadmin@connector ~]$ connectorctl  sshd kex show
Executing command:sshd
```

Step 3 To reinstate support for weak SSHD Key Exchange algorithms on this device, use the **connectorctl sshd kex reset** command. Run the **connectorctl sshd kex show** command to verify that weak SSHD Key Exchange algorithms are added back to the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd kex reset
Executing command:sshd
Command execution status:Success
-----------------------
Successfully reset Key Exchange algorithms configuration

[spacesadmin@connector ~]$ connectorctl  sshd kex show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported Key Exchange algorithms is:
kexalgorithms
ecdh-sha2-nistp256,
ecdh-sha2-nistp384,
ecdh-sha2-nistp521,
diffie-hellman-group14-sha1,
diffie-hellman-group14-sha256,
diffie-hellman-group16-sha512
```

Weak Host Key (hostkey) Algorithms

Host key algorithms are used to verify the server's identity to the client. The server uses its private key to authenticate itself, and the client uses the server's public key to verify this identity.

However, some Host Key algorithms are considered weak for many reasons. Here are a few reasons:

- Short key length (RSA keys that are less than 2048 bits).
- Outdated algorithms (Digital Signature Algorithms [DSA] is limited to a maximum key size of 1024 bits).
- Weak hash functions (MD5 or SHA-1) can compromise the security of the entire key exchange process.

Disabling weak Host Key algorithms is essential to secure SSH connections and prevent potential vulnerabilities.

Disable Weak Host Key (hostkey) Algorithms

Procedure

Step 1 Display the list of Host Key algorithms using the **connectorctl sshd hostkey show** command.

```
[spacesadmin@connector ~]$ connectorctl  sshd hostkey show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported host Key algorithms is:
hostkeyalgorithms *
```

Step 2 To remove support for weak Host Key algorithms from this device, use the **connectorctl sshd hostkey remove** command. Run the **connectorctl sshd hostkey show** command to verify that weak Host Key algorithms are removed from the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd hostkey remove -a
Executing command:sshd
Command execution status:Success
-----------------------
Removing all unsupported weak algorithms
Successfully removed
x509v3-ecdsa-sha2-nistp256,
x509v3-ecdsa-sha2-nistp384,
x509v3-ecdsa-sha2-nistp521,
x509v3-ssh-rsa,
x509v3-rsa2048-sha256,
x509v3-sign-rsa,,,
ssh-ed25519,,
ecdsa-sha2-nistp256,
ecdsa-sha2-nistp384,
ecdsa-sha2-nistp521,
ssh-rsa key exchange algorithm(s)
```

Step 3 To reinstate support for weak Host Key algorithms on this device, use the **connectorctl sshd hostkey reset** command. Run the **connectorctl sshd hostkey show** command to verify that weak Host Key algorithms are added back to the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd hostkey reset
Executing command:sshd
Command execution status:Success
-----------------------
Successfully reset key exchange algorithms configuration
```

Weak Cipher Algorithms

Cipher algorithms are used to encrypt data transmitted between the client and the server, ensuring confidentiality.

However, some cipher algorithms are considered weak for many reasons. Here are a few reasons:

- Ciphers with short key lengths are susceptible to force attacks.
- Some ciphers (such as RC4) have biases in the output that make it susceptible to cryptanalysis.
- Weak algorithm structure (use of small block sizes can lead to vulnerabilities when encrypting large amount of data).
- Even strong ciphers can be rendered weak, if they are implemented poorly.

Disabling weak cipher algorithms is essential to ensure secure communications.

Disable Cipher Algorithms

Procedure

Step 1 Display the list of cipher algorithms using the **connectorctl sshd cipher show** command.

```
[spacesadmin@connector ~]$ connectorctl  sshd cipher show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported Cipher algorithms is:
ciphers chacha20-poly1305@openssh.com,
aes128-ctr,
aes192-ctr,
aes256-ctr,
aes128-gcm@openssh.com,
aes256-gcm@openssh.com
```

Step 2 To remove support for weak cipher algorithms from this device, use the **connectorctl sshd cipher remove** command. Run the **connectorctl sshd cipher show** command to verify that weak cipher algorithms are removed from the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd cipher remove -a
Executing command:sshd
Command execution status:Success
-----------------------
Removing all unsupported cipher algorithms
Successfully removed -3des-cbc key exchange algorithm(s)
```

Step 3 To reinstate support for weak cipher algorithms on this device, use the **connectorctl sshd cipher reset** command. Run the **connectorctl sshd cipher show** command to verify that weak cipher algorithms are added back to the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd cipher reset
Executing command:sshd
Command execution status:Success
-----------------------
Successfully reset Cipher algorithms configuration
```

Weak Message Authentication Code (MAC) Algorithms

Message Authentication Code (MAC) algorithms are used to ensure the integrity and authenticity of the transmitted data, protecting it against tampering.

However, some MAC algorithms are considered weak for many reasons. Here are a few reasons:

- Many MAC algorithms, like HMAC, rely on hash functions to generate message digests. If the underlying hash function is weak (such as MD5 or SHA-1), the MAC algorithm inherits these weaknesses.
- Insufficient key length (shorter keys are more susceptible to attacks).
- Predictable key management (security of MAC algorithm is compromised with predictable keys).

Disabling weak MAC algorithms is essential to maintain the integrity and authenticity of communications.

Disable Message Authentication Code (MAC) Algorithms

Procedure

Step 1 Display the list of MAC algorithms using the **connectorctl sshd mac show** command.

```
[spacesadmin@connector ~]$ connectorctl sshd mac show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported MAC algorithms is:
macs hmac-sha2-256-etm@openssh.com,
hmac-sha2-512-etm@openssh.com,
hmac-sha2-256,
hmac-sha2-512,
hmac-sha1
```

Step 2 To remove support for weak MAC algorithms from this device, use the **connectorctl sshd mac remove** command. Run the **connectorctl sshd mac show** command to verify that weak MAC algorithms are removed from the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl sshd mac remove -a
Executing command:sshd
Command execution status:Success
-----------------------
Removing all unsupported weak mac algorithms
Successfully removed -umac-64-etm@openssh.com,
hmac-sha1-etm@openssh.com,
umac-64@openssh.com,
hmac-sha1 key exchange algorithm(s)

Successfully removed weak MAC configuration
```

Step 3 To reinstate support for weak MAC algorithms on this device, use the **connectorctl sshd mac reset** command. Run the **connectorctl sshd mac show** command to verify that weak MAC algorithms are added back to the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl sshd mac reset
Executing command:sshd
Command execution status:Success
-----------------------
Successfully reset weak MAC configuration
```

Weak Public Key (pubkey) Algorithms

Public Key algorithms are used for user authentication, allowing users to log in without a password by proving ownership of a private key.

However, some Public Key algorithms are considered weak for many reasons. Here are a few reasons:

- Insufficient key length (RSA keys less than 2048 bits are considered insecure).
- Random number generation process can lead to predictable keys that are easier to attack.

Disable Public Key (pubkey) Algorithms

Procedure

Step 1 Display the list of Public Key algorithms using the **connectorctl sshd pubkey show** command.

```
[spacesadmin@connector ~]$ connectorctl  sshd pubkey show
Executing command:sshd
Command execution status:Success
-----------------------
List of supported Public key algorithms is:
pubkeyalgorithms *
```

Step 2 To remove support for weak Public Key algorithms from this device, use the **connectorctl sshd pubkey remove** command. Run the **connectorctl sshd pubkey show** command to verify that weak Public Key algorithms are removed from the supported algorithm list.

```
[spacesadmin@connector ~]$ connectorctl  sshd pubkey remove -a
Executing command:sshd
Command execution status:Success
-----------------------
Removing all unsupported weak algorithms
Successfully removed x509v3-ecdsa-sha2-nistp256,
x509v3-ecdsa-sha2-nistp384,
x509v3-ecdsa-sha2-nistp521,
x509v3-ssh-rsa,
x509v3-rsa2048-sha256,
x509v3-sign-rsa,,,
ssh-ed25519,
ssh-ed25519-cert-v01@openssh.com,,
ecdsa-sha2-nistp256,
ecdsa-sha2-nistp384,
ecdsa-sha2-nistp521,
ssh-rsa-cert-v01@openssh.com,
rsa-sha2-256-cert-v01@openssh.com,
rsa-sha2-512-cert-v01@openssh.com,
-cert-v01@openssh.com,
ecdsa-sha2-nistp256-cert-v01@openssh.com,
ecdsa-sha2-nistp384-cert-v01@openssh.com,
ecdsa-sha2-nistp521-cert-v01@openssh.com,
ssh-rsa,
rsa-sha2-256,
rsa-sha2-512 key exchange algorithm(s)
```

Step 3 To reinstate support for weak Public Key algorithms on this device, use the **connectorctl sshd pubkey reset** command. Run the **connectorctl sshd pubkey show** command to verify that weak Public Key algorithms are added back to the supported algorithm list.

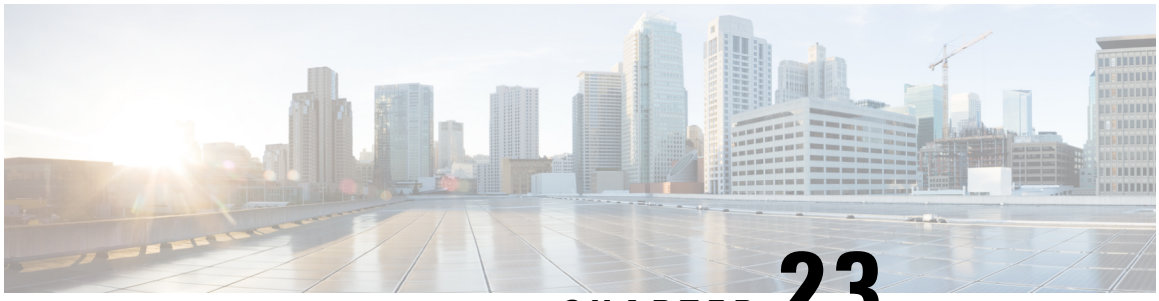
```
[spacesadmin@connector ~]$ connectorctl  sshd pubkey reset
Executing command:sshd
Command execution status:Success
-----------------------
Successfully reset key exchange algorithms configuration
```



PART IV

Services

- Gateway Service with Sensor Connect, on page 187
- Location Service, on page 191
- IoT Service (Wireless) , on page 197
- Connector Gateway with Sensor Connect, on page 211
- IoT Service (Wired) , on page 215
- Hotspot Service, on page 233
- Local Firehose, on page 237



CHAPTER 23

Gateway Service with Sensor Connect

- Gateway Service overview and operating model, on page 187
- Gateway Service Operating Principles, on page 187
- Configure Gateway Service with Sensor Connect, on page 188
- Gateway Service Components, on page 189

Gateway Service overview and operating model

The Cisco Spaces: Gateway Service is an edge service that runs on Cisco Spaces: Connector 3. It connects Cisco wireless infrastructure, including access points and Cisco Catalyst 9800 Series Wireless Controllers, to the Cisco Spaces cloud.

Gateway Service receives telemetry from access points and wireless controllers. It forwards telemetry to Cisco Spaces and supports local telemetry distribution to partner applications. Gateway Service serves as the primary telemetry ingestion service for AP data on Connector 3.

Note This service works with Sensor Connect. For more information, refer to Connector Gateway with Sensor Connect and Cisco Sensor Connect for IoT Services.

Gateway Service supports the following telemetry and management workflows:

- Receives BLE, UWB ad GNSS telemetry from access points through gRPC streams.
- Receives TDL telemetry subscription data from Cisco Catalyst 9800 Series Wireless Controllers.
- Forwards telemetry to Cisco Spaces through WebSocket data channels.
- Sends telemetry locally to partner applications through Local Firehose.

Gateway Service Operating Principles

Each supported access point establishes one gRPC session to the Gateway Service. The gRPC session uses bidirectional streaming and requires certificate validation and token-based authentication.

Cisco Catalyst 9800 Series Wireless Controllers are used to deliver gRPC configuration to access points and to send TDL telemetry to the Gateway Service. The connector uses NETCONF to onboard controllers and configure the required certificates and telemetry settings.

In high availability deployments, two Gateway Service instances share a virtual IP address. The active instance handles gRPC and cloud traffic. The backup instance monitors health and takes over if the active instance fails.

Configure Gateway Service with Sensor Connect

Use this procedure to add the Gateway Service to Cisco Spaces: Connector 3 and to onboard a Cisco Catalyst 9800 Series Wireless Controller.

Before you begin

Ensure that these prerequisites are met.

- Cisco Spaces: Connector 3 July 2026 OVA and above is deployed.
- The connector has network reachability to Cisco Spaces.
- The connector can reach a DNS server.
- If you use an explicit proxy, Connector 3 can communicate through it.
- The Cisco Catalyst 9800 Series Wireless Controller has NETCONF enabled and the minimum required version for Controller is 26.1.2.
- The controller is reachable from the connector.
- Firewall rules allow the required Cisco Spaces endpoints and ports.



Note **service-manager** is selected by default. Existing IoT Wireless Customers need to remove IoT Service before adding Gateway Service.

Procedure

Step 1 In the Cisco Spaces dashboard, click **Setup**, and choose **Wireless Networks**.

Step 2 In the Connect your wireless network window, go to **Step 2**, and click **View Connectors**.

Step 3 In the connector details window, click **Add Services**.

Step 4 In the **Add Services** window, choose **Gateway**, and click **Add**.

Step 5 After the **Gateway** service is installed and running, click **Add Controller** service.

Step 6 Enter the Cisco Catalyst 9800 Series Wireless Controller details in the required fields:

- **Controller IP Address**
- **Username**
- **Password**

Step 7 Submit the details to onboard the controller.

The connector onboards the controller through NETCONF.

Step 8 Verify the onboarding status from the connector CLI:

For example,

```
connectorctl -s gateway onboarding list
  {
    "ctlrConnectId": "ee1e9eb9-b56d-40bf-825a-d5d6cf256eeb",
    "host": "172.19.32.20",
    "status": "SUCCESS",
    "currentStep": "COMPLETE",
    "attempts": 1,
    "otpStatus": "CONFIGURED",
    "otpMessage": "OTP installed successfully on controller",
    "netconfStatus": "SUCCESS",
    "netconfMessage": "Connection with eWLC is successful"
  }
connectorctl -s gateway onboarding status -i <ctlrConnectId>
{
  "ctlrConnectId": "ee1e9eb9-b56d-40bf-825a-d5d6cf256eeb",
  "host": "172.19.32.20",
  "status": "SUCCESS",
  "currentStep": "COMPLETE",
  "metrics": "OnboardingMetrics{host='172.19.32.20', ctlrConnectId='ee1e9eb9-...', attempts=1,
duration=15234ms, status=SUCCESS, failure='null'}"
}
```

Gateway Service Components

Table 14: Gateway Service Components

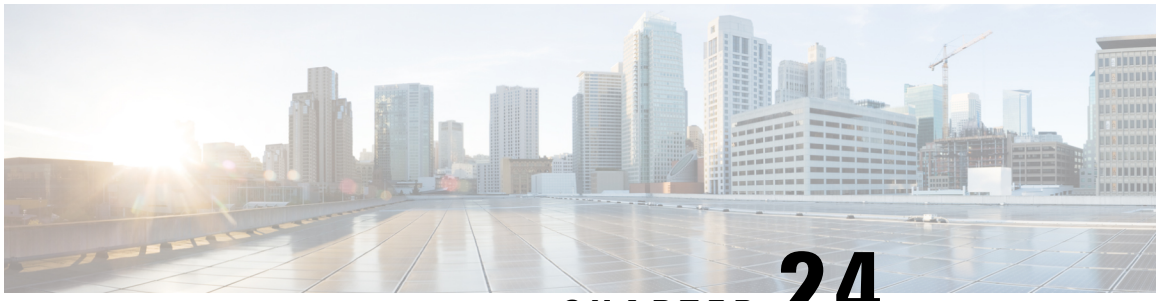
Component	Description
Access Points	Supported Cisco access points establish gRPC sessions with Gateway Service and send BLE and UWB.
Cisco Catalyst 9800 Series Wireless Controllers	Controllers provide AP gRPC configuration through CAPWAP and send TDL telemetry through gRPC Model-Driven Telemetry.
Cisco Spaces: Connector	Hosts Gateway Service as a Docker container and manages onboarding, telemetry forwarding, status reporting, and local data distribution.
Cisco Spaces Cloud	Receives telemetry from Gateway Service through WebSocket data channels.

Table 15: Supported Connector Versions

Component	Minimum Version
Cisco Spaces: Connector OVA	July2026 OVA and later
Cisco Spaces: Connector Docker	3.3 and later

Supported Access Points

Gateway Service supports gRPC telemetry streams from all the access points that are currently supported by IoT Wireless.



CHAPTER 24

Location Service

- Compatibility Matrix for Cisco Spaces: Connector: Location service, on page 191
- Open Ports for Location Service, on page 195

Compatibility Matrix for Cisco Spaces: Connector: Location service

This section covers the following:

- Location Service (Non-FIPS)
- Location Service (FIPS)

Location Service (Non-FIPS)

Table 16: Location Service (Non-FIPS)

Hardware or Application Name	Support for Cisco Spaces: Connector
Cisco Catalyst 9800 Series Wireless Controllers	• Supported on 17.6.8 and later releases. **Note** • Use the latest software version or maintenance release for each listed release. See Recommended Cisco IOS XE Releases for Catalyst 9800 Wireless LAN Controllers. • 16.12.8 and 17.3.8a are end-of-life (EOL). We recommend that you migrate to one of the recommended releases as per the Guidelines for Cisco Wireless Software Release Product Bulletin.

Hardware or Application Name	Support for Cisco Spaces: Connector
Cisco AireOS Wireless Controller	**Note** • Use the latest software or maintenance release version for each listed release. See Recommended AireOS Wireless LAN Controller Releases. • 8.3, 8.5, and 8.8 are end-of-life (EOL). We recommend that you migrate to one of the recommended releases as per the Guidelines for Cisco Wireless Software Release Product Bulletin. • 8.10 is end-of-life (EOL).
Cisco Embedded Wireless Controller on Cisco Catalyst Access Points (Cisco EWC-AP)	Supported versions are: • 17.6.1 **Note** • Use the latest software or maintenance release version for ea • Cisco Wireless Embedded Wireless Controller (EWC) on Ac announces the end-of-sale and end-of-life dates for the Cisco 17.15.x will be the final IOS-XE software supporting EWC-A EWC running on Catalyst switches in Software Defined Acc For more information, see https://www.cisco.com/c/en/us/products/collateral/wireless/emb Supported access points are: • Cisco Catalyst 9115 Series Access Points • Cisco Catalyst 9117 Series Access Points • Cisco Catalyst 9120 Series Access Points • Cisco Catalyst 9130 Series Access Points
Cisco Catalyst 9300 and 9400 Series Switches	Supported versions are 17.3.3 and later
Cisco Prime Infrastructure	Supported
Catalyst Center	Supported
Supported wireless controllers for Cisco FastLocate	• Supported on all releases of Cisco Catalyst 9800 Series Wireless Controllers
Supported wireless controllers for Cisco Hyperlocation	• Supported on Cisco Catalyst 9800 Series Wireless Controllers

Hardware or Application Name	Support for Cisco Spaces: Connector
Connector Active-Active Mode	• Not supported on Cisco Embedded Wireless Controller on Cisco Catalyst Access Points (Cisco EWC-AP) • Supported on Cisco Catalyst 9800 Series Wireless Controllers • Supported on Cisco AireOS Wireless Controller
Tested VMware Environments	• VMware vSphere Client Version 7.0.x and 8.0 • VMware vCenter Server Appliance 7.0.x and 8.0

Tested Proxies	• Squid proxy
	  • Forward-only mode (SSL tunneling)
	  • Squid-in-the-middle mode (SSL tunneling with intercept capabilities)
	• McAfee
	• Cisco web security appliance

Tested Access Points for Cisco FastLocate	• Cisco Aironet 2800 Series Access Points • Cisco Aironet 3800 Series Access Points • Cisco Aironet 4800 Series Access Points
Tested Access Points for Cisco FastLocate (Wi-Fi 7)	• Cisco Wireless 9178 Series Access Points • Cisco Wireless 9176 Series Access Points
Tested Access Points for Cisco FastLocate (Wi-Fi 6)	• Cisco Catalyst 9120 Series Access Points • Cisco Catalyst 9130 Series Access Points • Cisco Catalyst 9164 Series Access Points • Cisco Catalyst 9166 (I/D1) Series Access Points • Cisco Catalyst IW9167I Heavy Duty Access Points
Tested Access Points for Cisco Hyperlocation	• Cisco Aironet 4800 Series Access Point

Hardware or Application Name	Support for Cisco Spaces: Connector
Tested Access Points	• Cisco Catalyst 9105AX (I/W) Series Access Points • Cisco Catalyst 9115AX (I/E) Series Access Points • Cisco Catalyst 9117AX (I) Series Access Points • Cisco Catalyst 9136 (I) Series Access Points • Cisco Catalyst 9162 (I) Series Access Points • Cisco Catalyst 9164 (I) Series Access Points • Cisco Catalyst 9166 (I/D1) Series Access Points • Cisco Catalyst IW9167 (E/I) Heavy Duty Series Access Points • Cisco Catalyst IW9165D Heavy Duty Access Points • Cisco Catalyst IW9165E Rugged Access Points • Cisco Wireless 9172 Series Access Points

Location Service (FIPS)

Table 17: Location Service (FIPS)

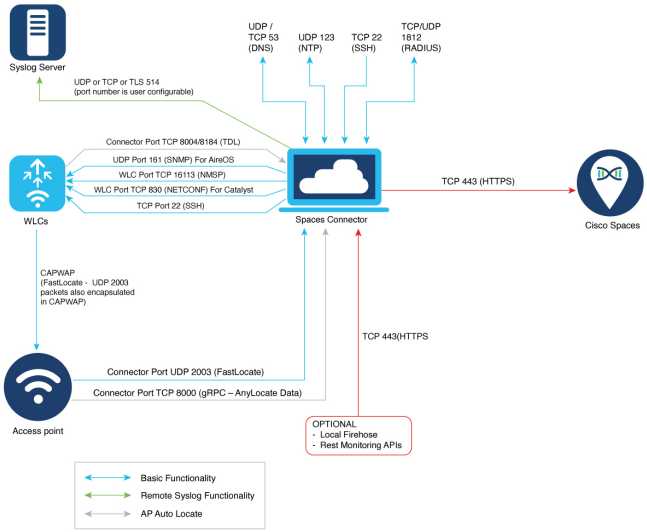
Hardware or Application Name	Support for Cisco Spaces: Connector
Cisco Catalyst 9800 Series Wireless Controllers	• 17.15.3 • 17.12.6 **Note** Use the latest software version or maintenance release for each listed release. See Recommended Cisco IOS XE Releases for Catalyst 9800 Wireless LAN Controllers.
Tested Access Points for Cisco FastLocate (Wi-Fi 7)	• Cisco Wireless 9178 Series Access Points • Cisco Wireless 9176 Series Access Points

Hardware or Application Name	Support for Cisco Spaces: Connector
Tested Access Points for Cisco FastLocate (Wi-Fi 6)	• Cisco Catalyst 9120 Series Access Points • Cisco Catalyst 9130 Series Access Points • Cisco Catalyst 9164 Series Access Points • Cisco Catalyst 9166 (I/D1) Series Access Points • Cisco Catalyst IW9167I Heavy Duty Access Points
Tested Access Points	• Cisco Catalyst 9105AX (I/W) Series Access Points • Cisco Catalyst 9115AX (I/E) Series Access Points • Cisco Catalyst 9117AX (I) Series Access Points • Cisco Catalyst 9136 (I) Series Access Points • Cisco Catalyst 9162 (I) Series Access Points • Cisco Catalyst 9164 (I) Series Access Points • Cisco Catalyst 9166 (I/D1) Series Access Points • Cisco Catalyst IW9167 (E/I) Heavy Duty Series Access Points • Cisco Catalyst IW9165D Heavy Duty Access Points • Cisco Catalyst IW9165E Rugged Access Points • Cisco Wireless 9172 Series Access Points

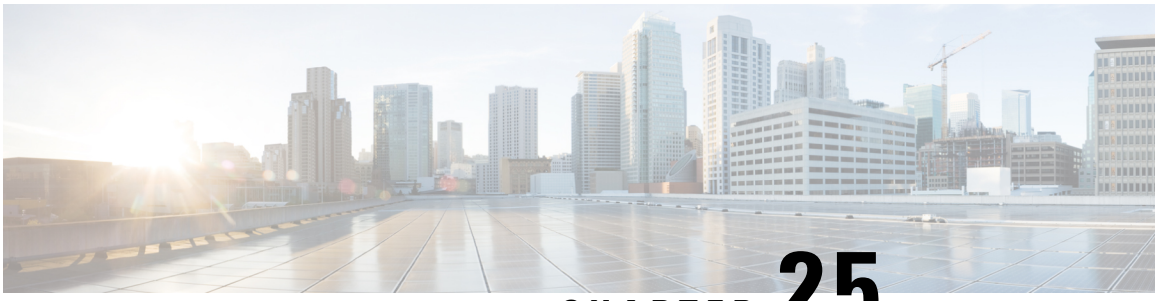
Open Ports for Location Service

This section lists the connector ports that must be open for the proper functioning of location service.

Figure 182: Open Ports for Location Service



Test the connectivity between the connector and the wireless controller. See Configure and Test Connectivity between the Connector 3 and AireOS controller or Configure and Test the Connectivity between a Connector 3 and a Catalyst 9800 controller.



CHAPTER 25

IoT Service (Wireless)

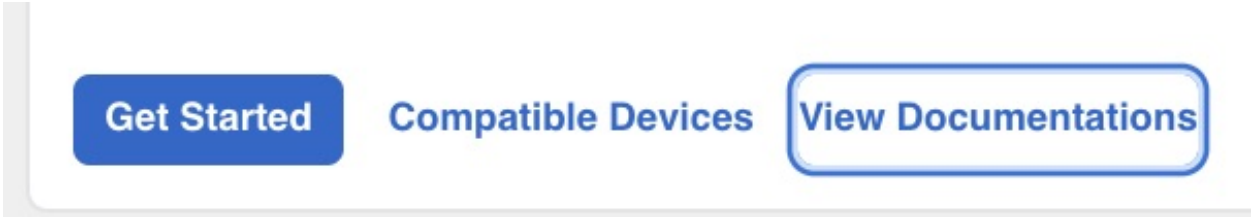
- Overview of Cisco Spaces: IoT Service (Wireless), on page 197

Overview of Cisco Spaces: IoT Service (Wireless)

Cisco Spaces: IoT Service (Wireless) is a platform service within Cisco Spaces that enables you to claim, manage, and monitor IoT devices using Cisco's wireless infrastructure. IoT Service is designed to enable management of IoT devices across vendors, form factors, and technology protocols. Bluetooth Low Energy (BLE) is the first technology available for management using IoT services.

IoT service (wireless) encompasses hardware, software, and partner components to enable the management of devices that support critical business outcomes. IoT service (wireless) uses Cisco Catalyst 9800 Series Wireless Controllers, Cisco Spaces: Connector, Cisco Wi-Fi6 access points, and Cisco Spaces. IoT service (wireless) adopts a next-generation approach to manage complexity in an enterprise environment.

Figure 183: Spaces IoT in a nutshell tab options



Click the **Get Started** button to learn more about getting started. This action redirects to the Cisco Spaces: IoT Service Configuration Guide (Wireless).

Click the **Compatible Devices** button to view the list of supported devices. This action redirects to the Cisco Spaces IoT Device Marketplace website.

Click the **View Documentation** button to see how to set up the product. This action redirects to the Cisco Spaces Setup Guide.

Using the IoT service (wireless), you can perform these IoT management activities:

- Deploy BLE gateways on supported APs in your network.
- Claim the BLE beacons that you acquired from Cisco Spaces: IoT Device Marketplace.
- Configure APs and manage floor beacons.

- Monitor device attributes such as location, telemetry, battery status, and movement status.

The section **Outcome Apps** describes these two sections:

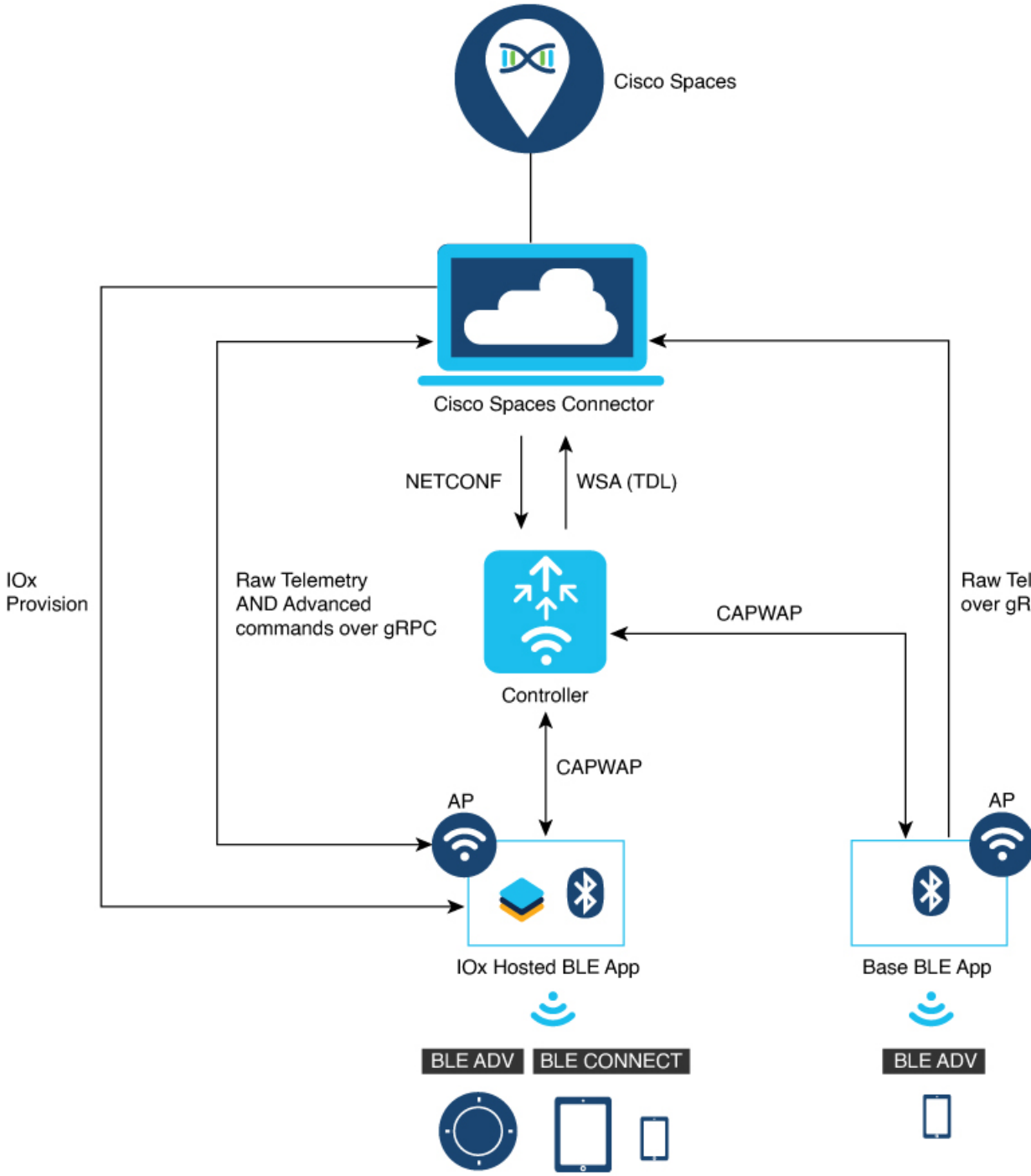
1. **Smart Workspaces**: Smart Workspaces helps you prepare your workspace for hybrid work. Click on the **Know More** button to visit the Cisco Spaces: Smart Workspaces page.
2. **Detect and Locate**: Detect and Locate simplifies IoT with eBLE technology powered by Cisco Spaces. Click on the **Know More** button to visit the Cisco Spaces: Detect and Locate page.

Components of Cisco Spaces: IoT Service

The section describes the various components that work to complete the Cisco Spaces: IoT Service solution.

The Cisco Catalyst 9100 Series Family of Access Points acts as a gateway of communication between Cisco Spaces and the IoT devices. Cisco Spaces: IoT Service can then use a range of common APIs to communicate with edge devices and apps. The Cisco Spaces: IoT Service collects data from devices and apps, and passes it to Cisco-partnered websites that manage these devices far more extensively (referred to in this document as Device Manager websites). These Device Manager websites can use edge-device signals to enable outcomes specialized and targeted for each industry.

Figure 184: Components of IoT Service



Access Points

You can configure access points as gateways in Cisco Spaces. You can find the list of supported APs in the **Compatibility Matrix** section.

Depending on your Cisco AP model, configure the AP as one of these types of BLE gateways:

- **Base BLE Gateway**: This is an AP that does not run Cisco IOx App. You can configure this AP in **Scan**, **Transmit**, or **Dual** mode.
- **Advanced BLE Gateway**: This is an AP installed with the Cisco IOx App. The Cisco IOx App brings the ability to connect, configure, and manage third-party BLE floor beacons using the Cisco Spaces dashboard. Moreover, the Cisco IOx App can be used to perform additional BLE filtering on the AP. You can configure this AP in **Scan**, **Transmit**, or **Dual** mode.

Cisco Catalyst 9800 Series Wireless Controllers

The Cisco Catalyst 9800 Series Wireless Controller (Catalyst 9800 controller) combines RF excellence with Cisco IOS-XE benefits, and comes in physical or virtual form factor. This wireless controller is reliable and highly secure. You can manage this Catalyst 9800 controller using CLI, GUI, NETCONF, Yang, or the Catalyst Center.

The Catalyst 9800 controller is the single point for configuring and managing a wireless network and access points. The Catalyst 9800 controller configures and manages APs using the CAPWAP protocol.

The Catalyst 9800 controller receives BLE configuration from Cisco Spaces over NETCONF and passes the configuration to AP over CAPWAP. The feedback path from the AP to the wireless controller is through CAPWAP, and from the Catalyst 9800 controller to Cisco Spaces through Telemetry data logger (TDL) telemetry streaming. The gRPC configuration from Cisco Spaces also goes through the Catalyst 9800 controller, and from there to the corresponding AP. The configuration sets up the gRPC channel between the AP and Cisco Spaces. The AP sends the gRPC channel statistics to the Catalyst 9800 controller, and you can view these statistics on the Catalyst 9800 controller.

Note

- You can have only one gRPC session between an AP and connector.
- Cisco Catalyst 9800 Series Wireless Controller running Cisco IOS XE Amsterdam 17.3.x supports only one of the following:
  - IoT service (wireless) with Cisco Spaces.
  - Network Assurance solution on Catalyst Center using Intelligent Capture (iCAP)

IoT service (wireless) and Intelligent Capture (iCAP) can co-exist from Cisco IOS XE Cupertino 17.7.x or higher.

Cisco Spaces: IoT Device Marketplace

Cisco Spaces: IoT Device Marketplace is a platform where you can discover, research, and purchase IoT devices. IoT Device Marketplace is a part of the Cisco Spaces full-stack partner ecosystem. Each device is preconfigured to give the customer an out-of-the-box experience with sensors, tags, wearables, and more. All the devices are compatible with the applications in the App Center. Current devices in the IoT Device Marketplace leverage BLE to transmit telemetry, with plans to add other technology in the future, such as Ultra Wide Band (UWB) and Zigbee.

Cisco Spaces: Connector

Cisco Spaces: Connector allows Cisco Spaces to communicate with more than one

- Cisco AireOS Wireless Controllers, and
- Cisco Catalyst 9800 Series Wireless Controllers.

APs connect to connector using the gRPC framework.

The APs establish a connection to connector using the gRPC protocol. The gRPC protocol configures floor beacons and receives telemetry data from the floor beacons. gRPC is a bidirectional streaming service, and requires a certificate to validate the host connection and a token for authentication. Each AP creates a gRPC connection. Connector can thus support many simultaneous connections.

Compatibility Matrix for IoT Service (Wireless)

Application Name	Support for Cisco Spaces: IoT Service
Supported wireless controllers	• Supported on Cisco Catalyst 9800 Series Wireless Controllers, Release 17.9.6 or 17.12.4 • Not supported on Cisco AireOS Wireless Controller • Not supported on Cisco Embedded Wireless Controller on Cisco Catalyst Access Points (Cisco EWC-AP) • Supported on Catalyst 9800 Controller running on Catalyst Switches in SD-Access mode (ECA) **Note** This support is conditional, and dependent on whether you have applied the fix described in CSCwk66790
Cisco Spaces: Connector Docker	2.0.455 and later
Cisco Spaces: Connector OVA	3.x and later
Cisco Prime Infrastructure	Cisco Prime Infrastructure Release 3.8 MR1 and later
Catalyst Center (for map import)	Catalyst Center Release 2.1.1 and later

Application Name	Support for Cisco Spaces: IoT Service
Access Points for advanced BLE gateway (Wi-Fi 6)	• Cisco Catalyst 9105 Series Access Points • Cisco Catalyst 9115 Series Access Points • Cisco Catalyst 9117 Series Access Points • Cisco Catalyst 9120 Series Access Points • Cisco Catalyst 9130 Series Access Points • Cisco Catalyst 9136 Series Access Points • Cisco Catalyst 9162 Series Access Points • Cisco Catalyst 9164 Series Access Points • Cisco Catalyst 9166 Series Access Points • Cisco Aironet 4800 Series Access Points • Cisco Catalyst IW9167 (E/I) Heavy Duty Series Access Points
Access points for basic BLE gateway	• Cisco Aironet 1815 Series Access Points • Cisco Aironet 2800 Series Access Points (USB dongle needed. No in-built USB radio) • Cisco Aironet 3800 Series Access Points (USB dongle needed. No in-built USB radio)
Cisco IOx App Version	1.0.46 and later **Note** For Cisco Catalyst 9800 Series Wireless Controllers Cisco IOS XE Cupertino 17.7.x, ensure that the IoX Application version is upgraded to Version 1.3.x

Note IoT Service (Wireless) is not supported with a directly connected controller, CMX Tethering, and AireOS connector. The only supported configuration is the Cisco Catalyst 9800 Wireless Controller and the connector.

This table lists the compatibility of the Advanced BLE Gateway for BLE and the Base BLE Gateway App with various AP modes. This table is not applicable to Cisco Embedded Wireless Controller on Cisco Catalyst Access Points (Cisco EWC-AP).

Table 18: AP Modes and app support

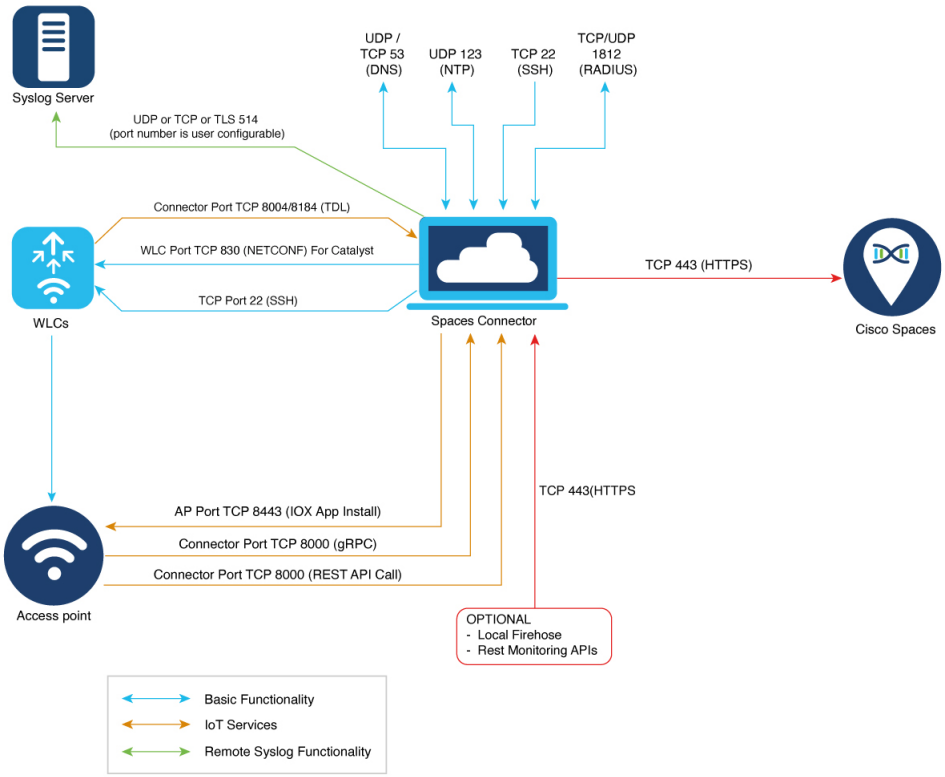
AP Mode	Advanced BLE Gateway App	Base BLE Gateway App
PI: Local	• 802.11ax: Supported • COS APs/Wave 2: Not supported	• 802.11ax: Supported • COS APs/Wave 2: Supported
P1: Flex	• 802.11ax: Supported • COS APs/Wave 2: Not supported	• 802.11ax: Supported • COS APs/Wave 2: Supported
P2: Fabric	• 802.11ax: Supported • COS APs/Wave 2: Not supported	• 802.11ax: Supported • COS APs/Wave 2: Supported
P3: Mesh	• 802.11ax: Supported • COS APs/Wave 2: Not supported	• 802.11ax: Supported • COS APs/Wave 2: Supported

Prerequisites of IoT Service (Wireless)

Open Ports for IoT Service (Wireless)

This section lists the connector ports that must be open for the proper functioning of IoT service (wireless).

Figure 185: Open Ports for IoT service (wireless)



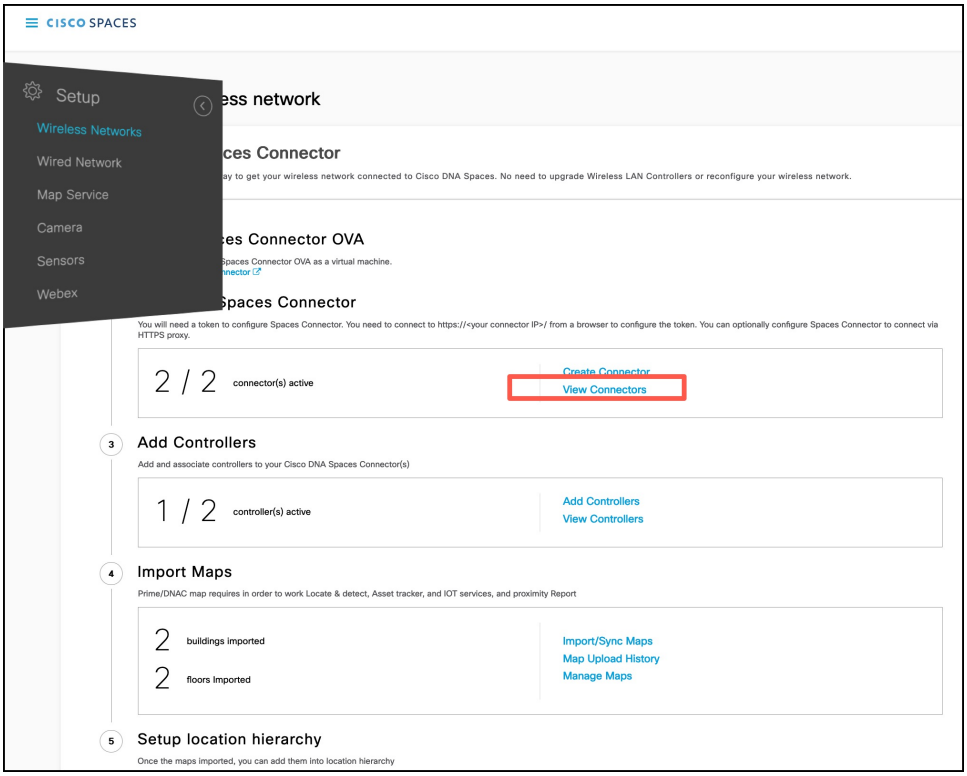
Configure IoT Service (Wireless)

Procedure

Step 1 In the Cisco Spaces dashboard left navigation pane, click **Setup** and choose **Wireless Networks**.

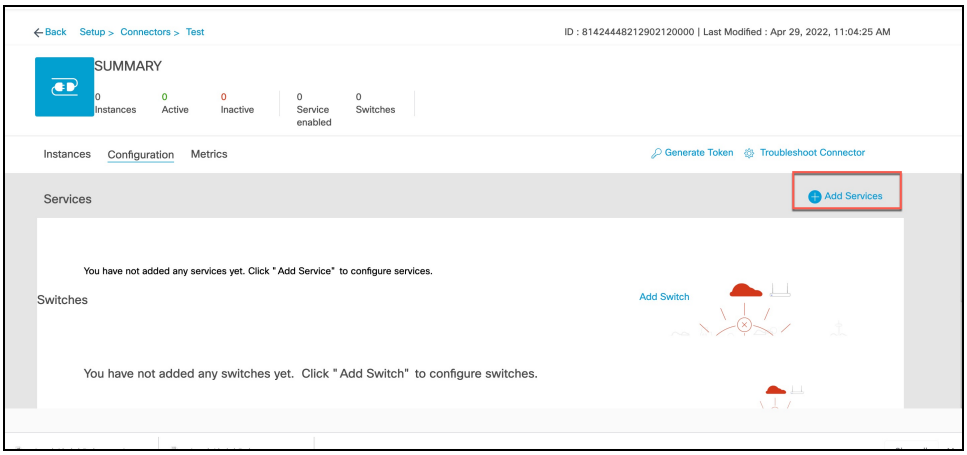
Step 2 In the **Connect your wireless network** window that is displayed, go to the **Step 2** area and click **View Connectors**.

Figure 186: View Connectors



Step 3 In the connector details window that is displayed, click **Add Services**.

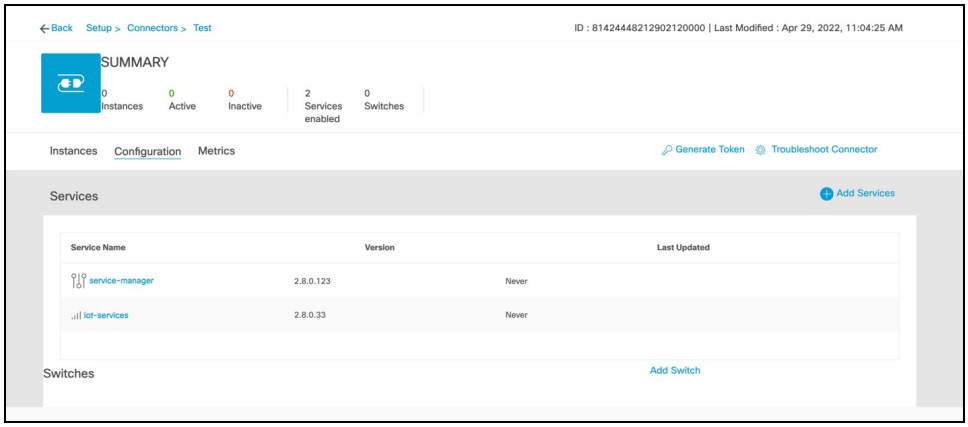
Figure 187: Add Services



Step 4 In the **Add Services** window that is displayed, choose **IoT Wireless** and click **Add**.

Note
service-manager is chosen by default.

Figure 188: Connector Details



In the **Connector Details** window, you can see that the number of services that are enabled has increased.

Verify IoT Streams for Catalyst 9800 Controller

This task is for troubleshooting purposes only. IoT streams are automatically enabled for all the wireless controllers associated with the IoT service (wireless) service of a connector.

This task helps you troubleshoot IoT streams of a Catalyst 9800 controller. If your APs are not visible, you can manually enable or disable the IoT streams of Cisco Spaces.

Procedure

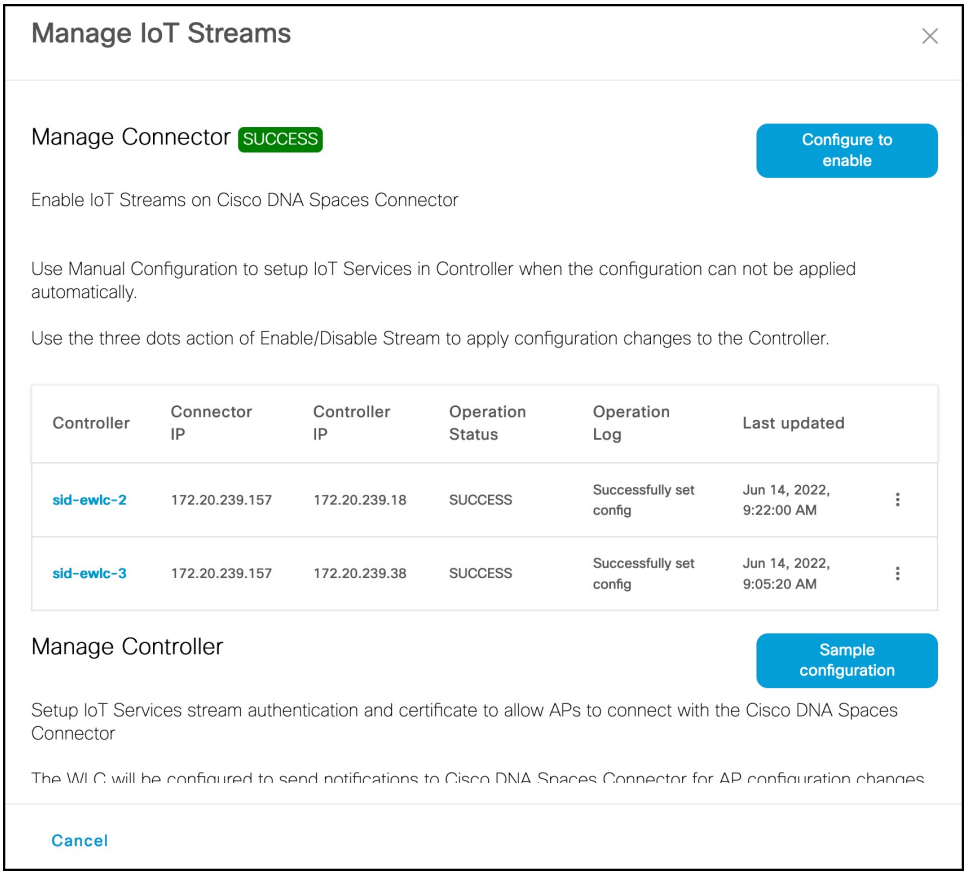
Step 1 In the Cisco Spaces dashboard left navigation pane, choose **Setup > Wireless Network**.

Step 2 In the **Configure via Spaces Connector** area titled **Step 2: Add Controllers**, click **View Connectors**.

Step 3 Click the connector of your choice.

Step 4 In the **Services** tab, in the **Actions** column, click the gear icon near IoT service (wireless) to open the **Manage IoT Streams** window.

Figure 189: Troubleshooting IoT Streams



Verify access points

This procedure helps you verify if IoT service (wireless) has synchronized and listed the APs in your network on the GUI.

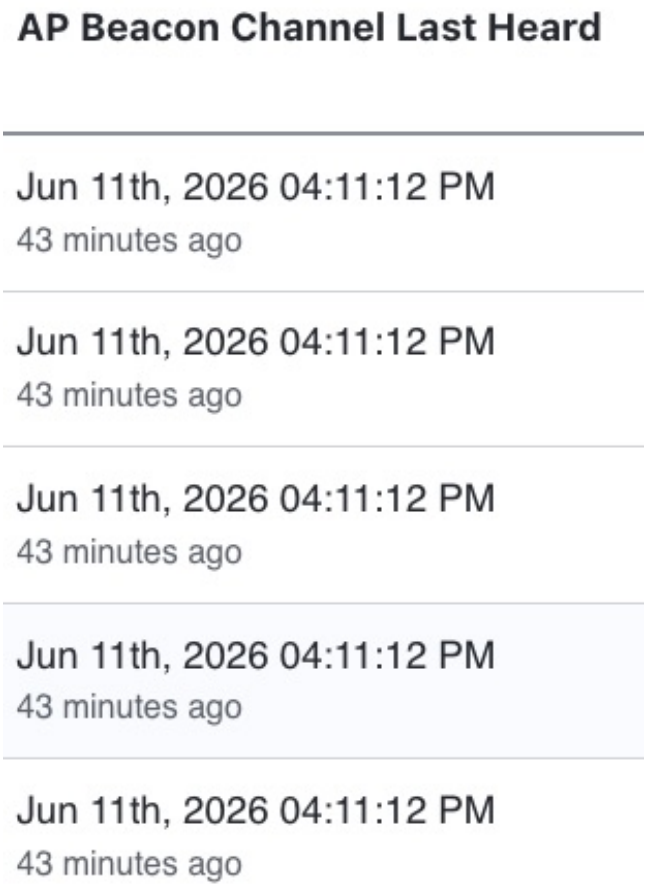
Procedure

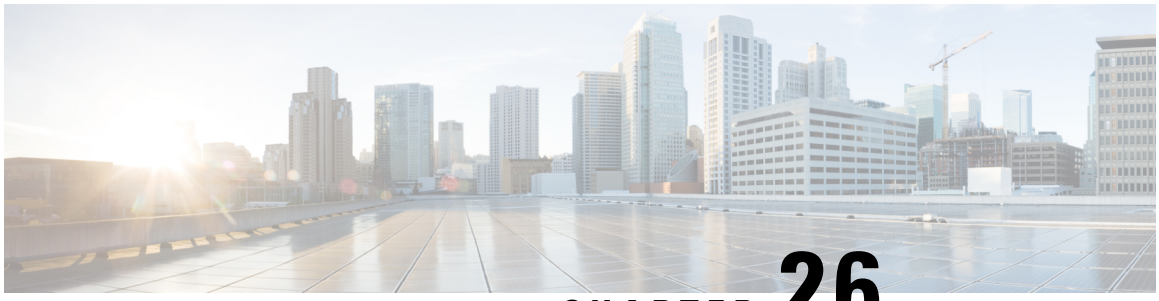
Step 1 In the Cisco Spaces dashboard left-navigation pane, choose **IoT Services** > **Manage** > **Wireless APs** > **AP Gateways**.

Step 2 Verify if IoT service (wireless) has synchronized and listed the APs in your network. Check the **Floor Beacon Channel Status** and **AP Beacon Channel Last Heard** columns.

Figure 190: Floor Beacon Channel Status

Figure 191: AP Beacon Channel Last Heard





CHAPTER 26

Connector Gateway with Sensor Connect

- Connector Gateway with Sensor Connect, on page 211
- Create a Connector with Sensor Connect, on page 211
- Add a Controller for Sensor Connect, on page 212
- Install Sensor Connect, on page 212
- Manage Sensor Connect After Installation, on page 213
- View Sensor Connect Metrics, on page 213

Connector Gateway with Sensor Connect

Cisco Spaces enables you to use Cisco Spaces Connector as a gateway with Sensor Connect. In this workflow, the connector gateway service is added first, a wireless controller is added to the connector, and Sensor Connect is installed and managed from the controller row.

The setup starts from **Setup** > **Wireless Networks** > **Connect via Spaces Connector** and continues on the Connector Details page. The video also shows the controller-side Sensor Connect interface, which is separate from the Cisco Spaces dashboard.

Create a Connector with Sensor Connect

Before you begin

Ensure that:

- You can log in to Cisco Spaces.
- You have permission to manage **Setup** > **Wireless Networks** and Cisco Spaces Connector.
- Cisco Spaces Connector is deployed or ready to activate.
- You have the controller access details, including the controller IP address, username, and password.
- The controller and Sensor Connect image are supported for this workflow.

Procedure

Step 1 Log in to Cisco Spaces.
Step 2 From the left navigation pane, choose **Setup** > **Wireless Networks**.
Step 3 Click **Connect via Spaces Connector** to expand the bar.
Step 4 In **Configure Spaces Connector** section, click **Create Connector**.
Step 5 In the **Create Connector** slide-in window, enter the connector name.
Step 6 Enable the required service toggles.
Step 7 Check the **Use connector as a gateway with Sensor Connect** check box.
Step 8 Click **Create**.

Add a Controller for Sensor Connect

After the connector is created, open the Connector Details page. Confirm that Sensor Connect is enabled, then add the controller that will host Sensor Connect.

Procedure

Step 1 On the **Connector Details** page, verify that the Sensor Connect gateway service is enabled.
Step 2 In the **Wireless Controllers** area, click **Add Controller**.
Step 3 In the **Add Controller** window, enter the controller name.
Step 4 Enter the controller IP address, controller type, username, and password.
Step 5 Confirm the required network interface details.
Step 6 Click **Save**.

The controller is displayed in the **Wireless Controllers** table. Wait until the controller status is ready before installing Sensor Connect.

Install Sensor Connect

Procedure

Step 1 In the **Wireless Controllers** table, locate the required controller.
Step 2 In the **Sensor Connect** column, click **Install**.

Alternatively, click the **Actions** menu for the controller and choose **Install Sensor Connect**.

Step 3 Wait for Cisco Spaces to generate the NAT configuration.

Step 4 Click **Apply Configuration and Install**.

Cisco Spaces applies the NAT configuration and starts the Sensor Connect installation workflow.

Note The installation timeline runs automatically. Cisco Spaces shows progress for applying the NAT configuration, downloading the Sensor Connect image, and installing or starting Sensor Connect.

Manage Sensor Connect After Installation

After Sensor Connect is installed successfully, the **Sensor Connect** status changes to **Running**. The options in the controller row **Actions** menu change based on the Sensor Connect status.

When Sensor Connect is running, the **Actions** menu includes the following options:

- Restart
- Stop
- View Metrics
- Upload Logs to Cloud
- Uninstall Sensor Connect

View Sensor Connect Metrics

Procedure

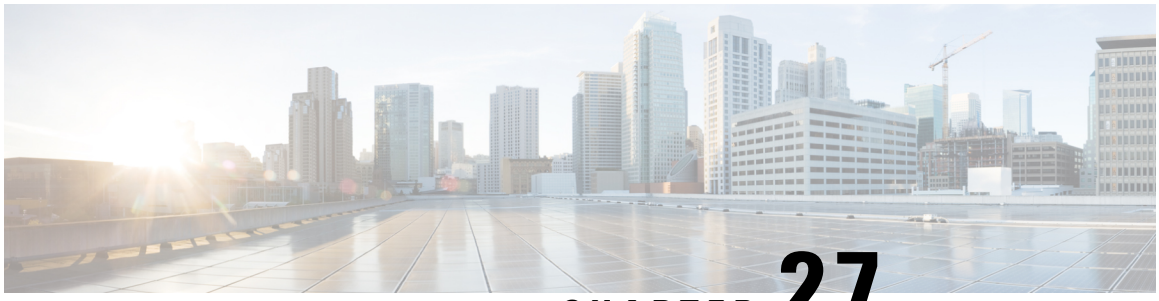
Step 1 In the **Wireless Controllers** table, click the controller name.

Step 2 Click the **Sensor Connector** tab.

Step 3 Review the **State**, **Version**, **Channel Status**, and **Last Heard** values.

Step 4 Click **24 h** or **7 days** to change the metrics period.

The metrics area displays CPU, memory, and disk charts.



CHAPTER 27

IoT Service (Wired)

- Overview, on page 215

Overview

Overview of IoT Service (Wired)

Cisco Spaces enables end-to-end wired and wireless IoT device management, monitoring, and business outcome delivery at an enterprise scale using the following:

- Cisco Spaces: IoT Service
- Cisco Spaces: IoT Device Marketplace
- Cisco Spaces App Center

In addition to serving as the management hub for wireless IoT devices, IoT Service can now integrate with Cisco Catalyst 9300 and 9400 Series Switches from Release 17.9.5 or later to receive IoT service (wired) data from sensors, such as:

- Passive infrared (PIR) sensors for presence detection
- Sensors with telemetry such as temperature, humidity, CO2, and air quality for environmental analytics
- Ethernet port information or status for occupancy and energy usage
- Smart power distribution unit (PDUs) for energy efficiency tracking

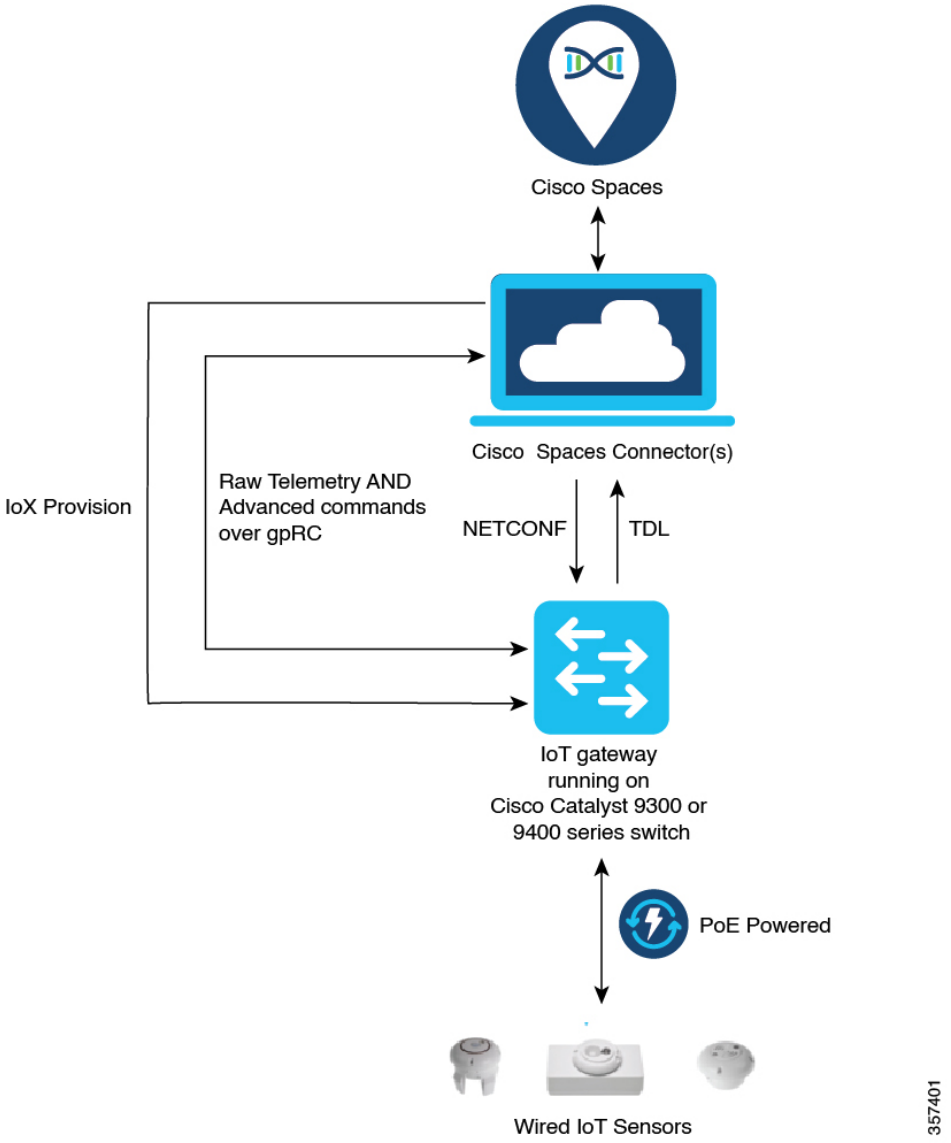
Integrating IoT service (wired) with the Cisco Catalyst 9300 and 9400 Series Switches series platform requires the following:

- Cisco Spaces: Connector
- A IoT service (wired) gateway deployed and managed by Cisco Spaces

Cisco Catalyst 9300 and 9400 Series Switches can send critical IoT data to IoT service (wired). IoT service (wired) can then transmit the information to:

- Business outcome applications on Cisco Spaces
- Cisco Spaces App Center using the Firehose API

Figure 192: Data flow in IoT Service (Wired)



Compatibility matrix for IoT Service (Wired)

Application Name	Support for IoT Service (Wired)
Cisco Catalyst 9300 Series Switches	• Cisco IOS XE Cupertino 17.9.5 • Cisco IOS XE Dublin 17.12.4
Cisco Catalyst 9400 Series Switches	• Cisco IOS XE Cupertino 17.9.5 • Cisco IOS XE Dublin 17.12.4

Application Name	Support for IoT Service (Wired)
Wired Docker Service	3.2. and later
Wired IOX Application	1.2.3 and later

IoT service (wired) is not supported with Cisco Spaces tenants or deployments leveraging the following configurations:

- Connecting directly with controller
- CMX Tethering
- Sensor Connect for IoT (IoT Orchestrator)

Prerequisites for Cisco Spaces: IoT Service (Wired)

The following are the necessary prerequisites to get you started with Cisco Spaces: IoT Service (Wired):

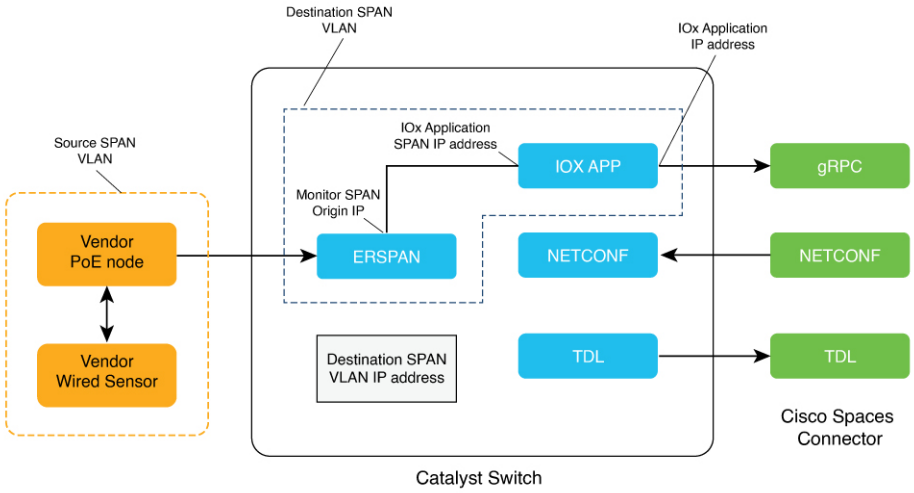
- Install Cisco Spaces: Connector in your network.
- Configure a network with one or more Cisco Catalyst 9300 and 9400 Series Switches, Release 17.9.5 or later.
- Switches must have **Cisco DNA Advantage** subscription. Cisco Spaces tenants must have either Spaces Extend, Act, Unlimited, or Spaces-Advantage license.
- Deploy wired sensors in your network. See Compatibility matrix for IoT Service (Wired) , on page 216.
- Ensure that Cisco Spaces is configured with maps either from Cisco Prime Infrastructure or Catalyst Center.
- Configure AAA on aCisco Catalyst 9300 Series Switches or a Cisco Catalyst 9400 Series Switches before adding it to Cisco Spaces by running these commands in:
  - **aaa new-model**
  - **aaa authentication login default local**
  - **aaa authorization exec default local**
- Perform NTP synchronization across wireless controllers, Cisco Spaces: Connectors, and switches in the network.
- Enable NETCONF on Cisco Catalyst 9300 or 9400 Series Switches on port 830, along with permission to use NETCONF.

Note Cisco Catalyst 9300 and 9400 Series Switches require a privilege level 15 user to use and push configurations over NETCONF. Additionally, the user must be a password-protected user, because public-key authentication is not supported.

Design Prerequisites

Ensure you have the following information handy before proceeding:

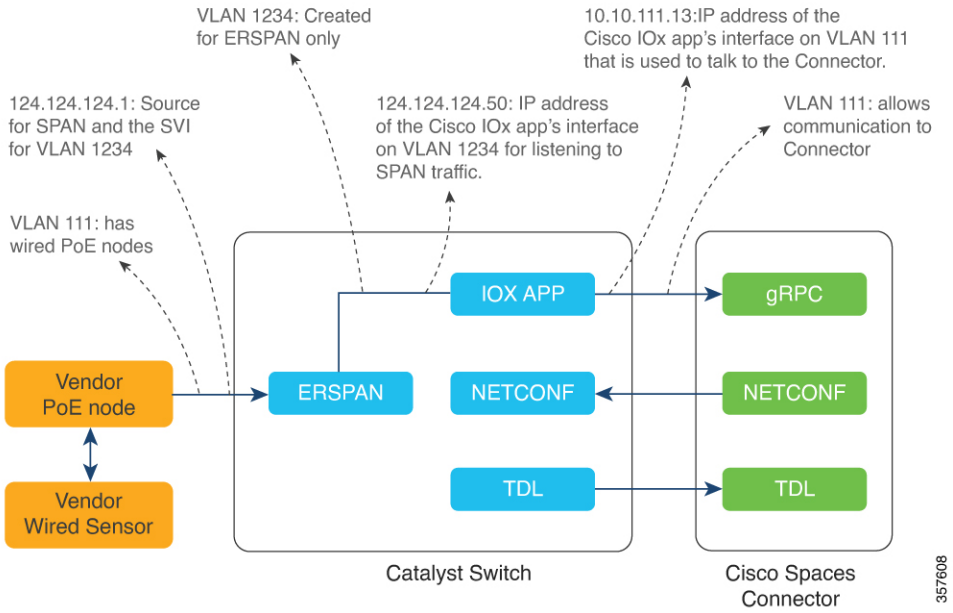
Figure 193: Design Prerequisites



- **Destination SPAN VLAN**: The VLAN used to send Encapsulated Remote Switched Port Analyzer (ERSPAN) traffic from Power over Ethernet (PoE) nodes to Cisco IOx App. You can use an existing VLAN or create a new one. This VLAN can also be local to the switch.
- **Destination SPAN VLAN IP address**: This is the Switched Virtual Interface (SVI) or the IP address of the destination VLAN that can be used to route traffic. If you are using an existing VLAN, you can provide the same IP address. We recommend that you create a new VLAN so that you can keep the ERSPAN traffic local without impacting the existing configuration. Note that this VLAN is used only within the switch for the SPAN traffic.
- **Source SPAN VLAN list:** List of VLANs to which the wired devices are connected. The traffic on these VLANs are monitored. If the wired devices are connected to multiple VLANs, enter the VLANs separated by a comma.
- **Monitor SPAN origin IP address**: This is the source IP address of the monitor session. This can be from the SPAN VLAN. This can also be the same as the destination VLAN IP address.
- **IoX application Span IP Address**
- **Application Cisco Spaces Connector VLAN**: This is the VLAN on which the connector is reachable (for management or data). You can configure the Cisco IOx App's second interface to use this VLAN to send traffic to the connector. This VLAN can be the same as the wired PoE node VLAN. The connector must be permitted to accept communications from the Cisco IOx application.
- **DHCP**: When enabled, DHCP allocates an IP address from the **Application DNA Spaces Connector VLAN** to the Cisco IOx App's second interface.
- **IoX application IP address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the Connector. This is not required if you select DHCP.

- **IoX application netmask**: This is the IP subnet mask that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.
- **IoX application gateway address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.

Figure 194: Sample Configuration



Prerequisites for Cisco Spaces: IoT Service (Wired)

The following are the necessary prerequisites to get you started with Cisco Spaces: IoT Service (Wired):

- Install Cisco Spaces: Connector in your network.
- Configure a network with one or more Cisco Catalyst 9300 and 9400 Series Switches, Release 17.9.5 or later.
- Switches must have **Cisco DNA Advantage** subscription. Cisco Spaces tenants must have either Spaces Extend, Act, Unlimited, or Spaces-Advantage license.
- Deploy wired sensors in your network. See Compatibility matrix for IoT Service (Wired) , on page 216.
- Ensure that Cisco Spaces is configured with maps either from Cisco Prime Infrastructure or Catalyst Center.
- Configure AAA on aCisco Catalyst 9300 Series Switches or a Cisco Catalyst 9400 Series Switches before adding it to Cisco Spaces by running these commands in:
  - **aaa new-model**
  - **aaa authentication login default local**

- **aaa authorization exec default local**

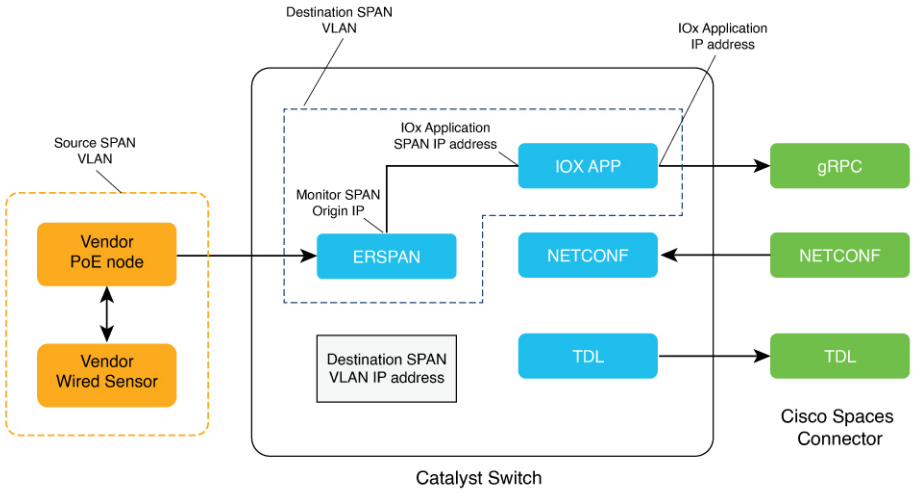
- Perform NTP synchronization across wireless controllers, Cisco Spaces: Connectors, and switches in the network.
- Enable NETCONF on Cisco Catalyst 9300 or 9400 Series Switches on port 830, along with permission to use NETCONF.

Note Cisco Catalyst 9300 and 9400 Series Switches require a privilege level 15 user to use and push configurations over NETCONF. Additionally, the user must be a password-protected user, because public-key authentication is not supported.

Design Prerequisites

Ensure you have the following information handy before proceeding:

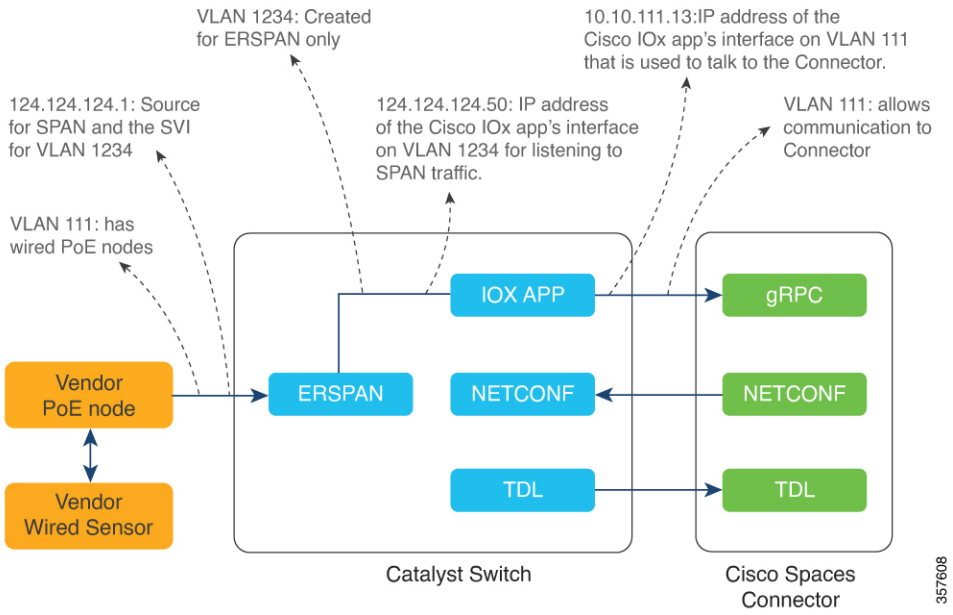
Figure 195: Design Prerequisites



- **Destination SPAN VLAN**: The VLAN used to send Encapsulated Remote Switched Port Analyzer (ERSPAN) traffic from Power over Ethernet (PoE) nodes to Cisco IOx App. You can use an existing VLAN or create a new one. This VLAN can also be local to the switch.
- **Destination SPAN VLAN IP address**: This is the Switched Virtual Interface (SVI) or the IP address of the destination VLAN that can be used to route traffic. If you are using an existing VLAN, you can provide the same IP address. We recommend that you create a new VLAN so that you can keep the ERSPAN traffic local without impacting the existing configuration. Note that this VLAN is used only within the switch for the SPAN traffic.
- **Source SPAN VLAN list:** List of VLANs to which the wired devices are connected. The traffic on these VLANs are monitored. If the wired devices are connected to multiple VLANs, enter the VLANs separated by a comma.
- **Monitor SPAN origin IP address**: This is the source IP address of the monitor session. This can be from the SPAN VLAN. This can also be the same as the destination VLAN IP address.

- **IoX application Span IP Address**
- **Application Cisco Spaces Connector VLAN**: This is the VLAN on which the connector is reachable (for management or data). You can configure the Cisco IOx App's second interface to use this VLAN to send traffic to the connector. This VLAN can be the same as the wired PoE node VLAN. The connector must be permitted to accept communications from the Cisco IOx application.
- **DHCP**: When enabled, DHCP allocates an IP address from the **Application DNA Spaces Connector VLAN** to the Cisco IOx App's second interface.
- **IoX application IP address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the Connector. This is not required if you select DHCP.
- **IoX application netmask**: This is the IP subnet mask that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.
- **IoX application gateway address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.

Figure 196: Sample Configuration



Open ports for IoT service (wired)

This section lists the connector ports that must be open for the proper functioning of each service or protocol.

Figure 197: Open Ports for IoT Service (Wired) with the IoT Gateway

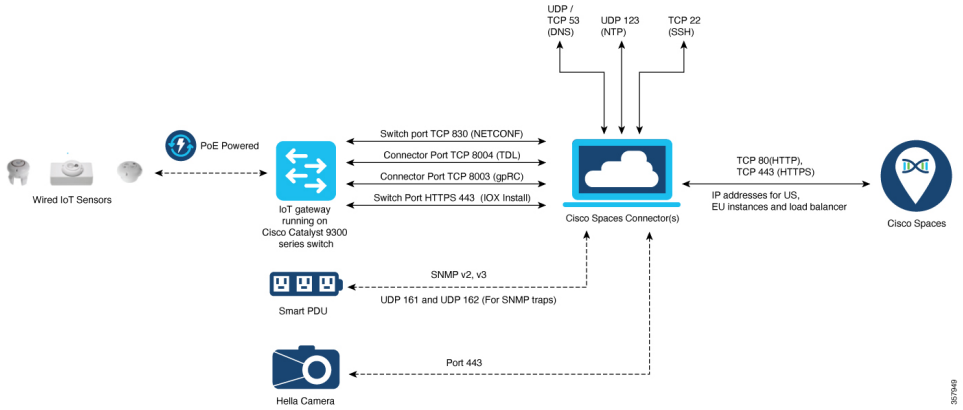
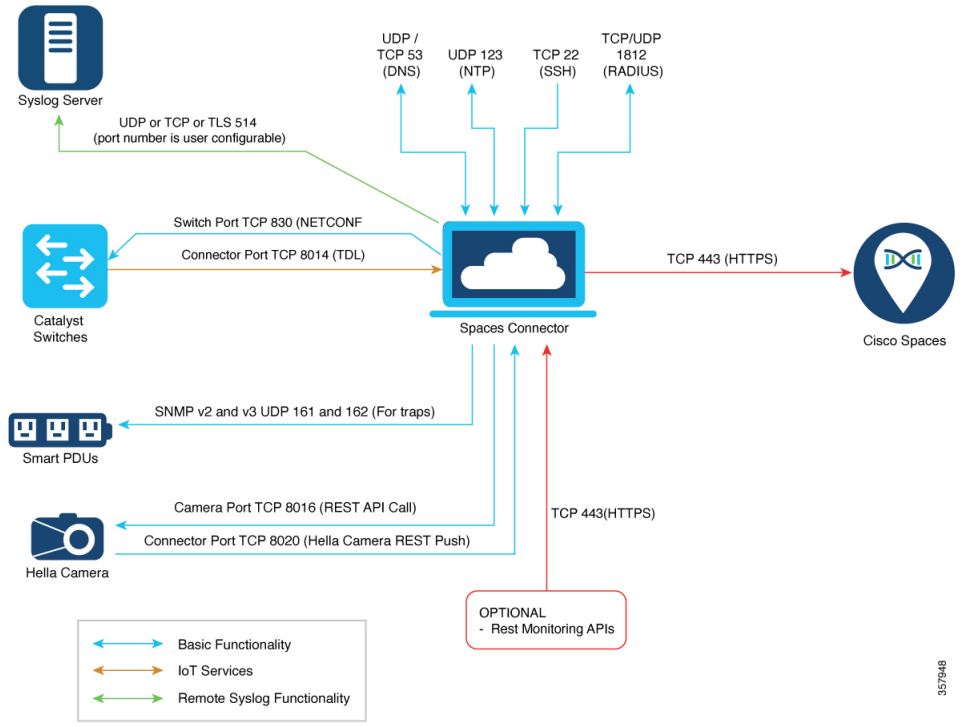


Figure 198: Open Ports for IoT Service (Wired) without the IoT Gateway



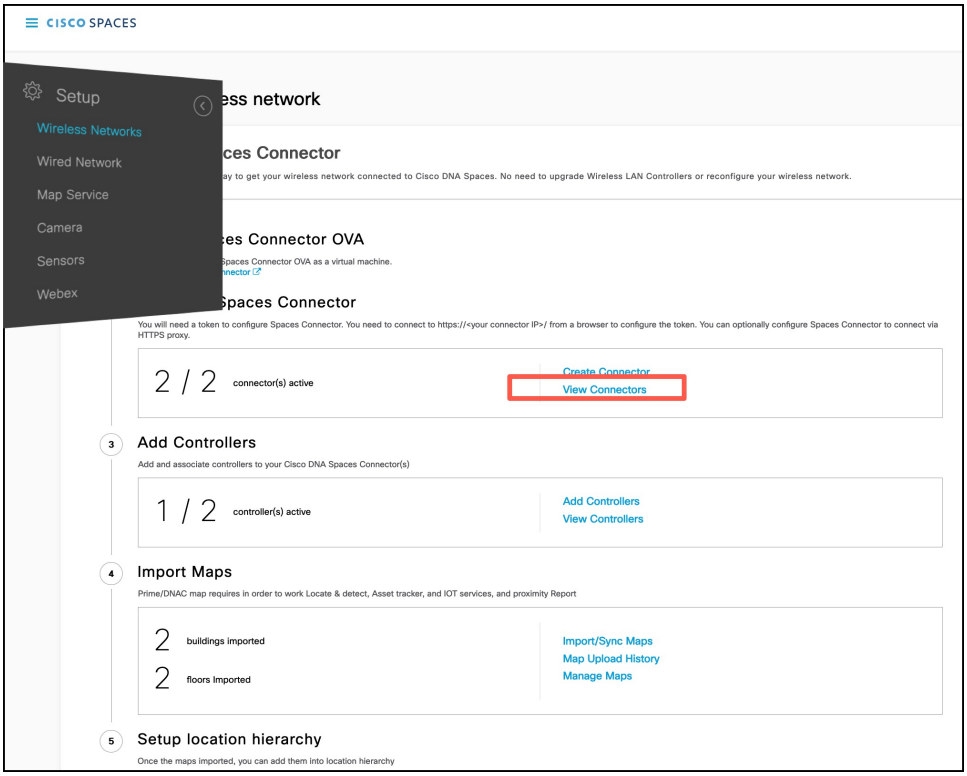
Configure IoT Service (Wired)

Procedure

Step 1 From the Cisco Spaces dashboard left-navigation pane, click **Setup** and choose **Wired Networks**.

Step 2 From the **Connect your wireless network** window that is displayed, go to the **Step 2** area and click **View Connectors**.

Figure 199: View Connectors

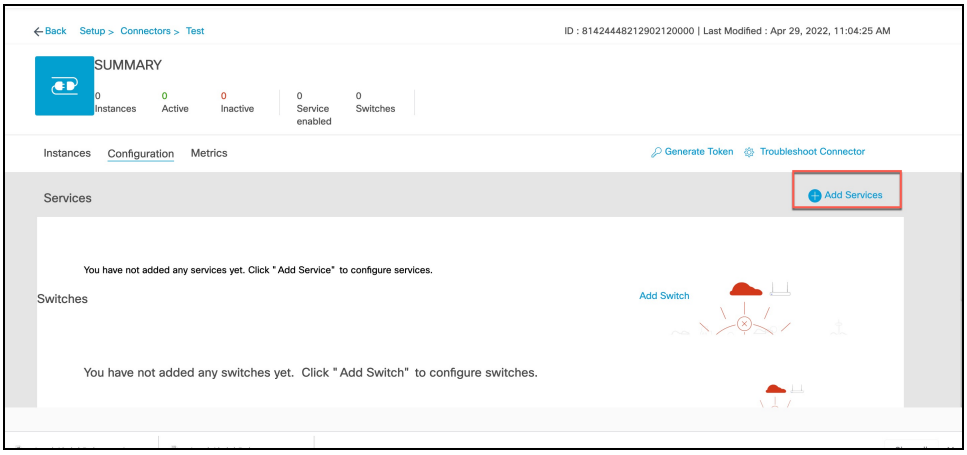


Step 3 Click a connector 3 of your choice.

Note
You can use the same connector that you used for Cisco Spaces: IoT Service (Wireless).

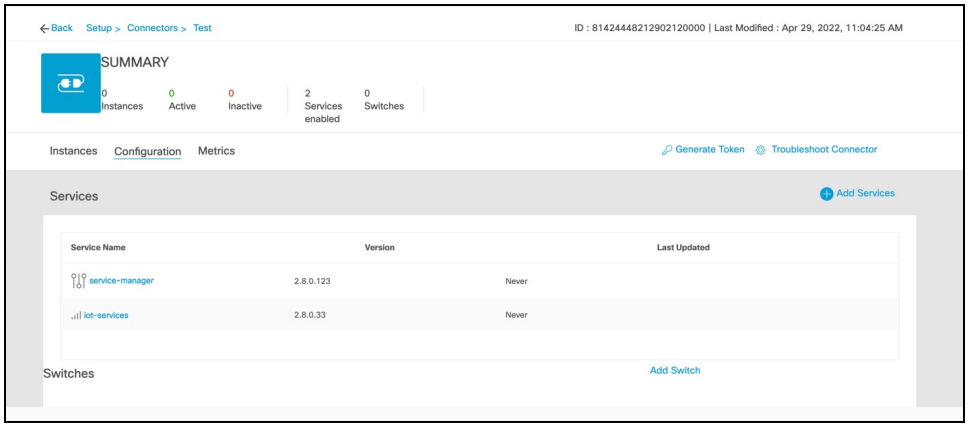
Step 4 In the connector details window that is displayed, click **Add Services**.

Figure 200: Add Services



Step 5 In the **Add Service** window that is displayed, choose **IoT Wired** and click **Add**.

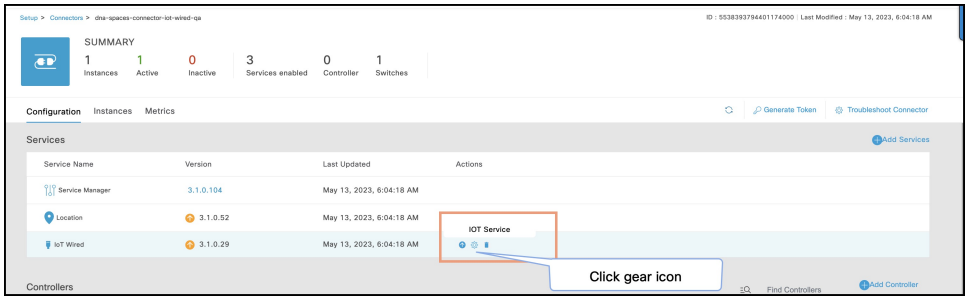
Figure 201: Adding a Service



In the **Connector Details** window, you can see that the **IoT Wired** service has been added.

Step 6 Click the gear icon near the **IoT Wired** row.

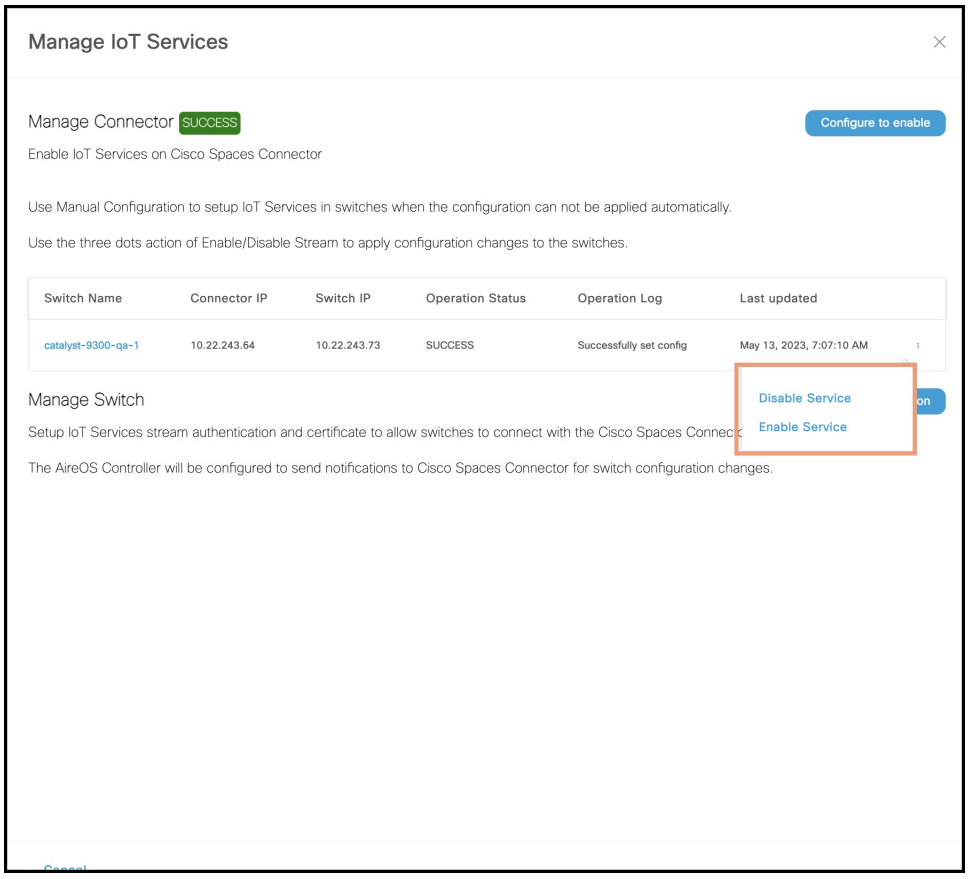
Figure 202: Gear Icon of IoT Wired



Step 7 (Optional) In the **Manage IoT Streams** window that is displayed, check if the connector is not already enabled, and if it is not, click **Configure to Enable**.

Step 8 From the list of switches, click the vertical three-dot icon adjacent to the switch and select **Enable Service**.

Figure 203: Enable Service



Note
If you are using the same connector for both wired and wireless IoT services, the connector is already enabled.

Step 9 Enter the SPAN VLAN and the Cisco IOx App details.

- **Destination SPAN VLAN**: The VLAN used to send Encapsulated Remote Switched Port Analyzer (ERSPAN) traffic from Power over Ethernet (PoE) nodes to Cisco IOx App. You can use an existing VLAN or create a new one. This VLAN can also be local to the switch.
- **Destination SPAN VLAN IP address**: This is the Switched Virtual Interface (SVI) or the IP address of the destination VLAN that can be used to route traffic. If you are using an existing VLAN, you can provide the same IP address. We recommend that you create a new VLAN so that you can keep the ERSPAN traffic local without impacting the existing configuration. Note that this VLAN is used only within the switch for the SPAN traffic.
- **Source SPAN VLAN list:** List of VLANs to which the wired devices are connected. The traffic on these VLANs are monitored. If the wired devices are connected to multiple VLANs, enter the VLANs separated by a comma.
- **Monitor SPAN origin IP address**: This is the source IP address of the monitor session. This can be from the SPAN VLAN. This can also be the same as the destination VLAN IP address.
- **IoX application Span IP Address**
- **Application Cisco Spaces Connector VLAN**: This is the VLAN on which the connector is reachable (for management or data). You can configure the Cisco IOx App's second interface to use this VLAN to send traffic

to the connector. This VLAN can be the same as the wired PoE node VLAN. The connector must be permitted to accept communications from the Cisco IOx application.

- **DHCP**: When enabled, DHCP allocates an IP address from the **Application DNA Spaces Connector VLAN** to the Cisco IOx App's second interface.
- **IoX application IP address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the Connector. This is not required if you select DHCP.
- **IoX application netmask**: This is the IP subnet mask that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.
- **IoX application gateway address**: This is the IP address that you must manually configure for the Cisco IOx App's second interface, and is used to communicate with the connector. This is not required if you select DHCP.

Figure 204: Configure Switch

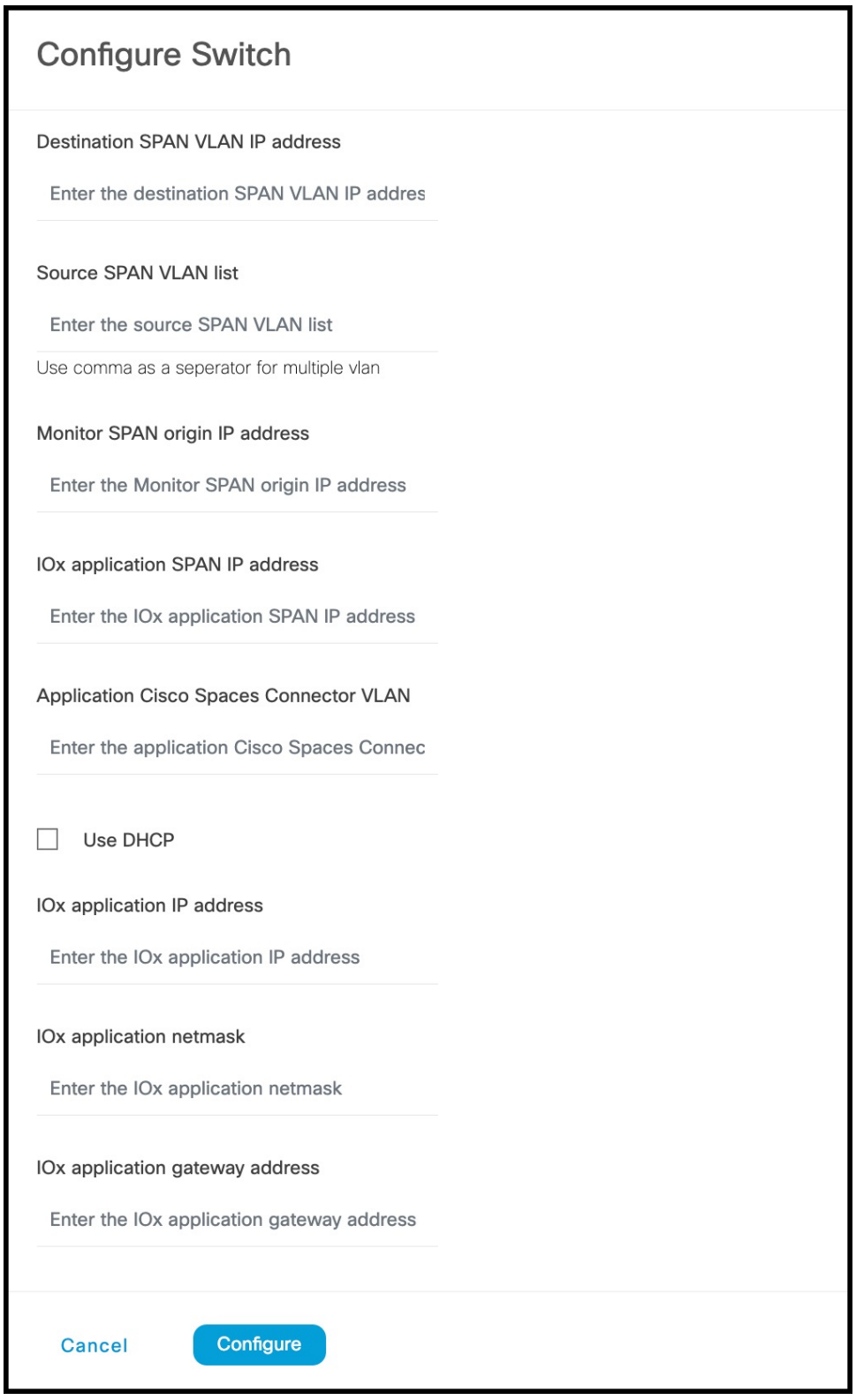


Figure 205: Configure Switch

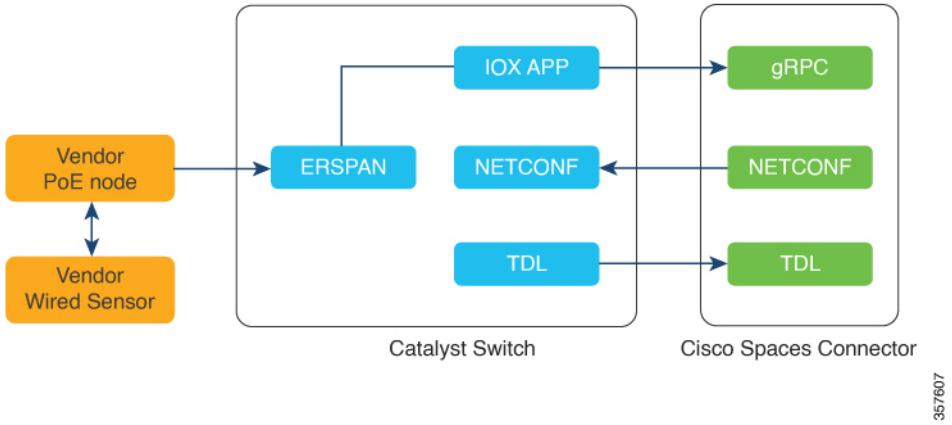
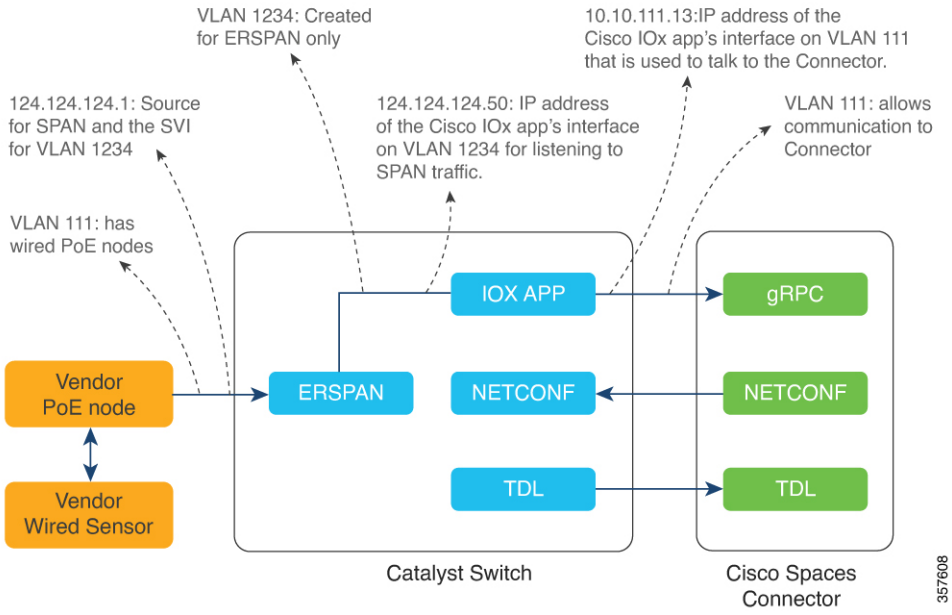


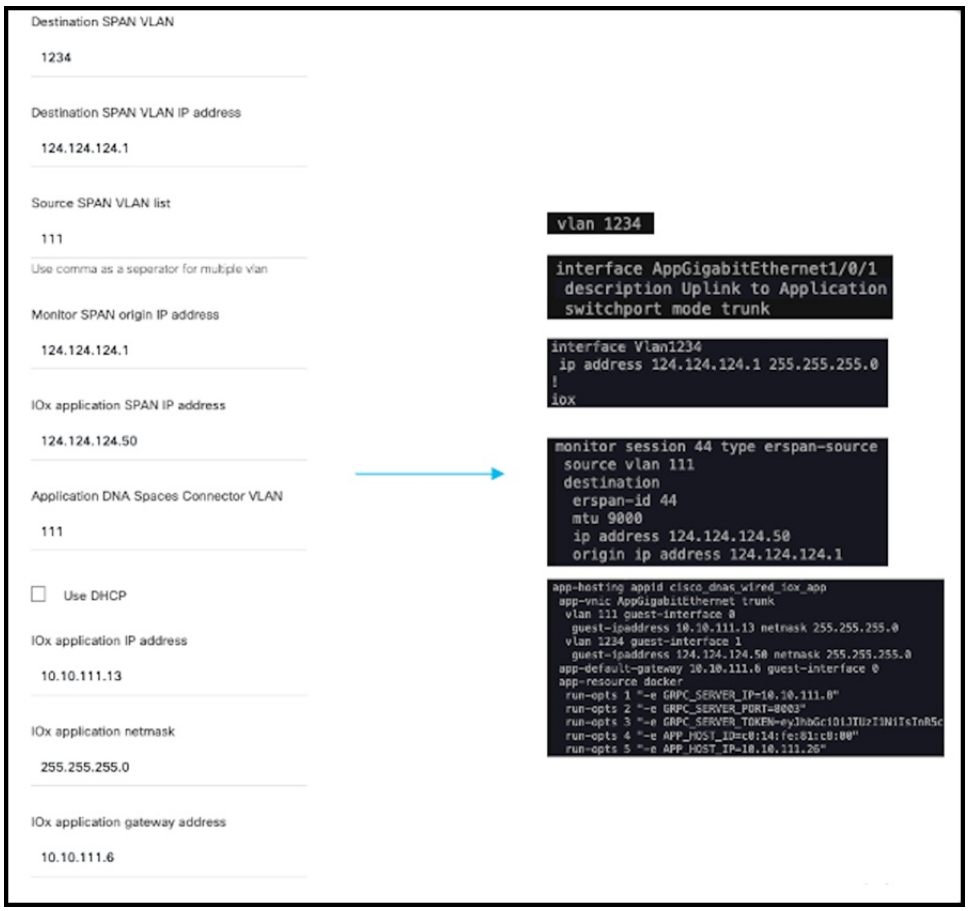
Figure 206: Sample Configuration



Step 10 Click **Configure**.

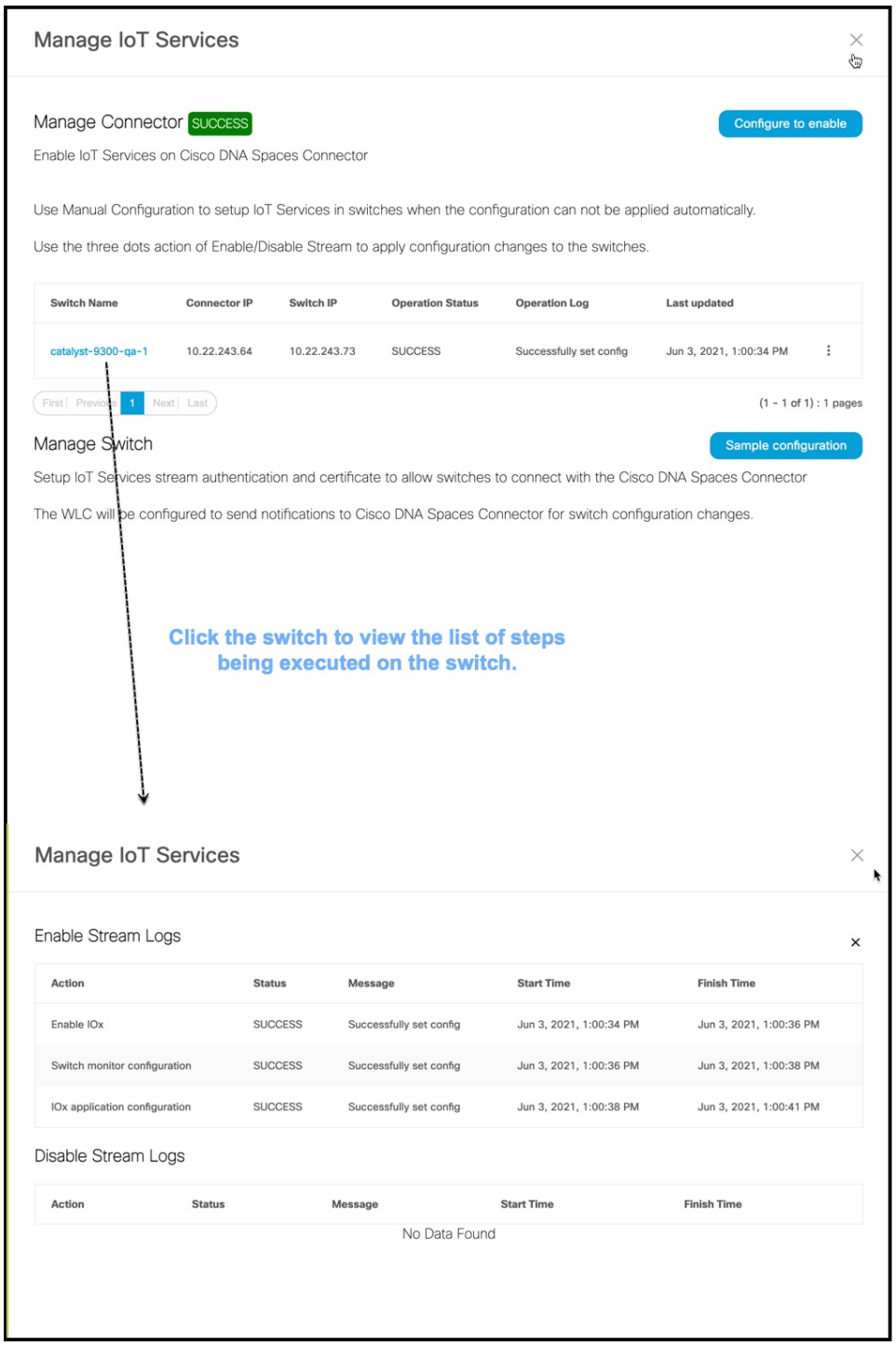
The configurations are deployed on the switch. The following diagram shows the corresponding CLI commands you can use in place of the GUI configuration.

Figure 207: GUI-Command Line Mapping



Step 11 In the **Manage IoT Services** window that you are taken to, you can click on a name of the switch to see the list of steps executed on that switch.

Figure 208: Manage IoT Services



Verify if Cisco Catalyst 9300 and 9400 Series Switches are Added to the Connector

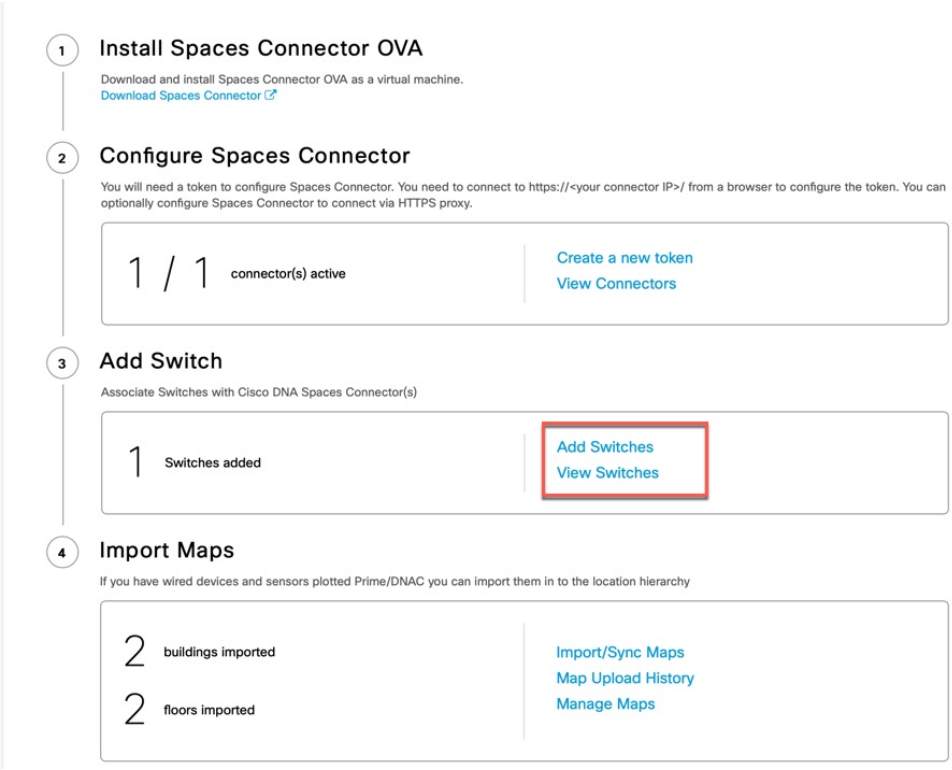
This procedure helps you verify if a Cisco Catalyst 9300 or 9400 Series Switches are deployed and active. This is a necessary prerequisite for proper functioning of Cisco Spaces: IoT Service (Wired).

Procedure

Step 1 In the Cisco Spaces dashboard left navigation pane, choose **Setup > Wired Network**.

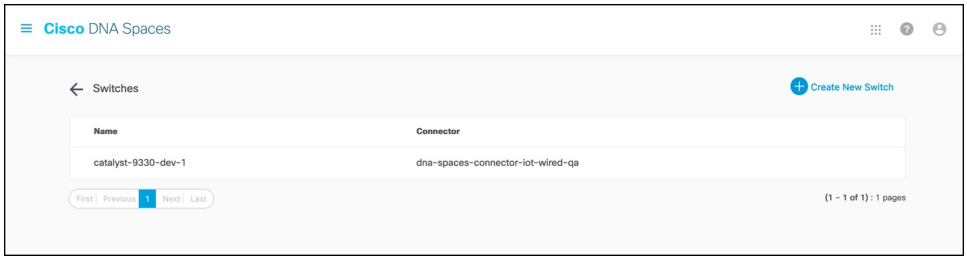
Step 2 In the **Add Switch** area, click **View Switches**.

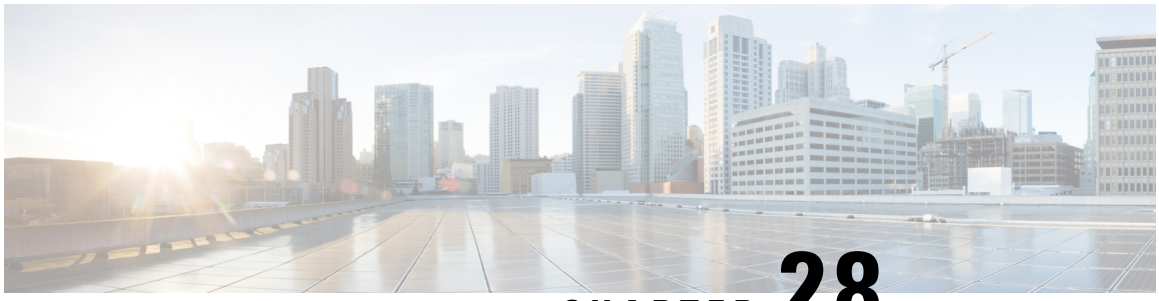
Figure 209: View Switches



Step 3 Ensure that a switch is listed here, and is connected to a Cisco Spaces: Connector.

Figure 210: View Switches





CHAPTER 28

Hotspot Service

- Configure Hotspot Service, on page 233
- Connector Dashboard: Hotspot service, on page 235
- Open Ports for Hotspot Service, on page 235

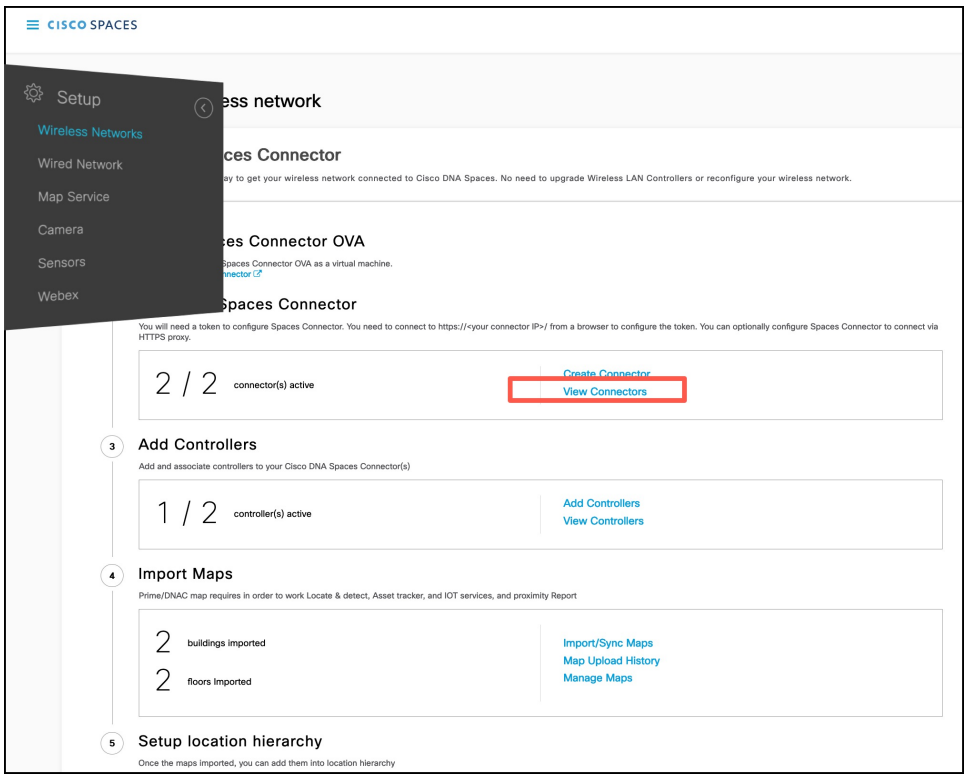
Configure Hotspot Service

Procedure

Step 1 In the Cisco Spaces dashboard left navigation pane, click **Setup** and choose **Wireless Networks**.

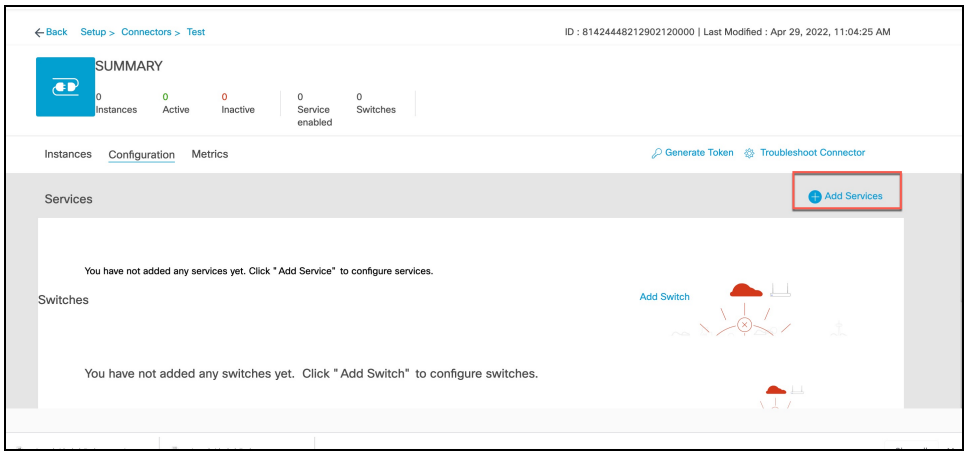
Step 2 In the **Connect your wireless network** window that is displayed, go to the **Step 2** area and click **View Connectors**.

Figure 211: View Connectors



Step 3 In the connector details window that is displayed, choose a connector and click **Add Services**.

Figure 212: Add Service



Step 4 In the **Add Service** window that is displayed, choose **hotspot** and click **Add**.

Note
service-manager is added by default.

In the **Connector Details** window, you can see that the number of services enabled has increased.

Connector Dashboard: Hotspot service

Figure 213: Hotspot Service

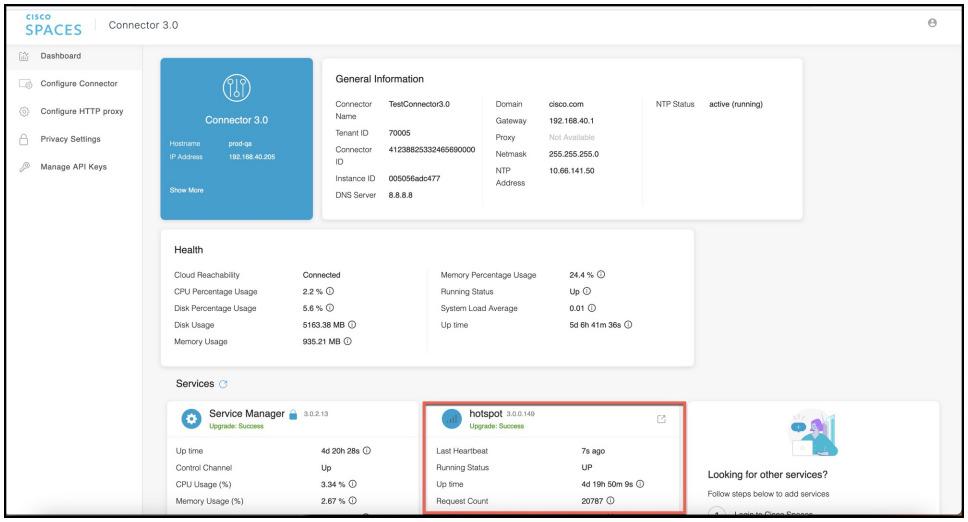
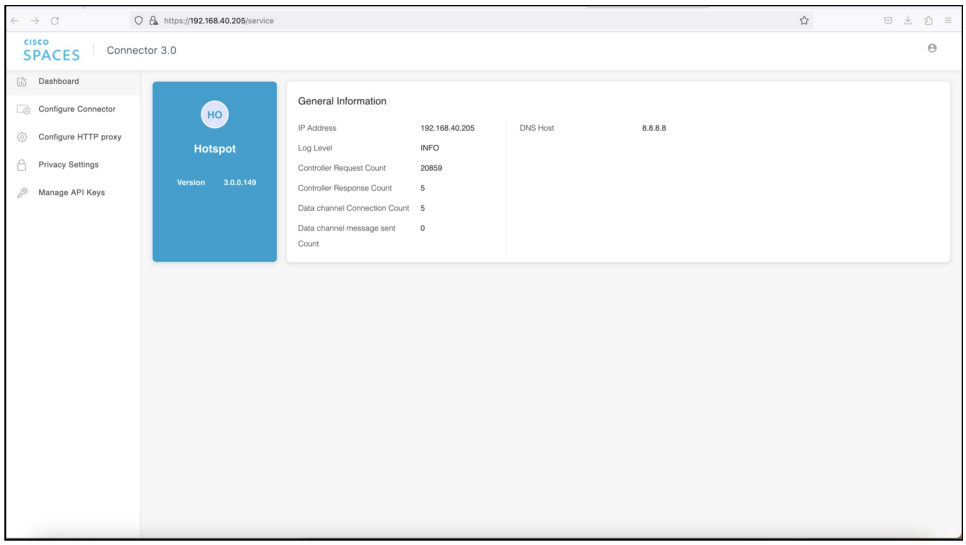


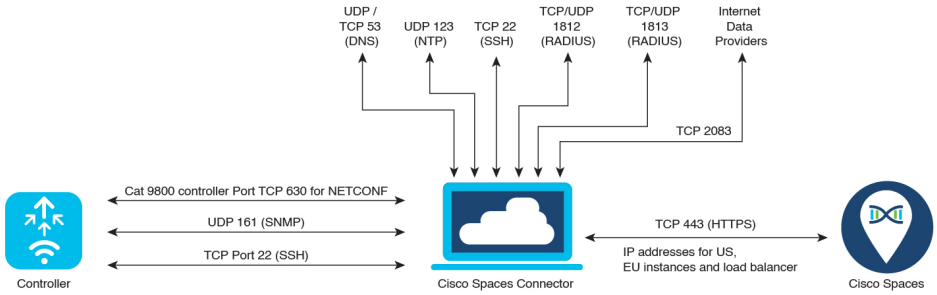
Figure 214: Hotspot Service: Details



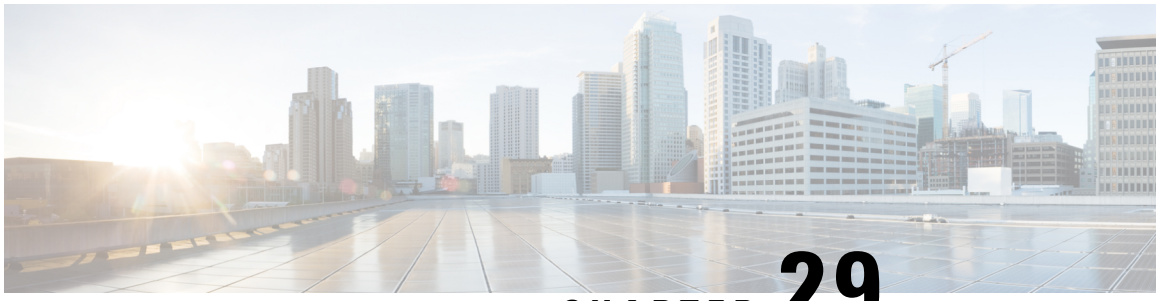
Open Ports for Hotspot Service

This section lists the connector ports that must be open for the proper functioning of the hotspot service.

Figure 215: Open Ports for Hotspot Service



Test the connectivity between the connector and the wireless controller. See Configure and Test Connectivity between the Connector 3 and AireOS controller or Configure and Test the Connectivity between a Connector 3 and a Catalyst 9800 controller.



CHAPTER 29

Local Firehose

- Local Firehose Service, on page 237
- Configure Local Firehose Service, on page 237
- Connector Dashboard: Local Firehose Service, on page 240

Local Firehose Service

The partner's location engine must be configured with the IP address of the connector.

If two connectors are configured in high-availability (either active-active or VIP-paired mode), ensure that both connector IP addresses are configured on the partner's location engine. In such a configuration, you can see that Radio Frequency Identification (RFID) tag information is received on both the connector channels, but Bluetooth Low Energy (BLE) tag information is received only on the Active connector channel.

Warning Do not configure the virtual IP address (VIP) of VIP-paired connectors on the partner's location engine.

IoT Service supports high availability only in the VIP-paired mode.

Note For creation and activation of a partner app, refer to the On-Prem Partner App

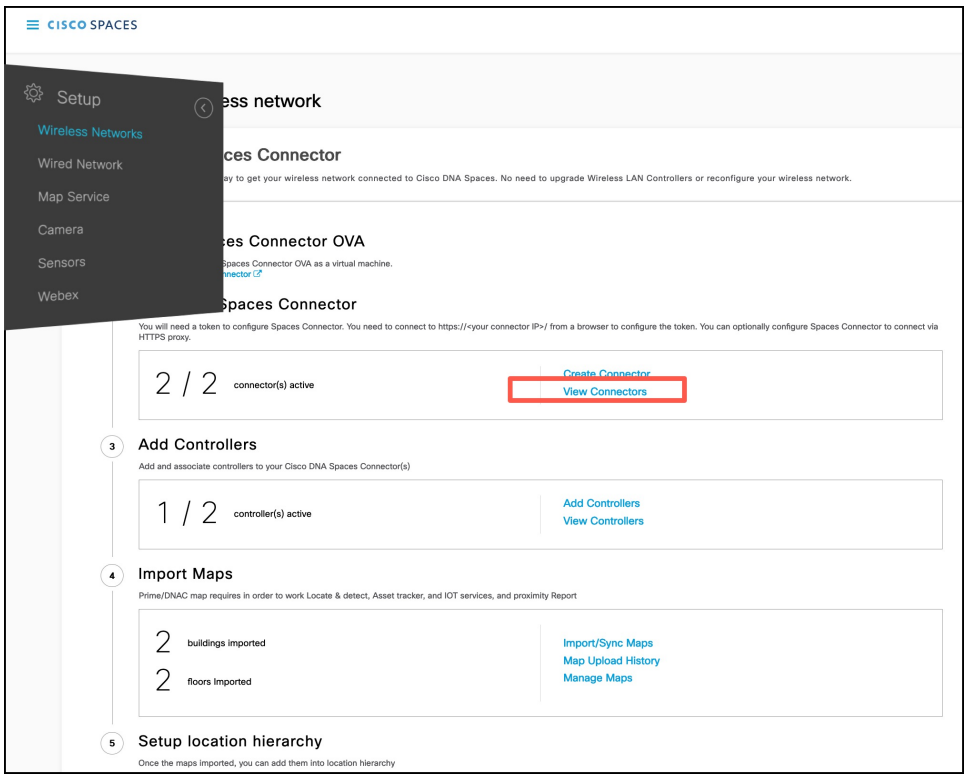
Configure Local Firehose Service

Procedure

Step 1 In the Cisco Spaces dashboard left navigation pane, click **Setup** and choose **Wireless Networks**.

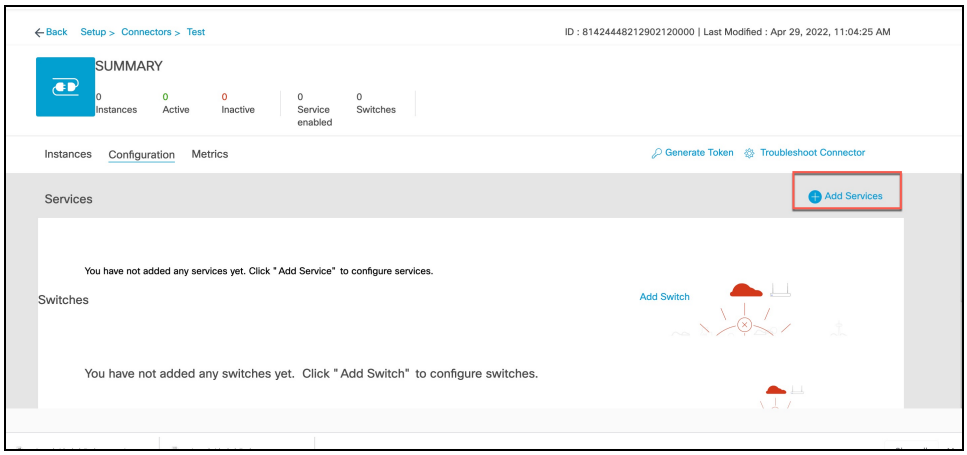
Step 2 In the **Connect your wireless network** window that is displayed, go to the **Step 2** area and click **View Connectors**.

Figure 216: View Connectors



Step 3 In the connector details window that is displayed, choose a connector and click **Add Services**.

Figure 217: Add Service



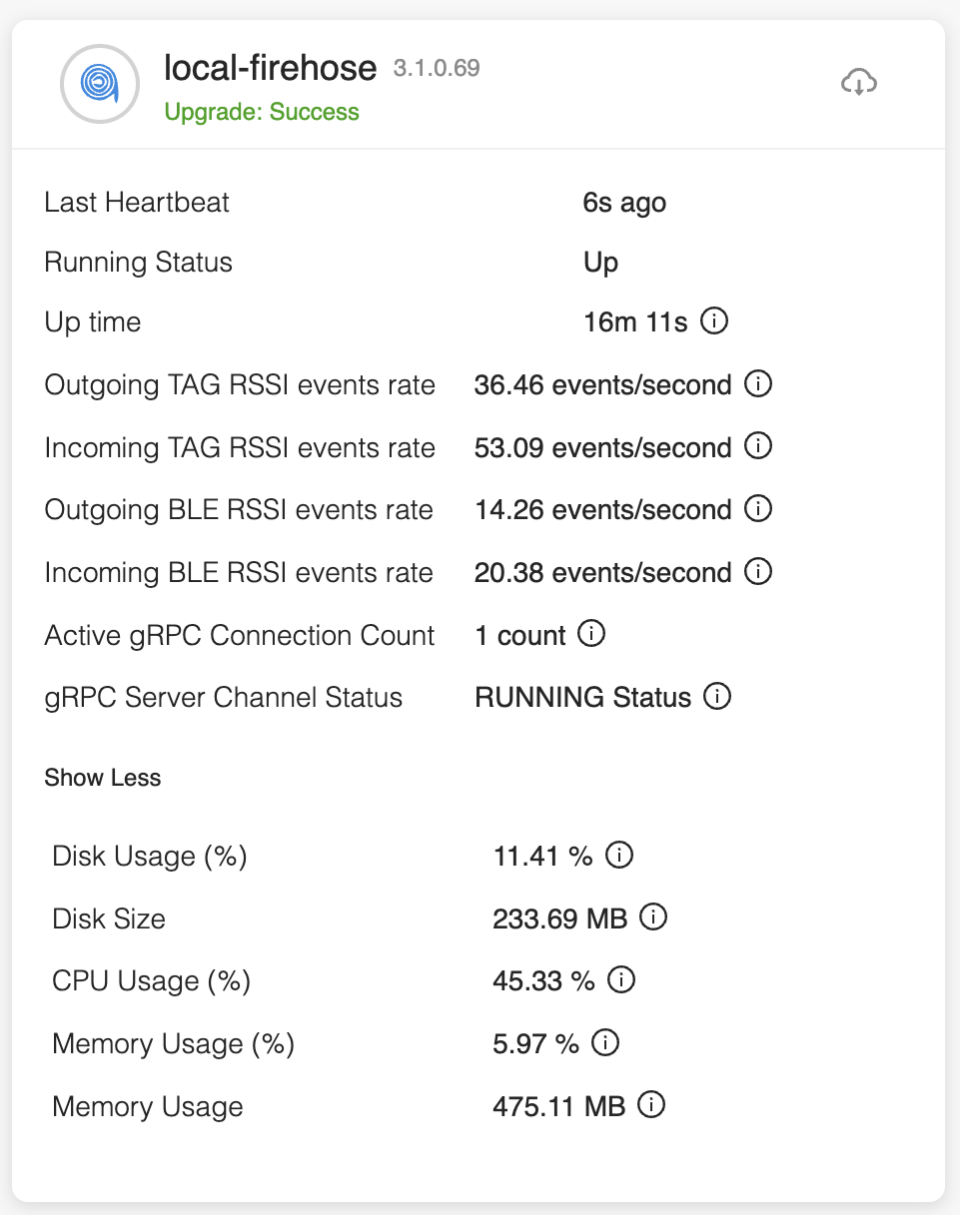
Step 4 In the **Add Service** window that is displayed, choose **local-firehose** and click **Add**.

Note
To receive events such as Device_RSSI for Received Signal Strength Indicator (RSSI)-based tags and Device_BLE events for Bluetooth Low Energy (BLE) tags, ensure that **location** and **iot-services** services are also added.

You can see that the number of services enabled has increased.

Step 5 Login to the Connector GUI. Scroll downwards to the **local-firehose** tile. Verify if the running status is **Up**.

Figure 218: local-firehose



Connector Dashboard: Local Firehose Service

Figure 219: Local firehose service: Details on the Connector

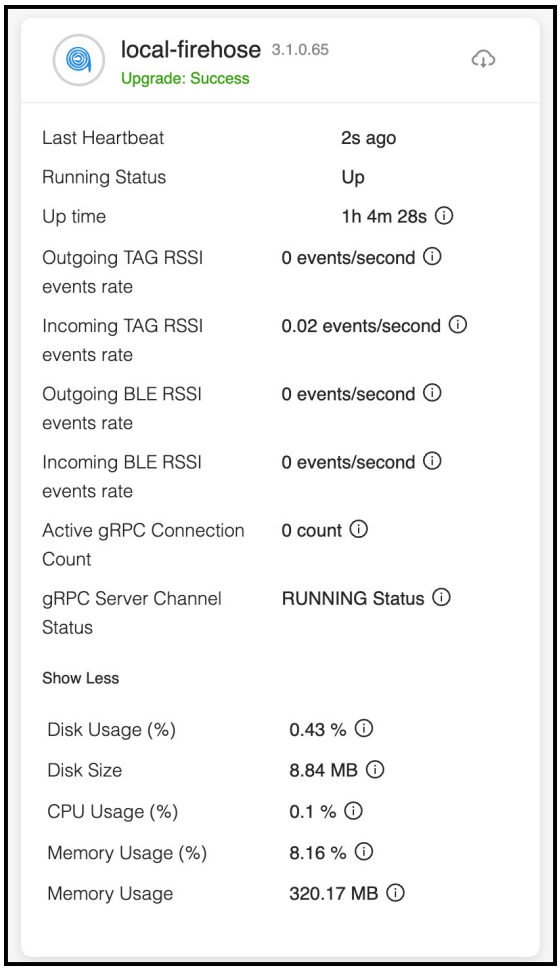
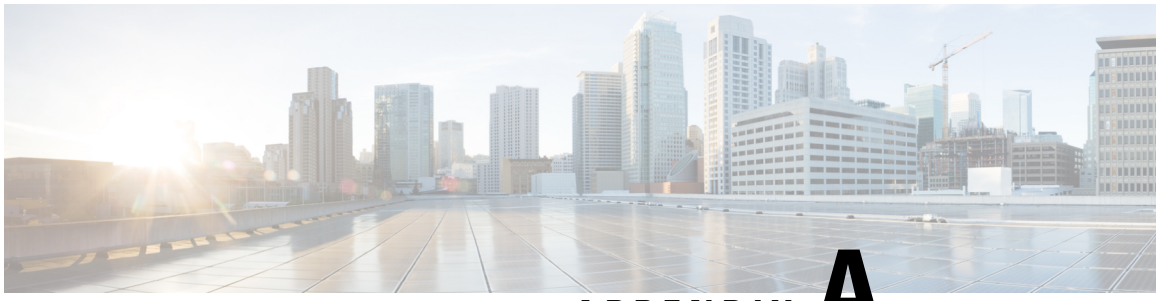


Table 19: Local Firehose Service Metrics

Display Field	Information
Active gRPC connection count	Number of connections from the partner's location engine
Outgoing TAG RSSI events rate	Number of RFID RSSI events sent from local-firehose-service to the partner's location engine
Incoming TAG RSSI events rate	Number of Radio Frequency Identification (RFID) Received Signal Strength Indicator (RSSI) events received from the location-service to local-firehose-service

Display Field	Information
Outgoing BLE RSSI events rate	Number of BLE RSSI Events sent from local-firehose-service to partner's location engine
Incoming BLE RSSI events rate	Number of Bluetooth Low Energy (BLE) RSSI Events received from iot-service to local-firehose- service



APPENDIX A

Connect Connector to Cisco AireOS Wireless Controller

• Configure and Test Connectivity Between a Connector and AireOS Controller, on page 243

Configure and Test Connectivity Between a Connector and AireOS Controller

Before you begin

- Deploy a connector OVA and activate it using a token from Cisco Spaces.
- Ensure that the IP address of a Cisco AireOS Wireless Controller is reachable from the Cisco Spaces: Connector.

Restriction

- In the context of CSCvk38081, we recommend that you do not add connector on the same subnet as the dynamic interface of the AireOS controller. However, if you cannot follow this recommendation, you can add the AireOS controller to connector and configure all the SNMP queries to the IP address of the dynamic interface of the controller.
- We also recommend that you do not add connector on the same subnet as the service port of the AireOS controller. However, if you cannot follow this recommendation, you can add the AireOS controller to connector and configure all the SNMP queries to the IP address of the service port of the controller.
- This restriction is a result of a limitation in the AireOS controller. While SNMP queries are usually made to the management IP address, the SNMP response packets are returned with a source IP address field that is configured with the IP address of the dynamic interface or source port.

Procedure

Step 1 Log in to **Cisco Spaces**.

Note
The Cisco Spaces URL is region-dependent.

Step 2 In the Cisco Spaces dashboard, choose **Setup** > **Wireless Networks**.

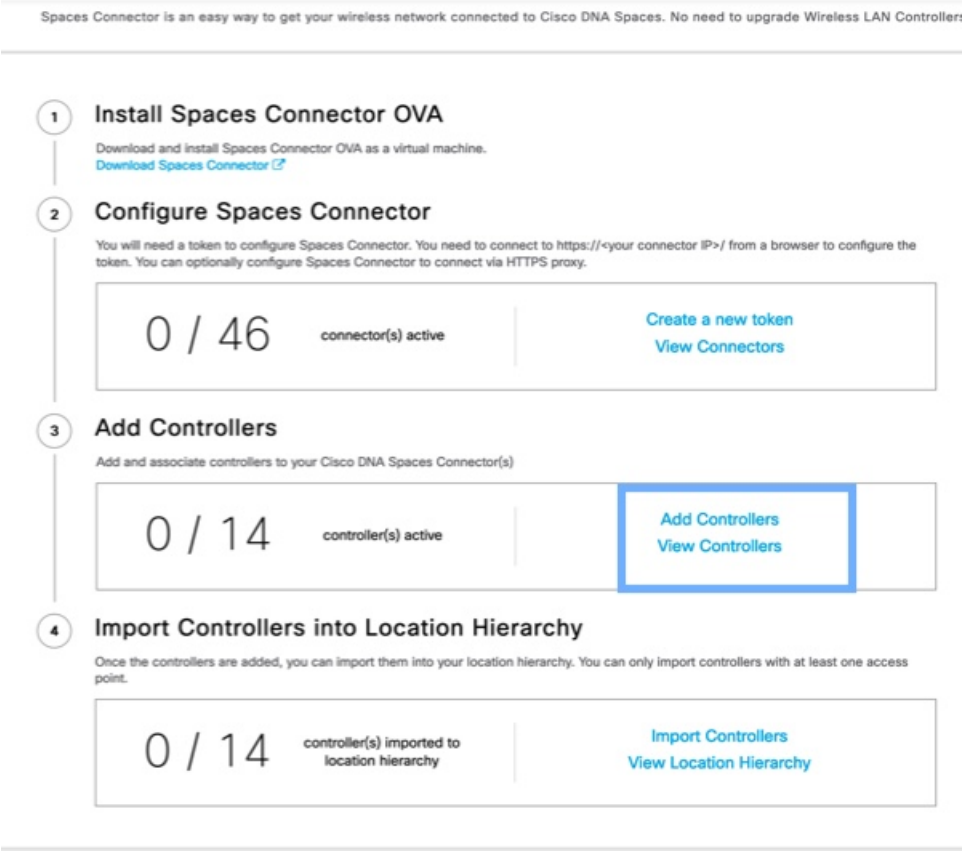
Step 3 Expand the **Connect via Spaces Connector** area using the respective drop-down arrow to display a list of steps.

Step 4 To test the connectivity from the Connector to an existing AireOS controller, click **View Controllers** in the **Step 3** area, and do the following steps:

a) Click the pencil icon to edit an AireOS controller.
b) Choose an active Connector from the **Connector** drop-down list to enable the **Test Connectivity** button.
c) Go to this step to test the connectivity to an existing AireOS controller.

Step 5 To add a new AireOS controller, click **Add Controllers** from the **Step 3** area.

Figure 220: Add a New AireOS controller



Step 6 From the **Connector** drop-down list, choose a Connector.

Step 7 Enter the **Controller IP** address and **Controller Name**, and from the **Controller Type** drop-down list, choose **WLC (AireOS)** to connect to an AireOS controller.

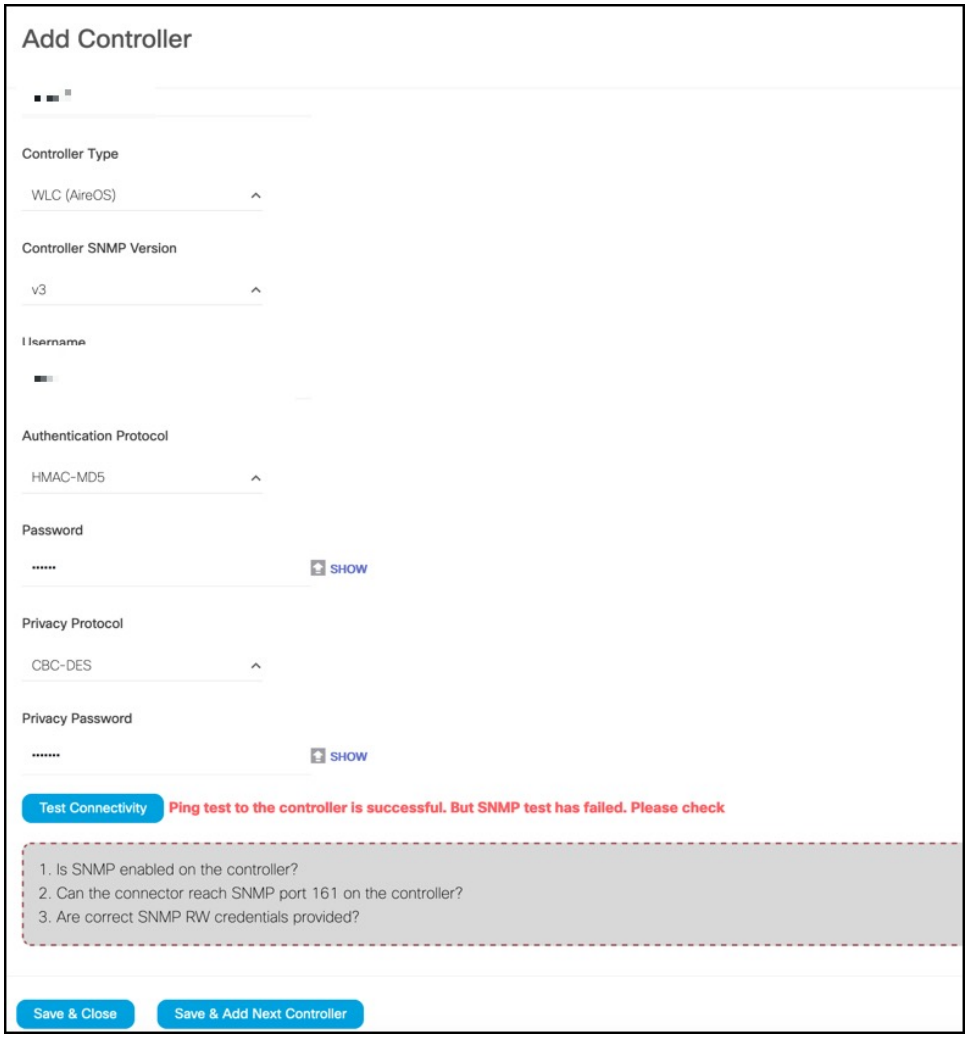
Step 8 From the **Controller SNMP Version** drop-down list, choose the SNMP version of the AireOS controller.

- If you choose the **SNMP** version as **v2C**, specify the SNMP read-write community.

- If you choose the **SNMP** version as **v3**, specify the SNMP v3 version username, password, and authentication protocol credentials. Ensure that SNMP v3 has read-write permissions in the AireOS controller.

Note
Both SNMP v2c and SNMP v3 must have read-write permission in the AireOS controller to register the Connector certificate in the AireOS controller. The Connector doesn't support SNMP v1.

Figure 221: Add a New AireOS controller



Step 9 Click **Test Connectivity** . Connector issues ping and SNMP commands to check the connectivity to Cisco Spaces using the credentials provided.

Note
Test Connectivity is enabled only when an active Connector is chosen.

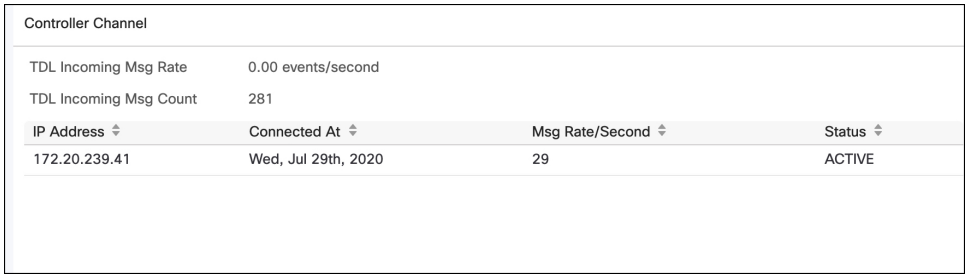
Table 20: Error Description

Status of PING	Status of SNMP Test	Displayed Test Connectivity Message
SUCCESSFUL	SUCCESSFUL	Connectivity test is successful

Status of PING	Status of SNMP Test	Displayed Test Connectivity Message
SUCCESSFUL	FAILED	Ping test is successful, but SNMP test failed. Check the following: Ping test to the AireOS controller is successful, but SNMP test has failed. Check the following: • If you are using v2c SNMP, check if the community strings are valid. • If you are using v3 SNMP, check if the credentials are correct. • Check if v2c or v3 mode is enabled in the controller.
FAILED	FAILED	Both ping and SSH test to the AireOS controller have failed. Check the following: • Is there IP connectivity between a Connector and a controller? • Is SSH enabled on the AireOS controller? • Is the SSH port 22 of the AireOS controller reachable from the Connector? • Have you provided accurate SSH credentials? • Is AAA enabled with local authentication? • Are you using an interface that is *not* the wireless management interface for NMSP and SSH connectivity?

Step 10 Click **Save**, and then click **Close**.
You can see the new Catalyst 9800 controller in the **Controller Channel** area of the Connector GUI. The Catalyst 9800 controller that is connected successfully to the Connector appears as **Active**. It takes approximately five minutes for the wireless controller to change to the **Active** state. Refresh your window to view the status change. The added Catalyst 9800 controller is also listed in the **Controller Channel** area of the Connector.

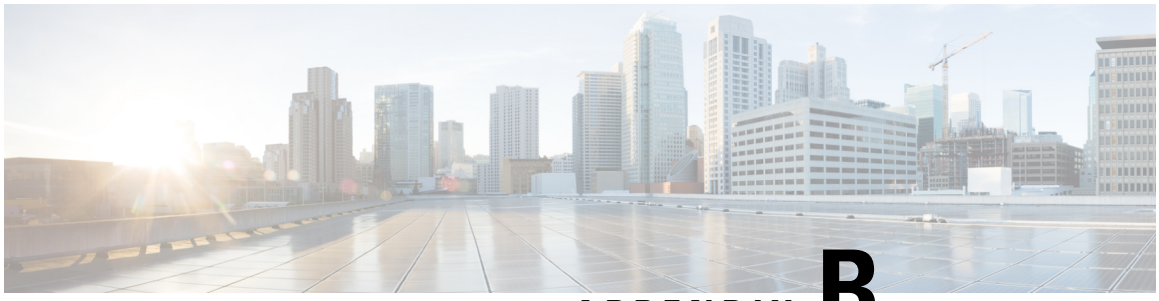
Figure 222: Details of the Catalyst 9800 controller



What to do next

You can import the added Catalyst 9800 controller to the Cisco Spaces location hierarchy.

Connect Connector to Cisco AireOS Wireless Controller



APPENDIX B

Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers

• Configure and Test the Connection Between Connector and Catalyst 9800 Controller, on page 249

Configure and Test the Connection Between Connector and Catalyst 9800 Controller

Before you begin

1. Deploy a connector OVA and activate it using a token from Cisco Spaces.
2. Note down the IP address of a Catalyst 9800 controller that is reachable from the Cisco Spaces: Connector.
3. On the Catalyst 9800 controller CLI, enter the config mode and enable AAA with local authentication using the **aaa authorization exec default local** and **aaa authentication login default local** commands.

   On the Catalyst 9800 controller CLI, run the following command in the **enable** mode:

   ```
   show run | sec  aaa
   ```

   From the output that is displayed, copy the configuration for **aaa authorization exec default**. In the **config** mode, append the configuration for local authentication to the copied configuration and configure the appended configuration.

   For instance, if the output displays **aaa authorization exec default group dnac-network-tacacs-group**, the appended configuration is **aaa authorization exec default group dnac-network-tacacs-group local**. This ensures that the existing configuration is not overwritten.

Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers



Note

Any certificate imported to the controller for Wireless Management Interface(WMI) that has been signed with a signature algorithm weaker than SHA-256 is not supported. Verify your certificate before adding the controller using the **show wireless management trustpoint** command.

```
Device# show wireless management trustpoint
Trustpoint Name  : manual_certs                 <<<<<<<<<<< Get the name of the trustpoint
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : f7900ae5e35473b5e32343d4ea9556176e71a63a
Private key Info : Available
FIPS suitability : Not Applicable
```

You can also verify the same using the **show crypto pki certificates verbose** command. In the output displayed, verify the content of the following fields (also highlighted in bold in the output):

- **Signature Algorithm**: Ensure that nothing less than SHA-256 is displayed here.
- **Associated Trustpoints**: Ensure that the signature algorithm is for the required trustpoint.

.

```
Device# show crypto pki certificates verbose
...
CA Certificate
  Status: Available
  Version: 3
  Certificate Serial Number (hex): 00AE697E4C7EEBE3E4
  Certificate Usage: Signature
  Issuer:
    e=support@vwlc.com
    cn=CA-vWLC-manual
    ou=Cisco DevX Wireless Simulator
    o=Cisco Virtual Wireless LAN Controller
    l=San Jose
    st=California
    c=US
  Subject:
    e=support@vwlc.com
    cn=CA-vWLC-manual
    ou=Cisco DevX Wireless Simulator
    o=Cisco Virtual Wireless LAN Controller
    l=San Jose
    st=California
    c=US
  Validity Date:
    start date: 18:08:16 Pacific Aug 27 2019
    end   date: 18:08:16 Pacific Aug 24 2029
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (4096 bit)
  Signature Algorithm: SHA256 with RSA Encryption
  Fingerprint MD5: 623E2FA4 7F908675 5422FF3C 257179F9
  Fingerprint SHA1: 05E3D17C 841AA033 C503D7BA 443CC2C2 1C510538
  X509v3 extensions:
    X509v3 Key Usage: 6000000
      Key Cert Sign
      CRL Signature
    X509v3 Subject Key ID: 1AE21C76 1B86780A B4E0AE43 205052BE EA0E4B4A
    X509v3 Basic Constraints:
        CA: TRUE
    X509v3 Authority Key ID: 1AE21C76 1B86780A B4E0AE43 205052BE EA0E4B4A
    Authority Info Access:
  Cert install time: 23:51:54 Pacific Jun 7 2024
```

Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers

```
  Associated Trustpoints: manual_certs
  Storage: nvram:supportvwlcc#E3E4CA.cer
”
...
```

Procedure

Step 1 Login to Cisco Spaces.

Step 2 In the Cisco Spaces dashboard, choose **Setup** > **Wireless Networks**.

Step 3 Expand the **Connect via Spaces Connector** area using the respective drop-down arrow to display a list of steps.

Step 4 To test the connectivity from the Connector to an existing Catalyst 9800 controller, click **View Controllers** in the **Step 3** Area.

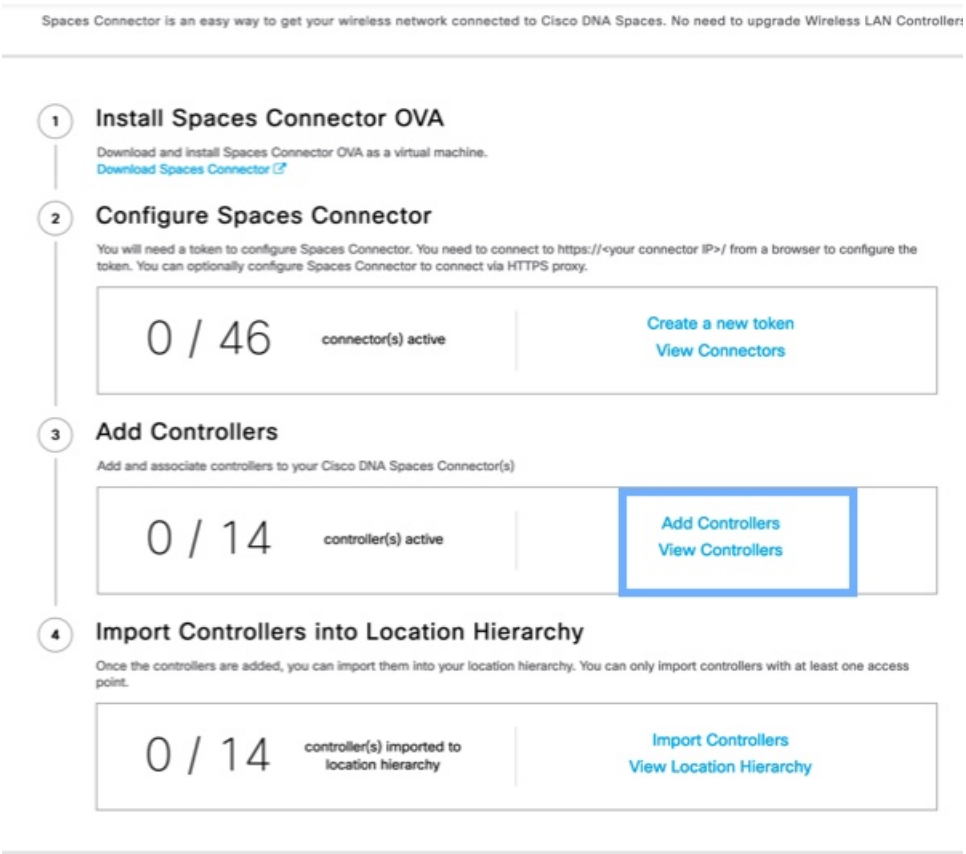
a) Click the pencil icon to edit a Catalyst 9800 controller.

b) Choose an active Connector from the **Connector** drop-down list to enable the **Test Connectivity** button.

c) Go to this step to test the connectivity to an existing AireOS controller.

Step 5 To add a new Catalyst 9800 controller, click **Add Controllers** from the **Step 3** Area.

Figure 223: Add a New Catalyst 9800 controller



Step 6 From the **Connector** drop-down list, choose a Connector.

Step 7 Enter the **Controller IP** address, **Controller Name**, and from the **Controller Type** drop-down list, choose **Catalyst WLC** to connect to a Cisco Catalyst 9800 Series Wireless Controllers.

Note
Ensure that the Controller IP address is not in the same subnet as the docker service network. You can validate this from the Connector CLI, where you can issue the **connectorctl dockersubnet show** command to verify the subnets used.

Step 8 Do one of the following:

- Enter **Netconf username**, **Netconf password**, and **Enable password**. This choice allows the Connector to recover gracefully from NMSP drops and push a fresh configuration to the Catalyst 9800 controller whenever required. If you have not configured an **enable** password in Catalyst 9800 controller you can skip configuring the **Enable password** in this step.
- Copy the configuration commands in the **Catalyst WLC CLI commands** section and run them manually on the Catalyst 9800 controller CLI.

Step 9 (Optional) Run the PING and SSH functionalities to test the reachability to the Catalyst 9800 controller and the credentials by clicking **Test Connectivity**. Note that **Test Connectivity** is available only for an active Connector.

Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers

Figure 224: Add a New Catalyst 9800 controller

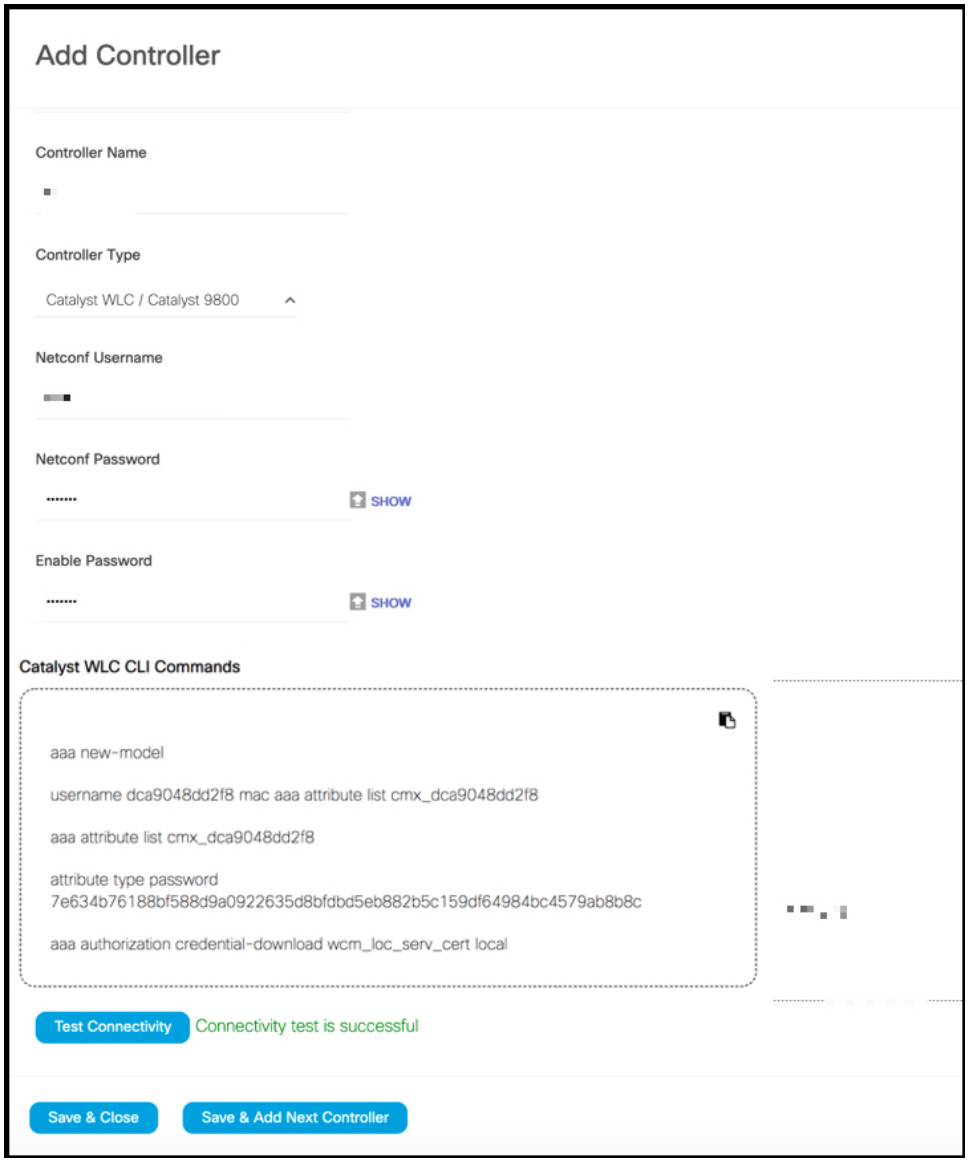


Table 21: Error Description

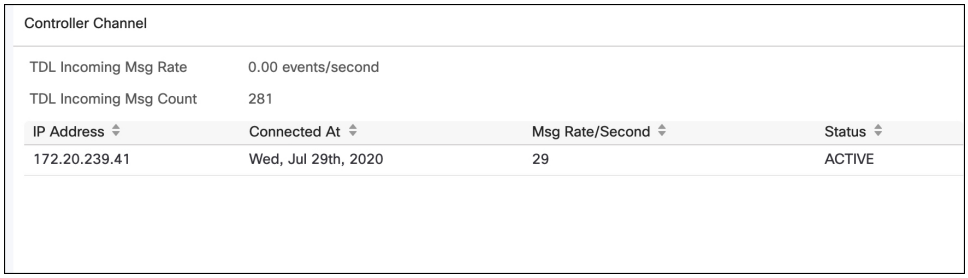
Status of PING	Status of SSH Credential Test	Meaning of status message combination and possible checks.
SUCCESSFUL	SUCCESSFUL	Connectivity test is successful.

Status of PING	Status of SSH Credential Test	Meaning of status message combination and possible checks.
SUCCESSFUL	FAILED	Ping test to the Catalyst 9800 controller is successful. But SSH test has failed. Check the following: **a.** Is SSH enabled on the controller? **b.** Is the SSH port 22 of the Catalyst 9800 controller reachable from the Connector? **c.** Have you provided accurate SSH read-write credentials?
FAILED	SUCCESSFUL	Connectivity test is successful.
FAILED	FAILED	Both Ping and SSH test to the Catalyst 9800 controller have failed. Check the following: **a.** Is there IP connectivity between Connector and controller? **b.** Is SSH enabled on the Catalyst 9800 controller? **c.** Is the SSH port 22 of the Catalyst 9800 controller reachable from the Connector? **d.** Have you provided accurate SSH credentials? **e.** Is AAA enabled with local authentication? **f.** Are you using an interface that is NOT the wireless management interface for NMSP and SSH connectivity?

Step 10 Click **Save**, and then click **Close**.
You can see the new Catalyst 9800 controller in the **Controller Channel** area of the Connector GUI. The Catalyst 9800 controller that is connected successfully to the Connector appears as **Active**. It takes approximately five minutes for the wireless controller to change to the **Active** state. Refresh your window to view the status change. The added Catalyst 9800 controller is also listed in the **Controller Channel** area of the Connector.

Connect Connector to Cisco Catalyst 9800 Series Wireless Controllers

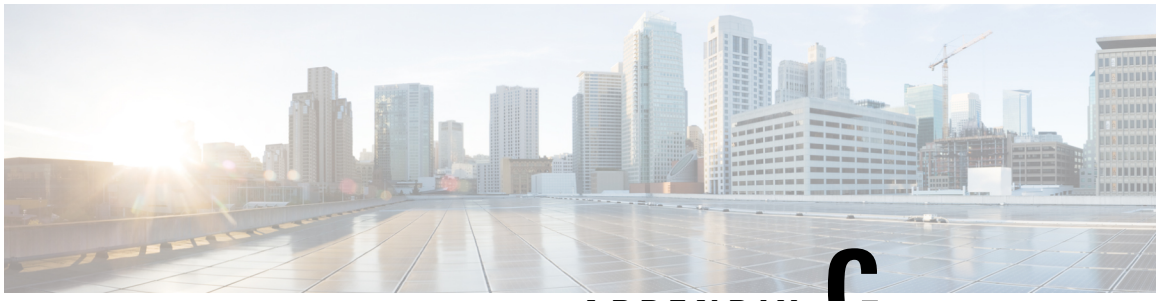
Figure 225: Details of the Catalyst 9800 controller



You can multiple Catalyst 9800 controllers to a Connector.

What to do next

You can import the added Catalyst 9800 controller to the Cisco Spaces location hierarchy.



APPENDIX C

Connect Connector to Cisco Catalyst 9300 or 9400 Series Switches

• Connecting a connector to Cisco Catalyst 9300 and 9400 Series Switches , on page 257

Connecting a connector to Cisco Catalyst 9300 and 9400 Series Switches

For certain use cases such as energy utilization or occupancy, the following steps are sufficient (and there is no further need to configure the IOX app). However, ensure that location services are enabled.

Before you begin

- Deploy a connector OVA and activate it using a token from Cisco Spaces.
- The IP address of a Cisco Catalyst 9300 and 9400 Series Switches that is reachable from the Cisco Spaces: Connector.
- Test the Netconf commands on the Cisco Catalyst 9300 and 9400 Series Switches

SUMMARY STEPS

1. Login to Cisco Spaces.
2. In the Cisco Spaces dashboard, choose **Setup > Wired Networks**.
3. From the **Step 3: Add Switches** area, click **Add Switch**.
4. From the **Add Switches** page, select the connector, enter a name to identify the switch, the switch IP address. **Netconf username**, **Netconf password**, and click the checkbox to acknowledge that you have tested these commands on the switch.
5. Click **Test** to see if the connection to the switch.
6. Do one of the following:
   - Click **Save & Add Next Switch**
   - Click **Save & Close**

Connect Connector to Cisco Catalyst 9300 or 9400 Series Switches

DETAILED STEPS

Procedure

Step 1 Login to Cisco Spaces.

Step 2 In the Cisco Spaces dashboard, choose **Setup** > **Wired Networks**.

Step 3 From the **Step 3: Add Switches** area, click **Add Switch**.

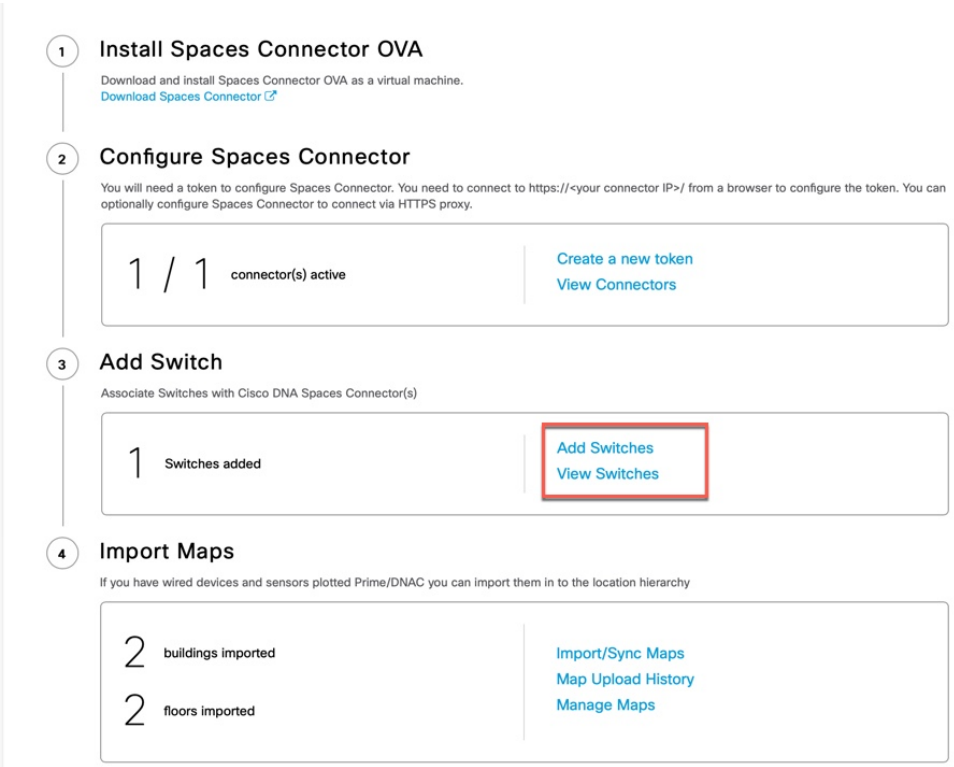


Figure 226:

Step 4 From the **Add Switches** page, select the connector, enter a name to identify the switch, the switch IP address. **Netconf username**, **Netconf password**, and click the checkbox to acknowledge that you have tested these commands on the switch.

Note
Ensure that the Controller IP address is not in the same subnet as the docker service network. You can validate this from the Connector CLI, where you can issue the **connectorctl dockersubnet show** command to verify the subnets used.

Step 5 Click **Test** to see if the connection to the switch.

Step 6 Do one of the following:

- Click **Save & Add Next Switch**
- Click **Save & Close**