



# Fabric Services System

Release 24.5.1

## User Guide

---

3HE 20072 AAAA TQZZA

Edition: 1  
May 2024

© 2024 Nokia.

Use subject to Terms available at: [www.nokia.com/terms](http://www.nokia.com/terms).

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

---

This document includes Nokia proprietary and confidential information, which may not be distributed or disclosed to any third parties without the prior written consent of Nokia.

This document is intended for use by Nokia's customers ("You"/"Your") in connection with a product purchased or licensed from any company within Nokia Group of Companies. Use this document as agreed. You agree to notify Nokia of any errors you may find in this document; however, should you elect to use this document for any purpose(s) for which it is not intended, You understand and warrant that any determinations You may make or actions You may take will be based upon Your independent judgment and analysis of the content of this document.

Nokia reserves the right to make changes to this document without notice. At all times, the controlling version is the one available on Nokia's site.

No part of this document may be modified.

NO WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF AVAILABILITY, ACCURACY, RELIABILITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, IS MADE IN RELATION TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT WILL NOKIA BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT, EVEN IN THE CASE OF ERRORS IN OR OMISSIONS FROM THIS DOCUMENT OR ITS CONTENT.

Copyright and trademark: Nokia is a registered trademark of Nokia Corporation. Other product names mentioned in this document may be trademarks of their respective owners.

© 2024 Nokia.

# Table of contents

<b>1</b>	<b>About this document.....</b>	<b>14</b>
1.1	What's new.....	14
1.2	Precautionary and information messages.....	15
1.3	Conventions.....	16
<b>2</b>	<b>Overview.....</b>	<b>17</b>
2.1	Concepts.....	17
2.2	Life cycle of a fabric.....	18
2.2.1	Design the fabric intent.....	18
2.2.2	Model in the Digital Sandbox.....	19
2.2.3	Deploy the fabric intent.....	20
2.2.4	Monitor the fabric intent.....	21
2.2.5	Repeat the cycle.....	21
2.3	Workload VPN intents.....	21
2.4	Maintenance intents.....	21
2.5	Additional resources.....	22
2.5.1	Documentation resources.....	22
2.5.2	Accessing the API descriptions from the system GUI.....	23
<b>3</b>	<b>User interface basics.....</b>	<b>24</b>
3.1	Signing in to the Fabric Services System.....	24
3.2	Signing out of the Fabric Services System.....	24
3.3	The dashboard.....	24
3.3.1	Notifications.....	28
3.4	The main menu.....	29
3.5	Detail panels.....	32
3.6	Lists.....	33
3.6.1	Customizing filters.....	35
<b>4</b>	<b>Deployment regions.....</b>	<b>36</b>
4.1	Region properties.....	38
4.2	Creating a region.....	41
4.3	Modifying a region.....	42
4.3.1	Implications of modifying a region.....	43

4.4	Updating a region to use iBGP.....	44
4.4.1	Updating region settings to use iBGP.....	44
4.4.2	Configuring fabric nodes as route reflectors.....	45
4.4.3	Updating existing fabric intents to use new iBGP settings.....	45
4.5	Viewing DHCP settings.....	46
4.6	The region map.....	47
4.6.1	Viewing and using the region map.....	48
4.6.2	Viewing and using the region list.....	48
4.6.3	Region map manipulation.....	49
4.7	The deployment pipeline.....	50
4.7.1	Viewing the deployment pipeline.....	52
4.7.2	Working with the deployment pipeline.....	52
4.8	The Fabric Intent List view.....	53
4.9	The Topology Map view.....	54
<b>5</b>	<b>Fabric intents.....</b>	<b>56</b>
5.1	Fabric topology.....	56
5.2	The design-and-deploy workflow.....	56
5.2.1	Create the fabric intent.....	57
5.2.2	Explore the fabric intent.....	58
5.2.3	Create and assign labels.....	58
5.2.4	Associate nodes with hardware.....	58
5.2.5	Add the fabric intent to the deployment pipeline.....	58
5.2.6	Deploy the fabric intent to hardware.....	58
5.3	Notable fabric intent configuration values.....	59
5.4	Supported hardware.....	59
5.4.1	Hardware-driven exceptions and special cases.....	60
5.5	Software and image catalogs.....	61
5.6	Fabric intents page.....	61
5.7	Manual fabric topologies.....	64
5.7.1	Elements of a topology file.....	65
5.7.2	Manual topology file examples.....	71
5.7.3	The Topologies page.....	86
5.7.4	Manual fabric topology parameters.....	89
5.7.5	Importing a manual topology.....	90
5.7.6	Viewing a manual topology.....	90

5.7.7	Exporting a manual topology.....	91
5.7.8	Creating a fabric intent using a manual topology.....	91
5.7.9	Updating a fabric with a new manual topology.....	93
5.8	Fabric intents with unmanaged nodes.....	94
5.8.1	Creating a fabric intent for unmanaged nodes.....	98
5.9	Viewing a fabric intent.....	100
5.9.1	Groups.....	100
5.9.2	Information displays.....	102
5.9.3	Error indicators.....	103
5.9.4	Viewing the event log.....	105
5.9.5	Viewing a fabric intent as code.....	106
5.9.6	Viewing the configuration file for a single node.....	107
5.9.7	Downloading the initial node configuration.....	107
5.9.8	Viewing the fabric inventory.....	108
5.9.9	Viewing fabric links.....	108
5.9.10	Viewing edge links.....	109
5.9.11	Generating a wiring plan.....	110
5.10	LAG management.....	110
5.10.1	Creating LAGs.....	110
5.11	Configuring a breakout port.....	113
5.12	Fabric intent modification.....	114
5.12.1	Editing a fabric intent.....	115
5.12.2	Discarding changes to a fabric intent.....	115
5.12.3	Creating a new version of a fabric intent.....	116
5.12.4	Duplicating a fabric intent.....	117
5.13	Deleting a fabric intent.....	118
5.14	Fabric intent deployment.....	118
5.14.1	Adding a fabric intent to the deployment pipeline.....	118
5.14.2	Deploying a fabric intent from the deployment pipeline.....	119
5.14.3	The Deployment Trigger Percentage setting.....	120
5.15	Deviations.....	121
5.15.1	Viewing deviations.....	122
5.15.2	Accepting or rejecting deviations.....	123
<b>6</b>	<b>Workload VPN intents.....</b>	<b>125</b>
6.1	Role of a workload VPN intent.....	125

6.2	Elements of a workload VPN intent.....	125
6.2.1	Fabrics.....	126
6.2.2	Subnets.....	127
6.2.3	Sub-interfaces.....	128
6.2.4	QoS profiles.....	128
6.2.5	ACL profiles.....	128
6.2.6	Routers.....	129
6.3	Viewing a workload VPN intent.....	129
6.3.1	Workload VPN intent view.....	130
6.3.2	Viewing a workload VPN intent as code.....	131
6.3.3	Viewing the workload VPN intent event log.....	132
6.4	Profile manager.....	133
6.4.1	Deploying the Profile Manager.....	133
6.4.2	Match groups.....	134
6.4.2.1	Creating a match group.....	134
6.4.2.2	Editing a match group.....	136
6.4.2.3	Deleting a match group.....	136
6.4.3	QoS profile management.....	137
6.4.3.1	Creating a QoS profile.....	137
6.4.3.2	Editing and deploying a QoS profile.....	139
6.4.3.3	Deleting a QoS profile.....	139
6.4.4	ACL profile management.....	140
6.4.4.1	Creating an ACL profile.....	141
6.4.4.2	Editing and deploying an ACL profile.....	144
6.4.4.3	Deleting an ACL profile.....	144
6.5	Workload VPN intent creation.....	145
6.5.1	Workload VPN intent parameter descriptions.....	146
6.5.1.1	Workload VPN intent parameters.....	146
6.5.1.2	Subnet parameters.....	147
6.5.1.3	Sub-interface parameters.....	151
6.5.1.4	Router parameters.....	153
6.5.2	Creating the basic workload VPN intent.....	154
6.5.3	Adding subnets to the workload VPN intent.....	155
6.5.4	Adding sub-interfaces to the workload VPN intent.....	157
6.5.5	Creating a router.....	159
6.5.6	Routing.....	159

6.5.6.1	Displaying the routing view for a node.....	160
6.5.6.2	BGP parameters.....	161
6.5.6.3	Configuring BGP.....	163
6.5.6.4	Static route parameters.....	165
6.5.6.5	Configuring static routes.....	167
6.5.6.6	Aggregate route parameters.....	169
6.5.6.7	Configuring aggregate routes.....	170
6.6	Workload VPN intent deployment.....	171
6.6.1	Adding a workload VPN intent to the deployment pipeline.....	172
6.6.2	Deploying a workload VPN intent from the deployment pipeline.....	172
6.7	Workload VPN intent modification.....	173
6.7.1	Editing a workload VPN intent.....	173
6.7.2	Creating a new version of a workload VPN intent.....	174
6.7.3	Updating a group of sub-interfaces by label reference.....	174
6.8	Deleting a workload VPN intent.....	175
<b>7</b>	<b>Configuration overrides.....</b>	<b>177</b>
7.1	Global configuration overrides.....	177
7.1.1	Global configuration override parameters.....	180
7.1.2	Creating a global configuration override.....	182
7.2	Contextual configuration overrides.....	184
7.2.1	Contextual configuration override parameters.....	186
7.2.2	Creating a contextual configuration override.....	187
7.3	Viewing and managing configuration overrides.....	190
<b>8</b>	<b>Traffic mirroring.....</b>	<b>193</b>
8.1	Mirror source groups.....	193
8.2	Mirror destinations.....	194
8.3	Configuration parameters.....	194
8.4	Configuring mirroring sources.....	197
8.5	Configuring a mirror destination.....	198
8.6	Configuring a mirroring instance.....	199
<b>9</b>	<b>Maintenance intents.....</b>	<b>200</b>
9.1	Viewing a maintenance intent.....	200
9.1.1	Elements of the maintenance intent Design view.....	201

9.1.2	Viewing affected nodes.....	202
9.1.3	Viewing the maintenance intent event log.....	202
9.2	Creating a maintenance label.....	203
9.3	Labeling objects for maintenance.....	204
9.4	Creating a maintenance intent.....	204
9.5	Duplicating a maintenance intent.....	207
9.6	Maintenance intent deployment.....	207
9.6.1	Adding a maintenance intent to the deployment pipeline.....	208
9.6.2	Progress of a deployed maintenance intent.....	208
9.6.3	Deploying a maintenance intent from the deployment pipeline.....	209
9.7	Removing a maintenance intent from the deployment pipeline.....	210
9.8	Aborting a maintenance intent.....	210
<b>10</b>	<b>Labels.....</b>	<b>212</b>
10.1	Label types.....	212
10.1.1	Nokia pre-defined labels.....	212
10.1.2	User-configured custom labels.....	213
10.2	The Label Factory.....	214
10.3	Viewing available labels.....	215
10.4	Creating a label.....	215
10.5	User-configured label manipulation.....	216
10.5.1	Editing an existing label.....	216
10.5.2	Deleting an existing label.....	216
10.6	Label assignments to fabric intent elements.....	217
10.6.1	Assigning labels to a fabric intent.....	217
10.6.2	Assigning a label to a specific node in a fabric intent.....	218
10.6.3	Assigning labels to a fabric link.....	218
10.6.4	Assigning labels to an edge link interface.....	219
10.6.5	Assigning labels to ISL interfaces.....	220
10.6.6	Removing a label assigned to a fabric intent.....	220
10.6.7	Removing a label assigned to a node in a fabric intent.....	221
10.6.8	Removing a label assigned to a fabric link.....	221
10.6.9	Removing a label assigned to an edge link interface.....	222
10.6.10	Removing a label assigned to an ISL interface.....	222
10.7	Label assignments to workload VPN intent elements.....	223
10.7.1	Assigning labels to a workload VPN intent.....	223

10.7.2	Assigning a label to a specific sub-interface.....	223
10.7.3	Removing a label assigned to a workload VPN intent.....	224
10.7.4	Removing a label assigned to a specific sub-interface.....	224
10.8	Label assignments to management profiles.....	225
10.8.1	Assigning a label to a management profile.....	225
10.8.2	Removing a label assigned to a management profile.....	225
10.9	Label assignment management.....	226
10.9.1	Viewing the assignments of a specific label.....	226
10.9.2	Querying the label assignment list.....	226
10.9.3	Label assignment queries.....	227
<b>11</b>	<b>Inventories.....</b>	<b>229</b>
11.1	Fabric elements inventory.....	229
11.1.1	Viewing the inventory of a single fabric intent.....	229
11.1.2	Viewing details about a node in the inventory.....	230
11.1.3	Node states.....	231
11.1.4	Inventory manipulation.....	232
11.1.4.1	Editing node information.....	232
11.1.4.2	Viewing the configuration file for a single node.....	233
11.1.4.3	Planned node and real-world hardware association.....	233
11.1.4.4	Disassociating planned nodes from real hardware.....	236
11.1.4.5	Changes to node associations.....	237
11.1.4.6	Updating the system name of a node in a fabric inventory.....	238
11.1.4.7	Viewing platform details for nodes in the inventory.....	238
11.1.4.8	Inventory items in spreadsheet format.....	239
11.1.4.9	Uploads of inventory items.....	241
11.1.5	Viewing the overall inventory.....	243
11.2	Management profiles.....	243
11.2.1	Management profile parameters.....	244
11.2.2	Creating a management profile.....	245
11.2.3	Editing a management profile.....	246
11.2.4	Assigning a management profile to a node.....	247
11.2.5	Deleting a management profile.....	247
<b>12</b>	<b>Alarms.....</b>	<b>249</b>
12.1	Displaying alarms.....	249

12.2	Customizing an alarm severity level.....	251
12.3	Third-party tool access to Fabric Services System alarms.....	253
12.3.1	Enabling Kafka alarms after software installation.....	253
12.3.2	Updating configuration for the external Kafka service.....	254
<b>13</b>	<b>Operations views.....</b>	<b>260</b>
13.1	Viewing the operational topology.....	263
13.2	Viewing operational insights.....	265
<b>14</b>	<b>Network resources.....</b>	<b>267</b>
14.1	The Network Resources page.....	268
14.2	Network Resource properties.....	269
14.3	Creating IP and Autonomous System pools.....	271
14.4	Managing Network Resources pools.....	273
<b>15</b>	<b>Digital Sandbox.....</b>	<b>275</b>
15.1	Integration with the Fabric Services System.....	275
15.1.1	Digital Sandbox status display.....	275
15.2	Creating a region in the Digital Sandbox.....	276
15.2.1	Modifying a region in the Digital Sandbox.....	276
15.3	Fabric intents and the Digital Sandbox.....	277
15.3.1	Creating a fabric intent in the Digital Sandbox.....	277
15.3.2	Updating the Digital Sandbox.....	278
15.3.3	Deploying a fabric intent in the Digital Sandbox.....	279
15.4	Workload VPN intents.....	279
15.4.1	Creating a workload VPN intent in the Digital Sandbox.....	279
15.4.2	Deploying a workload VPN intent to the Digital Sandbox.....	280
15.4.3	Updating the Digital Sandbox.....	280
<b>16</b>	<b>System administration.....</b>	<b>282</b>
16.1	Application settings.....	282
16.1.1	Viewing software and image catalogs.....	282
16.1.1.1	Adding a new software image.....	283
16.1.2	Common application settings.....	284
16.1.2.1	Configuring Geomap Tile Server settings.....	284
16.2	Adding a new network operating system version to the software catalog.....	284
16.3	Uploading SR Linux container images for Digital Sandbox.....	285

16.4	User and resource management.....	287
16.4.1	Roles.....	288
16.4.1.1	Viewing a list of existing roles.....	289
16.4.1.2	Predefined roles.....	289
16.4.1.3	Creating a role.....	289
16.4.1.4	Modifying the resource access permissions of a role.....	290
16.4.1.5	Deleting a role.....	290
16.4.2	Resource groups.....	291
16.4.2.1	Viewing a list of resource groups.....	291
16.4.2.2	Predefined resource groups.....	291
16.4.3	User groups.....	293
16.4.3.1	Viewing a list of existing user groups.....	293
16.4.3.2	Predefined user groups.....	294
16.4.3.3	Creating a user group.....	294
16.4.3.4	Assigning the role of a user group.....	295
16.4.3.5	Deleting a user group.....	296
16.4.4	Users.....	296
16.4.4.1	Viewing a list of existing users.....	296
16.4.4.2	Creating a new user.....	296
16.4.4.3	Assigning a user to a user group.....	297
16.4.4.4	Assigning a role to a user.....	298
16.4.4.5	Deleting a user.....	298
16.5	Health monitoring.....	299
16.5.1	Updating the health monitoring configuration after installation.....	301
16.6	Recovery after application node failure.....	302
16.6.1	Recovering an application after node failure.....	303
16.6.2	Recovering an application after node reboot.....	305
16.7	Repairing a Fabric Services System cluster.....	306
16.7.1	Updating the deployer installation file.....	306
16.7.2	Removing a node from a cluster.....	307
16.7.3	Adding a worker node to a cluster.....	308
16.7.4	Adding a master node to a cluster.....	310
16.8	Backup and restore.....	312
16.8.1	Backing up.....	314
16.8.2	Restore a backup and install the Fabric Services System application.....	316
16.8.2.1	Setting up new Fabric Services System nodes.....	316

16.8.2.2	Installing the Kubernetes cluster.....	317
16.8.2.3	Restoring a backup.....	317
16.8.2.4	Installing the Fabric Services System application.....	318
16.8.2.5	Restoring Digital Sandbox fabrics.....	319
16.9	Geo-redundancy.....	320
16.9.1	Geo-redundancy for systems integrations with Connect plugins.....	321
16.9.2	Geo-redundancy operations.....	321
16.9.2.1	REST API geo-redundancy operations.....	322
16.9.3	Geo-redundancy configuration.....	323
16.9.3.1	Configuring geo-redundancy information in deployer VMs.....	324
16.9.3.2	Verifying that the setup is ready for geo-redundancy using the deployer VMs..	325
16.9.3.3	Realigning certificates.....	329
16.9.3.4	Geo-redundancy parameters.....	331
16.9.3.5	Configuring geo-redundancy.....	332
16.9.4	Sync failure and recovery.....	334
16.9.4.1	Recovering from sync failure.....	335
16.9.4.2	Initiating failover: switching between the active and standby clusters.....	335
16.9.5	Converting a geo-redundant system to a standalone system.....	336
16.9.6	Geo-redundancy status and statistics.....	337
16.10	The technical support script.....	339
16.10.1	How to run the technical support script.....	339
16.11	Node discovery.....	340
16.11.1	Participants.....	340
16.11.2	The discovery and configuration process.....	341
16.11.3	Using an external DHCP server.....	342
<b>17</b>	<b>Security.....</b>	<b>343</b>
17.1	Platform password management.....	343
17.1.1	Changing passwords for internal services.....	343
17.2	User password management.....	346
17.2.1	Password policy parameters.....	347
17.2.2	Managing user password policies.....	349
17.2.3	Managing a user profile and password.....	350
17.2.4	Resetting internal passwords.....	350
17.3	Certificate management.....	351
17.3.1	Managing certificates.....	353

---

17.3.1.1	Deploying a user-provided CA certificate.....	353
17.3.1.2	Deploying a user-provided server certificate for northbound services.....	354
17.3.1.3	Deploying a user-provided node CA certificate.....	355
17.3.1.4	Generating a CSR.....	355
17.3.1.5	Generate a self-signed root CA certificate.....	356
17.3.1.6	Displaying certificates.....	357
17.3.2	Uploading a customer-generated root CA to the trust store.....	361
17.4	LDAP server integration.....	361
17.4.1	Federation Provider parameters.....	361
17.4.2	Integrating an LDAP server.....	364
17.4.3	Synchronizing with the LDAP server.....	365
17.4.4	Managing the Federation Provider.....	365
17.4.5	Configuring LDAP server details.....	366
17.5	Forwarding user audit logs to a remote server.....	366
17.5.1	Configuring a remote syslog server for user audit logs.....	367
<b>Appendix A: Supported alarms.....</b>		<b>369</b>
<b>Appendix B: Protobuf file message format.....</b>		<b>395</b>
<b>Appendix C: Acronyms.....</b>		<b>401</b>

# 1 About this document

This Fabric Services System *User Guide* describes the system's user interface (UI), and includes procedures that guide you through the design and deployment of a fabric intent.

This document is intended for network technicians, administrators, operators, service providers, and others who use the Fabric Services System.



**Note:**

This manual covers the current release and may also contain some content that will be released in later maintenance loads. See the *Fabric Services System Release Notes* for information about features supported in each load.

## 1.1 What's new

This section lists the changes that were made in this release.

*Table 1: What's new in Release 24.5.1*

Description	Location
System administration	
Support for geo-redundancy	<a href="#">Geo-redundancy</a> <a href="#">Sync failure and recovery</a> <a href="#">Geo-redundancy operations</a> <a href="#">REST API geo-redundancy operations</a> <a href="#">Geo-redundancy parameters</a> <a href="#">Realigning certificates</a> <a href="#">Configuring geo-redundancy</a> <a href="#">Initiating failover: switching between the active and standby clusters</a> <a href="#">Making the standby system active - standalone operation</a> <a href="#">Converting a geo-redundant system to a standalone system</a> <a href="#">Geo-redundancy status and statistics</a> <a href="#">Predefined resource groups</a> , added the RG-sync group <a href="#">Predefined roles</a> , added the geored user <a href="#">Restore a backup and install the Fabric Services System application</a>
Added a note about Keycloak handling of user passwords	<a href="#">Users</a>

Description	Location
Fabric intents	
<ul style="list-style-type: none"> <li>IXR-D4 is supported as a leaf and spine node.</li> <li>IXR-D5 is supported as a leaf, spine, and border-leaf node.</li> </ul>	<a href="#">Supported hardware</a>
Workload intents	
Support for static MAC address on subnet gateway (IRB)	<a href="#">Workload VPN intent parameter descriptions</a>
Support for configurable BFD timers	<a href="#">Workload VPN intent parameter descriptions</a>
Security	
Updated the procedure to include connect and geored users	<a href="#">Resetting internal passwords</a>
Updated the validity of northbound server certificates from 90 to 365 days	<a href="#">Certificate management</a>
Alarms	
New alarms are supported in this release	<a href="#">Appendix A: Supported alarms</a>

## 1.2 Precautionary and information messages

The following are information symbols used in the documentation.



**DANGER:** Danger warns that the described activity or situation may result in serious personal injury or death. An electric shock hazard could exist. Before you begin work on this equipment, be aware of hazards involving electrical circuitry, be familiar with networking environments, and implement accident prevention procedures.



**WARNING:** Warning indicates that the described activity or situation may, or will, cause equipment damage, serious performance problems, or loss of data.



**Caution:** Caution indicates that the described activity or situation may reduce your component or system performance.



**Note:** Note provides additional operational information.



**Tip:** Tip provides suggestions for use or best practices.

---

## 1.3 Conventions

Commands use the following conventions

- **Bold** type indicates a command that the user must enter.
- Input and output examples are displayed in `Courier` text.
- An open right angle bracket indicates a progression of menu choices or simple command sequence (often selected from a user interface). Example: **start > connect to**
- Angle brackets (< >) indicate an item that is not used verbatim. For example, for the command **show ethernet <name>**, **name** should be replaced with the name of the interface.
- A vertical bar (|) indicates a mutually exclusive argument.
- Square brackets ([ ]) indicate optional elements.
- Braces ({ }) indicate a required choice. When braces are contained within square brackets, they indicate a required choice within an optional element.
- *Italic* type indicates a variable.

Examples use generic IP addresses. Replace these with the appropriate IP addresses used in your system.

## 2 Overview

The Fabric Services System supports intent-based automation for the management of data center fabrics, including:

- planning, designing, and validating a prospective fabric. These are sometimes called "Day 0" activities, and precede actual deployment.
- deploying a planned fabric configuration to hardware for the first time; also, designating resources within your deployed fabric for allocation to specific sources of demand upon its traffic and processing capacity. These are sometimes called "Day 1" activities, and pertain to the deployment itself.
- telemetry monitoring, state monitoring, and subsequent re-configuration of a fabric. These are sometimes called "Day 2+" activities, and are performed some time after the initial deployment.

This *User Guide* describes how to use the system's Graphical User Interface (GUI) to design, test, and manage your data center fabric and the to manage the workloads that place storage and processing demands upon them.

### 2.1 Concepts

The Fabric Services System uses the following concepts for describing and managing the elements of a fabric:

- **Fabric:** a group of switches that are managed as a single logical unit. A single data center can include many complementary and mutually supporting fabrics. These fabrics form an structured "underlay" onto which connections and services can be superimposed.

Fabrics are designed in a hierarchical structure. Typically this structure extends:

- from one or more backbone nodes, providing a gateway that manages communication externally to a WAN
- to one or more spine nodes that aggregate and distribute traffic
- to a collection of leaf nodes each associated with one or more spine nodes
- and from those leaf nodes through edge link interfaces to termination points

Although that description is typical, the presence or absence, size, and role of each level varies widely in actual fabric designs.

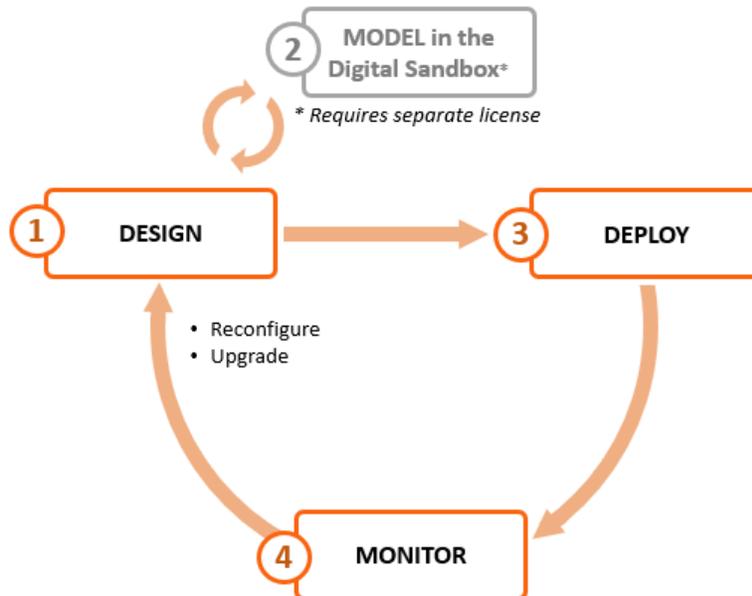
- **Fabric intent:** a set of node configurations that, when deployed to hardware, result in a functioning fabric. An intent gathers and deploys all of the required node configuration tasks into a single transaction, which then deploy successfully or not at all. If the deployment of any participating node's configuration fails at any point, the entire transaction is rolled back to restore all of the participating nodes to their state before the deployment of the intent began.
- **Workload:** a single source of demand upon a data center's fabrics. For example, all of the traffic from a single customer, or tenant, could be directed to only those resources encompassed by a particular workload VPN intent. A workload is an overlay that can be superimposed upon the fabric's previously configured underlay structure.

- **Workload VPN intent:** a set of node configurations pertaining to a specific subset of fabric resources (one or more fabrics, subnets, and sub-interfaces). After the workload VPN intent is defined and its configuration data has been deployed to participating nodes, the fabric resources encompassed by a workload VPN intent can be made available to manage a particular workload. Like a fabric intent, the Fabric Services System deploys a workload VPN intent to all participating nodes as a single transaction. This ensures that the entire deployment succeeds completely, or else fails completely and cleanly, leaving nodes in their pre-deployment states.

## 2.2 Life cycle of a fabric

The Fabric Services System provides tools to assist you in designing, deploying, and managing your data center fabric using the high-level cycle shown in [Figure 1: Fabric life cycle](#).

Figure 1: Fabric life cycle



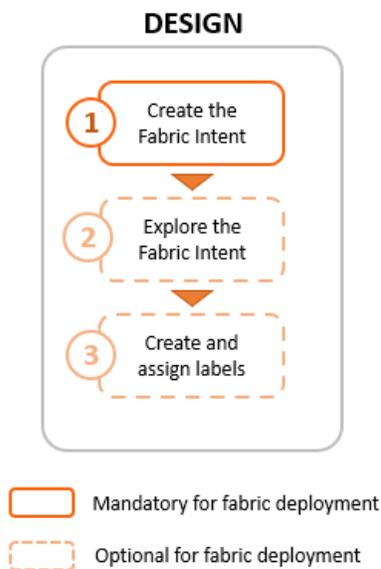
### 2.2.1 Design the fabric intent

Using the Fabric Services System, you can design a new fabric with a few clicks. The system then creates a detailed fabric intent that identifies all of the necessary nodes and includes all of the configuration files necessary to provision those nodes when they are available.

You can review and modify this fabric intent before proceeding with deployment, using the system's built-in deployment system to reconfigure participating nodes.

During design, you indicate whether you are designing a fabric intent for deployment to real hardware, or to the Fabric Services System Digital Sandbox, a simulator where you can validate prospective fabric designs. The Digital Sandbox is described in [Model in the Digital Sandbox](#).

Figure 2: Design workflow



To create the initial design for your fabric intent, you first import a topology description from a prepared file. The system then generates a recommended topology for the fabric you described. The procedure for creating a fabric intent is described in [Fabric intents](#).

After creating your fabric intent, you can (optionally) explore the fabric topology and view details about its constituent nodes and links.

Labels are tags that help you group and organize fabric objects according to specific criteria. You can create labels in the Label Factory and then assign them to fabric intents or to objects within a fabric intent.

Labels are a powerful tool you can use to identify a group of objects that can then be subject to collective actions, such as software updates. For example, you can create a label and apply it to a group of items that are subject to a particular upgrade. Then, you can apply the upgrade to all of the nodes with that label as a single action, instead of upgrading them individually.

### Related topics

[Digital Sandbox](#)

[Labels](#)

## 2.2.2 Model in the Digital Sandbox

If you have purchased the license for the Fabric Services System's Digital Sandbox, you can deploy your fabric to a virtual environment to evaluate its design without relying on physical hardware.

The Digital Sandbox is a network simulator that can emulate data center fabric designs (underlays) and the workload constraints configured upon those fabrics (overlays). Each SR Linux node within the fabric is emulated as its own virtual machine within the cluster, running its own copy of the SR Linux operating system.

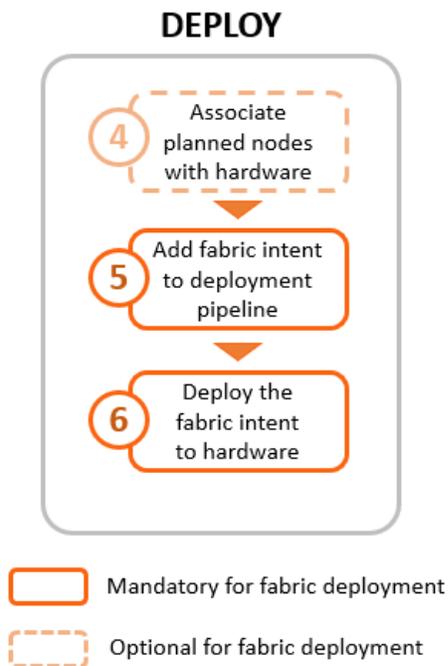
In its current form, the Digital Sandbox can emulate a region, the structures of fabrics within that region, and the workload constraints that are configured upon those fabrics (including the edge links that are

referred to by the workload). It does not yet simulate dynamic features like traffic flow between the simulated nodes and their endpoints.

### 2.2.3 Deploy the fabric intent

When you are satisfied with the fabric design, you can use the system to deploy your fabric intent to physical hardware.

Figure 3: Deploy workflow



Before deployment, you can associate any planned node in the fabric intent with its available, physical counterpart by providing the corresponding hardware serial number. You can associate individual nodes directly in the UI, or associate nodes in bulk by exporting a spreadsheet from the fabric intent, adding the necessary information for each node, and then re-importing the spreadsheet.

Whether you have performed this association or not, you can then add the fabric intent to the deployment pipeline. This pipeline manages all of the intents awaiting deployment, of which there can be many when multiple operators are simultaneously designing and maintaining multiple fabrics.

After the fabric intent is in the deployment pipeline, you can deploy the fabric. The system deploys configuration data to all of the participating, available nodes as a single transaction.

Even if none of the fabric's planned nodes are available when you design your fabric intent, you can still proceed with deployment. The system can deploy the planned fabric onto nodes as they become available; the fabric intent persists in the deployment pipeline until the entire deployment is complete. This allows you to plan a large fabric at the outset, and then deploy it in segments as the hardware become available over time.

#### Related topics

[Inventories](#)

## *Fabric intents*

### 2.2.4 Monitor the fabric intent

You can use the Operations features of the Fabric Services System to monitor the behavior of your fabric. These tools clearly signal any failure or design deviation, enabling you to take remedial steps as quickly as possible.

### 2.2.5 Repeat the cycle

As node replacement or software updates become necessary, you can use the system's design capabilities to again plan the necessary changes as a new version of the existing fabric intent. You can also model the planned update in the Digital Sandbox before deployment.

When the updated design is satisfactory, you can use the system's deployment capabilities to commit the changes to your working fabric.

You can repeat this cycle as often as needed throughout the lifetime of your fabric.

## 2.3 Workload VPN intents

The fabrics you deploy within a data center can carry traffic belonging to many customers (or "tenants"), all of which impose their own traffic, processing, and storage demands.

The Fabric Services System allows you to define a set of fabric resources that can be used to support an individual workload. This distributes the traffic and processing load efficiently over your fabric, and ensures that a tenant has reliable access to their share of the fabric's capacity. The tool that the Fabric Services System uses to allocate fabric resources is the workload VPN intent.

A workload VPN intent can encompass one or more fabrics. It identifies a set of subnets and their sub-interfaces within the participating fabrics to be made available to a particular source of demand. Sub-interfaces can be configured with Quality of Service (QoS) settings to prioritize traffic and Access Control Lists (ACLs) to accept or reject traffic originating from, or headed to, specific IP addresses.

The Fabric Services System uses the concept of an intent to encompass the set of node configurations that embody the workload VPN intent. The system deploys the workload VPN intent to all participating nodes as a single transaction. This approach ensures that the entire deployment succeeds completely, or else fails completely and cleanly to facilitate another attempt.

#### **Related topics**

[Workload VPN intents](#)

## 2.4 Maintenance intents

A maintenance intent identifies a set of nodes and a configuration change that the Fabric Services System deploys to those nodes.

The Fabric Services System supports two types of maintenance intent:

- software changes, which upgrades or downgrades the SR Linux software version running on one or more nodes within a single fabric.
- node replacement, which updates the hardware association for an existing node to a new piece of matching hardware with a different serial number, and downloads the necessary configuration files to the new node so that it can fully resume the role of the hardware it replaced.

### Related topics

[Maintenance intents](#)

## 2.5 Additional resources

You can obtain more information about the Fabric Services System user interface and its API from the other documents in the Fabric Services System documentation suite.

For the API, more information is available from the Swagger interface directly in the Fabric Services System UI.

### 2.5.1 Documentation resources

Table 2: Fabric Services System information sources

Document	Description
<i>Fabric Services System Release Notes</i>	Release notes accompany every release of the Fabric Services System. These highlight the most up-to-date information about supported features, supported hardware, known limitations, and resolved issues.
<i>Fabric Services System Software Installation Guide</i>	The <i>Software Installation Guide</i> documents the system requirements for the Fabric Services System and detailed procedures for obtaining and installing the Fabric Services System software.
Fabric Services System contextual help	Contextual help is available for many pages within the Fabric Services System UI. To view contextual help for the current page, click  near the upper right of the page.
<i>Fabric Services System User Guide</i>	The <i>User Guide</i> is the primary source for conceptual and practical information about everything you can do in the Fabric Services System GUI. It explores the concepts behind the Fabric Services System, describes the common features available throughout the UI, and provides detailed procedures you can follow to perform all of the main operations available within the system.

## 2.5.2 Accessing the API descriptions from the system GUI

### About this task

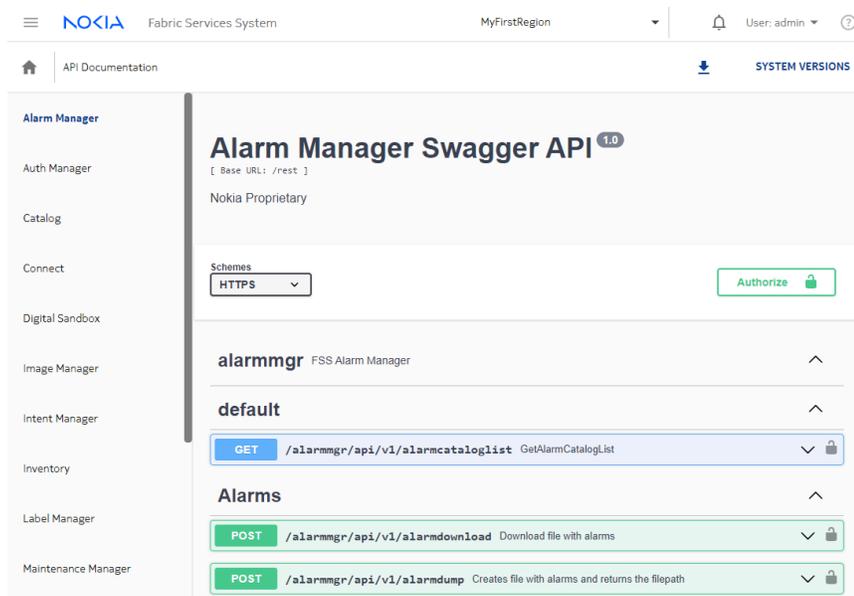
You can view information about API calls directly in the Fabric Services System GUI at any time. This view provides you with the most up-to-date list of APIs for your system and release.

From the system GUI, you can view currently supported APIs, read a description of their purpose, and test API calls using the Swagger interface.

### Procedure

**Step 1.** Sign in to the Fabric Services System.

**Step 2.** In your browser's URL field, enter the following to open the Swagger API interface: `<GUI URL>/apidocs`



## 3 User interface basics

The Fabric Services System graphical user interface (GUI) allows you to create, view, deploy, monitor, and update fabric intents, workload VPN intents, and maintenance intents. It also allows you to configure supporting systems that enable the management of those intents.

This section describes common operations in the user interface, and how to use some elements that recur throughout the GUI.

### 3.1 Signing in to the Fabric Services System

#### About this task

Follow this procedure to sign into the Fabric Services System user interface using your browser.

#### Procedure

**Step 1.** Open the login page in your browser by navigating to:

http://<server IP address>:8090/login

**Step 2.** Enter a username and password.

**Step 3.** Optional: Select the **Remember username** check box to reuse the same credentials for your next sign-in.

**Step 4.** Click **SIGN IN**.

The first time you sign in to the Fabric Services System, a welcome form displays. This form advises you to begin by creating a region to contain fabric intents and workload VPN intents. This form does not display in subsequent sessions, even if you have not yet created a region.

#### Expected outcome

The dashboard displays.

### 3.2 Signing out of the Fabric Services System

#### Procedure

**Step 1.** Click the username displayed at the upper right of any page.

**Step 2.** In the resulting drop-down list, click **SIGN OUT**.

### 3.3 The dashboard

The Fabric Services System dashboard shows high-level information about the fabric intents, workload VPN intents, and maintenance intents in the system, as well as information about the deployment pipeline.

Here you can see important status information about the intents and deployment pipeline at a glance, and the display can immediately guide you to critical issues that need your attention.

From the dashboard, you can navigate to anywhere within the system.



**Note:** The Fabric Services System organizes most of the objects it manages into containers called regions. These objects include everything that is displayed on the dashboard. The dashboard only displays information about the objects in a single region at a time. If you have configured more than one region in the Fabric Services System, you can choose among those regions to control which region's data is displayed on the dashboard. You choose the displayed region using the **Region selector** at the top of the page.



Figure 4: Fabric Services System dashboard

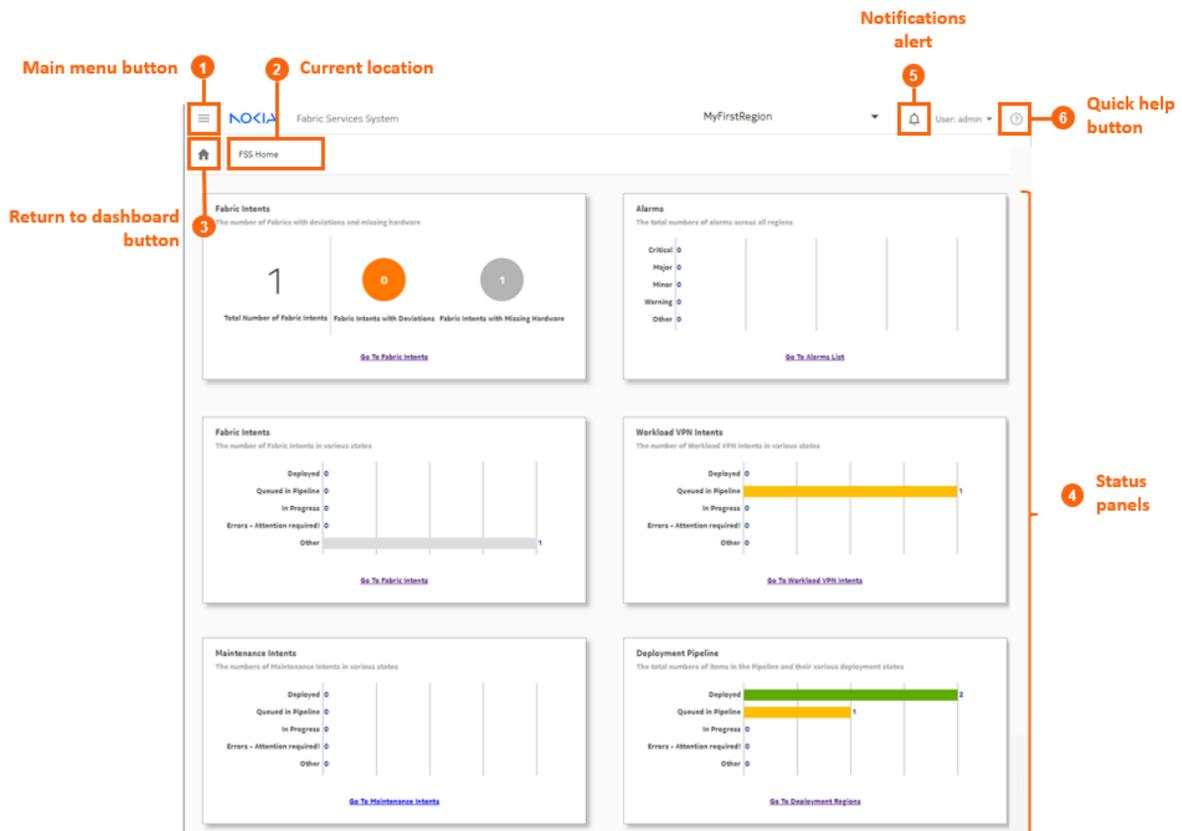


Figure 5: Dashboard deployment regions map

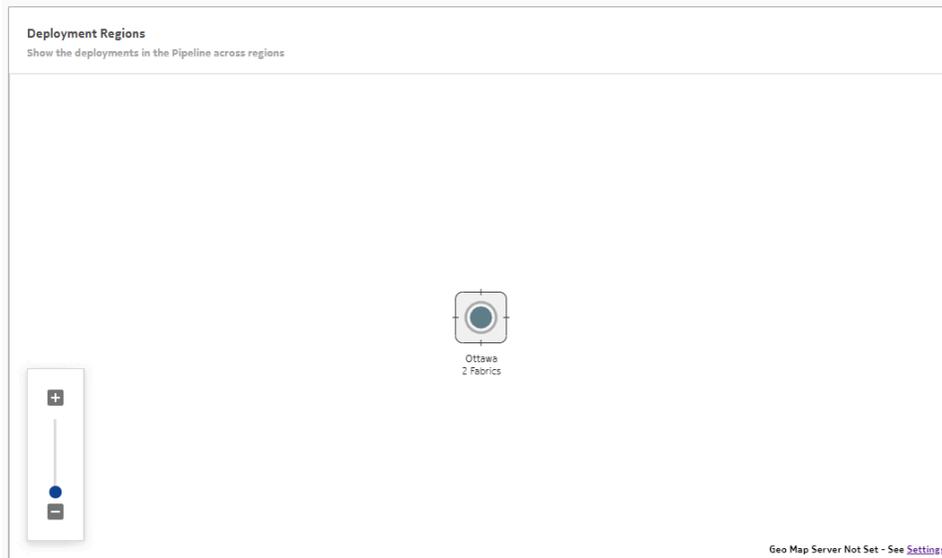


Table 3: Elements of the dashboard display

#	Description
1	The <b>Main menu button</b> opens the navigation panel on the left side of the display. See <a href="#">The main menu</a> for more information about this menu.
2	The <b>Current location</b> indicator displays the name of the current page.
3	The <b>Return to dashboard</b> button takes you directly back to the dashboard. This button displays on all screens within the UI.
4a	<p>The first <b>Fabric Intents</b> status panel displays a summary of the fabrics that are:</p> <ul style="list-style-type: none"> <li>the subject of deviations (the configuration of at least one participating node has been modified by an external user after the fabric was deployed)</li> <li>missing hardware</li> </ul> <p>A link at the bottom of the panel opens the <b>Fabric Intents</b> page, which shows a list of all fabric intents from which you can view full details about each intent in the list.</p> <p>Fabric intents are fully described in <a href="#">Fabric intents</a>.</p>
4b	<p>The <b>Alarms</b> panel indicates the number of current alarms of each possible severity level (Critical, Major, Minor, Warning, Other).</p> <p>A link at the bottom of the panel opens the <b>Alarms List</b> page, which shows a list of all alarms and from which you can view full details about each alarm in the list.</p> <p>Alarms are fully described in <a href="#">Alarms</a>.</p>

#	Description
4c	<p>The second <b>Fabric Intents</b> panel displays a chart showing the number of fabric intents in each of the following states.</p> <ul style="list-style-type: none"> <li>• Deployed: the fabric intent configuration has been deployed to participating nodes, and is in a functioning state with no configuration changes.</li> <li>• In Queue: the fabric intent has been added to the deployment queue, but is not yet deployed. It may be waiting for a user action to begin deployment, or may be waiting until preceding deployments are complete. The system does not begin a new deployment until the preceding intents have been fully deployed.</li> <li>• In Progress: the system is deploying the fabric intent but it is not complete.</li> <li>• Error: there is an error state that requires resolution. Consult the information panel for the affected fabric intent for additional details.</li> <li>• Other: The fabric is in some other state that likely requires attention. Consult the information panel for the affected fabric intent for additional details.</li> </ul> <p>A link at the bottom of the panel opens the Fabric Intents page, which shows a list of all fabric intents from which you can view full details about each intent in the list.</p> <p>Fabric intents are fully described in <a href="#">Fabric intents</a>.</p>
4d	<p>The <b>Workload VPN Intents</b> panel displays a chart showing the number of workload VPN intents in each of a series of states as described for fabric intents.</p> <p>A link at the bottom of the panel opens the Workload VPN Intents page, which shows a list of all workload VPN intents and from which you can view full details about each intent in the list.</p> <p>Workload intents are fully described in <a href="#">Workload VPN intents</a>.</p>
4e	<p>The <b>Maintenance Intents</b> panel displays a chart showing the number of maintenance intents in the same states described for fabric intents.</p> <p>A link at the bottom of the panel opens the Maintenance Intents page, which shows a list of all maintenance intents and from which you can view full details about each intent in the list.</p> <p>Maintenance intents are fully described in <a href="#">Maintenance intents</a>.</p>
4f	<p>The <b>Deployment Pipeline</b> panel displays a chart showing the number of intents that are currently in the system's deployment pipeline and are in each of the states described for fabric intents.</p> <p>A link at the bottom of the panel opens the Deployment Regions page, from which you can view the complete list of intents in the pipeline with additional details.</p> <p>Deployment regions, and the deployment pipeline, are fully described in <a href="#">Deployment regions</a>.</p>

#	Description
5	The <b>Notification alert</b> indicates whether there are active notifications pertaining to the Fabric Services System. When it is active, click this icon to see the active notifications.
6	The <b>Quick Help</b> ( ? ) button displays help topics pertaining to the current screen.
7	<b>Deployment regions map:</b> below the information panels, the system displays a geographic map showing the existing deployment regions. The Fabric Services System supports the creation of only one deployment region.  Double-click a region icon to open the <b>Regions</b> page. From there, you can view more information about the selected region.  This map works identically to the region map described in <a href="#">Deployment regions</a> .

### 3.3.1 Notifications

Notifications are alerts generated by different subsystems within the Fabric Services System. When a new notification is raised, the bell icon at the upper right of the UI displays a red circle. A counter beside the icon indicates how many new notifications have accumulated since you last viewed the list of individual items.

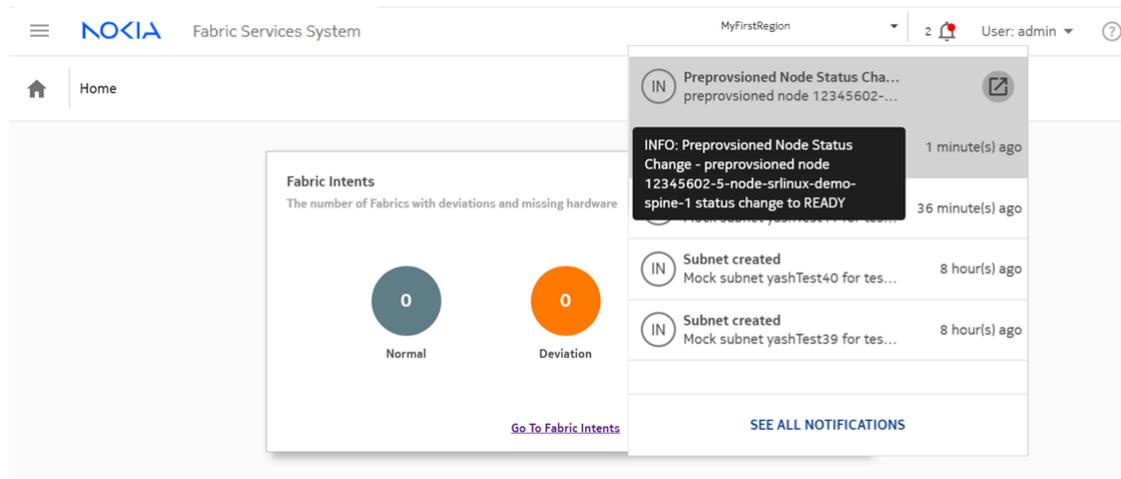
*Figure 6: Notifications indicator*



Clicking the bell icon displays the list of individual notifications. Hovering over a notification in the list displays additional details about that item.

Clicking on the notification takes you to the page in the UI that corresponds to the subsystem that triggered the notification. For example, in the illustration below, the inventory subsystem has triggered a notification when the status of a node changed from READY to NOT\_READY. The link in the notification takes you to the global inventory page.

Figure 7: Notifications list



Within the notifications list, items are sorted by order of occurrence with the most recent notifications displayed at the top of the list.

### Notification types

The system currently generates a notification for any device that changes its status to Not Ready.

### Notification severity

The system currently supports a single severity level: Alert.

## 3.4 The main menu

The main menu button (☰) opens a set of links to the main pages of the Fabric Services System. You can use the main menu to access major features from anywhere within the UI.

The main menu is divided into the following categories:

- DESIGN, which contains links to pages where you can create and manage Region, Fabric, Workload, and Maintenance intents.
- OPERATIONS, which contains links to monitor and maintain fabrics.
- COMMON RESOURCES, which contains links to supporting systems like profiles, the Label Factory, and the hardware inventory.
- Settings, for managing the system, authentication, and communications.

### DESIGN links

Design links consist of:

- **Deployment Regions:** this page shows a logical view of the fabrics managed by the system, divided first into regions, and then into smaller organizational units. From this page you can:
  - create, delete, or edit a region

- open a region to view its topology and other details
- access the deployment pipeline, from which you can deploy intents previously added to the pipeline
- **Topologies:** this page allows you to import manually created topology files, and manage a set of imported topologies. These manually created topologies are useful if you want to design your fabric outside of the Fabric Services System, and is necessary in cases where the nodes within the topology use an operating system other than SR Linux (which curtails the role of the Fabric Services System in configuring such nodes).
- **Fabric Intents:** this page shows a list of all fabric intents regardless of state, and includes basic information about each intent. From this page you can:
  - view a list of fabric intents
  - begin creating a new fabric intent
  - open a fabric intent to view its topology and other details
  - duplicate or create a new version of an existing fabric intent
  - manage the edge links associated with a fabric intent, including creating a LAG or configuring a breakout port
  - delete a fabric intent
  - add a fabric intent to the deployment pipeline
- **Workload VPN Intents:** this page shows a list of all workload VPN intents, and includes basic information about each intent. From this page, you can:
  - view a list of workload VPN intents
  - begin creating a new workload VPN intent
  - open a workload VPN intent in its own page to view additional details
  - delete an existing workload VPN intent
  - add a workload VPN intent to the deployment pipeline
- **Maintenance Intents:** this page shows a list of all maintenance intents, and includes basic information about each intent. From this page, you can:
  - begin creating a new maintenance intent
  - duplicate an existing maintenance intent
  - open a maintenance intent in its own page to view additional details
  - delete an existing maintenance intent
  - add a maintenance intent to the deployment pipeline
  - abort a maintenance intent that is underway
- **Overrides:** This page allows you to create special configurations that should be applied to one or more nodes within a fabric. These configurations can contain additional settings not included in the basic configuration that is generated by the overall fabric design.

## OPERATIONS links

Operations links consist of:

- **Operational and Health Insights:** this page displays a map of configured regions, and allows you to drill down to the fabric and node level to view health status and any deviations for displayed objects.

- **Alarms List:** this page displays information about all alarms currently affecting managed fabrics and the objects within them.
- **Traffic Mirroring:** This page shows all mirroring instances. From this page, you can:
  - create mirroring sources
  - create mirroring destinations
  - create a new mirroring instance

### COMMON RESOURCES links

- **Inventory:** this page shows a list of all hardware that the system is aware of. From this page you can:
  - add or delete hardware
  - associate planned nodes with real hardware when available
- **Label Factory:** this page shows a list of all labels that are available for assignment to objects displayed in the UI. From this page, you can:
  - create or delete labels
  - see which labels have already been assigned, and to which objects
- **Profiles:** this page displays a list of profiles available to the system for each of the following (on separate tabs):
  - Quality of Service (QoS)
  - Access Control List (ACL)
  - Match Groups
- **Policies:** this page allows you to customize the severity level for individual alarms.
- **Connect:** this page allows you to view the list of Connect deployments and plugins, and to add new deployments and plugins. For more information about this page, see the *Fabric Services System Connect Guide*.
- **Network Resources:** this page allows you to configure shared resources that are available when configuring a fabric intent or workload intent, such as:
  - System IP pools
  - Inter Switch Link IP pools
  - Management IP pools
  - Autonomous system pools



**Note:** These IP and AS pools are distinct from the single, default version of each pool that is configured as part of a deployment region.

- **User and Resource Management:** this page displays the Fabric Services System user, group, and permissions data.

### CLOUD INTEGRATIONS link

The Cloud Integrations **Connect** link opens a page that allows you to view the list of Connect deployments and plugins, and to add new deployments and plugins. For more information about this page, see the *Fabric Services System Connect Guide*.

## Settings

The Settings link opens a page from which you can configure some aspects of the system's behavior with regard to:

- **Image Management:** an interface for adding the SR Linux images to the system library
- **Software Catalog:** a list of SR Linux images currently in the system library
- **Common Application Settings:** an interface for managing the image displayed as a map background, and external services.

### Related topics

[Application settings](#)

## 3.5 Detail panels

Many pages in the Fabric Services System UI allow you to open panels on the right side of the page to display more information. This could be information about the overall display, or about selected objects.

Table 4: Detail panels

Icon	Panel	Description
	Deviations	For a fabric intent, this panel shows a cumulative list of deviations logged for the fabric intent. Each item in the deviations list identifies a configuration change originating from outside the Fabric Services system. For more information about deviations, see <a href="#">Deviations</a> .
	Information	This panel displays details about the object that is currently selected in the main display area.
	Legend	This panel displays the different shading used to highlight elements within the fabric intent, and their significance. The same panel also displays the set of icons that can appear in the main display area, and their significance.
	Errors	For a fabric intent, this panel displays a list of errors that require resolution. Double-click an error in the list for more information about that error that can guide troubleshooting. Controls in the Errors panel allow you to filter and sort the displayed list.
	Needs Attention	For a fabric intent, this panel displays a list of status issues that are not errors or deviations, but may require resolution. Double-click an item for more information about that item that can guide troubleshooting. Controls in the Needs Attention panel allow you to filter and sort the displayed list.

### 3.6 Lists

Many pages in the Fabric Services System UI display lists of objects. Wherever they appear, these lists include standard controls to help you sort and filter lists to focus on the information you need.

Figure 8: List controls

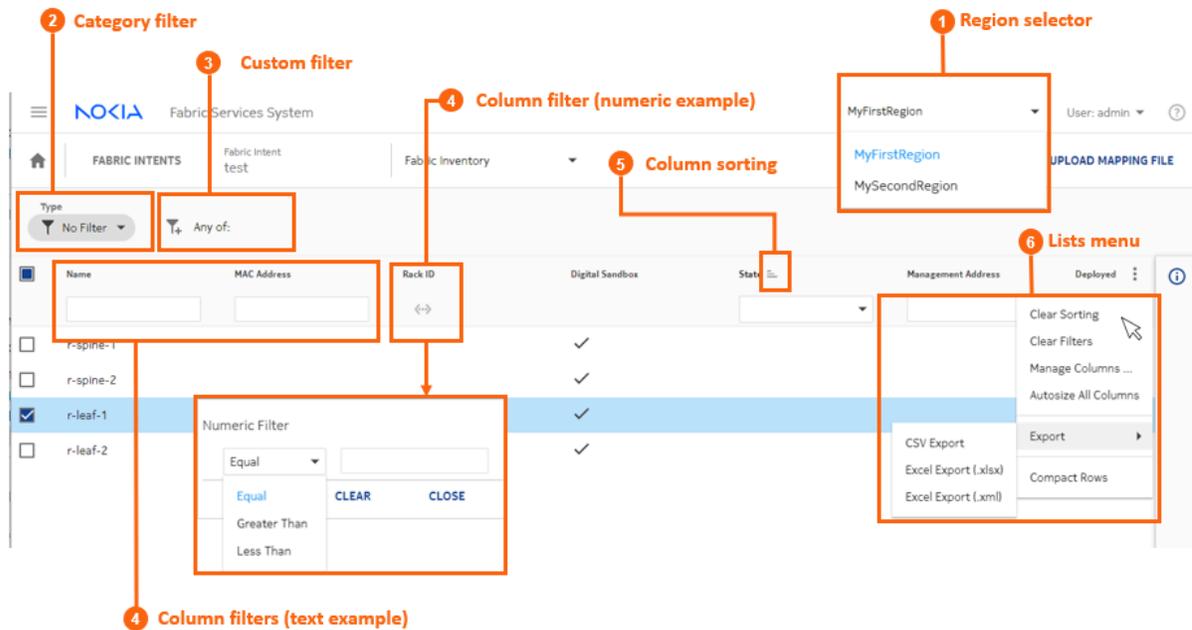


Table 5: List controls

#	Control type	Description
1	<b>Region selector</b>	<p>Most lists display a set of objects associated with a particular region. Use this drop-down to change the region whose information is currently displayed.</p> <p> <b>Note:</b> Each use account can be associated with one or more regions by an administrator. Only the regions your user account is associated with are available for selection in the <b>Region selector's</b> drop-down list.</p>
2	<b>Category filter</b>	<p>Some pages in UI can display two entirely different types of the same data. For example, the Inventory page can display a list of either physical hardware, or virtual hardware used in fabric simulations.</p>

#	Control type	Description
		The category field allows you to indicate which type of data you want the page to display.
3	<b>Custom filters</b>	You can build custom filters with multiple parameters using this control, as described in <a href="#">Customizing filters</a> .
4	<b>Column filters</b>	<p>You can constrain the list to show only items that match the values you specify in one or more column filters.</p> <p>Filtering is not supported for columns of check box values.</p> <ul style="list-style-type: none"> <li>• If the column filter is a field, type letters and numbers that you want matched in the values in that column.</li> <li>• If the column filter is a drop-down list, select a value that you want matched in the values in that column.</li> <li>• If the column filter is a calendar control, click the calendar icon and select a date that you want matched in the date values in that column.</li> <li>• If the column filter is a range control, click the range icon and enter minimum and maximum values to define a value range. Values for the column must fall within that range.</li> </ul>
5	<b>Column sorting</b>	<p>Click a column title to cycle between three sorting states:</p> <ul style="list-style-type: none"> <li>• Ascending order (  )</li> <li>• Descending order (  )</li> <li>• No sorting</li> </ul> <p>The system does not support sorting for columns of check box values.</p>
6	<b>Lists Menu</b>	Click the  icon to open a list of common list controls that allows you to control the behavior of most lists in the UI:
	<b>Clear Sorting</b>	Clears all sorting that has been set for all columns of the list.
	<b>Clear Filters</b>	Clears all filters that have been set for all columns of the list.
	Manage Columns...	Opens a list of all columns in the list, which you can then enable (to show) or disable (to hide). These selections persist for subsequent visits to the same page.
	<b>Autosize All Columns</b>	Sets the width of each column in the list to the minimum width required to display the heading without wrapping its text, and at least as wide as the filter text field for that column.

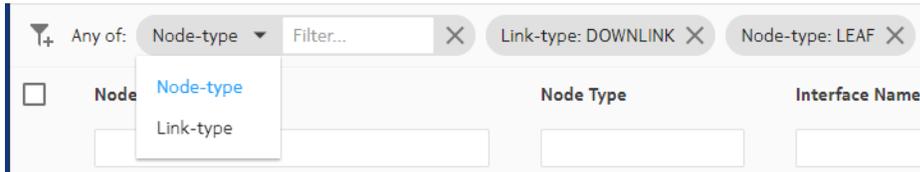
#	Control type	Description
	<b>Export to CSV, Export to XLSX, Export to XML</b>	Exports the entire table to either a Comma-separated Values (CSV) list, a spreadsheet in XLSX format, or a spreadsheet in XML format.
	<b>Compact Rows</b>	Reduces the white space allocated to each row, fitting more rows onto each page.

### 3.6.1 Customizing filters

#### About this task

You can define one or more filters to limit the set of rows displayed in any list. The fields on which you can filter, and the values to match in those fields, are determined by the system.

Figure 9: Adding custom filters to a list



To add a custom filter to a list:

#### Procedure

- Step 1.** In the panel across the top of the list, click the **Add Filter** icon (  ).
- Step 2.** In the **Filter** drop-down list, click a field name from the list.
- Step 3.** In the **Filter** field, click a value for the field from the resulting drop-down list.  
This is the value that must be matched in that field in order for a row to qualify for inclusion in the list.
- Step 4.** To add another filter, repeat steps 1 through 3.
- Step 5.** Continue adding filters until your custom filter is as precise as you need.

## 4 Deployment regions

In the Fabric Services System, a deployment region (or just "region") is a container for other intents. A region serves the following purposes:

- it logically groups intents together on the system map to make them more manageable.
- it is the level at which some connection properties are set; these properties are shared among all intents within the region.
- it maintains a deployment pipeline for all intents (fabric intents, workload VPN intents, and maintenance intents) within the region.

You cannot create an intent without assigning it to a region. For this reason, you must create a region before deploying any fabric intent, workload VPN intent, or maintenance intent.

### The deployment pipeline

The deployment pipeline tracks the set of intents that have been deployed to nodes, or are awaiting deployment to nodes, within a single region. This includes fabric intents, workload VPN intents, and maintenance intents.

The deployment pipeline is a key feature of a region. Because the Fabric Services System can manage complex data center fabrics encompassing many nodes and multiple locations, there is the potential for operators to concurrently design and deploy a large number of intents (whether fabric intents, workload VPN intents, or maintenance intents). The deployment pipeline ensures that resources are allocated to each intent deployment, in sequence, to prevent simultaneous deployments from interfering with each other.

You can view the deployment pipeline, trigger deployments, and manage the deployment process from the region's deployment pipeline page.

### Default pools

As part of region creation, you define a set of default pools for:

- System IP addresses
- Inter Switch Links (ISL) IP addresses
- Out of Band Management IP addresses
- Autonomous System numbers

These are the default pools that are available when creating fabric intents within the region. However you can create additional pools as part of the Network Resources available to objects within the Fabric Services System.

### Managing multiple regions

The Fabric Services System supports multiple regions. Keep the following in mind when working with multiple regions:

- Some types of objects within the Fabric Services System are associated with a single region, and are not available outside of that region. This includes:
  - Fabric intents

- Workload VPN intents
- Maintenance intents
- Network Resources (IPAM)



**Note:** An exception is that there cannot be any overlapping management IP CIDR blocks across the IP management pools of all regions.

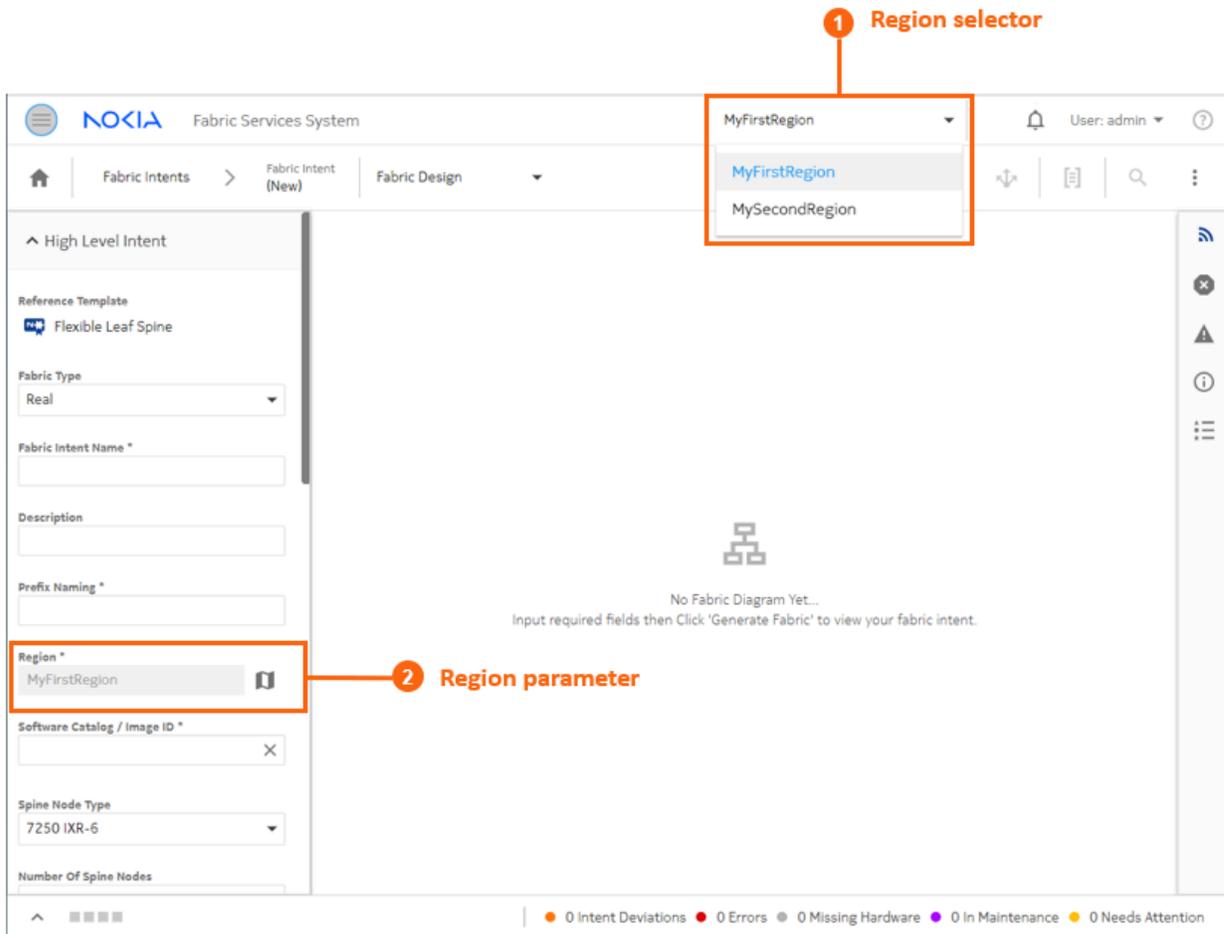
- ACL and QoS global profiles
- Alarms and alarm profiles
- Labels and the label manager
- Operational and health insights
- Traffic mirroring
- The node inventory, including management profiles
- Global Configuration Overrides (GCO) and Contextual Configuration Overrides (CCO)
- Other objects are global - that is, not associated with any region, and are available across all regions. This includes:
  - User accounts, and their supporting objects like user groups and roles
  - Anything that appears under **Settings** in the Fabric Services System menu:
    - software images
    - the software catalog
    - common application settings (currently pertaining to the geographical map)
    - certificates
  - Connect
- User accounts in the Fabric Services System can be associated with one or more regions. Within the Fabric Services System UI, a user only sees those region-specific items that belong to the regions to which that user has been granted access. For example, when viewing a list of fabric intents (a region-specific object), the list only includes fabric intents that belong to one selected region; and a user can only select a region to which their user account has been granted access.

### The Region selector

If two or more regions have been configured in the Fabric Services System, use the Region selector at the top of the page to select the correct region context for the data that is displayed on the current page.

This selection can also impact actions you are taking on that page.

For example, when configuring a fabric intent, you must specify the region to which the fabric intent belongs. Although there is a **Region** parameter among the settings for the fabric intent, you cannot set its value directly. Instead, you must select the appropriate region context using the region selector at the top of the page. This selection automatically sets the **Region** parameter for the fabric to match your selection.



## Related topics

[Network resources](#)

## 4.1 Region properties

Every region maintains a set of properties that are shared by all intents within that region. [Table 6: Region properties](#) lists these properties.

*Table 6: Region properties*

Property	Description
Name	Identifies this region uniquely within the Fabric Services System.
Description	Describes the region. This text is displayed in the <b>Additional information</b> panel for a region.
Location	Specifies the physical location represented by the region.

Property	Description
System IP Pool	<b>Pool name:</b> Specifies the name of the default System IP pool. This is set as "default" and cannot be changed.
	<b>IP Type:</b> Specifies whether the IP addresses within this pool use IPv4 or IPv6 format. This is set as "IPv4" and cannot be changed.
	<b>IP Blocks:</b> Contains one or more CIDR blocks representing IP addresses that can be assigned to the management interfaces of devices managed by the system. Enter these blocks using CIDR notation; for example, 192.0.2.0/24.  If you need more IP addresses for devices in your fabrics than the current CIDR blocks support, you can modify the region to add more CIDR blocks to the System IP pool.
Inter Switch Link IP Pool	<b>Pool name:</b> Specifies the name of the default Inter Switch Link IP pool. This is set as "default" and cannot be changed.
	<b>IP Type:</b> Specifies whether the IP addresses within this pool use IPv4 or IPv6 format. This is set as "IPv4" and cannot be changed.
	<b>IP Blocks:</b> Contains one or more CIDR blocks representing IP addresses that can be assigned to inter-switch links between devices in real (not Digital Sandbox) fabric intents throughout this region. Enter these blocks using CIDR notation. For example: 192.0.2.0/24.   <b>Note:</b> Each link within a fabric intent requires two IP addresses from this block; one for each endpoint. From the set of links that are possible with the IP addresses in the specified CIDR block, two are reserved for use by the system to represent the network IP address and the broadcast address. These are unavailable for inter-switch links. For example, a CIDR block ending in /26 that can support up to 32 links actually supports only up to 30 links within the fabric intents in this region.  If you create a fabric intent that requires more links than are available with the current pool, fabric generation fails. The event log for the fabric intent indicates that there are insufficient IP addresses for the required links, and shows the number of addresses required versus the number available.  If you need more IP addresses available to your fabrics than the current CIDR blocks support, you can modify the region to add more CIDR blocks and thereby support additional links.
Out of Band Management IP Pool	<b>Pool name:</b> Specifies the name of the default Out of Band Management IP pool. This is set as "default" and cannot be changed.
	<b>IP Type:</b> Specifies whether the IP addresses within this pool use IPv4 or IPv6 format. This is set as "IPv4" and cannot be changed.
	<b>IP Blocks:</b> Contains a CIDR block representing the IP addresses that will be assigned to the management interfaces of devices. Enter these blocks using CIDR notation; for example: 192.0.2.0/24.

Property	Description
	 <p><b>Note:</b> From the set of links that are possible with the IP addresses in the specified CIDR block, two are reserved for use by the system to represent the network IP address and the broadcast address. These are unavailable for the out-of-band management IP pool. If you need more links in your fabrics than the CIDR blocks you specified here support, you can modify the region to add more CIDR blocks to this pool.</p>
BGP ASN Numbers	<p>Determines the set of Autonomous System Numbers (ASNs) available for BGP.</p> <p>The Fabric Services System can maintain a pool of ASNs consisting of multiple blocks of numbers. ASNs are used to uniquely identify a network with a unique routing policy. The ASN must be unique so that IP address blocks appear to originate from a unique location to which BGP can determine a route.</p> <p>The single pool of numbers is assigned the label "default", and this cannot be altered.</p> <p>Within the pool, you can define one or more blocks of contiguous ASNs by providing a start and end number for each block. ASNs can be any number from 0 to 4294967295.</p> <p>Blocks of numbers within the same pool cannot contain overlapping values.</p> <p>As one block of ASNs is exhausted, the system begins assigning values from whichever remaining, unexhausted block contains the highest available values. Even if some numbers from the first block become available, the system continues allocating numbers from this second block until the second block is exhausted. Then the system again looks for the block with the highest available values from which to allocate new ASNs, and so on.</p> <p><b>Start:</b> the lowest permissible value in a range of a single block of ASNs.</p> <p><b>Start:</b> specifies the lowest permissible value within a block of ASNs.</p> <p><b>End:</b> specifies the highest permissible value within a block of ASNs.</p>
Digital Sandbox DHCP Configuration	<p><b>Digital Sandbox DHCP CIDR:</b> specifies the CIDR block use to assign IP addresses for the SR Linux container nodes within the Fabric Services System Digital Sandbox.</p> <p>You can create both IPv4 and IPv6 CIDR blocks for use by Digital Sandbox fabric intents.</p> <p>For more information about the Digital Sandbox, see <a href="#">Digital Sandbox</a>.</p>
Route Target	<p><b>Global Index:</b> specifies whether services created as part of a workload are unique within a region and across multiple regions.</p>
EVPN Profile	<p><b>Control Plane:</b> specifies the control plane type to use throughout the region. Supported values are:</p> <ul style="list-style-type: none"> <li>• None: no EVPN peering</li> <li>• eBGP</li> </ul>

Property	Description
	<ul style="list-style-type: none"> <li>iBGP</li> </ul> Default value: None
Protocol authentication	<b>BGP:</b> Enabled or Disabled Determines whether the Fabric Services System should create a unique MD5 authentication key. This encrypted key is required to authenticate communication between eBGP peers.  No user action is required to create the authentication key or to incorporate eBGP authentication requiring this key into subsequent node configurations. These actions are handled internally by the system.
EVI	Determines the range of the EVI pool: <b>Start:</b> the lowest value in the range of EVI values <b>End:</b> the highest value in the range of EVI values
VNI	Determines the range of the VNI pool: <b>Start:</b> the lowest value in the range of VNI values <b>End:</b> the highest value in the range of VNI values

## 4.2 Creating a region

### About this task

You must create at least one region to contain the intents you create using the Fabric Services System. The system supports multiple regions, but you must create at least one.

The first time you log into the system, you are prompted to create a region. The prompt does not appear for subsequent logins.

To create a new region:

### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** In the main menu, select **Deployment Regions**.
- Step 3.** Click the **+ CREATE A REGION** button.
- Step 4.** Enter values for the parameters described in [Table 6: Region properties](#), up to and including Route Target.
- Step 5.** To configure EVPN peering within the region, define the EVPN profile by doing the following:
  - a. If your region will not use EVPN peering, leave the default value of the **Control Plane** drop-down list as **None** and go to step [11](#).
  - b. If your region will use eBGP for EVPN peering, select eBGP from the **Control Plane** drop-down list and go to step [11](#).
  - c. If your region will use iBGP for EVPN peering, and you plan to use only nodes within a fabric created with the Fabric Services System as route reflectors, you must create those fabrics and configure those nodes before you can configure iBGP; go to step [7](#).

- d. If your region will use iBGP for EVPN peering and you plan to use existing nodes (that are not part of a fabric created with the Fabric Services System) as route reflectors, you can configure iBGP; go to step 6.

**Step 6.** Configure iBGP by doing the following:

- a. In the **Control Plane** drop-down list, select **iBGP**.
- b. Enter a valid Autonomous System (AS) value.
- c. Enter a Cluster ID to represent the SR OS cluster to which leaf nodes will be peered.  
The system currently supports a single SR OS cluster for iBGP.
- d. Add a route reflector by clicking **+ADD**, entering a route reflector IP address in the resulting form, and clicking **ADD**.
- e. If required for redundancy, add additional route reflector IP addresses using the same method.  
All route reflectors must point to the same SR OS cluster. Leaf nodes can use any of the reflectors in this list interchangeably to communicate with the SR OS cluster.
- f. Go to step 11.

**Step 7.** Leave the default value of the **Control Plane** drop-down list as None.

After you create a fabric that includes one or more nodes to act as route reflectors, complete [Updating a region to use iBGP](#). As part of that procedure, you change the Control Plane setting to iBGP and select your newly configured route reflectors.

**Step 8.** In the **Protocol Authentication** panel, configure eBGP authentication:

- if eBGP authentication is not required, leave the **BGP** toggle set to Disabled.
- if eBGP authentication is required, set the **BGP** toggle to **Enabled**. This causes the Fabric Services System to generate a unique key that is required for eBGP authentication and to incorporate the requirement for this authentication into subsequent node configurations.

**Step 9.** Configure start and end values to establish the range for the EVI pool.

**Step 10.** Configure start and end values to establish the range for the VNI pool.

**Step 11.** Click the **CREATE** button.

#### Related topics

[Modifying a region](#)

## 4.3 Modifying a region

### About this task

Follow this procedure to modify the properties of an existing region.



**Note:** To modify the System IP pools and blocks, Inter-Switch Link IP pools and blocks, Out-of-Band Management IP pools and blocks, and the Autonomous System Number pools, use the Network Resources Page.

### Procedure

**Step 1.** Click  to open the main menu.

- 
- Step 2.** In the main menu, select **Deployment Regions**. The deployment regions page opens, showing a graphical representation of regions already created.
- Step 3.** Right-click the region and select **Edit Details...** from the contextual menu.
- Step 4.** Modify any of the following settings for the region:
- General/Description
  - General/Location
  - EVPN Profile/Control Plane
  - Route Target/Global Index
  - Protocol Authentication/BGP
  - EVI
  - VNI
- Step 5.** Click **SAVE**. The system saves the new region details.

### 4.3.1 Implications of modifying a region

A region is a container for intents, and every fabric intent you create within a region inherits the region's properties (such as configurations for EVPN peering or eBGP authentication).

If you modify the region's settings, the system does not automatically propagate the changes to pre-existing fabric intents within the region.

To update existing fabric intents with the new region settings, you must regenerate each of the fabric intents.

- If you have saved, but not generated, a fabric intent, no action is required; the region settings are not inserted into the fabric's code until you generate the topology, so the latest settings are picked up immediately when you do generate the fabric topology.
- If you have generated but not deployed a fabric intent, you need to update and re-save the fabric intent, then regenerate the topology. The newly generated fabric intent includes the latest settings from the updated region.
- If you have deployed a fabric intent you need to create a new version of the intent, regenerate the topology, and deploy the result. Regenerating the new version of the fabric intent inserts the latest region settings into the fabric code, and deploying the fabric intent updates the working node configurations accordingly.

When you regenerate the fabric intent, the system updates the settings to match the current configuration of the region.

You can then add the new version of the fabric intent to the region's deployment pipeline, and deploy the fabric intent from the pipeline normally.

#### Related topics

[Creating a new version of a fabric intent](#)

## 4.4 Updating a region to use iBGP

If you plan to use only nodes within a fabric created with the Fabric Services System as route reflectors, you must create that fabric and configure those nodes before you can configure a region to use iBGP for EVPN peering.

When you have created at least one fabric within the region and configured nodes within that fabric as route reflectors, you can return to the **Deployment Regions** page to update the region settings to use iBGP.

This section assumes you are familiar with the process of creating and deploying fabrics in the Fabric Services System.

It also assumes you have the expertise to configure nodes as route reflectors, independently of the Fabric Services System.

### Related topics

[Fabric intents](#)

### 4.4.1 Updating region settings to use iBGP

#### Prerequisites

Before you begin this procedure, you must have created and deployed at least one fabric within the region, and then configured nodes within the region as route reflectors.

#### About this task

When there are route reflectors in a fabric, you can update the region settings to use those route reflectors for iBGP EVPN peering.

To update the region settings to use iBGP:

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** In the main menu, select **Deployment Regions**. The **Deployment Regions** page opens, showing a graphical representation of regions already created.
- Step 3.** Right-click the **Deployment Regions** map and select **Edit Details...** from the displayed list.
- Step 4.** From the **EVPN Profile** section, in **Control Plane** drop-down list, select iBGP.
- Step 5.** Enter a valid Local Autonomous System (AS) value.
- Step 6.** Enter a Cluster ID to represent the SR OS cluster to which leaf nodes are peered. The system currently supports a single SR OS cluster for iBGP.
- Step 7.** Add a previously configured route reflector to the **Route Reflector IP List** field by doing the following:
  - a. Click **+ADD**.
  - b. Enter a route reflector IP address in the resulting form.
  - c. Click **ADD**.
- Step 8.** If required for redundancy, repeat step 7 to add additional route reflector IP addresses.

All route reflectors must point to the same SR OS cluster. Leaf nodes use any of the reflectors in this list interchangeably to communicate with the SR OS cluster.

**Step 9.** Click the **SAVE** button. The system saves the new region settings.

#### What to do next

Proceed to [Updating existing fabric intents to use new iBGP settings](#).

## 4.4.2 Configuring fabric nodes as route reflectors

### Prerequisites

Before you begin this procedure, you must:

- have created a deployment region with EVPN peering set to "None", as described in [Creating a region](#)
- be familiar with the fabric intent creation and deployment process described in [Fabric intents](#)
- know how to configure nodes as route reflectors, independently of the Fabric Services System

### About this task

You can use the Fabric Services System to design and deploy fabrics within a region. You can then configure nodes within those fabrics to function as route reflectors. When there are route reflectors in the fabric, you can update the region settings to use those route reflectors for iBGP EVPN peering. To configure nodes within a fabric as route reflectors:

### Procedure

- Step 1.** Configure and deploy one or more fabric intents using the Fabric Services System.
- Step 2.** Note the IP address of each node you plan to configure as a route reflector.
- Step 3.** Manually configure each of these nodes as a route reflector, independently of the Fabric Services System.
- Step 4.** Accept these deviations.

### What to do next

Proceed to [Updating region settings to use iBGP](#).

## 4.4.3 Updating existing fabric intents to use new iBGP settings

### About this task

When you have modified the region to use iBGP, you must update all fabrics within the region so that they inherit these new settings.

To update your fabrics with the new iBGP settings, do the following for each fabric in the region:

### Procedure

- Step 1.** Open the fabric intent's Design page.
- Step 2.** Create a new version of the intent by selecting **Create a New Version** from the **More actions**  menu.
- Step 3.** Save the fabric intent.

**Step 4.** Click  **GENERATE FABRIC**.

**Step 5.** Deploy the new version of the fabric.

#### Related topics

[Implications of modifying a region](#)

[Deploying a fabric intent from the deployment pipeline](#)

## 4.5 Viewing DHCP settings

### About this task

The Fabric Services System maintains a set of properties that govern the way it, and nodes it manages, interact with a DHCP server. These properties include a DHCP default lease time, and a DHCP maximum lease time. By default, both are set to seven days (86,400 seconds).

These properties are stored in a pair of files on the Fabric Services System server:

- dhcpd.conf (for ipv4)
- dhcp6.conf (for ipv6)

For example:

```
# dhcpd.conf
log-facility local7;
default-lease-time 604800;
max-lease-time 604800;
shared-network fss {
}
```

```
# dhcpd6.conf
log-facility local7;
default-lease-time 604800;
max-lease-time 604800;
shared-network fss {
}
```

Although you cannot configure these properties from the Fabric Services System GUI, you can view their current values by following the steps in this procedure.

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** In the main menu, select **Deployment Regions**. The **Deployment Regions** page opens, showing a graphical representation of regions already created.

**Step 3.** Right-click the Deployment Region object on the map and select one of the following from the displayed list of actions:

- To view IPv4 DHCP settings, select **View DHCP Configuration**.
- To view IPv6 DHCP settings, click **View DHCP V6 Configuration**.

#### Expected outcome

An overlay displays, showing the contents of the corresponding configuration file.

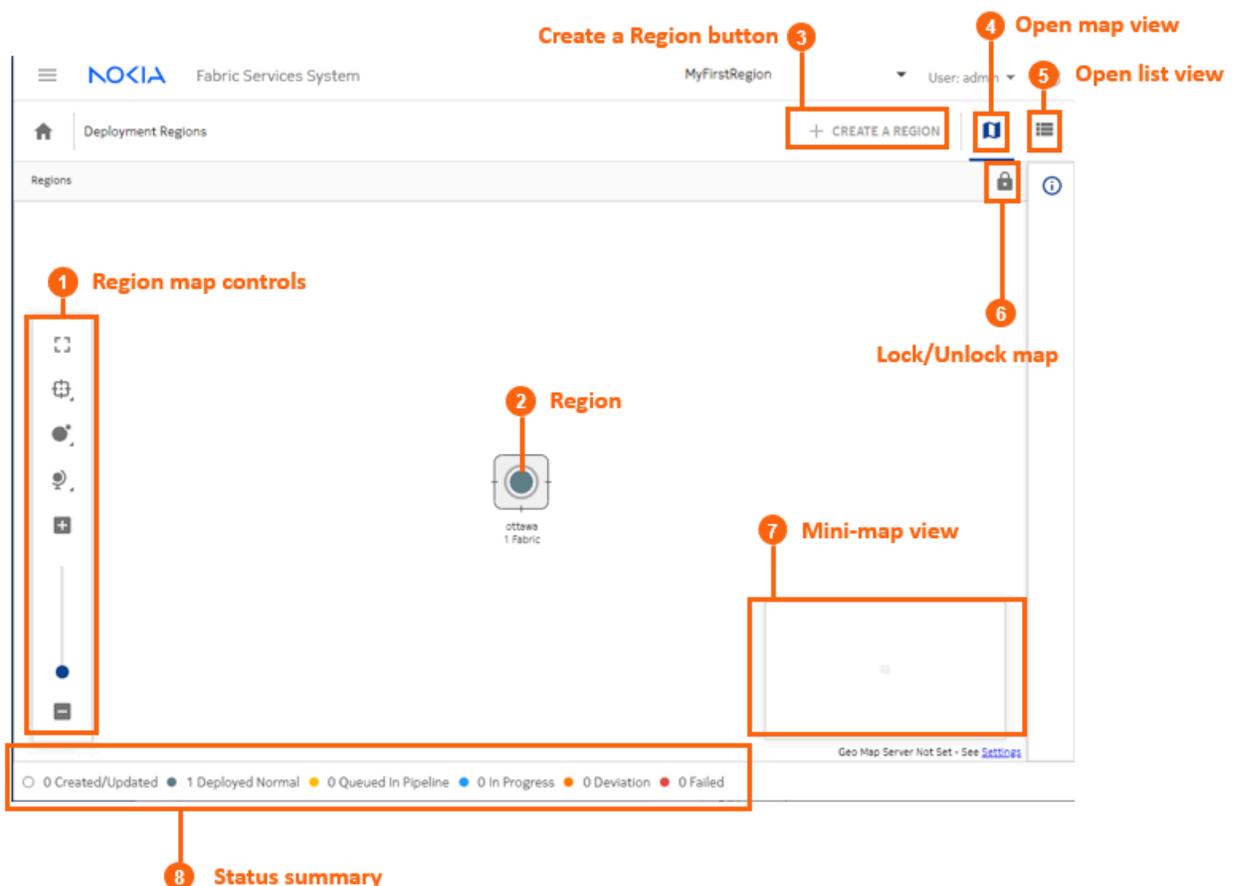
**Step 4.** Do any of the following:

- To save a copy of the configuration file, click .
- To copy the displayed information to your clipboard, click .
- To close the overlay, click the **X** icon.

## 4.6 The region map

The region map is a graphical representation of all regions managed by the system, superimposed over a map image.

Figure 10: The region map



You can change the arrangement of regions on the map, unless the administrator has chosen to lock the map for all users. The region map display is user-specific; your changes to the map persist the next time you log in with the same user account.

Administrators can manage their own versions of the map, and periodically deploy it so that the latest version is available to other users.

The background image displayed for the region map is determined by the system's **Geomap Tile Server** setting. To view or change this setting, go to **Settings** → **Common Application Settings**.

After you select an image, it displays the next time you open the region map. You can control the opacity and transparency of the image using the region map manipulation controls.

#### Related topics

[Configuring Geomap Tile Server settings](#)

[Region map manipulation](#)

### 4.6.1 Viewing and using the region map

#### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** In the main menu, select **Deployment Regions**. The deployment regions page opens, showing a graphical representation of region already created.

From this map, you can do the following:

- Lock the region icon in its current position by clicking the Lock button (). This replaces the button with an Unlock button (.
- View the set of fabric, workload, and maintenance deployments currently pending for a region by right-clicking on the region and selecting **Show Deployment Pipeline** from the contextual menu.
- View the list of Fabric Intents within a region by right-clicking on the region and selecting **Show Fabric Intent List** from the contextual menu.
- View and optionally edit details about a region by right-clicking on a region and selecting **Edit Details...** from the contextual menu.
- View information about DHCP settings by selecting **View DHCP Configuration...** from the contextual menu.
- Delete the region by right-clicking on the region and selecting **Delete...** from the contextual menu.  
You cannot delete a region that contains one or more fabrics.

### 4.6.2 Viewing and using the region list

#### About this task

Although the region displays as a map by default, you can switch the view to a list of regions instead.

However, in the current release, this is a list of one item.

To view a region list instead of a map:

#### Procedure

**Step 1.** Open the **Regions** map.

**Step 2.** Click the **Regional List** () button. The system replaces the map display with a list showing the single region already configured.

From this list you can access most of the options available from the region map by clicking on the **More actions** icon () at the right of the row in the region list and selecting one of the following:

- **Show Deployment Pipeline** to view the set of fabric, workload, and maintenance deployments currently pending for the region
- **Show Fabric Intent List** to view the list of Fabric Intents within the region
- **Edit Details...** to view and optionally edit details about a region.  
For information about the consequences of such edits, see [Implications of modifying a region](#).
- **View DHCP Configuration** to view the region's DHCP settings.
- **Delete** to delete the region.  
You cannot delete a region that contains one or more fabrics.

**Step 3.** To return to the map view from the list view, click the **Regional Map** ( ) button.

### 4.6.3 Region map manipulation

In addition to being able to drag the region within the map (unless the map has been locked), several controls allow you to manipulate the way elements are displayed on the regions map.

Table 7: Display controls

Icon	Description
	Lock: prevents you from changing the position of regions on the Regions map. When clicked, the button toggles to Unlock.
	Unlock: restores the ability to move regions on the Regions map. When clicked, the button toggles to Lock.
	Fit to screen: adjusts the magnification setting for the topology display so that the entire topology diagram fits inside the current window.
	Clustering controls: <ul style="list-style-type: none"> <li>• Sets the cluster health display method (no display, or as a pie chart, or as a solid circle).</li> <li>• Enables or disables clustering boundaries.</li> <li>• Enables or disables whether moving a region icon also changes the position of its contents on the underlying map.</li> </ul>
	Vertex controls: adjusts the size of the vertices to either small, medium, large, or extra large.
	Map controls: adjusts the following: <ul style="list-style-type: none"> <li>• Bird's eye view enables or disables the mini-map in the lower right.</li> <li>• Background layer opacity adjusts the background opacity in 5% increments.</li> </ul>
	Zoom in, zoom out: increases or decreases the magnification for the topology diagram.

## 4.7 The deployment pipeline

The Fabric Services System uses the deployment pipeline to manage the many concurrent fabric, workload, and maintenance intents that can be created when multiple users are working with the system simultaneously.

When you complete the design of a fabric, workload, or maintenance intent and are ready to deploy it, you add the intent to the region's deployment pipeline.

The intent then remains in the pipeline until you instruct the system to proceed with its deployment.

Figure 11: The Deployment Pipeline

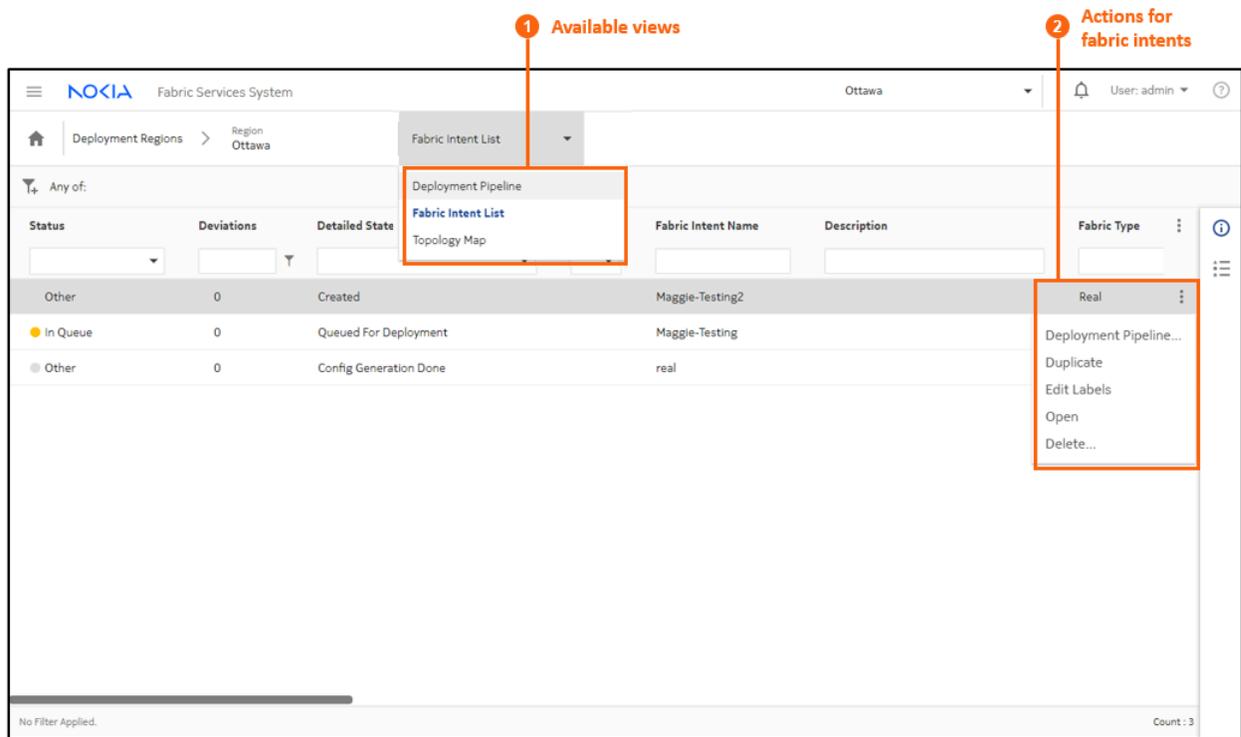


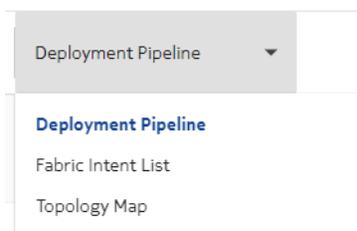
Table 8: Deployment pipeline columns

Column	Description
Sequence Number	The number indicating the sequence (from low to high) in which the displayed intents are deployed within the region. Deployment is currently strictly serial, so one deployment must complete entirely before the next begins.
Status	This indicates the progress the displayed intent has made toward deployment. The system currently supports the following deployment statuses: <ul style="list-style-type: none"> <li>Other: some status not accounted for below.</li> </ul>

Column	Description
	<ul style="list-style-type: none"> <li>• Deployed: deployment of the intent has completed successfully.</li> <li>• Error: deployment of the intent encountered an error and did not complete.</li> <li>• In Queue: the intent is awaiting deployment, likely waiting for another intent's deployment to complete.</li> <li>• In Progress: deployment of the intent is underway.</li> </ul>
Detailed State	This indicates the progress the displayed intent has made toward deployment.
Pipeline User	The login ID of the user who added the intent to the pipeline.
Source Type	The originating intent type: Fabric, Workload, or Maintenance.
Last Updated Time	The last time the intent was updated.
Source User	The login ID of the user who designed the intent.
Source Name	The name of the intent, as specified in its Name property.
Source Version	The version number of the intent.
Auto Deploy	True or False: indicates whether the auto-deploy option has been enabled.
Status Reason	Additional details for the Status field.

From the **Deployment Pipeline** page, you can also use the **View** drop-down list to display the fabric intent list or view the topology map for the region.

*Figure 12: Choosing another view*



From the **Deployment Regions** page, selecting **Fabric Intent List** switches the view to the list of all fabric intents contained within the current region regardless of state of the fabric. From that list, you can perform the following, limited set of fabric operations:

- Open a fabric intent
- Open the deployment pipeline

- Duplicate a fabric intent
- Delete a fabric intent

#### Related topics

[Adding a fabric intent to the deployment pipeline](#)

[Adding a workload VPN intent to the deployment pipeline](#)

[Adding a maintenance intent to the deployment pipeline](#)

[The Fabric Intent List view](#)

[The Topology Map view](#)

[Viewing the deployment pipeline](#)

[Duplicating a fabric intent](#)

[Deleting a fabric intent](#)

## 4.7.1 Viewing the deployment pipeline

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** In the main menu, select **Deployment Regions**. The **Deployment Regions** page opens, showing a graphical representation of regions already created.

**Step 3.** From this page, do either of the following:

- Right-click a region on the map and select **Show Deployment Pipeline** from the contextual menu.
- Click the **Regional List** () button to view the set of regions as a list; then click the **More actions** icon () at the right edge of a region's row and select **Show Deployment Pipeline** from the actions menu.

## 4.7.2 Working with the deployment pipeline

### About this task

From the deployment pipeline, you can manage the set of deployments that are pending for the region. From the deployment pipeline you can:

- view the design of an intent that is awaiting deployment
- deploy an intent
- remove an intent from the deployment pipeline
- abort a deployment that is underway
- delete a deployed intent from the list

To take any action available for the intents in the deployment queue, do the following:

### Procedure

**Step 1.** Open the deployment pipeline.

- Step 2.** Select an intent from the displayed list and click the **More actions** icon (  ) at the right edge of the row to open the actions menu.
- Step 3.** Do any of the following for a fabric intent:
- Select **Open Fabric Design** to leave the deployment queue and view the fabric intent in detail.
  - To deploy a fabric intent, select **Deploy**.
- Step 4.** Do any of the following for a workload VPN intent:
- Select **Open Workload Design** to leave the deployment queue and view the workload VPN intent in detail.
  - Select **Deploy** to deploy the workload VPN intent.
- Step 5.** Do any of the following for a maintenance intent:
- Select **Open Maintenance** to leave the deployment queue and view the workload VPN intent in detail.
  - Select **Deploy** to deploy the maintenance intent.
- Step 6.** To remove any intent that has not yet been deployed from the deployment pipeline, do the following:
- a. Select **Remove from pipeline...** in the actions menu. A confirmation form displays.
  - b. Click **REMOVE FROM PIPELINE...** in the confirmation form. The system removes the selected intent from the deployment pipeline.  
To resume deployment of the intent, you must add it back to the deployment queue from the intent page.
- Step 7.** To cancel the deployment that is underway for any intent, do the following:
- a. Select **Abort Deployment...**
  - b. Click **OK** in the confirmation form. The system stops the deployment and rolls back any configured nodes to their preceding state.
- Step 8.** To remove any deployed intent from the list, do the following:
- a. Select **Delete...** A confirmation form displays.
  - b. Click **OK** in the confirmation form. The system removes the selected intent from the deployment pipeline list.

#### Related topics

[Viewing the deployment pipeline](#)

[Adding a fabric intent to the deployment pipeline](#)

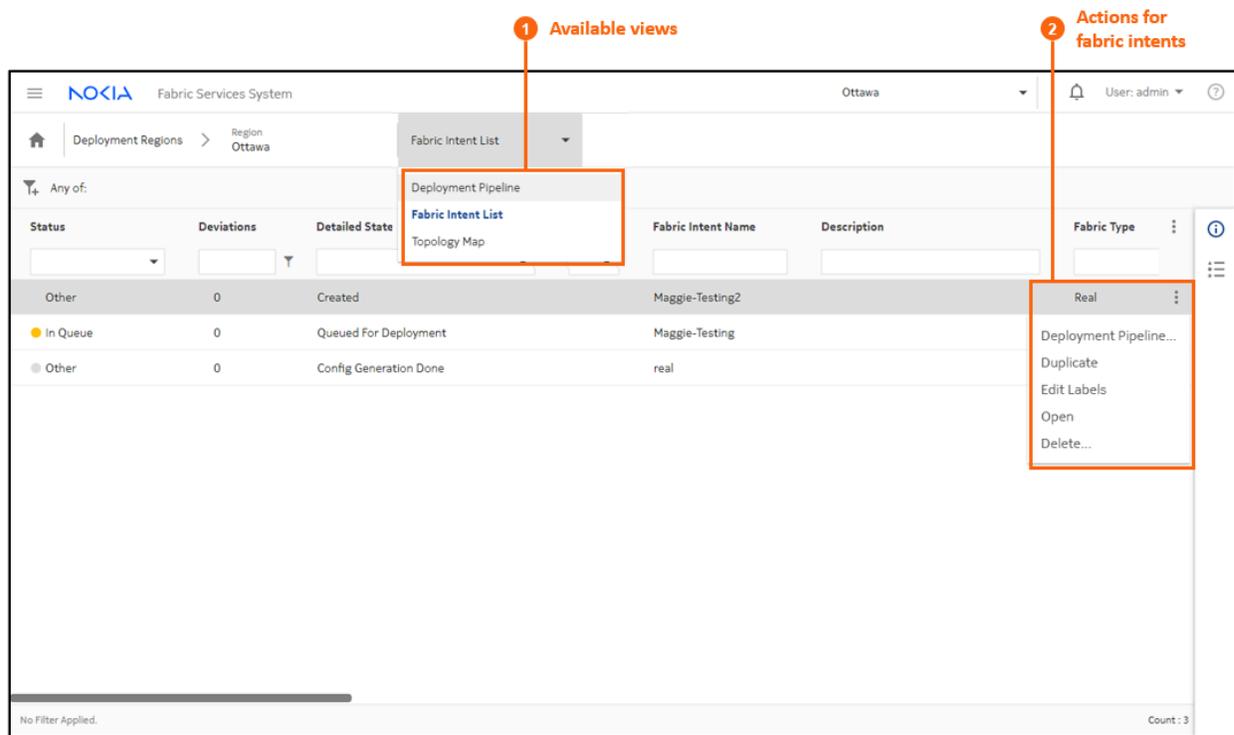
[Adding a workload VPN intent to the deployment pipeline](#)

[Adding a maintenance intent to the deployment pipeline](#)

## 4.8 The Fabric Intent List view

The **Deployment Pipeline** view includes a drop-down list that allows you to switch to alternate views, including the **Fabric Intent List** view.

Figure 13: The Fabric Intent List view



The **Fabric Intent List** view shows the list of all fabric intents in the region, along with their current states. From this view you can use selections in the **More actions** (  ) menu to switch back to the **Deployment Pipeline** view, or take actions for individual fabric intents including:

- duplicate the selected fabric intent
- edit labels for the selected fabric intent
- open the fabric intent in the **Fabric Design** view
- delete the fabric intent

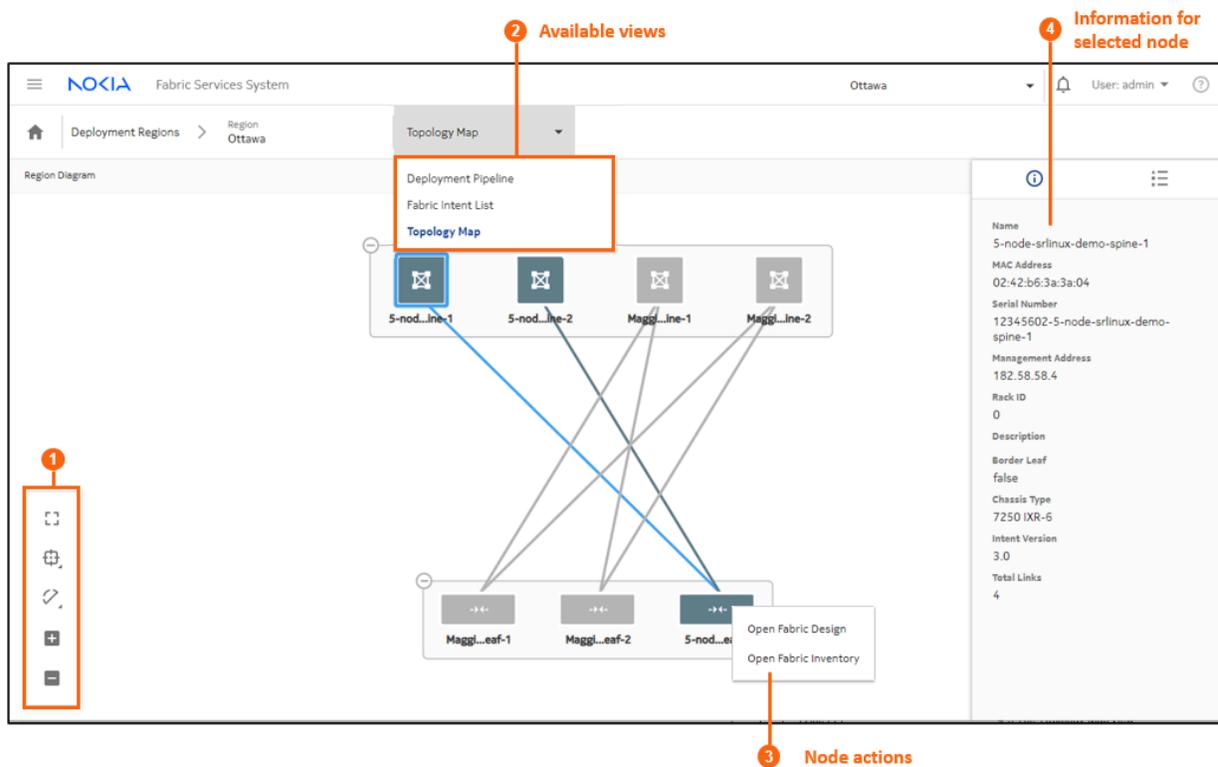
#### Related topics

[Fabric intents](#)

## 4.9 The Topology Map view

The **Deployment Pipeline** view includes a drop-down list that allows you to switch to alternate views, including the **Topology Map** view.

Figure 14: The Topology Map view



The **Topology Map** view shows the fabrics within the region, and the connections between the leaf, spine, and backbone nodes in those fabrics.

1. Map controls, like those described in [Table 7: Display controls](#), allow you to control the topology display scale, the number of nodes collectively represented by a group icon, and whether links are displayed.
2. The views drop-down list allows you to switch to the **Deployment Pipeline** or **Fabric Intent List** views.
3. Right-clicking a node allows you to open the **Fabric Intent Design** or **Fabric Inventory** views.
4. Expanding the information panel and clicking on the icon displays information about the currently selected object in the topology view.

### Related topics

[Fabric intents](#)

[Inventories](#)

## 5 Fabric intents

A fabric is a group of switches that are managed as a single logical unit. A single data center can include many complementary and mutually supporting fabrics.

With the Fabric Services System, you create a new fabric (or plan changes to an existing fabric) by designing and deploying a fabric intent: a detailed plan for a fabric's topology, whose node configuration files are deployed as a single transaction.

Each fabric exists within a region, a logical entity that stores configuration data common to the fabrics within it and that manages the orderly deployment of various types of intents. A fabric that is created within one region is not visible within, or available to, other regions.

After you deploy the fabric intent, the resulting fabric serves as the foundation for additional management templates that you can superimpose to manage the distribution of traffic across the fabric.

### Related topics

[Deployment regions](#)

### 5.1 Fabric topology

The topological design of fabrics for data centers can vary widely. To accommodate diverse fabric structures, the Fabric Services System uses a modular, tiered approach to representing a fabric's hierarchical layers and their interconnections:

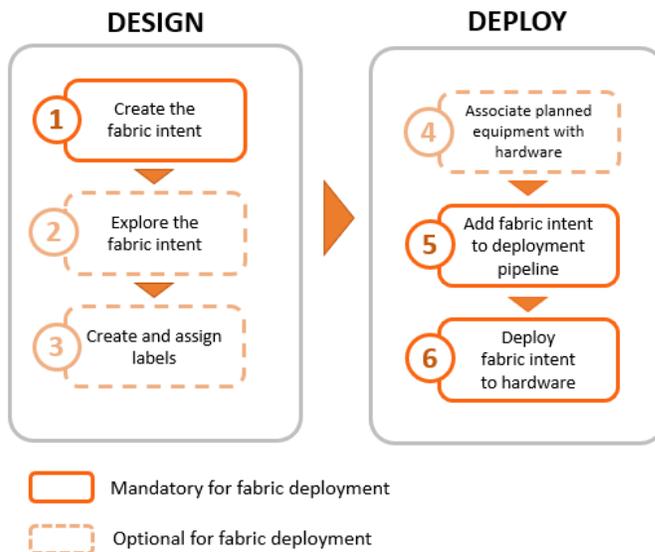
- Tier 1: leaf nodes, with edge-link connectivity
- Tier 2: spine nodes, which connect multiple leaf nodes together
- Tier 3: backbone nodes, which connect multiple fabrics together
- Tier 4: regional backbones, connecting backbone nodes together

This flexible, modular approach allows the Fabric Services System to model a wide range of topologies, grouping sets of leaves, spines, and backbone nodes in various ways and distributing connections between and within these groups in whichever way operators deem most efficient.

### 5.2 The design-and-deploy workflow

This section describes the workflow for designing, exploring, and deploying a fabric intent.

Figure 15: Fabric design and deployment workflow



Before you can create any fabric intents, you must:

- Create a region (a logical container for fabrics that manages the deployment of intents).
- Ensure that the software catalog includes the SR Linux image you intend for deployment to all nodes in the fabric.

#### Related topics

[Deployment regions](#)

[Software and image catalogs](#)

### 5.2.1 Create the fabric intent

Fabric intent creation begins by importing a description of the fabric topology from a prepared file.

After you import your topology and provide some basic parameters, you save the design and, with one click, generate the detailed fabric intent. The intent includes a fabric topology based on your selections, identifies the types of hardware required for each node, includes detailed configuration code for each node, and creates a wiring plan that technicians can use to connect the hardware together.

The Fabric Services System supports the creation of fabrics containing two types of nodes:

- real nodes - if this is a fabric you intend to deploy onto existing or planned real-world hardware
- virtual nodes - the system emulates with software in the Digital Sandbox. Such detailed simulations are useful for exploring, testing, and validating a potential fabric designs.

The Digital Sandbox is a separate component that requires its own license.

#### Related topics

[Digital Sandbox](#)

[Creating a fabric intent using a manual topology](#)

## 5.2.2 Explore the fabric intent

After you create a new fabric intent, you can explore the intent to verify that it appears as expected.

The Fabric Services System includes various tools you can use to explore the resulting fabric topology:

- the graphical fabric topology display
- the fabric-as-code view, that shows the precise configuration data for every element of the fabric.

### Related topics

[Viewing a fabric intent](#)

[Viewing a fabric intent as code](#)

[Viewing the configuration file for a single node](#)

## 5.2.3 Create and assign labels

The Fabric Services System supports an extensive labeling system you can use to tag the elements within a fabric intent. These tags are more than just passive information displays; applied systematically, tags can identify groups of logically related entities that can then be the subject of collective operations in the future.

### Related topics

[Labels](#)

## 5.2.4 Associate nodes with hardware

When you are satisfied with the fabric design, you can associate the abstracted, planned nodes in the fabric intent with their real-world counterparts using the hardware serial number.

### Related topics

[Planned node and real-world hardware association](#)

## 5.2.5 Add the fabric intent to the deployment pipeline

The first step in deploying a completed fabric design is to add it to the region's deployment queue.

Because a large number of intents may require deployment at any one time, the system uses regions to maintain a fixed queue of pending deployments. This deployment pipeline helps the system manage deployments in an orderly sequence and prevents them from interfering with each other.

### Related topics

[Adding a fabric intent to the deployment pipeline](#)

## 5.2.6 Deploy the fabric intent to hardware

Finally, you can go to the region's deployment pipeline and manually trigger the deployment of the fabric. This downloads the necessary configuration files to each of the participating nodes that you associated previously.

## Related topics

[Deploying a fabric intent from the deployment pipeline](#)

## 5.3 Notable fabric intent configuration values

When generating node configurations for any fabric intent, the Fabric Services System automatically assigns values to numerous parameters. Some of these values are selected to better support certain anticipated network configurations.

This section describes some of those configurations and their purpose.



**Note:** To view the full configuration details for any node within a fabric intent, including all automatic inclusions described here, right-click the node in the fabric intent topology view and click **Inspect Configuration**.

### Bidirectional forwarding detection (BFD)

The Fabric Services System enables BFD for all ISL interfaces:

```
"bfd": {
  {
    "admin-state": 'enable'
    "id": <interface>
  }
}
```

To support BFD on the iBGP peering sessions for EVPN route advertisement, the system also enables BFD on the management interface (system0.0):

```
bfd subinterface system0.0 admin-state enable
bfd subinterface system0.0 desired-minimum-transmit-interval 100000
bfd subinterface system0.0 required-minimum-receive 100000
bfd subinterface system0.0 detection-multiplier 3
bfd subinterface system0.0 minimum-echo-receive-interval 0
```



**Note:** The automatic enabling of BFD on management interface was introduced as part of the Fabric Services System Release 22.8. If you upgrade to Release 22.8, existing fabrics are not automatically modified to enable BFD in this way. For an existing fabric to benefit from this automatic configuration, you must regenerate and redeploy the fabric intent.

## 5.4 Supported hardware

The Fabric Services System currently supports hardware in the leaf, spine, backbone, and border leaf roles as shown in [Table 9: Supported hardware and roles](#).

Table 9: Supported hardware and roles

Hardware type	EVPN capable?	Eligible for these roles			
		T1 (Leaf)	T2 (Spine)	T3 (Back-bone)	T3 (Border Leaf)
7250 IXR-6	-	-	Yes	Yes	-
7250 IXR-10	-	-	Yes	Yes	-
7220 IXR-D1 <sup>1</sup>	Yes	Yes	-	-	-
7220 IXR-D2	Yes	Yes	-	-	-
7220 IXR-D2L	Yes	Yes	-	-	-
7220 IXR-D3	Yes	Yes	Yes	Yes	Yes
7220 IXR-D3L	Yes	Yes	Yes	Yes	Yes
7220 IXR-D4	-	Yes	Yes	-	-
7220 IXR-D5 <sup>2</sup>	-	Yes	Yes	-	Yes
7220 IXR-H2	-	-	Yes	Yes	-
7220 IXR-H3 <sup>3</sup>	-	-	Yes	Yes	-
210 WBX-32Q <sup>4</sup>	Yes	Yes	Yew	Yes	Yes

1. IXR-D1 nodes are supported only as unmanaged nodes.
2. The IXR-D5 is supported only for manual fabric topologies.
3. The system does not configure port speed or support breakout capability on 7220 IXR-H3 nodes.
4. The 210 WBX 32Q is supported only for manual fabric topologies. Supported releases are 20.10.R10 and 6.0.17.

### 5.4.1 Hardware-driven exceptions and special cases

The type of hardware used in a fabric intent can impact the Fabric Services System features available to you.

For WBX nodes:

- manual topology only: you must import the fabric topology, it cannot be automatically generated by the system.
- updates to the fabric topology require importing a new topology file with the new configuration, which can then be associated with a new candidate version of the fabric intent.

- although network resources pools are active and can be selected in the Fabric Services GUI during fabric creation, these pools are not used for WBX nodes and should not be selected in the UI for WBX fabrics.
- LAG configurations must be specified within the manual topology; you cannot configure LAGs for WBX nodes using the Fabric Services System GUI.
- the JSON configuration for WBX nodes cannot be displayed on the Fabric Intents page.

## 5.5 Software and image catalogs

When you create a fabric, you must specify what operating system and version should be present on all of the nodes that participate in the fabric. You select a particular OS software vendor, type, and version number as part of the fabric design, and the system deploys that software to each node as part of the overall fabric deployment.

Because you select only one image file for the fabric intent as a whole, the same software version is deployed to all nodes participating in the fabric. If a node is already running a different software version, it downloads the new software image file and uses it to replace its current version when you deploy the fabric.

### Related topics

[Viewing software and image catalogs](#)

## 5.6 Fabric intents page

The fabric intents page in the Fabric Services System GUI allows you to create and manage individual fabrics.

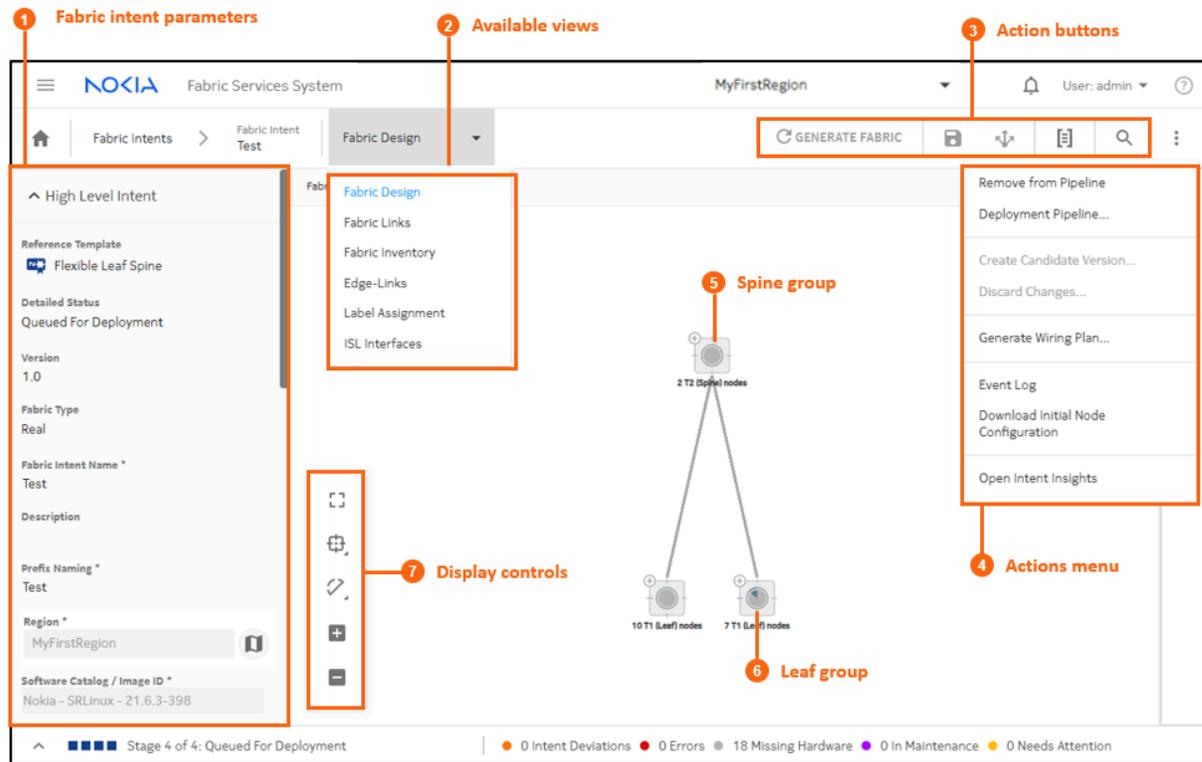


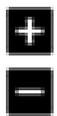
Table 10: Fabric Intents page elements

#	Description
1	Fabric intent parameters: when creating a fabric intent, enter basic parameters here. When viewing a fabric intent, configured parameters are displayed here.
2	Available views: several different views available from this page allows you to interact with different aspects of the fabric intent: <ul style="list-style-type: none"> <li>• <b>Fabric Design:</b> the basic view showing a map of the fabric topology.</li> <li>• <b>Fabric Links:</b> displays a list of links within the fabric topology, and information about each link.</li> <li>• <b>Fabric Inventory:</b> displays a list of all planned nodes within the fabric intent topology, and information about each node.</li> <li>• <b>Edge Links:</b> available only for version 2.0 or greater for a fabric intent, allows you to view and configure the edge links on leaf nodes. From the Edge Links view you can configure LAGs and breakout ports.</li> <li>• <b>Label Assignment:</b> allows you to assign labels to various elements of the fabric intent.</li> <li>• <b>ISL Interfaces:</b> displays a list of ISL interfaces and labels attached to them, if any.</li> </ul>
3	Action buttons: <ul style="list-style-type: none"> <li>• <b>GENERATE FABRIC:</b> use to generate a topology map for a saved fabric intent.</li> </ul>

#	Description
	<ul style="list-style-type: none"> <li>• <b>Save</b> (💾): use to save a fabric intent design.</li> <li>• <b>Deploy</b> (📤): use to send a generated fabric intent to the region's deployment queue.</li> <li>• <b>View fabric as code</b> (📄): use to view the configuration code for the current fabric; from that view you can compare this version's code against a previous version of the same fabric intent, or view the configuration code for an individual node within the fabric.</li> <li>• <b>Search for node</b> (🔍): click to search for a node on the topology map; the selected node is highlighted.</li> </ul>
4	<p>Actions menu:</p> <ul style="list-style-type: none"> <li>• <b>Remove from Pipeline</b>: if you have added a fabric intent to the region's deployment pipeline, this action removes it.</li> <li>• <b>Deployment Pipeline...</b>: this action takes you to the Deployment Pipeline page.</li> <li>• <b>Create Candidate Version...</b>: for a deployed fabric intent, this action creates a new version of the same fabric intent.</li> <li>• <b>Discard Changes...</b>: if you are in the midst of creating a new version of a fabric intent, this action discards your changes and resets the page to display the latest version of the intent.</li> <li>• <b>Generate Wiring Plan...</b>: this action creates a downloadable .csv file describing the wiring plan for the current fabric intent.</li> <li>• <b>Event Log</b>: this action opens the event log.</li> <li>• <b>Download Initial Node Configuration</b>: this action creates a downloadable .csv file containing the initial configuration file for each node in the fabric intent.</li> <li>• <b>Open Intent Insights</b>: this action opens the Operational and Health Insights page.</li> <li>• <b>Update Digital Sandbox</b>: if creating a fabric intent destined for the Digital Sandbox, this action updates the Digital Sandbox with the latest configuration data for the fabric intent.</li> <li>• <b>Cancel Digital Sandbox Update</b>: if you have begun an update of the Digital Sandbox, this action cancels that update.</li> </ul>
5	<b>Spine group</b> : denotes a collection of nodes acting as the fabric intent's spine. Click the + to expand the cluster and view individual spine nodes. For more information see <a href="#">Groups</a>
6	<b>Leaf group</b> : denotes a collection of nodes acting as the fabric intent's leaves. Click the + to expand the cluster and view individual leaf nodes.
7	<b>Display controls</b> : allows you to configure the topology display.

Table 11: Display controls

Icon	Description
	Fit to screen: Adjusts the magnification setting for the topology display so that the entire topology diagram fits inside the current window.

Icon	Description
	Clustering controls: <ul style="list-style-type: none"> <li>enables or disables the use of groups on the map</li> <li>sets the grouping density (the maximum number of nodes represented by a single group)</li> </ul>
	Adjust links: Allows you to control the way links are displayed in the fabric topology.
	Zoom in, Zoom out: Increases or decreases the magnification for the topology diagram.

## 5.7 Manual fabric topologies

You can use the Fabric Services System to import a predefined topology to serve as the basis for future fabric intents, or for a fabric intent you are in the process of designing.

This capability assumes that you have already created a file that describes a fabric topology. You can then:

- use the Topologies page to import the topology for use in a future fabric intent
- use the Fabric Design page to import that topology for use in the current fabric intent.

The topology file you import must be in JSON or YAML format, and must include information about nodes, links, and device profiles. After you import the topology, you can use it as the basis for any number of fabric intents you create with the Fabric Services System.

For node configurations contained within the manual topology, the software version specified for each node must be among the software versions supported in the Fabric Service System's software catalog.

This capability is supported for a topology consisting of nodes that run SR Linux; but it is also supported for topologies composed of hardware that does not run SR Linux. The following node types are currently supported for manual topologies:

- 7250-IXR and 7220-IXR nodes running SR Linux (see [Supported hardware](#))
- 210 WBX-32Q nodes

The types of configurations supported in such manual topologies vary depending on the hardware involved. For example, a manual topology consisting of WBX nodes can include LAGs, but one of SR Linux nodes cannot. (For SR Linux topologies, you can add LAGs afterward using conventional methods within the Fabric Services System).

After you create a fabric intent using a manual, imported topology, you can include it in workload intents as you would any other fabric intent. The special considerations for manual, imported topologies extend only to the design and updates for the fabric intent itself; subsequent workflow involving such fabric intents is unchanged.

### 5.7.1 Elements of a topology file

The fabric topology described in a JSON configuration file can include the following elements, some of which are optional while others are mandatory:

- identifying information for the topology itself
- a set of nodes
- a set of links
- a set of device profiles
- a set of interface profiles
- LAG descriptions (for WBX only; this is not supported for an SR Linux topology)

The following tables describe each of these elements in detail.

*Table 12: Topology identifying elements*

Field Name	Default Value	Mandatory/ Optional	Description of use/requirement
name		Mandatory	The file must include a name for the topology.
description		Optional	Optionally, you can include a description for the topology.

*Table 13: Node description*

Field Names	Default Value	Mandatory/ Optional	Description of use/requirement
deviceProfile		Mandatory	You must include a reference to a device profile which must also be specified as part of the topology template. If the device profile does not exist within the template file, the import fails.
systemName		Mandatory	You must include the system name of the node. Upon import for a particular fabric, the fabric intent prefix is prepended to this name. Only one entry with the same system name is allowed per template.
domainName		Optional	To support the ability to manually set the Fully Qualified Domain Name (FQDN) for a given node, manual topologies support the setting of a domain name as well as a system name.
deviceVendor		Mandatory	You must specify the following attributes for the node's device vendor: <ul style="list-style-type: none"> <li>• operating system (for example, SRLinux)</li> <li>• software version (for example, 22.6.1)</li> <li>• vendor (for example, Nokia)</li> </ul>

			The system validates that the specified software is supported by the version in the software catalogue for the chassis type of the device profile.
isManaged		Optional	<p>This optional parameter is used to indicate when a node should not be directly managed by the Fabric Services System, but is instead managed by some external process. When this is the case, this parameter must be included and its value set to "false".</p> <p> <b>Note:</b> You cannot mix both managed and unmanaged nodes within a single topology. The Fabric Services System does not check whether you have adhered to this restriction when you upload a topology file. It is your responsibility to ensure that any topology does not mix both managed and unmanaged nodes.</p>
lags		Optional	<p>You can include a list of LAGs that already exist on the device.</p> <p>This value is only allowed if the node is considered unmanaged, meaning that the Fabric Services System itself is not managing the node configuration. For example, this value is supported for WBX nodes, but not for SR Linux nodes.</p> <p>For more information about LAG data, see the table <a href="#">LAGs</a>.</p>
pod		Optional	You can include a label to identify a group of nodes.
rack		Optional	You can include a label to identify a grouping of nodes which reside in the same physical rack.
role		Mandatory	<p>You must include a role for the node.</p> <p>Several roles are available to identify a node's position within the fabric hierarchy:</p> <ul style="list-style-type: none"> <li>• T1_LEAF</li> <li>• T2_SPINE</li> <li>• T3</li> <li>• BORDERLEAF</li> </ul>

Table 14: LAGs

Field Names	Default Value	Mandatory/Optional	Description of use/requirement
name		Mandatory	You must include a text string that is the name for a LAG.

localName		Mandatory	You must include the name of the LAG as specified on the device itself; for example, lag1   <b>Note:</b> A node must not include more than one LAG with the same local name.
isMultiHome	True	Mandatory	You must indicate whether the LAG is multi-homed.
ports		Mandatory	You must include a list of ports that constitute this particular LAG.
 <b>Note:</b> LAG information is supported only for unmanaged topologies, such as those consisting of 210 WBX nodes. For managed topologies, such as those consisting of SR Linux nodes, configure LAGs conventionally as you would for a non-imported topology.			

Table 15: Device profile

Field Name	Default Value	Mandatory/Optional	Description of use/requirement
name		Mandatory	You must include a name for the device profile.
description		Optional	You can include a description for the device profile
chassis type		Mandatory	You must include one of the following chassis types: <ul style="list-style-type: none"> <li>• 210-WBX-32Q</li> <li>• 7220 IXR-D1</li> <li>• 7220 IXR-D2</li> <li>• 7220 IXR-D2L</li> <li>• 7220 IXR D3</li> <li>• 7220 IXR D3L</li> <li>• 7220 IXR-D5</li> <li>• 7220 IXR-H2</li> <li>• 7220 IXR-H3</li> <li>• 7250 IXR-10</li> <li>• 7250 IXR-6</li> </ul>
edge links		Optional	You can include a list of edge links on nodes acting as a leaf, border leaf, or spine. Only interfaces which are valid according to the inventory for the chassis type are accepted.   <b>Note:</b> Edge links can be configured on spine nodes as part of a manual topology links, and these links will be available for use as subinterfaces in workload intents. This capability is not currently supported for fabric

			<p>intents configured in the Fabric Services System GUI.</p> <p>If a configured interface is not valid, the device profile is not accepted and an error results when generating the fabric configuration.</p> <p>An edge link interface should not be allowed to overlap with an interface configured in a Link object. Overlapping entries are rejected and an error results when generating the fabric configuration.</p> <p>An edge link should only be specified once for a device profile. Duplicate entries are rejected and an error results when generating the fabric configuration.</p>
loopbacks		Optional	You can include set of loopback interfaces for the device.
interfaceGroups		Optional	Interface groups are used to group a set of interfaces within a device and apply an interface profile to the list of interfaces within the group.

Table 16: Interface groups

Field Names	Default Value	Mandatory/Optional	Description of use/requirement
type		Mandatory	Defines the type of interface. Only "ISL" is currently supported.
interfaces		Mandatory	A list of interfaces belonging to the group. Interfaces cannot belong to multiple interface groups.
interfaceProfile		Mandatory	A reference to an interface profile. If this is specified the profile must also be present within the topology definition. If the interface_profile is not specified then all interface parameters for the list of interfaces must use their default values (such as speed). Interface profiles can be referenced by multiple interface groups.

Table 17: Interface profiles

Field Names	Default Value	Mandatory/Optional	Description of use/requirement
name		Mandatory	This is the name of the interface profile
breakout		Optional	Breakout is a nested object which, if it contains any configuration, will dictate that the interface should use breakouts. The nested breakout object is defined separately.

forwardError Correction	rs-528	Optional	<p>For ISL ports, enables or disables Forward Error Correction (FEC) on the interface. Supported values are:</p> <ul style="list-style-type: none"> <li>• base-r</li> <li>• disabled</li> <li>• rs-108</li> <li>• rs-528</li> <li>• rs-544</li> </ul> <p> <b>Note:</b> The same port cannot be configured both as a breakout port and with FEC as part of the same interface profile. To configure a port as both broken-out and with FEC:</p> <ol style="list-style-type: none"> <li>1. First apply an interface profile that breaks out the port into multiple broken-out sub-ports.</li> <li>2. Then apply a different interface profile that includes FEC on the broken-out sub-port. Individual sub-ports could be configured this way with different FEC settings.</li> </ol>
-------------------------	--------	----------	--

Table 18: Breakout

Field Names	Default Value	Mandatory/ Optional	Description of use/requirement
numChannel		Mandatory	The number of channels for the interface breakout. Values are validated per interface.
channelSpeed		Mandatory	The speed of each channel. Values are validated per interface.

Table 19: Links

Field Name	Default Value	Mandatory/ Optional	Description of use/requirement
localNode		Mandatory	You must include the local node name. This must be a valid and existing node name in the topology template.
localPort		Mandatory	You must identify the interface on the local node that is attached to the remote port on the remote node. This interface cannot belong to more than one link and cannot be an edge link. If the native interface has been broken out with a configuration in interfaceGroups, identify the broken-out port numbers.

remoteNode		Mandatory	You must include the remote node name. This must be a valid and existing node name in the topology template. If the native interface has been broken out with a configuration in interfaceGroups, identify the broken-out port numbers.
remotePort		Mandatory	You must identify the interface on the remote node which is attached to the local port on the local node. This interface cannot belong to more than one link and cannot be an edge link.
isActive		Optional	You can indicate with a value of true or false whether the links should be active. If false, an IP address is allocated to the link but the Administrative status is not set to Up.
speed		Optional	You can indicate a speed for the link.
role		Mandatory	You must indicate the relationship between the nodes in this link. Any link that does not fall into one of the existing roles can be assigned a generic ISL role.  Supported roles are: <ul style="list-style-type: none"> <li>• T1_ISL_T2</li> <li>• T2_ISL_T3</li> <li>• T3_ISL_T4</li> <li>• ISL</li> </ul>

### Additional manual topology settings

When you create a fabric intent using a manual topology, you first import that topology and then configure the fabric intent using the Fabric Services System GUI. You have the opportunity on the Fabric Intents page to configure additional settings for the manual topology, including the selection of IP and ASN pools other than the default pools defined for the region.

For more information, see the procedure for creating a fabric using a manual topology.

### Topology validation

When you import a topology into the Fabric Services System, the system performs several validation checks on the data in the selected topology.

For node-level data, the system verifies that:

- If there are duplicate system names, only the first instance is accepted.
- Every node has a valid device profile name that is defined in the device profile section.
- For each chassis type, the software image version is equal to or greater than the version supported in the system's software catalog.

For link data, the system verifies that:

- a valid node name is provided for the local node
- a valid port is provided for the local port
- a valid node name is provided for the remote node

- a valid port is provided for the remote port
- all ports for the local and remote nodes are unique
- Any edge link removed as part of the imported topology were not configured with breakout in the previous version of the intent
- any edge links removed as part of the imported topology are not already participating in a LAG
- any edge links removed as part of the imported topology are not already participating in a workload
- for ports participating in an inter-switch link (ISL), the configured port speed is supported by the ports on both ends of the link.



**Note:** If a broken-out port is used in an ISL and the speed attribute is set on that link, the Fabric Services System checks that the speed value is the same as the channel speed of that broken-out port.

For edge port data, the system verifies that:

- the port is valid based on the catalog for the device type.
- for managed nodes only, each port used for edge links and ISLs is unique



**Note:** This check is not performed for unmanaged nodes.

For LAG data, the system verifies that:

- each LAG name is locally unique.
- for managed nodes only, no LAG data is provided for an imported topology intended for SR Linux nodes.



**Note:** This check is not performed for unmanaged nodes.

For breakout ports, the system verifies that:

- The number of channels for the interface breakout is supported by the interface.
- The speed of each channel is supported by the interface.

## 5.7.2 Manual topology file examples

### Example 1: Leaf Spine topology with 210 WBX nodes

```
{
  {
    "links": [{
      "localNode": "t1-leaf1",
      "localPort": "1/1/9",
      "remoteNode": "t2-spine1",
      "remotePort": "1/1/9",
      "role": "ISL",
      "isActive": true
    }],
    {
```

```
"localNode": "t1-leaf1",
"localPort": "1/1/13",
"remoteNode": "t2-spine2",
"remotePort": "1/1/13",
"role": "ISL",
"isActive": true
},
{
"localNode": "t1-leaf2",
"localPort": "1/1/17",
"remoteNode": "t2-spine2",
"remotePort": "1/1/17",
"role": "T1_ISL_T2",
"isActive": true
},
{
"localNode": "t1-leaf3",
"localPort": "1/1/21",
"remoteNode": "t2-spine1",
"remotePort": "1/1/21",
"role": "T1_ISL_T2",
"isActive": true
},
{
"localNode": "t1-leaf3",
"localPort": "1/1/25",
"remoteNode": "t2-spine2",
"remotePort": "1/1/25",
"role": "T1_ISL_T2",
"isActive": true
},
{
"localNode": "t1-leaf4",
"localPort": "1/1/29",
"remoteNode": "t2-spine3",
"remotePort": "1/1/29",
"role": "T1_ISL_T2",
"isActive": true
},
{
"localNode": "t2-spine1",
"localPort": "2/1/9",
"remoteNode": "t3-spine1",
"remotePort": "2/1/9",
"role": "T2_ISL_T3",
"isActive": true
},
{
"localNode": "t2-spine1",
"localPort": "2/1/13",
"remoteNode": "t3-spine2",
"remotePort": "2/1/13",
"role": "T2_ISL_T3",
"isActive": true
},
{
"localNode": "t3-spine1",
"localPort": "2/1/17",
"remoteNode": "t4-bleaf1",
"remotePort": "2/1/17",
"role": "T3_ISL_T4",
"isActive": true
},
{
```

```

    "localNode": "t3-spine2",
    "localPort": "2/1/21",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/21",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine1",
    "localPort": "2/1/25",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/25",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine2",
    "localPort": "2/1/29",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/29",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine2",
    "localPort": "2/1/33",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/33",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine2",
    "localPort": "2/1/37",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/37",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine2",
    "localPort": "2/1/41",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/41",
    "role": "T3_ISL_T4",
    "isActive": true
  },
  {
    "localNode": "t3-spine2",
    "localPort": "2/1/45",
    "remoteNode": "t4-bleaf2",
    "remotePort": "2/1/45",
    "role": "T3_ISL_T4",
    "isActive": true
  }
],
"nodes": [{
  "deviceProfile": "wbx_t1",
  "role": "T1_LEAF",
  "sequence": 1,
  "systemName": "t1-leaf1",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {

```

```

    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  },
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
  ],
  {
    "deviceProfile": "wbx_t1",
    "role": "T1_LEAF",
    "sequence": 1,
    "systemName": "t1-leaf2",
    "pod": "pod1",
    "rack": "1",
    "deviceVendor": {
      "operatingSystem": "SROS210WBX",
      "softwareVersion": "22.6.2-24",
      "vendor": "Nokia"
    },
    "lags": [{
      "name": "lag-1",
      "localName": "lag-1",
      "isMultiHome": false,
      "ports": [
        "1/1/22",
        "1/1/23"
      ]
    },
    {
      "name": "lag-10",
      "localName": "lag-10",
      "isMultiHome": false,
      "ports": [
        "1/1/25",
        "1/1/24"
      ]
    }
  ]
  },
  {
    "deviceProfile": "wbx_t1",
    "role": "T1_LEAF",
    "sequence": 1,
    "systemName": "t1-leaf3",
    "pod": "pod1",
    "rack": "1",
    "deviceVendor": {

```

```

    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  },
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
],
{
  "deviceProfile": "wbx_t1",
  "role": "T1_LEAF",
  "sequence": 1,
  "systemName": "t1-leaf4",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  },
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
],
{
  "deviceProfile": "wbx_t1",
  "role": "T1_LEAF",
  "sequence": 1,
  "systemName": "t1-leaf5",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {

```

```

    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  },
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
  ],
  {
    "deviceProfile": "wbx_t1",
    "role": "T1_LEAF",
    "sequence": 1,
    "systemName": "t1-leaf6",
    "pod": "pod1",
    "rack": "1",
    "deviceVendor": {
      "operatingSystem": "SROS210WBX",
      "softwareVersion": "22.6.2-24",
      "vendor": "Nokia"
    },
    "lags": [{
      "name": "lag-1",
      "localName": "lag-1",
      "isMultiHome": false,
      "ports": [
        "1/1/22",
        "1/1/23"
      ]
    },
    {
      "name": "lag-10",
      "localName": "lag-10",
      "isMultiHome": false,
      "ports": [
        "1/1/25",
        "1/1/24"
      ]
    }
  ]
  },
  {
    "deviceProfile": "wbx_t2",
    "role": "T2_SPINE",
    "sequence": 1,
    "systemName": "t2-spine1",
    "pod": "pod1",
    "rack": "1",
    "deviceVendor": {

```

```

    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": []
},
{
  "deviceProfile": "wbx_t2",
  "role": "T2_SPINE",
  "sequence": 1,
  "systemName": "t2-spine2",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": []
},
{
  "deviceProfile": "wbx_t2",
  "role": "T2_SPINE",
  "sequence": 1,
  "systemName": "t2-spine3",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": []
},
{
  "deviceProfile": "wbx_t2",
  "role": "T2_SPINE",
  "sequence": 1,
  "systemName": "t2-spine4",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": []
},
{
  "deviceProfile": "wbx_t3",
  "role": "T3",
  "sequence": 1,
  "systemName": "t3-spine1",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": []
},
{
  "deviceProfile": "wbx_t3",

```

```

"role": "T3",
"sequence": 1,
"systemName": "t3-spine2",
"pod": "pod1",
"rack": "1",
"deviceVendor": {
  "operatingSystem": "SR0S210WBX",
  "softwareVersion": "22.6.2-24",
  "vendor": "Nokia"
},
"lags": []
},
{
"deviceProfile": "wbx_t3",
"role": "T3",
"sequence": 1,
"systemName": "t3-spine3",
"pod": "pod1",
"rack": "1",
"deviceVendor": {
  "operatingSystem": "SR0S210WBX",
  "softwareVersion": "22.6.2-24",
  "vendor": "Nokia"
},
"lags": []
},
{
"deviceProfile": "wbx_t3",
"role": "T3",
"sequence": 1,
"systemName": "t3-spine4",
"pod": "pod1",
"rack": "1",
"deviceVendor": {
  "operatingSystem": "SR0S210WBX",
  "softwareVersion": "22.6.2-24",
  "vendor": "Nokia"
},
"lags": []
},
{
"deviceProfile": "wbx_t4",
"role": "BORDERLEAF",
"sequence": 1,
"systemName": "t4-bleaf1",
"pod": "pod1",
"rack": "1",
"deviceVendor": {
  "operatingSystem": "SR0S210WBX",
  "softwareVersion": "22.6.2-24",
  "vendor": "Nokia"
},
"lags": [{
  "name": "lag-1",
  "localName": "lag-1",
  "isMultiHome": false,
  "ports": [
    "1/1/22",
    "1/1/23"
  ]
}],
{
  "name": "lag-10",
  "localName": "lag-10",

```

```

    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
],
{
  "deviceProfile": "wbx_t4",
  "role": "BORDERLEAF",
  "sequence": 1,
  "systemName": "t4-bleaf2",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  }],
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
],
{
  "deviceProfile": "wbx_t4",
  "role": "BORDERLEAF",
  "sequence": 1,
  "systemName": "t4-bleaf3",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SROS210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  }],
  {
    "name": "lag-10",
    "localName": "lag-10",

```

```

    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
],
{
  "deviceProfile": "wbx_t4",
  "role": "BORDERLEAF",
  "sequence": 1,
  "systemName": "t4-bleaf4",
  "pod": "pod1",
  "rack": "1",
  "deviceVendor": {
    "operatingSystem": "SR0S210WBX",
    "softwareVersion": "22.6.2-24",
    "vendor": "Nokia"
  },
  "lags": [{
    "name": "lag-1",
    "localName": "lag-1",
    "isMultiHome": false,
    "ports": [
      "1/1/22",
      "1/1/23"
    ]
  },
  {
    "name": "lag-10",
    "localName": "lag-10",
    "isMultiHome": false,
    "ports": [
      "1/1/25",
      "1/1/24"
    ]
  }
]
},
],
"deviceProfiles": [{
  "name": "wbx_t3",
  "chassisType": "210-WBX-32Q",
  "description": "wbx_t3",
  "edgeLinks": []
},
{
  "name": "wbx_t4",
  "chassisType": "210-WBX-32Q",
  "description": "wbx_t4",
  "edgeLinks": [
    "1/1/1",
    "1/1/2",
    "1/1/3",
    "1/1/4",
    "1/1/5",
    "1/1/46",
    "1/1/47",
    "1/1/48",
    "1/1/6",
    "1/1/7",
    "1/1/8",
    "1/1/35",

```

```

    "1/1/36",
    "1/1/37",
    "1/1/38",
    "1/1/39"
  ]
},
{
  "name": "wbx_t1",
  "chassisType": "210-WBX-32Q",
  "description": "wbx_t1",
  "edgeLinks": [
    "1/1/1",
    "1/1/2",
    "1/1/3",
    "1/1/4",
    "1/1/5",
    "1/1/46",
    "1/1/47",
    "1/1/48",
    "1/1/6",
    "1/1/7",
    "1/1/8",
    "1/1/35",
    "1/1/36",
    "1/1/37",
    "1/1/38",
    "1/1/39"
  ]
},
{
  "name": "wbx_t2",
  "chassisType": "210-WBX-32Q",
  "description": "wbx_t2",
  "edgeLinks": []
}
],
"template": {
  "name": "wbx_topology_t1_t2_t3_t4"
}
}

```

### Example 2: 7220 IXR-D5 with breakout ports and FEC

```

{
  "interfaceProfiles": [
    {
      "name": "ipG100-2",
      "breakout": {
        "channelSpeed": "G100",
        "numChannels": 2
      }
    },
    {
      "name": "ipG100-4",
      "breakout": {
        "channelSpeed": "G100",
        "numChannels": 4
      }
    }
  ],
  {
    "forwardErrorCorrection": "base-r",
    "name": "fec-profile1"
  },
}

```

```

    {
      "forwardErrorCorrection": "rs-528",
      "name": "fec-profile2"
    }
  ],
  "deviceProfiles": [
    {
      "name": "d5-spine",
      "chassisType": "7220 IXR-D5",
      "description": "spine-7220 IXR-D5",
      "edgeLinks": [],
      "interfaceGroups": [
        {
          "interfaceProfile": "fec-profile1",
          "interfaces": [
            "ethernet-1/29/1",
            "ethernet-1/29/2",
            "ethernet-1/7/1",
            "ethernet-1/7/2"
          ],
          "type": "ISL"
        },
        {
          "interfaceProfile": "ipG100-2",
          "interfaces": [
            "ethernet-1/29",
            "ethernet-1/7"
          ],
          "type": "ISL"
        }
      ]
    },
    {
      "name": "h3-spine",
      "chassisType": "7220 IXR-H3",
      "description": "spine-7220 IXR-H3",
      "edgeLinks": [],
      "interfaceGroups": [
        {
          "interfaceProfile": "fec-profile2",
          "interfaces": [
            "ethernet-1/24/1",
            "ethernet-1/24/2",
            "ethernet-1/24/3",
            "ethernet-1/24/4"
          ],
          "type": "ISL"
        },
        {
          "interfaceProfile": "ipG100-4",
          "interfaces": [
            "ethernet-1/24"
          ],
          "type": "ISL"
        }
      ]
    },
    {
      "name": "d3l-leaf",
      "chassisType": "7220 IXR-D3L",
      "description": "d3lleaf",
      "edgeLinks": [
        "ethernet-1/1",
        "ethernet-1/2"
      ]
    }
  ]
}

```

```

    ],
    "interfaceGroups": [
      {
        "interfaceProfile": "fec-profile1",
        "interfaces": [
          "ethernet-1/4",
          "ethernet-1/5"
        ],
        "type": "ISL"
      },
      {
        "interfaceProfile": "fec-profile2",
        "interfaces": [
          "ethernet-1/6",
          "ethernet-1/7"
        ],
        "type": "ISL"
      }
    ]
  },
  ],
  "nodes": [
    {
      "deviceProfile": "d3l-leaf",
      "role": "T1_LEAF",
      "sequence": 1,
      "systemName": "leaf1",
      "pod": "pod1",
      "rack": "1",
      "deviceVendor": {
        "operatingSystem": "SRLinux",
        "softwareVersion": "22.11.1-184",
        "vendor": "Nokia"
      }
    },
    {
      "deviceProfile": "d3l-leaf",
      "role": "T1_LEAF",
      "sequence": 1,
      "systemName": "leaf2",
      "pod": "pod1",
      "rack": "1",
      "deviceVendor": {
        "operatingSystem": "SRLinux",
        "softwareVersion": "22.11.1-184",
        "vendor": "Nokia"
      }
    },
    {
      "deviceProfile": "d5-spine",
      "role": "T2_SPINE",
      "sequence": 1,
      "systemName": "spine1",
      "pod": "pod1",
      "rack": "1",
      "deviceVendor": {
        "operatingSystem": "SRLinux",
        "softwareVersion": "22.11.1-184",
        "vendor": "Nokia"
      }
    },
    {
      "deviceProfile": "h3-spine",
      "role": "T2_SPINE",

```

```

    "sequence": 1,
    "systemName": "spine2",
    "pod": "pod1",
    "rack": "1",
    "deviceVendor": {
      "operatingSystem": "SRLinux",
      "softwareVersion": "22.11.1-184",
      "vendor": "Nokia"
    }
  },
],
"links": [
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/4",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/29/1",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/4",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/29/2",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/5",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/7/1",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/5",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/7/2",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/8",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/27",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G40"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/8",
    "remoteNode": "spine1",
    "remotePort": "ethernet-1/28",
    "role": "T1_ISL_T2",
    "isActive": true,

```

```

    "speed": "G40"
  },
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/6",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/24/1",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/6",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/24/2",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/7",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/24/3",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/7",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/24/4",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G100"
  },
  {
    "localNode": "leaf1",
    "localPort": "ethernet-1/9",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/27",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G40"
  },
  {
    "localNode": "leaf2",
    "localPort": "ethernet-1/9",
    "remoteNode": "spine2",
    "remotePort": "ethernet-1/28",
    "role": "T1_ISL_T2",
    "isActive": true,
    "speed": "G40"
  }
},
"template": {
  "description": "leaf-spine topology",
  "name": "leaf-spine"
}
}

```

### 5.7.3 The Topologies page

The Topologies page of the Fabric Services System GUI allows you to:

- view a list of topologies already loaded into the system
- examine any single topology in more detail, either in a graphical view or as a JSON file
- begin creating a new fabric intent based on a selected topology
- delete a topology from the system
- export a selected topology to a JSON file

Figure 16: Topologies page

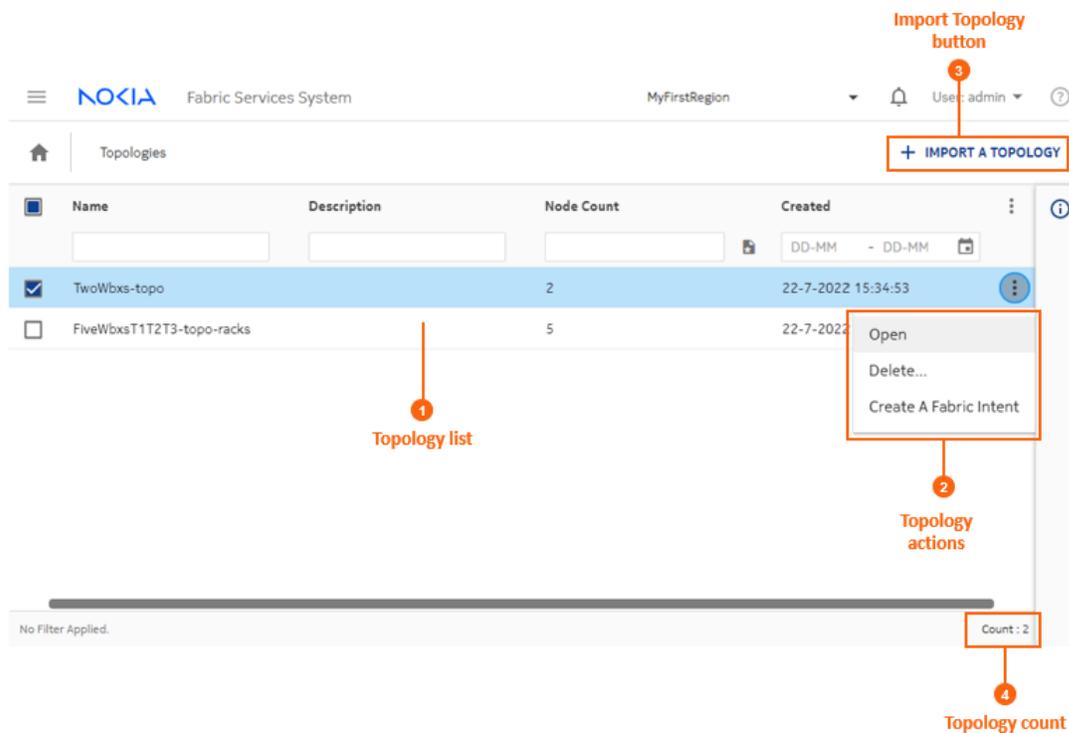


Table 20: Topologies page elements

#	Description
1	This area lists all of the topologies that have been imported into the system.
2	Topology actions: these are actions available for individual topologies in the list. <ul style="list-style-type: none"> <li>• <b>Open:</b> opens a new form displaying a graphical view of the topology, showing its constituent nodes and their links. From this form you can open a JSON view of the same topology, and export that data to a .json file.</li> <li>• <b>Delete...:</b> deletes the selected topology from the list.</li> </ul>

#	Description
	 <p><b>Note:</b> Deleting a topology does not affect any fabric intents that are already using that topology. The topology data for that fabric is fully captured in the fabric intent configuration.</p> <ul style="list-style-type: none"> <li>• <b>Create a Fabric Intent:</b> Opens the Fabric Intents page for the creation of a new fabric intent based on a manual topology, and already configured to use the selected topology file as its basis. See <a href="#">Creating a fabric intent using a manual topology</a>. If you used this method to create your fabric intent, you can begin that procedure at step 6 because the template and the topology are already selected for you.</li> </ul>
3	<b>Import topology</b> button: use this button to import a new topology in the form of a JSON file.
4	<b>Topology count:</b> the number of topologies currently displayed in the list.

Table 21: Topologies page columns

#	Description
<b>Name</b>	The name assigned to the topology, based on the name element in the originating topology file.
<b>Description</b>	The description of the topology based on the description element within the originating topology file, if present.
<b>Node count</b>	The number of nodes within the topology.
<b>Created</b>	The date on which the topology was imported into the Fabric Services System.

### The topology view

When viewing an individual topology, the Topology View page displays.

Figure 17: Topology view

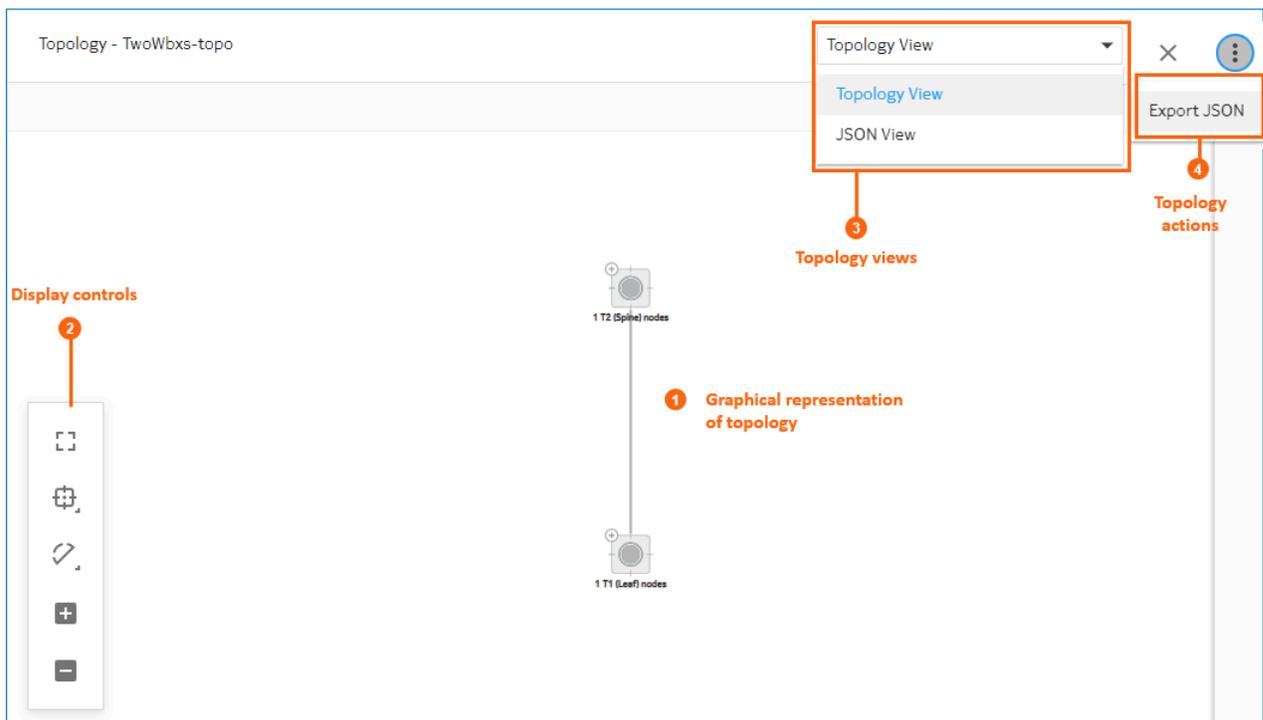


Table 22: Topologies page elements

#	Description
1	This area shows a graphical representation of the topology.
2	Display controls; these behave as described in <a href="#">Table 7: Display controls</a> .
3	<b>Views</b> control allows you to switch between the following: <ul style="list-style-type: none"> <li>• <b>Topology view</b>: the graphical representation of the manual topology</li> <li>• <b>JSON view</b>: a view of the JSON code that describes the topology</li> </ul>
4	Topology actions: <b>Export JSON</b> allows you to export the current topology to a JSON file.

### Related topics

[Lists](#)

[Region map manipulation](#)

## 5.7.4 Manual fabric topology parameters

Table 23: Basic parameters for manual topology fabric intent

Parameter	Description
<b>Fabric Type</b>	Indicates whether this is a Real or Digital Sandbox fabric. Select Real. This creates a fabric intent intended for deployment to real-world hardware. The Digital Sandbox option is used to create a virtual fabric to test and validating prospective designs. This option is described in <a href="#">Digital Sandbox</a> .
<b>Fabric Intent Name</b>	Specifies the name that identifies this fabric.
<b>Imported Topology From File</b>	Specifies the name of a previously imported topology.
<b>Description</b>	Provides a description of the fabric intent. Optional.
<b>Prefix Naming</b>	Specifies a string to be added at the beginning of the name of every node in the fabric intent. The rest of the node name is automatically generated. For example, enter "A01" here to assign nodes names such as "A01-leaf-1" and "A01-spine-1". The Prefix Naming string must be unique to each fabric.
<b>Region</b>	Specifies an already-created region. This identifies the region that contains the new fabric intent.
<b>Domain</b>	<p>Optionally, provides a single domain name to be used as part of the fully qualified domain name (FQDN) for all of the nodes within the fabric. Multiple domain names separated by commas are not supported.</p> <p>When a domain name is provided, the Fabric Services System includes it within the <b>system name</b> parameter of the node configuration:</p> <pre> "system": {   "name": {     "host-name": &lt;fss-sysname&gt;     "domain-name": &lt;domain-name-from-fabric&gt;   } } </pre> <p>The Fabric Services System also updates its DHCP server with the specified domain name and host name.</p> <p>Like other aspects of node configuration, the inclusion of the domain name can be verified by right-clicking the node in the fabric design view and selecting <b>Inspect Configuration</b>.</p> <p> <b>Note:</b> This parameter is supported only for fabric intents created using a manual topology.</p>
<b>Labels</b>	This area is disabled when creating the initial version of any fabric intent.

Parameter	Description
<b>Deploy Trigger Percentage</b>	<p>Specifies the minimum percentage of nodes within the fabric intent that must be physically installed before the system allows you to deploy the intent to hardware.</p> <p>For all real fabrics, this value is fixed at 0%.</p> <p>For a full explanation of the impact of this setting, see <a href="#">The Deployment Trigger Percentage setting</a>.</p>
<b>VLANs on ISL</b>	<p>Specifies whether VLANs are permitted on inter-switch links (ISLs)</p> <p>Enabling this toggle allows the configuration of VLANs on ISLs in the Fabric Services System. This capability is unique to fabric intents using a manual topology; conventionally configured fabric intents do not currently support VLANs on ISLs.</p> <p>Enabling this toggle displays a <b>VLAN ID</b> field. Use this field to specify the VLAN tag that should be created on the ISL endpoints. Only a single VLAN is currently supported.</p> <p>If required you can change the value for this VLAN tag by creating a new version of the fabric intent, selecting a new value here, and deploying the new version.</p> <p>This setting is disabled by default.</p>

## 5.7.5 Importing a manual topology

### Prerequisites

A JSON file that describes a fabric topology must already exist.

### About this task

This procedure describes how to import a topology in the form of a previously created JSON or YAML file. When imported, this topology can be used as the basis for creating new fabrics.

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Topologies** to open the Topologies page.
- Step 3.** Click **+IMPORT A TOPOLOGY**.
- Step 4.** Use the resulting file navigation form to locate the JSON or YAML file describing your topology and click **Open**.

### Expected outcome

The topology described in the selected JSON file is imported into the Fabric Services System, and it now appears as an item in the list of available topologies. You can now create a fabric that uses this topology.

## 5.7.6 Viewing a manual topology

### Prerequisites

You must have imported a topology.

### About this task

This procedure describes how to view one of the manual topologies that has been imported into the Fabric Services System.

#### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Topologies** to open the Topologies page.
- Step 3.** Select a topology from the list.
- Step 4.** Click the **More** icon (  ) at the right edge of the row and select Open from the actions list to open a graphical display of the topology.

## 5.7.7 Exporting a manual topology

### Prerequisites

You must have imported a manual topology.

### About this task

Use this procedure to export a manual topology from the Fabric Services System to a JSON file.

#### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Topologies** to open the Topologies page.
- Step 3.** Select a topology from the list.
- Step 4.** Click the **More** icon (  ) at the right edge of the row and select Open from the actions list to open a graphical display of the topology.
- Step 5.** At the upper right of the page, click the **More** icon (  ) and select Export JSON.

#### Expected outcome

The system saves the current topology as a JSON file.

## 5.7.8 Creating a fabric intent using a manual topology

### Prerequisites

Before you create a new fabric intent based on a manually configured topology file, ensure you have done the following:

- Created a region to which the fabric can belong; see [Creating a region](#).
- Imported a JSON file that describes the topology into the Fabric Services System.

### About this task

This procedure describes how to create a fabric intent based on a previously defined topology that is described in a JSON file.

## Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** In the main menu, select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the fabric intent.
- Step 4.** Click the **+ CREATE A FABRIC INTENT** button to open the Fabric Design page.
- Step 5.** Select or import the topology file on which this fabric is based:
- To select an already-imported topology file, click in the Imported Topology From File field and select the topology from the displayed list.
  - To import a new topology from a file, click the **Import** icon and select the topology file. The topology in that file is automatically selected as the basis for this fabric intent.
- Step 6.** On the left-side panel, enter or select the basic parameters that define your intended fabric as described in [Table 23: Basic parameters for manual topology fabric intent](#).
- Step 7.** Optionally, select an ASN pool and IP pools other than the default pools for the region as described in [Table: Optional parameters for fabric intents](#):
- **ASN Pool Name**
  - **Inter Switch Link Pool Name**
  - **Management Pool Name**
  - **System Pool Name**
- Step 8.** Click  to save the fabric intent. When you save the fabric intent, the system:
- updates the state of the fabric intent to Created.
  - updates the version number of the fabric intent to 1.0.
  - enables the  **GENERATE FABRIC** button.
- Step 9.** Click  **GENERATE FABRIC**.

## Expected outcome

The system generates a recommended topology for your fabric based on the template you selected and the parameters you provided.

When generating the topology, the system also generates the various cable connections and the individual node configurations required to support this fabric topology.

During fabric generation, the fabric intent state advances through the following:

- Cable Map in Progress
- Config Generation in Progress
- Configuration Generated

After the generation is complete, the resulting topology displays in the main area of the **Fabric Intents** page.

## 5.7.9 Updating a fabric with a new manual topology

### Prerequisites

- You must have created a JSON file that describes the intended new topology.
- You must have imported that topology file into the Fabric Services System so that it appears in the list of available manual topologies.

### About this task

After you have deployed a fabric intent that is based on a manual, imported topology, you can update the fabric intent's topology by selecting and applying a different imported topology.

Like any change to a deployed fabric intent, you begin by creating a new candidate version of the fabric intent. A new candidate version of a fabric intent keeps the same fabric intent name, but the system assigns an incremented version number, date, and (if necessary) a new user association. You then apply the new topology to this new version of the fabric intent.

You can update a fabric topology in this way even if the fabric intent is already being used as part of an existing workload intent. However if the new topology deletes a node that is participating in a workload intent, the deployment of the new fabric intent fails. The reason for the failure is captured in the event log.

For this reason, before applying a topology update that deletes a node, ensure that any workloads using that node are first updated to eliminate that node from their configuration.

To update a fabric intent with a new manual topology:

### Procedure

**Step 1.** Click the  menu.

**Step 2.** Select **Fabric Intents**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the fabric intent.

**Step 4.** To open a specific fabric intent from the list, do one of the following:

- Double-click the row for that fabric intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

**Step 5.** Click the  menu at the upper right of the page.

**Step 6.** Select **Create Candidate Version**.

#### Expected outcome

The system saves a new version of the current fabric intent.

You can now edit the fabric intent's parameters and save the result without affecting the original version.

**Step 7.** Use the **Imported Topology From File** drop-down to select a new topology from the topology list.

If the new topology adds new nodes, the existing nodes retain their IDs in the new topology. Only new nodes are assigned new IDs.

**Step 8.** Click  to save the fabric intent. When you save the fabric intent, the system:

- updates the state of the fabric intent to Created.

- updates the version number of the fabric intent to 1.0.
- enables the  **GENERATE FABRIC** button.

**Step 9.** Click **GENERATE FABRIC** to generate a new fabric diagram.

You can now view the fabric intent, and eventually add it to the deployment pipeline and deploy it as you would for a new fabric intent.

At any time before deploying the new version, you can discard this candidate version and revert to the previously deployed version by clicking on the **More actions** icon (  ) in the Fabric Design view and selecting **Discard Changes**.

## 5.8 Fabric intents with unmanaged nodes

Typically, the process of creating a fabric intent assumes that the Fabric Services System itself is responsible for managing all of the nodes participating in the fabric. The Fabric Services System creates an initial configuration for each node, and then deploys and later updates that node configuration as part of the ongoing development and maintenance of the fabric. Typically, the Fabric Services System also uses its internal DHCP server to manage the IP addresses assigned to nodes.

In some cases, however, the management of the nodes within the fabric and the links between them might be reserved exclusively for some process external to the Fabric Services System, and the network also maintains its own, external DHCP server. This scenario is described by the Fabric Services System as a fabric consisting of "unmanaged nodes".

In such a case, all of the capabilities of the Fabric Services System to monitor existing fabrics and to create and manage workloads can still be used, but any alteration to the configuration of the nodes and links that constitute the underlying fabric itself is scrupulously avoided, and the existing, externally managed configurations of all nodes is carefully protected.

The procedure to create a fabric intent consisting of unmanaged nodes is described in [Creating a fabric intent for unmanaged nodes](#). Once fabric intent creation is complete, the Fabric Services System discovers all of the nodes within the fabric, including details regarding any workloads that may already exist on that fabric.

From that point onward the Fabric Services System assumes responsibility for any node configuration pertaining to workloads. It also offers, outside of fabric configuration, the full set of features that would be available for a typical fabric consisting of managed nodes. But the system leaves the underlying fabric configuration untouched.



**Note:** Currently, the Fabric Services System only supports fabrics consisting entirely of managed nodes, or entirely of unmanaged nodes. Fabrics consisting of some managed nodes and some unmanaged nodes are not supported.



**Notice:** Unmanaged nodes must be running a version of SR Linux that is supported by the current release of the Fabric Services System. This includes any software version that has been manually added to the Fabric Service System's software catalog. It is important that you identify the SR Linux software version correctly in the manual topology file for every unmanaged node. It is the operator's responsibility to ensure that the software is identified correctly; the system does not validate the software version during deployment.

If the SR Linux software running on the unmanaged node does not match that indicated in the manual topology file, the system will continue to attempt automatic deployment to the node

indefinitely. As a result, the region's deployment pipeline could become filled with failed automatic deployments and this could irretrievably affect the behavior of the system.

### Manual topology files for fabrics with unmanaged nodes

Any fabric consisting of unmanaged nodes must be created within the Fabric Services System as a fabric intent based upon an imported, manual topology.

There are two unique aspects of any manual topology file that describes a fabric of unmanaged nodes:

- Node descriptions in manual topologies support the optional "IsManaged" property. For unmanaged nodes, this property is mandatory; it must be included in the node description and its value set to "false":  
`"IsManaged": false`
- Because the Fabric Services System does not manage the Inter-Switch Links (ISLs) between nodes, it is not necessary to include ISL data in the manual topology file; only node data is mandatory. If link data is not present in the topology file, then no links are displayed in the topology view when creating the fabric intent; the nodes will appear disconnected. This will not affect the Fabric Services System's ability to work with the fabric, since the links exist and are managed by an external process. If link data is present in the topology file, then the links will display within the Fabric Services System normally when you create the fabric intent. However, this link data is not refreshed within the Fabric Services System if it is altered on the node.

### Node configuration data stored within the Fabric Services System

Although the Fabric Services System does not directly manage the nodes within the fabric, it does store the following sets of node configuration data that are required to support its capabilities:

- the management IP address of each unmanaged node in the fabric, stored as part of the fabric's node inventory. This information, along with the necessary certificate data, is enough for the Fabric Services System to discover an unmanaged node, establish a gNMI connection, learn the full node configuration details, and consider the node to be in a Ready state.



**Note:** For unmanaged nodes, the management IP address is mandatory but the serial number is optional. For more typical, managed nodes, the serial number is mandatory.

- the complete set of configuration data for each node, stored as a system-generated Global Configuration Override (GCO). Typically GCOs are used to store expected variations in a node's configuration. But for unmanaged nodes, the GCO is used to store the entire node configuration, where it is available for consultation by components of the Fabric Services System. For example, it is from this configuration data that the Fabric Services System creates initial configuration files as part of a maintenance intent.



**Note:** Any system-generated GCO is called a "system GCO", to distinguish it from those created manually by an operator.

- when it is required by a maintenance intent, a separate set of basic, initial configuration data parsed from the full configuration of each node. This configuration data consists of the network instance, management interface, and system information, and is required to use the provisioning processes that are part of maintenance intent deployment.

## Deployment of fabric intents with unmanaged nodes

The Fabric Services System does not alter any fabric configuration data for unmanaged nodes. Nevertheless, after loading the manual topology of an unmanaged fabric, it is necessary to "deploy" the resulting fabric intent. Although no configuration data is sent to the node during deployment to an unmanaged node, the act of "deploying" the fabric intent satisfies certain internal requirements of the Fabric Services System.

For the deployment of such a fabric intent to proceed, two conditions must be met:

- all participating nodes must be in a Ready state
- none of the system GCOs that represent the nodes within the fabric can be empty

If either of these conditions is not met, the attempt to deploy the fabric intent fails and results in an error message in the system log. Deployment completes for those nodes that satisfy the conditions, but are suspended for those that do not. Once the conditions are satisfied for any of the remaining nodes, the deployment on that node automatically resumes.



**Note:** An empty system GCO for an unmanaged node usually indicates some kind of delay in obtaining the configuration data from the node. This problem typically resolves itself when regular communication with the node is established.



**Note:** This check to ensure the system GCO is not empty is performed when creating the first version of a fabric intent that uses unmanaged nodes. However, any subsequent deployments will not repeat this check. For this reason, even though it is possible to manually delete any system GCO, it is important not to delete the system GCO corresponding to any deployed, unmanaged node. If you do inadvertently delete such a system GCO, replace that data with an equivalent user-created GCO.

## Subsequent updates to fabric configuration

If the external process that is managing the "unmanaged" nodes makes any change to their configuration, the Fabric Services System detects these changes as deviations. The system automatically accepts these deviations and incorporates them into the stored system GCO. However, because the Fabric Services System does not deploy fabric configurations to unmanaged nodes, the system does not deploy any updated version of the fabric intent after absorbing these deviations in to the system GCO.

## Workload VPN intents

Workloads VPN intents on fabrics consisting of unmanaged nodes are created, deployed, and managed exactly like workload VPN intents on conventional fabrics.

The Fabric Services System does not discover pre-existing workloads on unmanaged nodes. However the configuration is accepted as part of system GCO.

New workloads created with the Fabric Services System can be deployed to unmanaged fabrics as long as no conflicting configuration is present in the fabric from pre-existing workloads. If such a conflicting configuration is present, delete the pre-existing workload configuration on the node before deploying the new workload from the Fabric Services System.

## Maintenance intents

Maintenance intents allow you to manage the replacement of nodes within a fabric, and the upgrading or downgrading of SR Linux software on existing nodes.

Maintenance intents for fabrics consisting of unmanaged nodes are created, deployed, and managed exactly like maintenance intents on conventional fabrics.

However, assigning a serial number to an unmanaged node, which is a requirement for maintenance intents to function, results in an entry for each of the affected nodes in the Fabric Services System's own, internal DHCP server. This is necessary for the function of the SR Linux Zero Touch Provisioning (ZTP) capability that is used by maintenance intents.

Any maintenance intent also requires an initial configuration file for each node affected by the intent. The Fabric Services System creates this initial configuration file by reading directly from the system GCO that contains the node's configuration immediately before the creation of the maintenance intent. The following node data is extracted from the system GCO to create the initial configuration file, which is required for the processes that deploy the maintenance intent:

- network instance management information
- interface management information
- system information

After deployment, this initial configuration data is sufficient to allow the existing, external process to resume management of the node after the maintenance intent has completed successfully.

### Alterations to workflows when using unmanaged nodes

For the most part, other than the management of the underlay fabric itself, operations within the Fabric Services System are unchanged when working with a fabric consisting of unmanaged nodes. Workload VPN intents and maintenance intents are managed almost identically, and other features for monitoring the fabric behave just as they do for fabrics consisting of nodes managed by the Fabric Services System.

However, there are a few exceptions to standard settings or procedures when working with a fabric of unmanaged nodes.

- When configuring a Management IP Pool for use with unmanaged nodes, any CIDR block within that pool must have the **Is Managed** property disabled. This is a unique requirement for unmanaged nodes.
- Domain names are typically set once for a fabric, and so the same domain name is configured on all of the nodes within the fabric. However, for unmanaged nodes, the Fabric Services System reads the domain name individually from each node configuration. As a result, for unmanaged nodes, domain names may vary among nodes within a fabric.
- Global Configuration Overrides (GCOs) can only add new configurations that are not already present on the node (and reflected in the corresponding system GCO). Any modification or deletion of existing configurations contained within this system GCO is not supported.
- When configuring a mirror destination for traffic mirroring, you must manually provide a **Source IP** address if the source is an unmanaged node.
- For unmanaged nodes, the Operational Deviation view of the Operational Health and Insights page does not display deviations pertaining to Inter-Switch Links (ISLs). The tracking of deviations on ISLs is exclusively supported for managed nodes.
- It is the operator's responsibility to resolve any conflict between an already-created or discovered workload in the system GCO, and the workload created by the system.

#### Related topics

[Manual fabric topologies](#)

[Creating a fabric intent for unmanaged nodes](#)

## 5.8.1 Creating a fabric intent for unmanaged nodes

### Prerequisites

Ensure that:

- the participating nodes and their links are already configured as a functioning fabric
- an external DHCP server is in place
- a manual topology file has been created describing the intended fabric



**Note:** It is important that this topology file is an accurate representation of the unmanaged fabric. It is your responsibility to ensure that the information in the topology file is complete and accurate.



**Note:** The "isManaged" flag must be disabled for all nodes in the topology file. The Fabric Services System does not support a fabric that contains a mixture of managed and unmanaged nodes.

- an association file has been created containing the serial number and management IP address for each node



**Note:** The management IP address is mandatory; the serial number is optional but recommended. For any node to be the subject of a future maintenance intent, the serial number is mandatory.

### About this task

Follow this procedure to create and deploy a fabric intent consisting of unmanaged nodes. Most steps refer you to existing procedures. Some steps include special requirements when performing those other procedures.

### Procedure

**Step 1.** Use the **Region Selector** at the top of the page to select the region in which to create the fabric intent.

**Step 2.** Import the manual topology file that describes the existing fabric into the Fabric Services System so that it is available for use when you create the fabric intent.  
See [Importing a manual topology](#).

**Step 3.** Ensure that certificates are in place for the participating nodes.

**Step 4.** Create a Management IP pool that contains one or more CIDR blocks intended for use by unmanaged nodes.



**Note:** When creating any CIDR block within this pool, you must disable the **Is Managed** property. This is a unique requirement for management IP pools intended for use with unmanaged nodes.

See [Creating IP and Autonomous System pools](#).

**Step 5.** Create a fabric intent that uses a manual topology, including saving the fabric intent and generating the fabric topology. This adds the set of nodes within the topology to the Fabric Services System inventory.



**Note:** For **Imported Topology From File**, select the manual fabric topology you previously uploaded.



**Note:** For the **Management Pool Name**, select the Management IP Pool you configured containing a CIDR block whose **Is Managed** property is disabled.



**Note:** After you generate the fabric, the topology display in the Fabric Design view of the fabric intents page will not display any links between the nodes of an unmanaged fabric if Inter-Switch Link (ISL) data was not included in the manual topology data.

See [Creating a fabric intent using a manual topology](#).

**Step 6.** Associate each planned node in the fabric with a node contained in the Fabric Services System inventory by assigning it a management IP address (mandatory) and a serial number (optional, but recommended). Typically, this association is achieved by importing a file with the association data that was prepared ahead of time.



**Note:** An unmanaged node must be running a supported version of SR Linux to be discovered and become Ready during this step.

See [Uploading association data to the fabric intent inventory](#)



**Note:** Immediately after you complete this step and the node enters a Ready state, the following occurs:

- A set of system GCOs is created: one for each node that is in a Ready state, containing the complete, discovered configuration for that node.
- If the region containing this fabric intent is set to use the Fabric Services System's internal DHCP server, a set of entries is created in the internal DHCP list for the region, one for each Ready node. These are required if you in the future create a maintenance intent for an unmanaged node.

**Step 7.** If the node used a Deployment Trigger of greater than 0%, deploy the fabric intent.

See [Deploy the fabric intent to hardware](#)



**Note:** If the fabric is configured with a 0% deployment trigger, deployment proceeds automatically.



**Note:** This deployment action does not send configuration data to the nodes participating in the fabric. This is purely an internal process that satisfies requirements within the Fabric Services System.

## Expected outcome

Upon completion of this procedure, the fabric consisting of unmanaged nodes is known to the Fabric Services System. Other capabilities of the Fabric Services System are now active with respect this fabric, including alarms, operational health displays, and so on.

## 5.9 Viewing a fabric intent

### About this task

Follow this procedure to view a previously created fabric intent.

### Procedure

- Step 1.** Click the ☰ menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the ⋮ icon at the right edge of that row, and select **Open** from the displayed action list.

### 5.9.1 Groups

Groups represent multiple nodes in your fabric's topology. When a fabric consists of many nodes, the topology can grow complicated. Groups simplify the display of such complicated structures to make them more comprehensible at a glance.

The maximum number of nodes represented by a single group is a variable you can set with the Clustering Controls display setting (as described in [Table: Display controls](#)). The system automatically adds more groups to the display to represent additional nodes beyond this threshold.

You can click the + control for any group to expand it; the node is replaced by the set of individual nodes it represented.

Figure 18: Node group

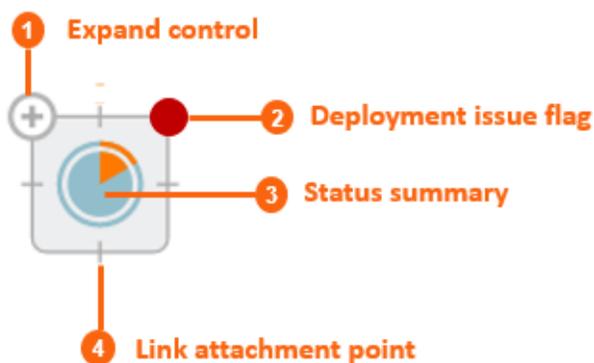


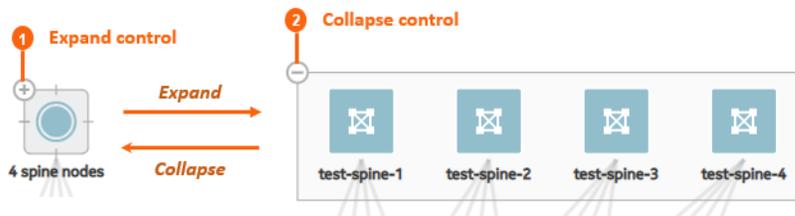
Table 24: Group elements

	Element	Description
1	Expand control	Click this control to expand the group and display its individual nodes.
2	Deployment issue flag	This displays only if there are issues associated with the deployment of configuration data to one or more nodes in the group.
3	Status summary	This is a pie chart that indicates the proportion of nodes in the group that are the subject of alerts, versus those that are not. Errors are represented by the orange segment.
4	Link attachment point	Links to other nodes originate from one of these points at the edge of the group icon.

To expand a group, click the **+** control at the top-left corner of the group.

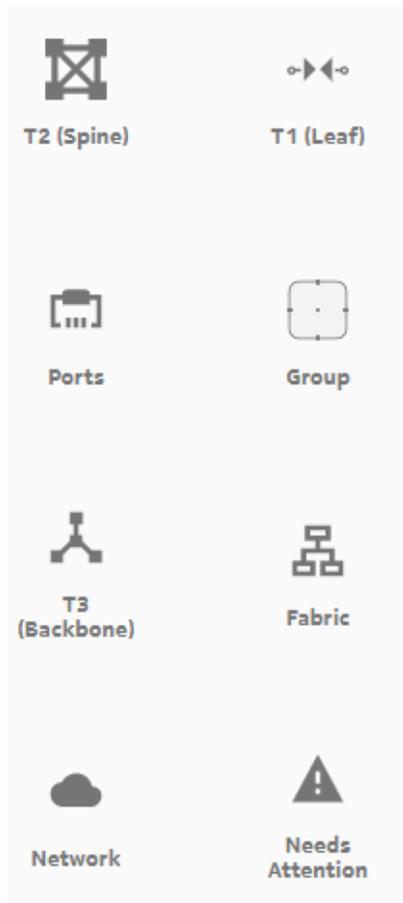
To collapse an expanded group, click the **—** control in the same position.

Figure 19: Expanded group



The symbol for each node indicates its role as spine, leaf, or backbone as shown in [Figure 20: Icons for fabric intents](#).

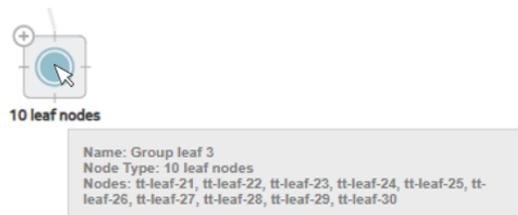
Figure 20: Icons for fabric intents



## 5.9.2 Information displays

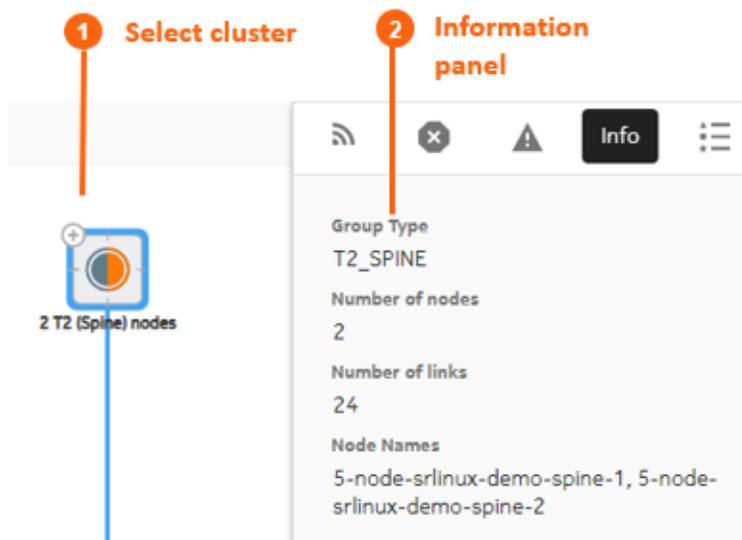
You can hover the cursor over a node or group in the fabric intent Design view to see details about that object.

Figure 21: Hovering over a group



You can also expand the information panel on the right side of the page by clicking ⓘ. The information panel shows more details about the currently selected object in the fabric topology (either a group, a node, or a link).

Figure 22: Information panel for a group

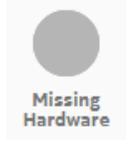
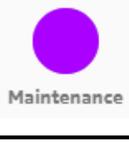


### 5.9.3 Error indicators

The Fabric Services System UI provides a number of cues to draw your attention to problem areas in your fabric intent. Highlights and symbols in the topology display identify groups and nodes that are the subject of one or more errors.

Table 25: Node shading

Element	Description
	Operational Normal: this node or group is in good order and in an operational state.
	Intent Matching: this node or group is configured as described in the original fabric intent.

Element	Description
	Intent Deviation: this node or group has been configured differently than specified in the original fabric intent. This arises when someone outside of the Fabric Services System alters the node configuration after the deployment of your fabric intent.
	Operational Issue: this node or group is not operational or its operation is impaired. Check the Alerts list to identify the operational issue that is affecting the object.
	Missing Hardware: this node or group is identified as "real" in the fabric intent, but no corresponding real-world hardware has been associated with it.
	Maintenance: this node or group is subject to a Maintenance intent (either to replace a node with new matching hardware, or to update the node's software load).

The Status Summary at the bottom of the page shows general information about the fabric intent's progress and status.

Figure 23: Status Summary (expanded)



- Stage: The current fabric intent's progress through its development stages toward deployment, including:
  - Created: A user has selected a template, supplied basic parameters, and elected to proceed with fabric generation.
  - Topology Generated: the system has generated the fabric topology.
  - Configuration Generated: the system has generated the configuration files for individual nodes participating in the fabric, based on the current fabric intent design.
  - Deployment Done: Tracked separately for the Digital Sandbox and real hardware, this indicates that the necessary configurations have been applied to the nodes participating in the fabric.
- Status: These indicators show a count of nodes within the current fabric that are in various states, as described in [Table 25: Node shading](#).

## 5.9.4 Viewing the event log

### About this task

The Fabric Services System maintains a unique event log for every fabric intent, showing significant events in the history of the fabric intent including a time stamp and the outcome of each event.

Figure 24: Event Log for a fabric intent

Event Log			
Timestamp:	2021-03-22T18:53:39Z	State:	Created
Message:	Create Success		
Timestamp:	2021-03-22T18:53:41Z	State:	CableMapInProgress
Message:	Generating Cable Map		
Timestamp:	2021-03-22T18:53:45Z	State:	CableMapDone
Message:	Cable Map Generation Done		
Timestamp:	2021-03-22T18:53:45Z	State:	ConfigGenInProgress
Message:	Generating Config		
Timestamp:	2021-03-22T18:53:45Z	State:	ConfigGenFailed
Message:	ipam precheck fails: Insufficient ISL IP addresses for Links. Request : 288, Total : 128, Available : 62, Unavailable : 2.		

**CLOSE**

To view the event log for a fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 4.** From the opened fabric intent, click the  menu at the upper right of the page.
- Step 5.** Select **Event Log** from the action list.
- Step 6.** To exit the event log, click the **CLOSE** button.

## 5.9.5 Viewing a fabric intent as code

### About this task

You can view the configuration code that the system has generated to represent the current fabric. This can be helpful for verifying the design in detail, and possibly revising the fabric design if needed.

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
- Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** Click  to open the code view.
- Step 6.** Optional: Do any of the following:
- To save the fabric code in a local file, click  and choose a location and name for the file.
  - To copy a portion of the code, select the code and click . The system adds the selection to your clipboard.
  - To find a particular string of text within the fabric code, click  and enter the text string. The first instance is highlighted; use the arrows to navigate forward or backward to additional instances, or click **ALL** to highlight all instances simultaneously.
- Step 7.** To compare the current fabric code against the code for a previous version of the same fabric:
- a. At the end of the "breadcrumb" list at the upper left of the overlay, in the drop-down list, click **Compare Versions**.
  - b. Click the **Compare To** drop-down list and select a different version number of the current fabric from the list.
  - c. Click the **Expand <number> lines** link to view the full fabric code displayed in two panels, one for each version.
  - d. Optional: Repeat step 7.b to select another version of the same fabric against which to compare the current version's code.
- Step 8.** To view the code for an individual node within the fabric:
- a. At the end of the "breadcrumb" list at the upper left of the overlay, in the drop-down list, click **Fabric Elements**.
  - b. Click a node in the left column to see its current configuration code.
  - c. Click the **Expand <number> lines** link to view the full fabric code displayed in two panels, one for each version.
  - d. Optional: To see the normalized version of the same node's code, click the drop-down list at the upper right of the overlay and select **Normalized**.

- Step 9.** When finished viewing the fabric as code, click **X** at the upper right of the overlay to return to the **Fabric Design** view.

## 5.9.6 Viewing the configuration file for a single node

### About this task

You can view the configuration file that the system has generated for each node in the inventory. Viewing the configuration file can be helpful for verifying the precise configuration that is planned for the node and possibly revising the configuration if needed.

Follow this procedure to view the current configuration planned for a single node.

### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Find the node and click  at the end of its row.
- Step 4.** Select **Inspect Configuration** from the displayed actions list.
- Step 5.** Optional: Do any of the following:
- To save the fabric configuration in a local file, click  and choose a location and name for the file.
  - To copy a portion of the fabric configuration, select the portion and click . The selection is added to your clipboard.
  - To find a particular string of text within the fabric configuration, click  and enter the text string. The first instance is highlighted; use the arrows to navigate forward or backward to additional instances, or click ALL to highlight all instances simultaneously.
- Step 6.** Click **X** at the upper right of the overlay to close the **Inspect Configuration** overlay.

## 5.9.7 Downloading the initial node configuration

### About this task

You can download a file containing the initial configuration code for all of the nodes participating in the current fabric intent.

To download the configuration file:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Select **Fabric Intents**.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
- Double-click the row for that fabric intent.

- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** From the opened fabric intent, click the  menu at the upper right of the page.
- Step 6.** Select **Download Initial Node Configuration** from the list. The system immediately downloads the file "initialNodeConfigs" to your Downloads folder.
- Step 7.** To view the configuration, open the initialNodeConfigs file in a text editor.

## 5.9.8 Viewing the fabric inventory

### About this task

You can open a fabric intent's Fabric Inventory view to see a list of all of the nodes that are included in the fabric design. The Inventory view displays extensive details about each node, including its role in the current fabric intent.

This list is a subset of the full inventory; it shows only the nodes participating in the current fabric intent.

From this view you can edit labels for nodes, associate planned nodes with real hardware, inspect an individual node's configuration code, and download a mapping file to help with node association. For more information about the complete inventory and the full set of actions available from any inventory view, see [Inventories](#).

To open the Inventory view for a single fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
- Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** Click the **View** drop-down list and select **Fabric Inventory**.
- Step 6.** To return to the **Fabric Design** view or open another view, make the corresponding selection from the **View** drop-down list.

## 5.9.9 Viewing fabric links

### About this task

You can open a fabric intent's Fabric Links view to see a list of all of the links that are included in the fabric design. The Fabric Links view displays information about each node, including:

- Role
- Endpoint 1 Node, IP Address, and Port

- Endpoint 2 Node, IP Address, and Port
- Any associated labels

From this view you can edit labels on individual links.

To open the Fabric Links view:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** Click the **View** drop-down list and select **Fabric Links**.
- Step 6.** To return to the **Fabric Design** view or open another view, make the corresponding selection from the **View** drop-down list.

### Related topics

[Labels](#)

## 5.9.10 Viewing edge links

### About this task

You can open a fabric intent's Edge-Links view to see a list of all of the links and LAGs that are included in the fabric design.

To open the Edge-Links view:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** Click the **View** drop-down list and select **Edge-Links**.

### Expected outcome

The display switches to the **Edge-Links** view.

This view displays the available ports in the current fabric intent, as well as any LAGs that have been created within the current fabric intent.

- Step 6.** To view additional details about any edge link:

- a. Select a row and click the more (  ) icon.
- b. Select **Open** from the drop-down list.

**Step 7.** Click **Close** to return to the **Edge-Links** view.

To return to the **Fabric Design** view or open another view, make the corresponding selection from the **View** drop-down list.

## 5.9.11 Generating a wiring plan

### About this task

You can generate a wiring plan to assist technicians when installing and connecting the fabric's planned, supporting hardware.

The wiring plan is in the form of a .csv file that identifies all of the connected pairs of nodes participating in the fabric, their roles (for example leaf or spine), and the ports on each node that serve as endpoints for their interconnections.

To generate a wiring plan:

### Procedure

**Step 1.** Click the  menu.

**Step 2.** Select **Fabric Intents**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.

**Step 4.** To open a specific fabric intent from the list, do one of the following:

- Double-click the row for that fabric intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

**Step 5.** Click the **More actions** icon (  ) to open the actions menu and select **Generate Wiring Plan...**

### Expected outcome

The system generates and saves a file with the name <fabric name>\_wiringPlan.csv.

## 5.10 LAG management

You can create link aggregation groups (LAGs) from the **Edge Links** view. The system supports the creation of LAGs that include multiple ports on a single leaf node, and also multi-home LAGs (MH-LAGs) that include ports on multiple leaf nodes. For MH-LAGS, the ports should be located on leaf nodes within the same fabric.

### 5.10.1 Creating LAGs

#### About this task

You can designate multiple edge links on the nodes managed by the Fabric Services System to be aggregated into a LAG. The system supports the creation of both LACP LAGs and Static LAGs.

**Note:**

LAG configuration is not supported on the initial version of a fabric intent. To configure a LAG, you must first save and deploy Version 1.0 of the fabric intent, and then create a new version of the intent. You can configure LAGs for Version 2.0 and greater of a fabric intent.

**Procedure**

**Step 1.** Choose one of the following:

- If you are configuring a LAG for a fabric intent that is already deployed, begin by creating a new candidate version of the existing fabric intent. Then go to step 2.
- If you are configuring a LAG for Version 2.0 or greater of a fabric intent that has not yet been deployed, go to step 2.

**Step 2.** Click the ☰ menu.

**Step 3.** Select **Fabric Intents**.

**Step 4.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.

**Step 5.** To open a specific fabric intent from the list, do one of the following:

- Double-click the row for that fabric intent.
- Select a row, click the ⋮ icon at the right edge of that row, and select **Open** from the displayed action list.

**Step 6.** Click the **View** drop-down list and select **Edge-Links**.

**Step 7.** Click **+ CREATE A LAG** to begin creating a LAG from the available downlink interfaces.

If **+ CREATE A LAG** is disabled, make sure your fabric intent has not been deployed. If it is deployed, you must create a new candidate version to create LAGs.

**Step 8.** In the **LAG Details** view, define basic properties for the LAG:

- **Name**
- **LAG Type:** either LACP or Static.
- **MultiHoming:** enable or disable the ability to include ports from different nodes in the same LAG.
- If you enabled multihoming, select a **LAG mode**. This can be either All Active, Single Active, or Port Active:
  - **All Active:** the default selection, this option leaves all links participating in the LAG active.
  - **Single Active:** optionally enable **Single Active** to keep a single link in the LAG active while the other is in standby mode.



**Note:** In Single active mode, the physical interfaces within the same LAG all remain operationally up. However each sub-interface associated with a network-instance has its operational state up or down based on whether it is selected to be the active or standby sub-interface.

After you enable **Single Active**, the **Preferred Active** drop-down list displays. After you add edge link interfaces in the LAG as described in step 9, you can use this drop-down to select a participating link within the LAG to be active.

Enable the **Revertive** option if you want traffic that has been switched to the standby link to return to the preferred active link after the fault is resolved.

- **Port Active:** optionally enable **Port Active** to keep a single link active while the other is in standby mode. If **Single Active** is disabled, both links are active.



**Note:** In Port Active mode, the active and standby function is handled at the interface level. Standby interfaces are operationally down and only forward traffic to the active interface.

After you enable **Port Active**, the **Preferred Active** drop-down list displays. After you add edge link interfaces in the LAG as described in step 9, you can use this drop-down to select a participating link within the LAG to be active.

Enable the **Revertive** option if you want traffic that has been switched to the standby link to return to the preferred active link after the fault is resolved.

- **Lag Speed**



**Note:** The system supports the configuration of LAG members with speeds of 10M, 100M, 1G, 10G, 25G, 40G, 50G, 100G, or 400G. Note that the 50G speed is supported only for IXR-D3 nodes.



**Note:** The system does not validate that the ports you select as part of the LAG can support the speed you select here. If the configuration is invalid, the system notifies you (and informs you of the reason for failure) when you try to deploy the fabric intent.

- Optionally configure Link Aggregation Control Protocol (LACP) settings:
  - **LACP Interval:** choose either Fast or Slow LACP internal modes.
  - **LACP Fallback:** enable to LACP Fallback to allow one or more designated links of an LACP-controlled LAG to go into forwarding mode if LACP is not operational after a configured timeout period.  
When enabled, the additional fields below are available.
  - **LACP Fallback Timeout:** Specify a timeout value in seconds after which the forwarding mode will be triggered.
  - **LACP Fallback Preferred Interface:** Select one interface within the LAG to become active if LACP fallback is triggered.



**Note:** You must add a set of edge links to this LAG before you can select one of them in this field.

**Step 9.** Add the edge link interfaces to constitute the LAG:

- Click + **ADD EDGE LINK INTERFACES**.
- Select any number of edge link interfaces by checking the box at the left edge of the downlink interface's row.



**Note:** If you did not enable MultiHoming, all of the interfaces you select must be located on the same node.

- When you have selected the participating edge link interfaces, click **ADD**.

**Expected outcome**

The system returns you to the **LAG Details** overlay, and the downlink interfaces you selected are now displayed in the **Edge Link Interfaces** panel.

**Step 10.** Click **SAVE**.



**Note:** You do not need to configure anything in the **Local LAG IDs** panel. That list updates automatically to show previously configured LAGs.

**Expected outcome**

The system returns you to the **Edge Links** view, and the LAG you created is now displayed in the list as an additional edge link.

**Step 11.** You can return to the **Fabric Design** view or open another view by making the corresponding selection from the **View** drop-down list.

**Related topics**

[Creating a new version of a fabric intent](#)

## 5.11 Configuring a breakout port

**About this task**

On 7220 IXR-D3 and D3L systems, the QSFP28 connector ports (ports 1/3-1/33) can operate in breakout mode. Four breakout ports, each operating at 10G or 25G, can be configured on each QSFP28 connector port operating in breakout mode.

In the Fabric Services System, you can enable breakout mode for supporting ports as part of Edge Link Interface configuration for a fabric intent.

When a port is configured in breakout mode, the original port itself can no longer be selected as a sub-interface for a workload VPN intent, nor can it be made part of a LAG. However, each of the broken-out sub-ports derived from it can:

- participate in a LAG; however, a LAG cannot include a mixture of regular ports and broken-out sub-ports. All of the ports in a LAG must be of one type of the other.
- be selected as a sub-interface for a workload VPN intent.

In lists of ports in the UI, the system displays the broken-out ports as a group of four sub-ports; for example, port 1/3 in breakout mode displays as ports 1/3/1 through 1/3/4.

You can un-configure breakout mode for a port, but only if none of the broken-out ports are currently members of a LAG or functioning as sub-interfaces.

**Note:**

Breakout port configuration is not supported on the initial version of a fabric intent. To configure a breakout port, you must first save and deploy Version 1.0 of the fabric intent, and then create a new version of the intent. You can configure breakout ports for Version 2.0 and greater of a fabric intent.

To configure a port in breakout mode, do the following:

**Procedure**

**Step 1.** Choose one of the following:

- If you are configuring ports for a fabric intent that is already deployed, begin by creating a new candidate version of the existing fabric intent. Then go to step 2.
- If you are configuring breakout ports for Version 2.0 or greater of a fabric intent that has not yet been deployed, go to step 2.

**Step 2.** Click the  menu.

**Step 3.** Select **Fabric Intents**.

**Step 4.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.

**Step 5.** To open a specific fabric intent from the list, do one of the following:

- Double-click the row for that fabric intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

**Step 6.** From the **View** drop-down list, select **Edge-Links**.

**Step 7.** In the list of edge links, select a port by checking the box at the left edge of its row.

**Step 8.** At the right edge of the row for the selected port, click the **More actions** icon () and select **Open**.

**Step 9.** Click the **Breakout Mode** toggle to enable Breakout Mode for the port.

#### **Expected outcome**

The UI displays the following new properties for the port:

- Num Channels (number of channels): set to 4.
- Channel Speed: use the dropdown to select either 10 or 25.

**Step 10.** Click **SAVE**.

#### **Related topics**

[Creating a new version of a fabric intent](#)

## 5.12 Fabric intent modification

The Fabric Services System provides a number of methods to alter a fabric intent. Options vary depending on whether a fabric intent has been saved, generated, or deployed.

The current configuration of a node based directly on the latest version of a fabric intent, and excluding any changes from configuration overrides, is referred to by the Fabric Services System as the *normalized* configuration for that node.

### **Locks to prevent concurrent changes to fabric intent data**

In addition to the direct modification of a fabric intent by Fabric Services System user, some other activities can require an update to a fabric's configuration. For example, if you configure traffic mirroring, the system must modify the affected nodes' normalized configurations to incorporate the mirroring data.

To avoid conflicts that could result from the concurrent modification of node configuration data, the Fabric Services System locks the fabric intent when generating a new configuration based on inputs from any source. This lock is released only when the changes underway are used to generate a new configuration,

and is deployed. Only then can the competing, concurrent configuration changes be applied to further modify the normalized configuration.

So, for example, an attempt to apply configuration changes that incorporate traffic mirroring data is prevented until any changes already underway from direct modification of the general fabric intent have been applied. That is, the new configuration must have been generated and the new version of the fabric intent deployed.

Similarly, if the system is in the process of updating the normalized configuration based on traffic mirroring configuration, modifications to the fabric intents using the Fabric Intents form are prevented until the mirroring configuration has been generated and that new version of the fabric intent deployed.

### Related topics

[Configuration overrides](#)

## 5.12.1 Editing a fabric intent

### About this task

You can edit the parameters for a fabric intent that you have previously saved or generated, but not yet deployed. Editing in this way does not create a new version of the fabric intent; the version number is not set, and changes do not increment the version number until you deploy the current fabric intent.

When editing a fabric intent, you cannot change the template that the fabric intent design is based on, but you can alter the description and topology-related values such as the number of leaves and spines. When you begin the edit operation, the system makes clear which fields can be altered and which cannot.

When you deploy a fabric intent, you can no longer edit its parameters. You must create a new version of a deployed fabric intent at which point you can save, regenerate, and possibly redeploy the new version.

To edit a saved fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** Edit parameters in the left panel as required.
- Step 6.** When you are finished editing parameters, click  to save the new configuration.

## 5.12.2 Discarding changes to a fabric intent

### Prerequisites

After you create a new version of a fabric intent, but before you deploy it, you can discard the changes you have made. This deletes the new version of the fabric intent entirely, and reverts the display show to the previously deployed version.

If you discard changes for the initial version of a fabric intent that was never deployed, it deletes the fabric intent entirely.

To discard changes for any version of a fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** To open a specific fabric intent from the list, do one of the following:
  - Double-click the row for that fabric intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- Step 5.** From the opened fabric intent, click the **More actions** icon (  ) at the upper right of the page.
- Step 6.** Select **Discard Changes**.
- Step 7.** Click **OK**.

### Expected outcome

The system deletes the current, in-progress candidate version of the fabric intent, and the display reverts to show the last-deployed version.

## 5.12.3 Creating a new version of a fabric intent

### About this task

After a fabric intent has been deployed, you can still revise its design. This creates a new, incremented version of the previously deployed fabric intent, which you can alter without affecting the currently deployed version. When you are ready, you can deploy the new version of the fabric intent to replace the previous version.

Some actions you perform in the system may require a change to the configurations of one or more fabrics. When this happens, the system may automatically create, save, and generate a new version of the affected fabric intents for you. You can then deploy the new version to apply the necessary configuration changes to the affected nodes.

A new version of a fabric intent keeps the same fabric intent name, but the system assigns an incremented version number, date, and (if necessary) a new user association.

Creating a new fabric intent is also a prerequisite for some operations that are not supported on the initial version of a fabric intent. Currently, the operations that are supported only on Version 2.0 or greater of a fabric intent are:

- configuring LAGs
- configuring breakout ports

To create a new version of an existing fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.

**Step 4.** To open a specific fabric intent from the list, do one of the following:

- Double-click the row for that fabric intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

**Step 5.** Click the  menu at the upper right of the page.

**Step 6.** Select **Create Candidate Version**.

#### **Expected outcome**

The system saves a new version of the current fabric intent.

You can now edit the fabric intent's parameters and save the result without affecting the original version.

**Step 7.** Click **GENERATE FABRIC** to generate a new fabric diagram.

You can now view the fabric intent, and eventually add it to the deployment pipeline and deploy it as you would for a new fabric intent.

At any time before deploying the new version, you can discard this candidate version and revert to the previously deployed version by clicking on the **More actions** icon (  ) in the Fabric Design view and selecting **Discard Changes**.

#### **Related topics**

[Configuring a breakout port](#)

[Creating LAGs](#)

## 5.12.4 Duplicating a fabric intent

### **About this task**

In addition to saving new, incremental versions of an existing fabric intent, you can save a completely independent copy of an existing intent. This new duplicate has its own version history that you can continue to develop independently of the original.

To duplicate a fabric intent:

### **Procedure**

**Step 1.** Click the  menu.

**Step 2.** Select **Fabric Intents**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.

**Step 4.** Find the fabric intent in the displayed list.

**Step 5.** Click the **More actions** icon (  ) at the right end of the row.

**Step 6.** Select **Duplicate** from the action list.

### **Expected outcome**

The system creates a new, independent copy (not a new version) of the selected fabric intent.

## 5.13 Deleting a fabric intent

### About this task

You can delete a fabric intent. This removes the fabric intent and all of its versions from the Fabric Services System database.

The system does not delete the management IP on real nodes when you delete a fabric intent.

To delete a copy of a fabric intent:

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** Find the fabric intent in the displayed list.
- Step 5.** Click the **More actions** icon (  ) at the right end of the row.
- Step 6.** Select **Delete...** from the action list and click **OK** in the confirmation form.

## 5.14 Fabric intent deployment

Deploying a fabric intent creates a functioning instance of the fabric on any associated, real hardware.

Before you can deploy your fabric, you must have saved the fabric intent and generated its configuration.

Deploying the fabric involves two procedures:

- [From the fabric intent page, add the fabric intent to the region's deployment pipeline.](#)
- [From the deployment pipeline, select the fabric intent and select Deploy from the actions menu.](#)

### 5.14.1 Adding a fabric intent to the deployment pipeline

#### Procedure

- Step 1.** Open the list of fabric intents.
- Step 2.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 3.** Click .
- Step 4.** Click the **ADD TO PIPELINE** button.

#### Expected outcome

The system adds the fabric intent to the deployment queue for the region, and updates the status of the fabric intent to Queued for Deployment.

#### Expected outcome

After you add a fabric intent to the deployment pipeline, it remains in the pipeline until you tell the system to proceed with the deployment. This can be useful if the hardware for the fabric is not yet in place; you

can "park" the fabric intent in the deployment queue until the hardware is present, and only then trigger deployment.

### What to do next

When you are ready to proceed with deployment, go to [Deploying a fabric intent from the deployment pipeline](#).

#### Related topics

[Deploying a fabric intent from the deployment pipeline](#)

[The Deployment Trigger Percentage setting](#)

## 5.14.2 Deploying a fabric intent from the deployment pipeline

### About this task

When you are ready to deploy the fabric intent to hardware, do the following:

#### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the fabric intent.
- Step 4.** Find the row corresponding to the fabric intent you want to deploy.
- Step 5.** Click the  icon at the right edge of the row and select **Deployment Pipeline** from the actions list.
- Step 6.** Find your fabric intent in the deployment pipeline list.
- Step 7.** At the right edge of the row click the **More actions** icon ().
- Step 8.** Select **Deploy** from the actions list.

### Expected outcome

When deployment is complete, the fabric's status advances to Deployed.

The timing of the deployment depends on the fabric intent's Deployment Trigger Percentage setting.



**Note:** If any issues arise during deployment, they appear as errors in the error panel at the right of the page. The system also highlights them as deployment issues in the status bar and by adding a red circle to the nodes and groups affected by the deployment error.



**Note:** If a node participating in the fabric is not Ready when deployment begins, deployment is delayed for that node.

- When the node becomes Ready (for example, when a node that has lost its gNMI connection restores that connection) deployment to that node resumes automatically provided it passes all node-level checks and intent-level checks. These checks are repeated every 60 seconds; so a node that fails any of these checks initially can pass them later, and deployment will resume for that node.

The node-level checks consist of the following:

- The node is in the Ready state

- The deployed fabric's version number is lower than version number of the candidate intent being deployed
- If the node is unmanaged, a system GCO exists to describe its configuration
- The node is running SR Linux

The intent-level checks consist of the following:

- The intent is in non-candidate mode, and is in either the "deploymentdone" or "deploymentfailed" state
- The intent is not the subject of any deviation, region lock, workload-related interface check failure, or maintenance lock

However, if the candidate intent is in a "stale" state (due to, for example, the creation of a mirror or GCO for which automatic deployment was not selected), automatic deployment will not proceed even if the target node adopts a Ready state. Instead, deployment of the intent must be performed manually.

- If multiple nodes are unready at deployment, and several become Ready at the same time, deployment automatically resumes for all of the newly Ready nodes as a group (but subject to the limitations described above).

### 5.14.3 The Deployment Trigger Percentage setting

When you create a fabric intent, you select a Deployment Trigger Percentage value. This value represents the percentage of nodes participating in the fabric intent that must be in a Ready state before the system begins deploying the fabric intent.

The Fabric Services System currently supports two values for the Deployment Trigger Percentage: 0% and 100%:

- All Real fabrics use the 0% value, and this cannot be changed.
- All Digital Sandbox fabric intents use the 100% value, and this cannot be changed.

Depending upon the fabric's Deployment Trigger Percentage value, the system does one of the following after you select **Deploy** in the **Deployment Pipeline** action menu:

- If the value is set to 0%, the system immediately begins deploying the configuration to any available Ready nodes.
  - The system deploys the necessary configuration to each Ready node. The system waits until the configuration is complete on one node before proceeding to the next node.
  - When the system runs out of Ready nodes to configure, including if there are no Ready nodes to begin with, it suspends the deployment process. The state of the fabric intent advances to Deployment Done.
  - Deployment of the fabric intent does not resume, even if additional nodes reach a Ready state, until you:
    - open the Deployment Pipeline
    - select the fabric intent
    - select **Deploy**

At that point the system resumes deployment until it again runs out of Ready nodes, or the entire fabric intent is deployed.

- If any node configuration fails, the system rolls back the entire fabric deployment. Each node that it reconfigured is restored to its previous state, as captured in the rollback save.
- if the value is set to 100%, the system waits until all nodes in the fabric intent are in a Ready state before beginning deployment.
  - After all nodes are Ready, the system deploys the necessary configuration to each node. The system waits until the configuration is complete on one node before proceeding to the next node.
  - The system saves a rollback configuration before deploying a configuration, in case it needs to roll back the deployment on the node.
  - If deployment of the entire fabric completes successfully, the system sets the State of the fabric to Deployment Done.
  - If any node configuration fails, the system rolls back the entire fabric deployment, one node at a time. Each node that it reconfigured is restored to its previous state, as captured in the rollback save.

## 5.15 Deviations

After you deploy a fabric intent, various circumstances may alter the node configurations from those expressed in your fabric intent's configuration files. The system continues to monitor all nodes in the fabric after deployment, and flags any configuration change from your last-deployed fabric design as a "deviation".

Deviations are highlighted in several places in the UI:

- A node whose configuration includes at least one deviation is highlighted in orange on the fabric diagram.
- A group that contains one or more nodes with deviations displays a status summary pie chart on the group icon; this chart is shaded orange in proportion to the number of nodes in the group that are subject to alerts, including deviations.
- The status summary at the bottom of the Fabric Design view shows the number of deviations detected throughout the fabric.
- Clicking the deviations display in the status summary opens the Alerts log, which lists individual deviations. Like any alert, you can double-click it to view additional details.

If you attempt to deploy a new version of a fabric intent when there are deviations in the currently-deployed version of that fabric, the system displays an error and prevents your deployment. The error message redirects you to the **Alerts** panel for fabric intent, where a list of deviations is displayed and you can accept or reject each deviation listed there.

Deviations that you accept are incorporated into the node configuration maintained by the Fabric Services System, and are automatically deployed to the node as a new version of the associated fabric intent.

Every deviation that you accept is also stored as a system-generated global configuration override. This ensures that the deviation-based configuration information is stored and managed in a manner consistent with other configuration exceptions that are created by the Fabric Services System user.

### Related topics

[Configuration overrides](#)

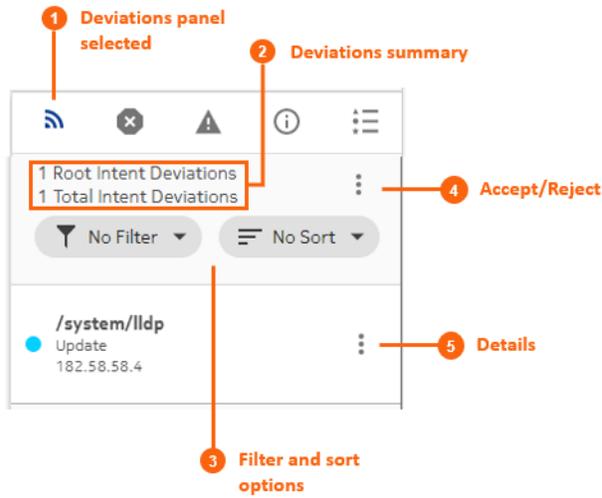
[Global configuration overrides](#)

## 5.15.1 Viewing deviations

### About this task

The Deviations (📶) panel displays a list of all variations between the fabric's last deployed configuration and its current configuration.

Figure 25: Deviations panel



For each item in the Deviations panel, you can view more information. You can also use the panel to either accept or reject deviations.

To view deviations using the Deviations panel:

### Procedure

**Step 1.** Click 📶 on the right side of the fabric intents page.

**Step 2.** Optional: In the **Filter** drop-down list, select a filter type:

- No Filter
- Delete
- Update

**Step 3.** Optional: In the **Sort** drop-down list, select one of the following sort methods:

- No Sort
- Ascending
- Descending

**Step 4.** Optional: Click the **More actions** icon (⋮) to the right of any deviation and click **Details** to view more information about that particular deviation.

### Related topics

[Accepting or rejecting deviations](#)

## 5.15.2 Accepting or rejecting deviations

### About this task

After the system notifies you of deviations that have been configured on the nodes within a fabric, you have the option to either accept or reject the deviations.

- If you accept the deviations, the system creates a new candidate version of the fabric intent that includes the affected node. The new candidate version includes an updated configuration that incorporates the deviation. The system automatically deploys the updated fabric intent.
- If you reject the deviations, the system creates a new candidate version of the previous fabric intent and automatically deploys it, overwriting the deviation and restoring each node to its pre-deviation configuration.

You can only accept or reject deviations while a fabric intent has been fully deployed (indicated by the Deployment Done state).

You can only accept or reject a set of deviations one time for a candidate version of a fabric intent. If additional deviations arise before you have deployed the current version, and you attempt to accept or reject any or all of them, the system displays an error message indicating that you cannot do so because the current candidate version of the fabric intent is not in a Deployment Done state.

After you accept or reject the deviations for a fabric intent, the system automatically deploys the resulting new configuration of the fabric intent.



**Note:** The system does not handle any new deviations while deployment is in progress for previous deviations.

To accept or reject deviations to a deployed fabric intent:

### Procedure

**Step 1.** Open a fabric intent.

**Step 2.** Click the Deviations icon (  ) to open the **Deviations** panel.

**Step 3.** Click the **More actions** icon (  ) to the upper right of the **Deviations** panel, and select **Accept/Reject** from the displayed menu. The **Accept/Reject Watches** overlay displays.

**Step 4.** In the displayed list of deviations, select one or more deviations by checking the box to the left of each row. Select the deviations that you want to either accept or reject as a whole.

Alternatively, you can click **Accept All** or **Reject All** to accept or reject the entire list of deviations.

**Step 5.** To accept the selected deviations:

- a. Click the **Accept** button at the bottom of the overlay. The system displays a confirmation form indicating that the selected items are accepted, and all other items in the list are rejected by implication.
- b. Click **OK** to confirm the action. The confirmation form closes.
- c. Click the **X** at the upper right of the overlay to return to the **Design** view. Note that the system creates a new candidate version of the fabric intent design.

**Step 6.** To reject the selected deviations:

- a. Click the **Reject** button at the bottom of the overlay. The system displays a confirmation form indicating that the selected items are rejected, and all other items in the list are accepted by implication.
- b. Click **OK** to confirm the action. The confirmation form closes.
- c. Click the **X** at the upper right of the overlay to return to the **Design** view. Note that the system creates a new candidate version of the fabric intent.

### **Expected outcome**

After you accept or reject the deviations for a fabric intent, the system automatically deploys the resulting new configuration of the fabric intent.

## 6 Workload VPN intents

A workload VPN intent defines a set of VPN resources that can be made available to serve a particular source of demand for a data center's traffic, processing, and storage capacity.

Just as with fabric intents, the Fabric Services System uses the concept of an "intent" to describe a set of configuration data that affects multiple nodes, and are deployed together.

When deploying a workload VPN intent:

- The system deploys all of the new configurations for participating nodes as a single transaction. This ensures that the entire deployment succeeds completely, or else fails completely and cleanly, leaving nodes in their pre-deployment states to facilitate another deployment attempt.
- The fabric intent remains in the same intent version.



**Note:** All workload VPN intents are specific to the region in which they are created. A workload VPN intent that is created within one region is not visible within, or available to, other regions.

### 6.1 Role of a workload VPN intent

A workload VPN intent represents a specific subset of fabric resources (one or more fabrics, subnets, and sub-interfaces). After it has been defined, a workload VPN intent can be allocated to a single source of demand upon the participating fabric or fabrics. For example, all of the traffic from a single customer, or tenant, can be directed to only those resources encompassed by a particular workload VPN intent.

The workload VPN intent is embodied in a set of node configuration files that, when the intent is deployed, cause the participating nodes to behave as described in the intent.

### 6.2 Elements of a workload VPN intent

A workload VPN intent must be associated with specific fabrics within a deployment region (or "region"). The region manages the deployment pipeline, which organizes the sequential deployment of configuration data to the various nodes participating in the intent.

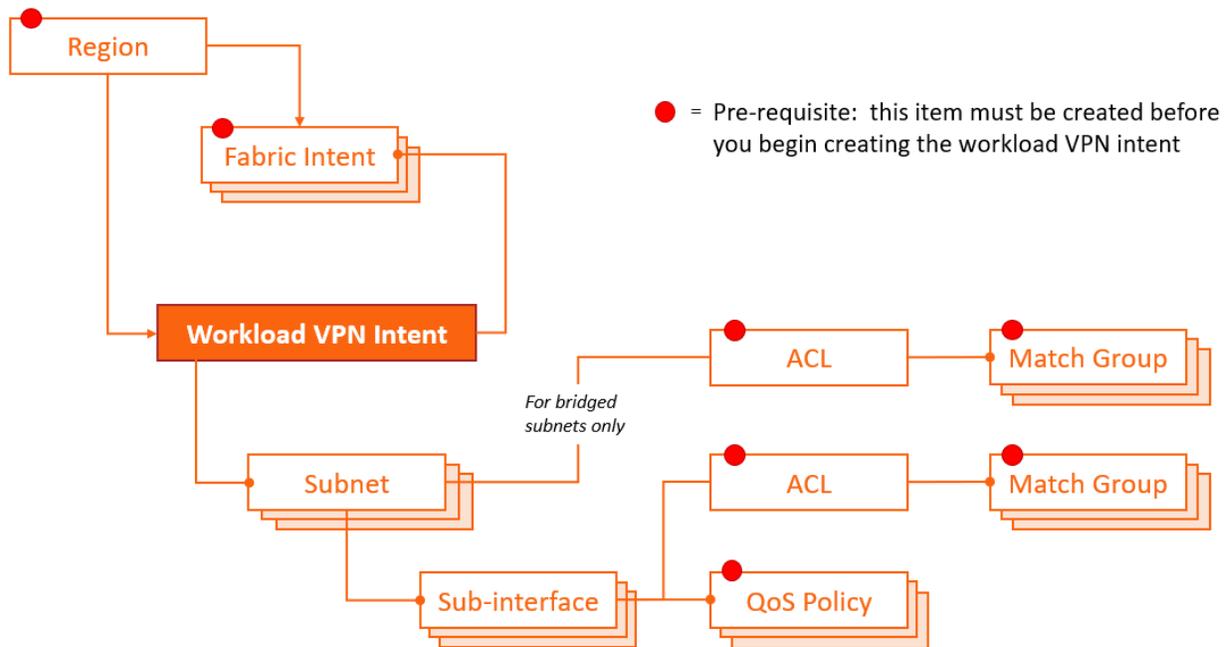
To represent the set of resources available to a workload, a workload VPN intent uses a hierarchical structure consisting of:

- **Fabrics:** a workload VPN intent can span one or more fabrics that you have already defined in the system.
- **Subnets:** each subnet defines a subset of the participating fabrics. Subnets can be either bridged or routed.

A bridged subnet with one or more IP addresses configured can be assigned an access control list (ACL), a set of rules that permit, or reject, any traffic that matches those IP addresses. These rules are applied on the integrated routing and bridging (IRB) interface. The set of IP addresses is stored as a match group; the ACL then assembles a set of match groups, adds further defining properties, and indicates whether to accept or reject the matching packets.

- Sub-interfaces: each sub-interface identifies a set of edge link ports and LAGs within a particular subnet, and across which the workload traffic can be bridged or routed.  
A sub-interface can be associated with:
  - Access Control Lists (ACL): each sub-interface can be assigned a set of rules that permit, or reject, any traffic that matches those IP addresses. The set of IP addresses is stored as a match group; the ACL then assembles a set of match groups, adds further defining properties, and indicates whether to accept or reject the matching packets.
  - QoS profiles: each sub-interface can be assigned a set of rules for prioritizing traffic.

Figure 26: Elements of a workload VPN intent



### Related topics

[Deployment regions](#)

## 6.2.1 Fabrics

Every workload VPN intent allocates resources that belong to one or more fabric intents that already exist in the system.

As part of workload VPN intent design, you must select the participating fabric intent or intents. All traffic managed by the workload VPN intent is confined to these participating fabric intents (and, more specifically, only to the subnets and sub-interfaces you select within those fabric intents).

A single fabric intent can support multiple workload VPN intents.

## 6.2.2 Subnets

A sub-network, or subnet, is a set of one or more nodes on a selected fabric.

### Types of subnets

When creating a workload VPN intent, you identify each subnet as either bridged, routed, or loopback.

- Bridged subnets can be assigned to a network-instance of type MAC-VRF. They are associated with a MAC-VRF network instance and allow for configuration of bridge table and VLAN ingress and egress mapping.

When adding a bridged subnet to a workload VPN intent, you associate an IP Anycast Gateway, and can associate an ACL.

With a gateway IP address, a MAC-VRF is attached to the default or ip-vrf network-instance by a single integrated routing and bridging (IRB) interface, which allows routing between different MAC-VRF instances of a particular tenant or group of servers.

Bidirectional forwarding detection (BFD) is supported on bridged subnets; a gateway IP address is required.

If enabled, MAC duplication detection monitors MAC addresses that move between sub-interfaces and between a sub-interface and an EVPN. A MAC address is considered a duplicate when the move count is greater than the number of moves within the configured monitoring window. Upon exceeding the specified number of moves, the system retains the prior local destination of the MAC and executes a specified action. You can enable monitoring for a subnet or for a sub-interface; for a sub-interface, MAC duplication detection must be enabled on the subnet to which the sub-interface belongs).

- Routed subnets can be assigned to a network-instance of type mgmt, default, or ip-vrf. They allow for configuration of IPv4 and IPv6 settings.
- Loopback subnets are assigned to loopback interfaces. Loopback interfaces are virtual interfaces that are always up, providing a stable source or destination from which packets can always be originated or received. When you create a loopback subnet, you must specify /32 (IPv4) or /128 (IPv6) host IP addresses to associate with a loopback interface.

### Enabling layer 2 proxy ARP

L2 proxy ARP is applicable to bridged subnets. When proxy ARP is enabled for a MAC-VRF, a table is created that contains entries related to the broadcast domain. The table can include entries that dynamically learned, EVPN-learned, or duplicate entries (if duplicate IP detection is enabled). Static entries are not supported.

You can modify the size of the proxy ARP table; by default, the table can have 250 entries.

### Enabling layer 3 proxy ARP and layer 3 proxy ARP ND

You can enable L3 proxy ARP for the following scenarios:

- a bridged subnet that is configured with a gateway IP address (IRB interface)
- routed subnets

### Related topics

[Adding subnets to the workload VPN intent](#)

### 6.2.3 Sub-interfaces

For each subnet, you identify the set of sub-interfaces over which tenant traffic can travel. These sub-interfaces can be physical or logical ports, including link aggregation groups (LAGs).

Each LAG you select as a sub-interface must have been configured previously within the system as part of the fabric intent design. The interfaces participating in a LAG can be located on a single node or on multiple nodes (constituting a multi-home LAG or MH-LAG).

For each sub-interface, you can define quality of service (QoS) profiles that prioritize traffic, and an access control list that identifies the packets to be accepted or rejected.

The Fabric Services system supports MAC duplication detection on sub-interfaces if MAC duplication detection is enabled on the subnet to which the sub-interface belongs.

#### Related topics

[Adding sub-interfaces to the workload VPN intent](#)

### 6.2.4 QoS profiles

The Fabric Services System supports quality of service (QoS) policies for assigning traffic to forwarding classes and remarking traffic at egress before it leaves a router. A QoS profile can be one of the following:

- QoS classifier profile  
A QoS classifier profile maps incoming packets to the appropriate forwarding classes.
- QoS rewrite-rule profile  
DSCP rewrite-rule policies mark outgoing packets with an appropriate DSCP value based on the forwarding class.

#### Related topics

[QoS profile management](#)

### 6.2.5 ACL profiles

An access control list (ACL) profile defines a set of packet types that should be either accepted or rejected.

An ACL profile is assembled from one or more match groups. Each match group defines a particular set of properties that could be possessed by packets. The ACL then uses these match groups to specify a set of packet properties, and then includes an instruction to either accept or reject packet that conform to those properties. Match groups thereby provide an easy way to define a set of IP addresses one time, and then re-use that information in multiple ACLs.

When assigned to a bridged subnet or a sub-interface in any type of subnet, the ACL defines the traffic that is permitted on those workload VPN intent resources. When an ACL is applied to a bridged subnet with a gateway configured, the IP filter is applied to the IRB interface.

#### Related topics

[Creating a match group](#)

[Creating an ACL profile](#)

[Adding subnets to the workload VPN intent](#)

[Adding sub-interfaces to the workload VPN intent](#)

## 6.2.6 Routers

A router is a logical object that represents a single EVPN VXLAN IP-VRF instance that can span multiple devices. Routers are responsible for routing between all subnets attached to the router. You can attach subnets and gateways within the subnets to routers. When you create a workload intent and assign IP gateway v4 and v6 addresses, a default router is created. You also can create additional routers objects.

By default, when you create a region, the system provides EVPN Instance (EVI) and Virtual Network Identifier (VNI) pools with a range of values from 1-65535. These range of values can be modified in the region properties of the Fabric Services System.

The VNI and EVI default values are derived from settings in the region properties. You can edit the VNI and specify whether the route targets are automatically derived or you can manually provide route targets.

When the route targets are automatically derived:

- You can set specific VNI and route targets per subnet or router object within a workload VPN intent. These settings are used when there are existing services in the data center (DC) on a DC-GW or other fabric where the existing services (network-instances) need to stretch to the new data center managed by the Fabric Services System. The route targets and VNI used for the network-instance must be specified.
- If you want the Fabric Services System to automatically derive a route target, but need to ensure that the values used do not overlap with existing services in the data center, you can set a specific pool of EVI or VNI from which the Fabric Services System allocates VNI and route targets for an ip-vrf or mac-vrf object within a workload intent.

### Related topics

[Adding subnets to the workload VPN intent](#)

[Creating a router](#)

## 6.3 Viewing a workload VPN intent

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** From the menu, select **Workload VPN Intents**.

### Expected outcome

The **Workload VPN Intents** page displays, showing a list of already-created workload VPN intents in the currently selected region.

**Step 3.** Use the **Region Selector** at the top of the page to select the region that includes the workload VPN intent you would like to view.

**Step 4.** To open a specific workload VPN intent from the list, do one of the following:

- Double-click the row for that workload VPN intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

### 6.3.1 Workload VPN intent view

Figure 27: Workload VPN intent Design view

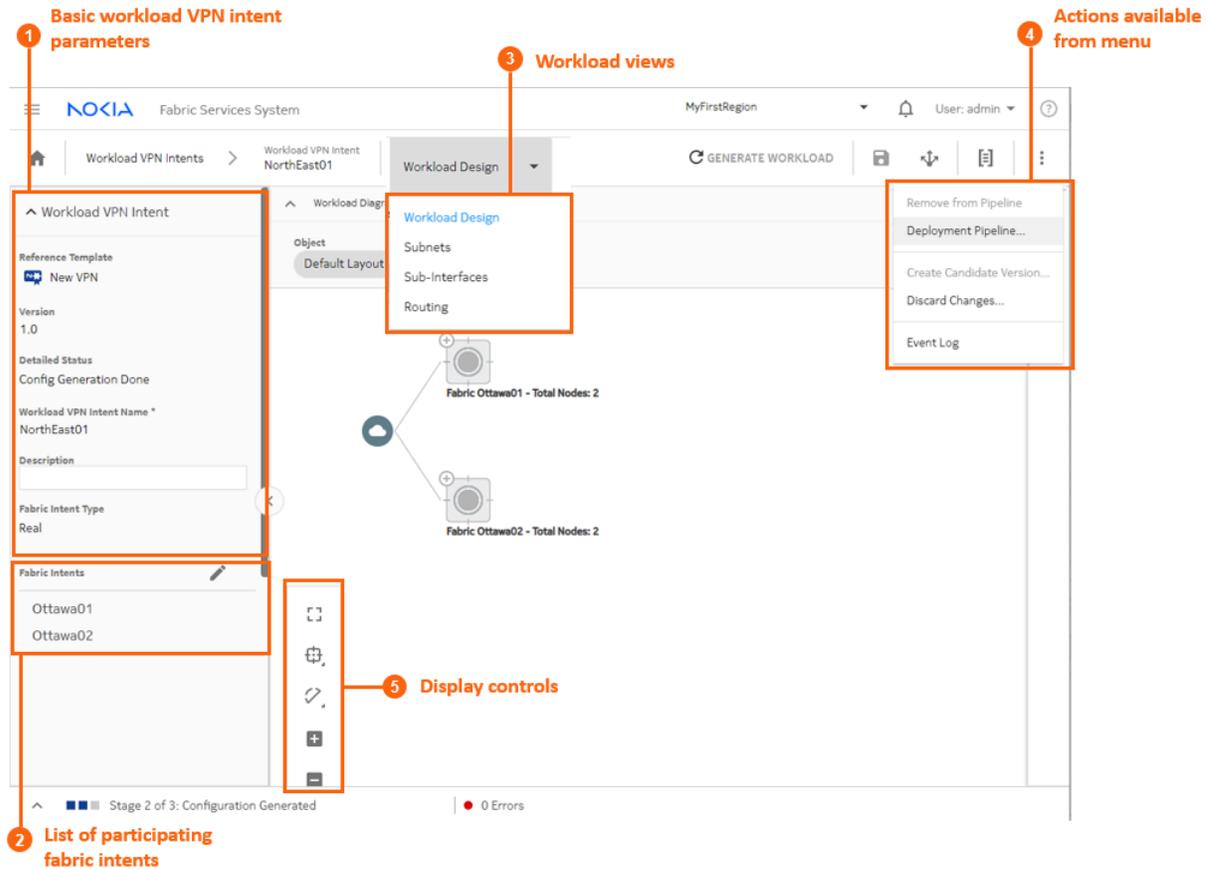
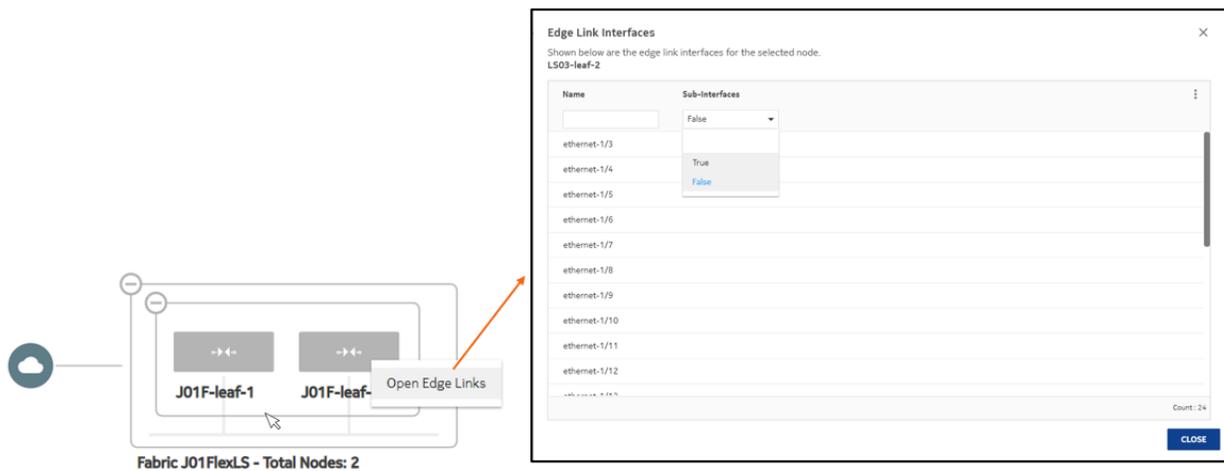


Table 26: Elements of workload VPN intent Design view

	Property	Description
1	Basic workload VPN intent parameters	Basic parameters include the VPN version number and state, as well as its name and a description.
2	List of participating fabric intents	This list shows the set of fabric intents over which the workload VPN intent can distribute traffic.
3	Workload views	The <b>View</b> drop-down list provides access to the different views necessary to see and design a workload VPN intent, its subnets, and its sub-interfaces. The <b>Routing</b> view allows you to configure BGP settings.

	Property	Description
4	Actions available from menu	From the actions menu you can manage deployment, create a new version of the intent, or view the intent's event log.
5	Display controls	These controls allow you to modify the way the system displays the workload VPN intent.

Figure 28: Workload VPN intent map objects



On the map display for workload VPN intents, you can expand backbone clusters to show the participating backbone nodes, and leaf clusters to show participating leaf nodes.

You can also right-click a leaf node and select **Open Edge Links** to open an overlay listing all of the edge links for the selected node.

## 6.3.2 Viewing a workload VPN intent as code

### About this task

You can view the detailed configuration code that the system has generated to represent the current workload VPN intent. This can be helpful for verifying the design in detail and possibly revising the workload VPN intent design if needed.

### Procedure

**Step 1.** Open a workload VPN intent.

**Step 2.** Click  to open the code view.

**Step 3.** Optional: Do any of the following:

- To save the workload VPN intent code in a local file, click  and choose a location and name for the file.

- To copy a portion of the code, select the code and click . The system adds the selection to your clipboard.
- To find a particular string of text within the workload VPN intent code, click  and enter the text string. The first instance is highlighted; use the arrows to navigate forward or backward to additional instances, or click **ALL** to highlight all instances simultaneously.

**Step 4.** Click  at the upper right of the overlay to return to the **Workload Design** view.

#### Related topics

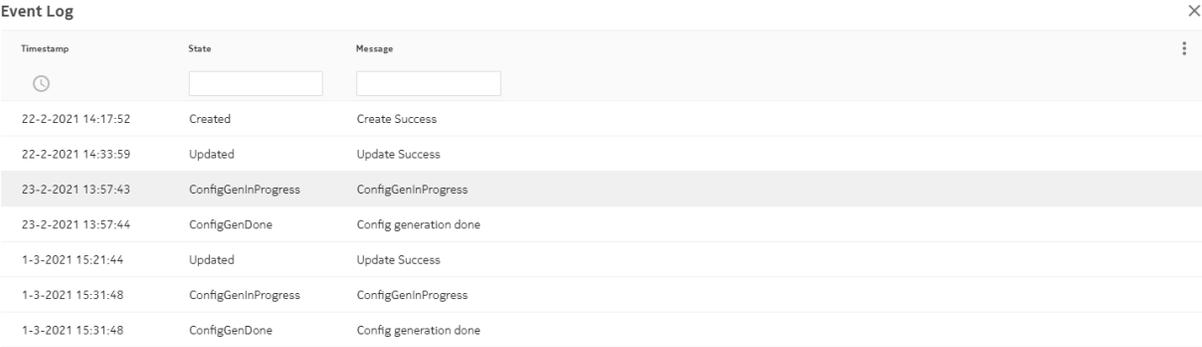
[Viewing a workload VPN intent](#)

### 6.3.3 Viewing the workload VPN intent event log

#### About this task

The event log retains a detailed history of all registered events that have occurred with respect to the current workload VPN intent. This can be useful when troubleshooting workload VPN intent issues, or just to verify that the workload history matches expectations.

*Figure 29: The workload VPN intent event log*



Timestamp	State	Message
	<input type="text"/>	<input type="text"/>
22-2-2021 14:17:52	Created	Create Success
22-2-2021 14:33:59	Updated	Update Success
23-2-2021 13:57:43	ConfigGenInProgress	ConfigGenInProgress
23-2-2021 13:57:44	ConfigGenDone	Config generation done
1-3-2021 15:21:44	Updated	Update Success
1-3-2021 15:31:48	ConfigGenInProgress	ConfigGenInProgress
1-3-2021 15:31:48	ConfigGenDone	Config generation done

Count: 7

**CLOSE**

The **More actions** () menu at the upper right provides access to the usual list management controls.

#### Procedure

- Step 1.** Open a workload VPN intent.
- Step 2.** Click the **More actions** icon () to open the actions list.
- Step 3.** Select **Event Log** from the drop-down list.
- Step 4.** Click  at the upper right of the overlay or the **CLOSE** button to return to the **Workload Design** view.

**Related topics**[Viewing a workload VPN intent](#)[Lists](#)

## 6.4 Profile manager

The Fabric Services System Profile Manager handles the life cycle of ACL and QoS profiles from creation to deployment.

At a high level, the steps for deploying a QoS or ACL profile is as follows:

1. Create a QoS or ACL profile.
2. Assign the QoS or ACL to the sub-interfaces or subnets (ACL profile only) in a workload VPN intent.
3. Deploy the Profile Manager.
 

You can either auto-deploy the Profile Manager or deploy it manually.

  - To auto-deploy the Profile Manager:
    - a. Add the workload VPN intent to the deployment pipeline
 

At this point Profile Manager is added to the deployment pipeline and is automatically deployed.
    - b. Deploy the workload VPN intent from the pipeline.
  - To deploy the Profile Manager manually:
    - a. Generate the configuration for profile manager.
    - b. Add the Profile Manager to the pipeline.
    - c. Deploy the Profile Manager from the pipeline.
    - d. Deploy the workload VPN intent from the pipeline.

To edit or delete an ACL or QoS profile, first you have to create a candidate version of the Profile Manager. Then, edit or delete the profile and manually deploy the Profile Manager.

### 6.4.1 Deploying the Profile Manager

**About this task**

Whenever you change the definition of an ACL or ACL profile, you must deploy the Profile Manager. You can auto-deploy the Profile Manager or deploy it manually.

**Procedure**

**Step 1.** Select the deployment method.

- To deploy the Profile Manager manually, go to Step 2.
- To auto-deploy the Profile Manager, go to Step 3.

**Step 2.** Deploy the Profile Manager manually.

- a. From the **Profile Manager** view, click **GENERATE PROFILE MANAGER**.
- b. Click  to add the Profile Manager to the deployment pipeline.
- c. On the upper right of the **Profile Manager** view, click  and select **Deployment pipeline**.

- d. In the deployment pipeline, find the Profile Manager that is in queue and click  at the end of its row.  
In the deployment pipeline, in the **Source Name** column, the Profile Manager is identified as Global Intent.
- e. Select **Deploy**.
- f. From the **Workload VPN Intents Workload Design** view, click  to add the workload intent to the deployment pipeline.
- g. On the upper right of the page, click  and select **Deployment pipeline**.
- h. Find the workload VPN intent that is in queue, click  at the end of its row and select **Deploy**.

**Step 3.** Auto-deploy the Profile Manager.

If you are creating a new QoS or ACL profile, after assigning the profile to a subnet or sub-interface, deploy the Profile Manager as follows:

- a. Go to the workload VPN intent.
- b. Execute [Adding a workload VPN intent to the deployment pipeline](#).

**Expected outcome**

This step auto-deploys the Profile Manager. In the deployment pipeline, the status of the Profile Manager is deployed. Note that the workload VPN intent is not yet deployed; continue with [Deploying a workload VPN intent from the deployment pipeline](#) to deploy it.

## 6.4.2 Match groups

Match groups allow you specify a profile for specific types of packets which can then be used to indicate their inclusion or exclusion from workload traffic.

In the Fabric Services System UI, you can:

- Create a match group.
- Edit a match group.
- Delete a match group.

**Related topics**

[ACL profiles](#)

### 6.4.2.1 Creating a match group

**Procedure**

- Step 1.** Click  to open the main menu and select **Profiles**.
- Step 2.** From the **Profiles** drop-down list, select **Match Groups**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the match group.
- Step 4.** Create an IPv4 or IPv6 match group.
  - To create an IPv4 match group, go to step [5](#).

- To create an IPv6 match group, go to step 9.

**Step 5.** Click + **CREATE IPV4 MATCH GROUP**.

**Step 6.** Enter general information about the match group:

- a. Enter a **Name** for the match group.
- b. Optional: Enter a **Description**.

**Step 7.** Enter IPv4 match entry information for the match group:

- a. In the **IPv4 Match Entry** panel, click **+ADD**.
- b. Enter an IP address in the resulting form.  
The IP address must be specified as a prefix; that is, the host section must be all zeros.
- c. Click **ADD**.
- d. Repeat steps 7.a through 7.c until the IPv4 Address list is complete.

Note that the **ACL Reference List** is empty. This list shows all of the ACL policies that are currently using this IPv4 match group; but because this is a new match group, no profiles are using it.

- e. At the lower right of the **Match Group** overlay, click **CREATE**.

#### **Expected outcome**

The system closes the **Match Group Creation** overlay and returns you to the **Profiles** page with the **Match Groups** view selected. The match group you just created is now included in the list of available IPv4 match groups.

Repeat this step until the IPv4 match entry list is complete.

**Step 8.** Do one of the following:

- To create an IPv6 match group, go to step 9.
- If you are finished creating match groups, go to 12.

**Step 9.** Click + **CREATE IPV6 MATCH GROUP**. The **Match Group Creation** overlay displays.

**Step 10.** Enter general information about the match group:

- a. Enter a **Name** for the match group.
- b. Optional: Enter a **Description**.

**Step 11.** Enter IPv6 match entry information for the match group:

- a. In the **IPv6 Match Entry** panel, click **+ADD**.
- b. Enter an IP address in the resulting form.  
The IP address must be specified as a prefix; that is, the host section must be all zeros.
- c. Click **ADD**.
- d. Repeat steps 11.a through 11.c until the IPv6 address list is complete.



**Note:** The **ACL Reference List** field is empty. This list shows all of the ACL policies that are currently using this IPv6 match group; but because this is a new match group, no profiles are using it.

- e. At the lower right of the **Match Group** overlay, click **CREATE**.

**Expected outcome**

The system closes the **Match Group Creation** overlay and returns you to the **Profiles** page with the **Match Group** view selected. The match group you just created is now included in the list of available match groups.

**Step 12.** You have completed this procedure.

### 6.4.2.2 Editing a match group

**About this task**

You can edit a match group at any time.

After you edit a match group, you must update ACLs that rely on that match group. To aid you in identifying the affected ACLs, these ACLs display a True flag in their Need update status. Open and save the ACL.

If the updated ACL profile is being used by a workload VPN intent, and that workload VPN intent has already been generated or deployed, then you must regenerate that workload VPN intent:

- If the workload VPN intent has been generated but is not yet deployed, you can re-save and regenerate the workload VPN intent without creating a new version. Regenerating the workload VPN intent incorporates the new ACL settings into its configuration.
- If the workload VPN intent has already been deployed, you need to create a new candidate version of the workload VPN intent before you can regenerate and redeploy it with the new ACL settings.

**Procedure**

**Step 1.** Click  to open the main menu and select **Profiles**.

**Step 2.** From the **Profiles** drop-down list, select **Match Groups**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region containing the match group.

**Step 4.** Select a match group from the list, click the **More actions** icon (  ) at the right edge of the row, and select **Open** from the drop-down list.

**Step 5.** Update parameters for the match group.

**Step 6.** At the lower right of the **Match Groups** overlay, click **SAVE**.

**Related topics**

[Creating a match group](#)

### 6.4.2.3 Deleting a match group

**About this task**

You can only delete a match group that is not being used by one or more ACL profiles. For any match group, a list of ACL profiles that are using it are listed in the ACL Reference List when viewing the match group's details.

To delete a match group:

**Procedure**

**Step 1.** Click  to open the main menu and select **Profiles**.

**Step 2.** In the **Profiles** drop-down list, click **Match Groups**.

- Step 3.** Use the **Region Selector** at the top of the page to select the region containing the match group.
- Step 4.** Select a match group from the list by clicking on the **More actions** icon (  ) at the right edge of the row, and select **Delete** from the drop-down list.
- Step 5.** In the confirmation form, click **OK**.

**Expected outcome**

The system deletes the selected match group and closes the confirmation form, returning you to the **Profiles** page with the **Match Groups** view selected. The match group you just deleted no longer appears in the list.

### 6.4.3 QoS profile management

QoS profiles allow you to shape the traffic managed by a workload VPN intent.

A QoS profile is global and can be re-used across workload intents. After you create QoS profiles, you then assign them to sub-interfaces of a workload VPN intent. The Profile Manager handles the deployment of QoS profiles. Before you deploy a workload intent, you must first deploy the Profile Manager.

If two workload intents reference the same QoS profile on the same node, only a single resulting QoS policy is created. The list of nodes on which an QoS policy is required is dynamically created based on the attachments of sub-interfaces to subnets where the QoS profile is referenced.

The configuration of a QoS policy is dynamically added to the devices when sub-interfaces that have QoS profiles referenced are added to subnets. The node list for a QoS profile is dynamically updated and the configuration is automatically deployed on the nodes as soon as the node is added to the list for a specified QoS profile.

You can generate the configuration of a QoS profile even if there are no nodes that require the configuration, which means that you can view the resulting QoS policies before they are associated with subnets or sub-interfaces in the QoS profile.

When you remove a QoS profile from a sub-interface of a node, the QoS policy configuration is automatically removed when the workload intent is deployed.

**Related topics**

[QoS profiles](#)

#### 6.4.3.1 Creating a QoS profile

**About this task**

You can create either a QoS classifier profile or a QoS rewrite rule profile.

**Procedure**

- Step 1.** Click  to open the main menu and select **Profiles**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the QoS profile.
- Step 3.** From the **Profiles** drop-down list, select **QoS**.
- Step 4.** If the Profile Manager is in the deployed state, create a candidate version.  
On the upper right of the **QoS** view, click  and select **Create Candidate Version**.

**Expected outcome**

The **Detailed Status** field shows **Created**.

- Step 5.** Click **+ADD QOS PROFILE**, then specify the type of profile you are creating.
- To create a QoS classifier profile, click **CREATE QOS CLASSIFIER**, then go to Step 6.
  - To create a QoS profile for rewrite rules, click **CREATE QOS REWRITE RULE**, then go to Step 7.
- Step 6.** Configure settings for a QoS classifier profile.
- a. In the **General** pane, provide a name for the QoS profile and an optional description.
  - b. In the **Classifiers** pane, click **+ DSCP POLICY**.
  - c. In the **Add DSCP Policy** form, set the following parameters for the policy.
    - **Value**
    - **Forwarding Class**
  - d. Click **ADD**.  
Continue adding DSCP policies as needed.
  - e. Go to 8.
- Step 7.** Configure settings for the QoS rewrite rules profile.
- a. In the **General** pane, provide a name for the QoS profile and an optional description.
  - b. In the **Rewrite Rules** panel, click **+ QOS MAPPING**.
  - c. In the **Add QoS Mapping** form, set the following parameters for the mapping entry:
    - **Forwarding Class**
    - **DSCP**
  - d. Click **ADD**.
  - e. Repeat steps 7.c and 7.d until the DSCP policy list is complete.



**Note:** To delete a DSCP policy from the list while you are executing this procedure, click  at the end of the policy's row and select **Delete**.

**Expected outcome**

The **Workload Reference List** field is empty. This field shows all of the workload VPN intents that are currently using this global QoS profile, but because this profile is new, no workload VPN intents are using it.

- Step 8.** At the lower right of the QoS overlay, click **CREATE**.

**Expected outcome**

The QoS profile is added to the **QoS** view.

### 6.4.3.2 Editing and deploying a QoS profile

#### About this task

If the Profile Manager is already deployed, to edit a QoS profile, first, create a candidate version of the Profile Manager. Then, edit the QoS profile. To push the changes to the nodes, add the Profile Manager to the deployment pipeline and then deploy it.

#### Procedure

- Step 1.** Click the main menu  and select **Profiles**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing the QoS profile.
- Step 3.** If the Profile Manager is in the deployed state (that is, the **Detailed Status** field shows Deployed), create a candidate version first.

On the upper right of the **Profile Manager** view, click  and select **Create Candidate Version**.

#### Expected outcome

The **Detailed Status** field shows **Created**.

- Step 4.** From the **Profiles** drop-down list, select **QoS**.
- Step 5.** Update the QoS profile.
  - a. Double-click the profile that you want to edit.
  - b. Update parameters as needed.
  - c. On the lower right of the **QoS** overlay, click **SAVE**.
- Step 6.** From the **Profile Manager** view, click **GENERATE PROFILE MANAGER**.
- Step 7.** Click  to add the Profile Manager to the deployment pipeline.
- Step 8.** Deploy the Profile Manager manually.

See Step 2 in [Deploying the Profile Manager](#).

### 6.4.3.3 Deleting a QoS profile

#### About this task

You can delete a QoS profile provided it has not been used in a workload VPN intent. If you try to delete a QoS profile that is in use, the system prevents the action and displays a message indicating why the profile cannot be deleted.

#### Procedure

- Step 1.** Click  to open the main menu, then select **Profiles**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing the QoS profile.
- Step 3.** If the Profile Manager is in the deployed state (that is, the **Detailed Status** field shows Deployed), create a candidate version first.

On the upper right of the **Profile Manager** view, click  and select **Create Candidate Version**.

#### Expected outcome

The **Detailed Status** field shows **Created**.

- Step 4.** From the **Profiles** drop-down list, select **QoS**.
- Step 5.** Find the profile from the list and click  at the end of its row.
- Step 6.** Select **Delete** from the drop-down list, then click **OK**.
- Step 7.** Deploy the Profile Manager manually.  
See Step 2 in [Deploying the Profile Manager](#).

### Expected outcome

The system deletes the selected profile and closes the confirmation form, returning you to the **Profiles** page with the **QoS** view selected. The profile you just deleted no longer appears in the list.

## 6.4.4 ACL profile management

ACL profiles allow you to create IP filters that can be applied on a node when you associate the profile with a workload intent subnet or sub-interface.

### ACL profile deployment

An ACL profile is global, which means that the IP filters that are created can be reused across workload intents. If two workload intents reference the same ACL profile on the same node, only a single resultant IP filter is created.

The high-level steps for deploying an ACL profile is as follows:

1. Assigning ACL profiles to the sub-interfaces and subnets in a workload intent.
2. Generating the configuration of the workload intent.
3. Adding the workload intent to the pipeline.
4. Auto-deploying the workload intent.  
At this point, the ACL profile is automatically deployed. The workload intent still has to be deployed.
5. Deploying the workload intent in the pipeline.

If a node no longer has any sub-interfaces in a subnet that references an ACL profile, the IP filter configuration is removed from the device.

The configuration of an IP filter is dynamically added to the devices when sub-interfaces are created on subnets that reference ACL profiles. That is, the node list for an ACL profile is dynamically updated and the configuration of the IP filter is automatically deployed on the nodes as soon as the node is added to the list for a specific ACL profile.

You can generate the configuration of an ACL profile even if there are no nodes that require the configuration, which means that you can view the resulting ACL policies before they are associated with subnets or sub-interfaces in the ACL profile.

When you remove an ACL profile from a subnet, the ACL policy configuration is automatically removed from the node when the workload intent is deployed.

### Validation of an ACL entry

The Fabric Services System does not validate the combinations of fields that constitute an ACL entry. You must configure these fields in accordance with the requirements of the target platform. For this reason, be aware of the applicable requirements, limitations, and dependencies of the participating nodes when configuring any ACL profile as part of a workload VPN intent.

For example, the Fabric Services System allows you to configure an IPv4 ACL entry with the following attributes:

- `Protocol = icmp`
- `Destination port = 80`

However, this configuration is not permitted by SR Linux because port numbers can only be configured for protocols such as UDP and TCP.

If you deploy a workload VPN intent that includes an ACL configuration that is unsupported by the target node, the deployment results in an error. A message from the platform indicates the conflicting configuration.

#### Related topics

[ACL profiles](#)

### 6.4.4.1 Creating an ACL profile

#### Prerequisites

Before you create an ACL profile, ensure that you have created match groups that describe the types of packets you want to accept or reject as part of the ACL.

#### About this task

When you create an ACL profile, you define a set of packets that should be accepted or rejected by the system. You can create an IPv4 ACL profile or an IPv6 ACL profile.

You define these packet sets by selecting one or more previously created match groups. Each match group already defines one set of possible packets properties; the ACL profile assembles the match groups to create a full profile of the packets deemed acceptable for the current profile.

Later, you can assign ACL profiles to workload VPN intents to represent the packets that are acceptable or unacceptable for the workload VPN intent. The workload VPN intent either accepts or rejects (depending on your selection) all packets that conform to the profiles encompassed by its assigned ACLs.

The total number of entries created for a single IPv4 or IPv6 ACL is the product of the following numbers:

- the number of IP address in the source match groups
- the number of IP addresses in destination match groups

#### Procedure

**Step 1.** Click  to open the main menu and click **Profiles**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the ACL profile.

**Step 3.** From the **Profiles** drop-down list, select **ACL**

**Step 4.** If the Profile Manager is in the deployed state, create a candidate first.

On the upper right of the **ACL** view, click  and select **Create Candidate Version**.

#### Expected outcome

The **Detailed Status** field shows **Created**.

**Step 5.** Click **+ADD ACL PROFILE**, then specify the type of profile you are creating.

- To create an IPv4 ACL profile, click **CREATE IPV4 ACL**, then go to [6](#).

- To create an IPv6 ACL profile, click **CREATE IPV6 ACL**, then go to [7](#).

**Step 6.** Configure the IPv4 ACL profile.

- a. In the **Match Group Mappings IPv4** section, click **+ADD**.
- b. In the **Match Group Mapping IPv4 Details** form, set the following parameters in the **General** section:
  - **Priority**
  - **Accept/Reject** drop-down list: select **Accept** or **Reject**. This setting determines whether the match group you are selecting is intended to define acceptable or unacceptable packet types.
- c. In the **Source Match Groups** panel, click the box to the left of each pre-existing IPv4 match group you want to associate with the packet source.

**Expected outcome**

The ACL now represents an instruction to either accept or reject (based on your selection above) any packet whose source IP address conforms to the information in the selected match groups.

- d. In the **Destination Match Groups** panel, click the box to the left of each pre-existing IPv4 match group you want to associate with the packet destination.

**Expected outcome**

The ACL now represents an instruction to either accept or reject (based on your selection above) any packet whose destination IP address conforms to the information in the selected match groups.

- e. In the **IPv4 Match Entry** section, set the following parameters for the packets to be accepted or rejected by this ACL:
  - **First Fragment**
  - **Fragment**
  - **Source Port Operator**
  - **Source Port Value**  
The setting for the **Source Port Value** parameter can either be a number or a text string associated with a predefined port. Enter a number or select a value from the drop-down list.
  - **Source Port Range Start**
  - **Source Port Range End**
  - **Destination Port Operator**
  - **Destination Port Value**
  - **Destination Port Range Start**
  - **Destination Port Range End**
  - **ICMP Code**: can support multiple values
  - **ICMP Type**
  - **Protocol**
  - **TCP Flags**

- f. Click **ADD**.
- g. Go to Step 8.

**Step 7.** Configure the IPv6 ACL profile.

- a. In the **Match Group Mappings IPv6** panel, click **+ADD**.
- b. In the **Match Group Mapping IPv6 Details** overlay, set the following parameters in the **General** section:
  - **Priority**
  - **Accept/Reject** drop-down list: select **Accept** or **Reject**. This setting determines whether the match group you are selecting is intended to define acceptable or unacceptable packet types.
- c. In the **Source Match Groups** panel, click the box to the left of each pre-existing IPv6 match group that you want to associate with the packet source.

The ACL now represents an instruction to either accept or reject (based on your selection above) any packets whose source IP address conforms to the information in the selected match groups.
- d. In the **Destination Match Groups** panel, click the box to the left of each pre-existing IPv6 match group you want to associate with the packet destination.

The ACL now represents an instruction to either accept or reject (based on your selection above) any packet whose destination IP address conforms to the information in the selected match groups.
- e. Set IPv6 Match Entry values for the packets to be accepted, or rejected, by this ACL:
  - **Source Port Operator**
  - **Source Port Value**

The setting for **Source Port Value** can either be a number or a text string associated with a predefined port. Type a number or select a value from the drop-down list.
  - **Source Port Range Start**
  - **Source Port Range End**
  - **Destination Port Operator**
  - **Destination Port Value**
  - **Destination Port Range Start**
  - **Destination Port Range End**
  - **ICMP Code**: can support multiple values
  - **ICMP Type**
  - **Next Header**
  - **TCP Flags**
- f. Click **ADD**. The system adds the match group to the **Match Group Mappings IPv6** list.

**Step 8.** Click **CREATE**.



**Note:** The **Workload Reference List** field shows all of the workloads that are currently using this ACL profile, but because this is a new ACL profile, it is empty because no profiles are using it.

#### Related topics

[Creating a match group](#)

### 6.4.4.2 Editing and deploying an ACL profile

#### About this task

If the Profile Manager is already deployed, to edit a QoS profile, create a candidate version of the Profile Manager first, then edit the QoS profile. Then, to push the changes to the nodes, add the Profile Manager to the deployment pipeline and deploy it.

#### Procedure

**Step 1.** Click the main menu  and select **Profiles**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing the ACL profile.

**Step 3.** If the Profile Manager is in the deployed state (that is, the **Detailed Status** field shows Deployed), create a candidate version first.

On the upper right of the **Profile Manager** view, click  and select **Create Candidate Version**.

#### Expected outcome

The **Detailed Status** field shows **Created**.

**Step 4.** From the **Profiles** drop-down list, select **ACL**.

**Step 5.** Update the ACL profile.

- a. Double-click the profile that you want to edit.
- b. Update parameters as needed.
- c. On the lower right of the **ACL** overlay, click **SAVE**.

**Step 6.** Deploy the Profile Manager manually.

See Step 2 in [Deploying the Profile Manager](#).

#### Related topics

[Creating an ACL profile](#)

### 6.4.4.3 Deleting an ACL profile

#### About this task

There are some restrictions in place when deleting an ACL profile to ensure that you do not invalidate any fabric intents that rely on it:

- If an ACL profile has been assigned to a bridged subnet or a sub-interface or to a routed sub-interface, the system prevents you from deleting the ACL.

- You cannot delete an ACL profile associated with a previous version of a deployed workload intent, even if you are designing a subsequent, undeployed version of that same intent that no longer relies on that ACL profile.
- After you deploy a workload VPN intent that no longer relies on an ACL profile, the system allows the deletion of the unassociated ACL profile (provided no other workload VPN intent still relies on it).

### Procedure

**Step 1.** Click the main menu  and select **Profiles**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing the ACL profile.

**Step 3.** If the Profile Manager is in the deployed state (that is, the **Detailed Status** field shows Deployed), create a candidate version first.

On the upper right of the **Profile Manager** view, click  and select **Create Candidate Version**.

#### Expected outcome

The **Detailed Status** field shows **Created**.

**Step 4.** From the **Profiles** drop-down list, click **ACL**.

**Step 5.** Select an ACL profile from the list and click  at the end of its row.

**Step 6.** Select **Delete** from the drop-down list, then click **OK**.

**Step 7.** Deploy the Profile Manager.

See Step 2 in [Deploying the Profile Manager](#).

#### Expected outcome

The system deletes the selected ACL profile and closes the confirmation form, returning you to the **Profiles** page with the ACL view selected. The ACL profile you just deleted no longer appears in the list.

## 6.5 Workload VPN intent creation

A workload VPN intent assigns fabric resources to specific sources of demand.

### Prerequisites

Before you create a new workload VPN intent, ensure the following:

- The region for the workload VPN intent has been created.
- All fabrics that you intend to use in with this workload VPN intent have been created and successfully deployed.
- The QoS profiles that you intend to use with this workload VPN intent have been created.
- The ACL profiles that you intend to use with this workload VPN intent have been created.
- The LAGs that you intend to act as sub-interfaces for your workload VPN intent have already been created within the system.

### Procedure overview

Creating a workload VPN intent involves the following sub-tasks, each consisting of multiple steps:

1. [Creating the basic workload VPN intent](#)
2. [Adding subnets to the workload VPN intent](#)
3. [Adding sub-interfaces to the workload VPN intent](#)
4. Configuring routing for the workload, as described in [Routing](#)

### Empty workload intents

At a high-level, the deployment of a workload intent involves the following tasks:

1. Generating the configuration
2. Adding the workload intent to the deployment queue
3. Deploying queue item

The system allows the creation and deployment an empty workload intent without generating an error message, including the following cases:

- there is no change in the candidate version
- there are one or more subnets and routers, but no sub-interfaces
- there a no subnets or routers and no sub-interfaces
- other situations that normally would not allow deployment that are not error states

#### Related topics

[Elements of a workload VPN intent](#)

[Deployment regions](#)

[Fabric intents](#)

[QoS profile management](#)

[ACL profile management](#)

[Creating LAGs](#)

## 6.5.1 Workload VPN intent parameter descriptions

This section describes the required and optional workload VPN intent parameters and the appropriate values that you can set in the platform.

### 6.5.1.1 Workload VPN intent parameters

Table 27: Workload design parameters

Parameter	Description	Values
<b>Workload VPN Intent Name</b>	Specifies a unique name for the workload VPN intent.	Any string value
<b>Description</b>	Specifies an optional description for the workload VPN intent.	Any string value
<b>Fabric Intent Type</b>	Specifies the fabric intent environment.	<b>Real or Digital Sandbox</b>

Parameter	Description	Values
<b>Fabric Intents</b>	Identifies one or more fabrics that you want to include in the workload VPN intent.  If no fabric is selected, all fabrics in the region are part of the workload intent, and you can create a sub-interface on any fabric within the region. Selecting a fabric intent creates a restriction list composed only of sub-interfaces that belong to the selected fabric intents.	Select from existing fabric intents or leave blank.
<b>Labels</b>	Specifies the labels to apply to the workload VPN intent  The labels are not selected during workload VPN intent creation, but you can apply labels to the workload VPN intent itself later.	Supported labels

### 6.5.1.2 Subnet parameters

Table 28: Subnet configuration parameters

Parameter	Description	Values
<b>Name</b>	Specifies the name of the subnet.	String
<b>Description</b>	Provides a description for the subnet.	String
<b>Type</b>	Specifies the type of subnet.	<b>Bridged, Routed, or Loopback</b>
<b>IP Anycast Gateway (V4/V6)</b>	<b>IP Gateway (V4/V6):</b> For bridged subnets, specifies an IP gateway to act as an IRB interface.	An IP address with a required CIDR
<b>Anycast Gateway MAC address</b>	Specifies an anycast gateway MAC address. This option is available if an IP anycast gateway is set.	MAC address
<b>Primary</b>	Sets an address as the primary address. The primary address is used to form a BGP peering session between a multinetted interface and a neighbor.	Default: disabled
<b>Router</b>	Specifies the router to attach this subnet.	the default router presented or select an existing router from the drop-down list
<b>BFD</b>	Enables or disables bidirectional forwarding detection (BFD) for the subnet.  When BFD is enabled, the default settings for <b>Desired Minimum Transit Interval</b> , <b>Desired Minimum Transit Interval</b> , and <b>Required Minimum Receive</b> apply.  BFD is applicable to bridged, routed, and loopback subnets.	—

Parameter	Description	Values
<b>Desired Minimum Transit Interval</b>	Specifies the minimum interval between the transmission of BFD control packets that this system wants to use.	Default: 1,000,000 microseconds
<b>Required Minimum Receive</b>	Specifies the required minimum interval between received BFC control packet that this system requires.	Default: 1,000,000 microseconds
<b>Detection Multiplier</b>	Specifies the multiplier used to determine the BFD detection interval. BFD detection interval = <i>Detection Multiplier</i> x <i>Desired Minimum Transit Interval</i>	1 to 5 Default: 3
<b>IP MTU</b>	For bridged subnets, specifies the maximum transmission unit allowed.	1500 or higher
<b>VNI</b>	Specifies the unique VXLAN network identifier (VNI) from the selected VNI pool. If no value is specified, the Fabric Services System assigns a VNI from the VNI pool.	—
<b>Provision Type</b>	Specifies whether the route targets are automatically derived or manually set. If set to <b>Manual</b> , you can set the following parameters: <ul style="list-style-type: none"> <li>• <b>Import Route Target</b></li> <li>• <b>Export Route Target</b></li> </ul>	<b>Automatically Derived</b> (the default) or <b>Manual</b>
<b>Import Route Target</b>	Specifies the name of a BGP policy to use as an import policy.	String
<b>Export Route Target</b>	Specifies the name of a BGP policy to use as an export policy.	String
Layer 2 proxy ARP		
<b>L2 proxy ARP</b>	Enables or disables Layer 2 proxy ARP on a bridged network. When this parameter is enabled, you can set the following parameters: <ul style="list-style-type: none"> <li>• <b>Table size</b></li> <li>• <b>Duplicate IP Detection</b></li> </ul>  <b>Note:</b> You cannot enable this parameter if a gateway has been attached to the subnet.	Default: disabled
<b>Table size</b>	Specifies the size of proxy ARP table, that is, the maximum number of entries.	Default: 250
<b>Duplicate IP Detection</b>	Enables duplicate IP address detection for the subnet. When this field is enabled, you can configure the following settings: <ul style="list-style-type: none"> <li>• <b>Hold Down Time</b></li> <li>• <b>Monitoring Window</b></li> <li>• <b>Num Moves</b></li> </ul>	Default: disabled

Parameter	Description	Values
<b>Hold Down Time</b>	Specifies the time from the moment an IP address is considered duplicate to the moment the IP address is removed from the proxy ARP table.	Integer Default: 9 minutes
<b>Monitoring Window</b>	Specifies the number of minutes that the system monitors a proxy ARP table entry following an IP address move.	Default: 3 minutes
<b>Num Moves</b>	Specifies the maximum number of moves a proxy ARP table entry can have during the monitoring window before the IP is considered duplicate.	Default: 5 moves
Layer 3 proxy ARP and related settings		
<b>L3 ProxyArp Enabled</b>	Enables Layer 3 proxy ARP for a bridged subnet that is configured with a gateway IP address. For a routed subnet, Layer 3 proxy ARP is enabled on the sub-interface.	Default: disabled
<b>IPv4 Host Route Enabled</b>	Enables the dynamic population of IPv4 host routes. When the <b>L3 ProxyArp Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3 ProxyArp Enabled</b> is disabled.	—
<b>IPv4 Learn Unsolicited ARP Enabled</b>	For IPv4 addresses within the subnet, enables the learning of ARP entries out of any ARP packet arriving at the IRB sub-interface, regardless of whether there was an ARP-Request sent from the IRB. When the <b>L3 ProxyArp Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3 ProxyArp Enabled</b> is disabled.	—
<b>L3ProxyND Enabled</b>	Enables Layer 3 proxy neighbor discovery (ND) for a bridged subnet that is configured with a gateway IP address. For a routed subnet, Layer 3 proxy ARP is enabled on the sub-interface.	Default: disabled
<b>IPv6 Host Route Enabled</b>	Enables the dynamic population of IPv6 host routes. When the <b>L3 ProxyArp Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3 ProxyArp Enabled</b> is disabled.	—
<b>IPv6 Learn Unsolicited ARP Enabled</b>	For IPv6 addresses within the subnet, enables the learning of Neighbor Discovery Request entries out of any Neighbor Discovery Request packet arriving at the IRB sub-interface, regardless of whether there was a Neighbor Discovery Request issued from the IRB.	Default: disabled
ACL parameters		
<b>Ingress ACL IPv4</b>	Specifies an existing profile that the system applies to the ingress IPv4 traffic on this subnet.	An existing IPv4 ACL profile from the drop-down list

Parameter	Description	Values
<b>Ingress ACL IPv6</b>	Specifies an existing profile that the system applies to the ingress IPv6 traffic on this subnet.	An existing IPv6 ACL profile from the drop-down list
<b>Egress ACL IPv4</b>	Specifies an existing profile that the system applies to the egress IPv4 traffic on this subnet.	An existing IPv4 ACL profile from the drop-down list
<b>Egress ACL IPv6</b>	Specifies an existing profile that the system applies to the egress IPv6 traffic on this subnet.	An existing IPv6 ACL profile from the drop-down list MAC duplication and detection parameters
<b>Mac Duplication Detection</b>	Enables MAC duplication detection for the subnet. When this parameter is enabled, you can set the following parameters: <b>Action</b> <b>Hold Down Time</b> <b>Monitoring Window</b> <b>Num Moves</b>	Default: disabled
<b>Action</b>	Specifies the action to take on the sub-interface upon detecting that at least one MAC address is a duplicate: <ul style="list-style-type: none"> <li>• <b>stop learning</b>: the MAC address is not relearned on this or any sub-interface</li> <li>• <b>blackhole</b>: frames received on this or any other sub-interface are dropped if the MAC sources address or if the MAC-VFR MAC destination address matches a blackhole MAC address (the MAC source address is still learned)</li> <li>• <b>oper-down</b>: the sub-interface is disabled with an mac-dup-detected error message; arriving frames on a different sub-interface with the same source address are dropped</li> </ul>	Default: <b>stop learning</b>
<b>Hold Down Time</b>	Specifies the time to wait from the moment a MAC address is declared duplicate before it is flushed from the bridge table, after which the monitoring process for the MAC address is restarted.	2 to 60 minutes Default: 9
<b>Monitoring Window</b>	Specifies the period, in minutes, during which the moves are observed.	1 to 15 Default: 3

### 6.5.1.3 Sub-interface parameters

Table 29: Sub-interface configuration parameters

Parameter	Description	Values
<b>Subnet</b>	Specifies the subnet with which this sub-interface is associated.	An existing subnet
<b>Description</b>	Describes the selected sub-interface.	String
<b>IP Gateway (V4/V6)</b>	Specifies the IP address of the forwarding device. If the IP address is the primary gateway, set the <b>Primary</b> field. To form a BGP peering session between a multi-netted interface and a neighbor, one of the gateway IP addresses must be set to primary.	IP address of the gateway device
<b>Encap Type</b>	Configures encapsulation settings for bridged subnets: <ul style="list-style-type: none"> <li>• <b>UnTagged</b> – specifies that untagged frames can be captured on tagged interfaces</li> <li>• <b>Single Tagged</b> – specify one of the following options: <ul style="list-style-type: none"> <li>– <b>Vlan ID Any</b> – specifies that non-configured VLAN IDs and untagged traffic are classified to a Layer 2 sub-interface</li> <li>– <b>Vlan ID</b> – specifies the VLAN ID, a value from 1 to 4094</li> </ul> </li> <li>• <b>Single Tagged Range</b> – click <b>+ ADD</b> to open the <b>ADD VLAN Range</b> form where you can enter the low end and high end of the VLAN range. You can add up to eight non-overlapping ranges.</li> </ul>	<b>UnTagged, Single Tagged, or Single Tagged Range</b>
<b>IP MTU</b>	Specifies the maximum transmission unit for the sub-interface; this is the maximum size for an IP packet that is not fragmented in the course of transmission.	1500 or higher
Association parameters		
<b>Association Type</b>	Specifies the method used to associate this sub-interface with its "parent" subnet.	<b>Node and Interface, Interface label selector</b>
<b>Node ID</b>	Specifies the node within the fabric on which the current sub-interface is located.	Select an existing leaf node within the fabric or fabrics associated with this workload VPN intent
<b>Interface Name</b>	Specifies the interface on the selected node with which this sub-interface is associated. This setting can be a LAG.	An interface
Layer 3 proxy ARP and related parameters		

Parameter	Description	Values
<b>L3 ProxyArp Enabled</b>	Enables L3 proxy ARP for a sub-interface attached to routed subnet.	Default: disabled
<b>IPv4 Host Route Enabled</b>	Enables the dynamic population of IPv4 host routes.	When the <b>L3 Proxy Arp Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3 ProxyArp Enabled</b> is disabled.
<b>IPv4 Learn Unsolicited ARP Enabled</b>	For IPv4 addresses within the subnet, enables the learning of ARP entries out of any ARP packet arriving at the IRB sub-interface, regardless of whether there was an ARP-Request sent from the IRB.	When the <b>L3 Proxy Arp Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3 ProxyArp Enabled</b> is disabled.
<b>L3ProxyND Enabled</b>	Enables L3 proxy ND for a sub-interface attached to routed subnet.	default: disabled
<b>IPv6 Host Route Enabled</b>	Enables the dynamic population of IPv6 host routes.	When the <b>L3ProxyND Enabled</b> parameter is enabled, this parameter is also enabled. You can disable it if <b>L3ProxyND Enabled</b> is disabled.
<b>IPv6 Learn Unsolicited ARP Enabled</b>	For IPv6 addresses within the subnet, enables the learning of Neighbor Discovery Request entries out of any Neighbor Discovery Request packet arriving at the IRB sub-interface, regardless of whether there was a Neighbor Discovery Request issued from the IRB.	default: disabled
MAC duplication and detection parameters (if enabled on the subnet)		
<b>Action</b>	<p>Specifies the action to take on the sub-interface upon detecting that at least one MAC addresses is duplicate on the sub-interface:</p> <ul style="list-style-type: none"> <li>• <b>stop learning</b> – the MAC address is not relearned on this or any subinterface</li> <li>• <b>blackhole</b> – frames received on this or any other sub interface are dropped if the MAC sources address or if the mac-vrf MAC destination address matches a blackhole MAC address (the MAC source address is still learned)</li> </ul>	—

Parameter	Description	Values
	<ul style="list-style-type: none"> <li><b>oper-down</b> – the sub-interface is disabled with an error mac-dup-detected; arriving frames on a different sub-interface with the same source address are dropped</li> </ul>	
ACL parameters		
<b>Ingress ACL IPv4</b>	Specifies an existing profile that the system applies to the ingress IPv4 traffic on this sub-interface.	An existing IPv4 ACL profile
<b>Ingress ACL IPv6</b>	Specifies an existing profile that the system applies to the ingress IPv6 traffic on this sub-interface.	An existing IPv4 ACL profile
<b>Egress ACL IPv4</b>	Specifies an existing profile that the system applies to the egress IPv4 traffic on this sub-interface.	An existing IPv4 ACL profile
<b>Egress ACL IPv6</b>	Specifies an existing profile that the system should apply to the egress IPv6 traffic on this sub-interface.	An existing IPv4 ACL profile
QoS parameters		
<b>QoS Classifier</b>	Specifies an existing QoS classifier profile that maps incoming packets to the appropriate forwarding classes.	An existing QoS profile
<b>QoS Rewrite Rule</b>	Specifies an existing QoS profile that defines the rewrite rule policies to mark outgoing packets with an appropriate DSCP value based on the forwarding class.	An existing QoS profile

### 6.5.1.4 Router parameters

Table 30: Router configuration parameters

Parameter	Description	Value
<b>Name</b>	This parameter specifies the name of the router.	String
<b>Description</b>	This parameter specifies the optional description for the router.	String
<b>VNI Pool</b>	<p>By default, the Fabric Services System deploys with a default VNI pool.</p> <p>For bridged subnets, you can:</p> <ul style="list-style-type: none"> <li>select from which VNI pool a VNI gets automatically allocated to a new subnet</li> <li>change the VNI pool after the subnet has been deployed.</li> </ul>	Default VNI pool
<b>VNI</b>	Specifies an available VNI from the selected VNI pool.	Default: system-assigned VNI from the pool

Parameter	Description	Value
<b>Provision Type</b>	By default, route targets are automatically derived. When this parameter is set to <b>Manual</b> , you can specify route targets for the subnet using following parameters: <ul style="list-style-type: none"> <li>• <b>Import Route Target</b></li> <li>• <b>Export Route Target</b></li> </ul>	<b>Automatically Derived</b> (the default) or <b>Manual</b>

#### Related topics

[Creating a router](#)

## 6.5.2 Creating the basic workload VPN intent

### Procedure

- Step 1.** Click  to open the main menu and select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the workload VPN intent.



**Note:** You cannot change the region selection after you begin creating the workload VPN intent. If you select a new region in the **Region Selector** while creating a workload VPN intent, the creation form closes and you are returned to the **Workload VPN Intents** page.

- Step 3.** Click **+ CREATE A WORKLOAD VPN INTENT** to display a set of fabric templates.

#### Expected outcome

Templates are displayed in a grid view by default. To switch to the list view, select  in the template selection screen. Click  to return to the grid view.

- Step 4.** Click on a VPN template, then click **CREATE**.

- Step 5.** Configure basic parameters.
- **Workload VPN Intent Name**
  - **Description**
  - **Fabric Intent Type**

- Step 6.** Optional: Select any number of fabrics intents or no fabric intents to participate in the workload intent.

If no fabric is selected, all fabrics in the region are part of the workload intent, and you can create a sub-interface on any fabric within the region. Selecting a fabric intent creates a restriction list composed only of sub-interfaces that belong to the selected fabric intents.

- Click  next to **Fabric Intents**.
- Check the box at the left edge of the row for each fabric you want to include as part of your workload intent.
- Click **SELECT INTENTS**.

**Step 7.** Click  to save the latest change to the workload design.

**Expected outcome**

The display updates to show the selected fabric intent's topology. The system advances the workload VPN intent's Detailed Status to Created and its Version to 1.0.

**What to do next**

Proceed to [Adding subnets to the workload VPN intent](#).

### 6.5.3 Adding subnets to the workload VPN intent

**Procedure**

**Step 1.** If you are not continuing directly from the procedure [Creating the basic workload VPN intent](#), first open the **Workload VPN Intent** view by doing the following:

- a. Click  to open the main menu.
- b. From the menu, select **Workload VPN Intents**.
- c. Use the **Region Selector** at the top of the page to select the region in which to create the workload VPN intent.

**Step 2.** In the view drop-down list, select **Subnets**.

**Step 3.** Click **+CREATE A SUBNET**.

**Step 4.** Configure the basic parameters for the subnet.

- **Name**
- **Description**

**Step 5.** In the **Type** drop-down list, specify the type of subnet.

- bridged subnet – click **Bridged**, then continue with step 6
- routed subnet – click **Routed**  
In the **Router** field, accept the default router or select an existing router. Then, continue to step 14.  
Do not add an IRB IP address here. Later, you connect the routed subnet to a sub-interface that attaches to a VRF instance.
- loopback subnet – click **Loopback**  
In the **Router** field, accept the default router or select an existing router. Then, continue to step 14.

**Step 6.** Configure parameters for the bridged subnet.

Set the following parameters:

- **IP Anycast Gateway (V4/V6)** – this IP address acts as an IRB interface  
The subnet can span one, two, or more nodes.  
Click **+ADD** to add an IP address. In the **Add IP Anycast Gateway** form that displays, add the IP address. If the IP address is the primary, click the **Primary** field. Click **ADD**. You can add up to four gateways.
- **Anycast Gateway MAC address**—This option is available if an IP anycast gateway is set.
- **Router** – for related information, see [Routers](#).

- Step 7.** Optional: For bridged subnets with a configured gateway, enable Layer 3 IPv4 proxy ARP, IPv6 proxy ND, and related settings.
- **L3 ProxyArp Enabled**  
Enabling Layer 3 IPv4 proxy ARP also enables the following parameters; when Layer 3 IPv4 proxy ARP is disabled, you can enable them independently:
    - **IPv4 Learn Unsolicited ARP Enabled**
    - **IPv4 Host Route Enabled**
  - **L3 ProxyND Enabled**  
Enabling Layer 3 IPv6 proxy ND also enables **IPv6 Learn Unsolicited ARP Enabled**; when Layer 3 IPv6 proxy ND is disabled, you can enable it independently.
- Step 8.** Optional: Enable BFD and related BFD timers.
- Step 9.** Optional: Accept the default or select a new value for the **IP MTU** parameter IP.
- Step 10.** Optional: Configure ACL settings.  
Select existing ACL profiles for the following parameters:
- **Ingress ACL Profile IPV4**
  - **Ingress ACL Profile IPv6**
  - **Egress ACL Profile IPV4**
  - **Egress ACL Profile IPv6**
- Step 11.** Optional: Set a specific pool VNI from which the Fabric Services System allocates VNI and route targets for an IP-VRF or MAC-VRF object within a workload VPN intent.  
You can use these settings to configure the Fabric Services System to automatically derive a route target, while ensuring that the values used do not overlap with existing services elsewhere in the data center. You can update the following fields:
- **VNI**
  - **Provision Type**
    - **Import Route Target**
    - **Export Route Target**
- Step 12.** Optional: For bridged subnets without a configured gateway, enable L2 proxy ARP settings.  
When you enable L2 proxy ARP, you can also set the L2 ARP table size. You can also configure the following duplicate IP detection parameters:
- **Hold Down Time**
  - **Monitoring Window**
  - **Num Moves**
- Step 13.** Optional: Enable MAC duplication detection.
- Step 14.** Click **CREATE**.
- Expected outcome**  
The newly added subnet appears in the **Subnets** view.
- Step 15.** In the view drop-down list, select **Workload Design**.
- Step 16.** Click  to save the latest change to the workload design.

## What to do next

Proceed to [Adding sub-interfaces to the workload VPN intent](#).

### Related topics

[Subnet parameters](#)

[Creating a router](#)

## 6.5.4 Adding sub-interfaces to the workload VPN intent

### Prerequisites

If you intend to select sub-interfaces by their label, you must have assigned labels to the intended sub-interfaces.

### About this task

Each sub-interface is associated with a previously created subnet. A workload sub-interface consists of an edge-link port or LAG with which you associate ACL and QoS policies.

The Fabric Services System supports two methods for selecting the edge-link port or LAG that constitutes a sub-interface:

- Node and interface – explicitly select a node and then an interface on that node
- Interface label selector – assign the Edge-Link label to a set of objects, and then select the label from among those previously created and assigned to underlay interfaces. All interfaces with the specified label are selected

### Procedure

**Step 1.** Open a **Create Sub-Interface** form.

- From the **Subnets** view, find the subnet, click  at the end of its row, and select **Create Sub-Interface**.
- From the Workload VPN intent's view menu, select **Sub-Interfaces** and click **+ CREATE A SUB-INTERFACE**.

**Step 2.** Provide an optional description for the sub-interface.

**Step 3.** Optional: Configure ACL settings.

Specify existing ACL profiles for the following parameters:

- **Ingress ACL Profile IPV4**
- **Ingress ACL Profile IPv6**
- **Egress ACL Profile IPV4**
- **Egress ACL Profile IPv6**

**Step 4.** For routed and loopback sub-interfaces, specify a gateway.

- a. In the **IP Gateway (V4/V6)** section, click **+ADD**.
- b. In the **IP Anycast Gateway** form, enter an IP address. The interface you select here can be a LAG, if the LAG has already been provisioned.
- c. Set the **Anycast Gateway MAC address**– field. This option is available if an IP anycast gateway is set.

- d. If the IP address is the primary gateway, set the **Primary** field.
- Step 5.** Optional: If the interface is for a routed subnet, enable layer 3 proxy ARP and proxy ND settings.
- **L3 ProxyArp Enabled**  
Enabling L3 IPv4 proxy ARP also enables the following parameters; when L3 IPv4 proxy ARP is disabled, you can enable them independently:
    - **IPv4 Learn Unsolicited ARP Enabled**
    - **IPv4 Host Route Enabled**
  - **L3 ProxyND Enabled**  
Enabling L3 IPv6 proxy ND also enables **IPv6 Learn Unsolicited ARP Enabled**; when L3 IPv6 proxy ND is disabled, you can enable it independently.
- Step 6.** In the **Association Type** drop-down list, specify the type of association.
- To select sub-interfaces by label, select **Interface Label Selector** and go to step 7.
  - To select sub-interfaces by selecting individual nodes and ports, select **Node and Interface**, then go to step 8.
- Step 7.** In the **Associations** panel, select **Interface Label Selector**.
- a. In the **Interface Label Selector** field, click  to open the **Label Picker** form.
  - b. From the list of labels, locate the "Edge-Link" label you created previously to identify the edge link ports. Click  on the left end of the row beside the label.
  - c. Click **SELECT** to close the **Label Picker** form.
  - d. Repeat sub-steps 7.a through 7.c until you have selected all of the intended sub-interfaces.
  - e. Go to step 9.
- Step 8.** In the **Association** pane, select the node ID and interface.
- a. In the **Node ID** field, select a node ID associated with a leaf node.  
You must select a leaf node here, because only leaf nodes possess the edge link connections required by the eventual workload.
  - b. In the **Interface Name** field, select an interface to identify a specific interface on the selected node.
  - c. If the subnet is a loopback subnet, select a loopback interface from the manual topology fabric shown.
- Step 9.** Optional: For bridged subnets, if MAC duplication detection is enabled for the subnet to which this sub-interface belongs, set the **Action** field.
- Step 10.** Optional: Assign QoS profiles.  
Specify a profile for the following fields:
- **Qos DSCP Classifier**
  - **Qos DSCP Rewrite Rules**
- Step 11.** Click **CREATE**.
- Step 12.** In the view drop-down list, click **Workload Design**.
- Step 13.** Click  to save the latest change to the workload design.
- Step 14.** Click  **GENERATE WORKLOAD**.

**Expected outcome**

The system generates configuration data for the nodes involved in the workload VPN intent and advances the workload state to Configuration Generated. The workload version remains 1.0.

**Related topics**

[Assigning labels to a workload VPN intent](#)

[Sub-interface parameters](#)

## 6.5.5 Creating a router

**About this task**

Use this procedure to create a router.

**Procedure**

- Step 1.** From the main menu , select **Workload VPN Intents**.
- Step 2.** In the view drop-down list, select **Routers**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the router.
- Step 4.** Click **+CREATE WORKLOAD ROUTER**.
- Step 5.** In the **General** pane, set the following parameters:
  - **Name**
  - **Description**
- Step 6.** In the **Router Definition** pane, accept the default settings or set the following parameters:
  - **VNI**
  - **Provision Type**
  - If **Provision Type** is set to **Manual**, configure the following parameters:
    - **Import Route Target**
    - **Export Route Target**

**Related topics**

[Adding subnets to the workload VPN intent](#)

[Routers](#)

[Router parameters](#)

## 6.5.6 Routing

BGP, static routes, and aggregate routes are configured within the routing section a workload intent.

**BGP**

BGP is an inter-AS routing protocol. An AS is a network or a group of routers logically organized and controlled by common network administration. BGP enables routers to exchange network reachability information, including information about other AS that traffic must traverse to reach other routers in another AS.

When you use BGP as the provider edge (PE) or customer edge (CE) routing protocol, you configure external peering between the provider's AS and the customer network AS.

When you create eBGP links between leaf nodes and customer autonomous systems, the customer autonomous systems may learn of routes through the fabric from different sources. The eBGP links created with the Fabric Services System are configured so that a customer AS prefers the route it learns from its local peer, because that is likely the most efficient path. This setting is achieved using the BGP Local Preference attribute, which the Fabric Services System sets to a value of 130 for links between peers (while other links generally have a preference value of 100). This behavior is automatic and is not configurable.

## Static routes

The Fabric Services System supports static routes to next-hop addresses.

A static route is made up of two parts:

- one or more next-hop groups
- routes that reference the next-hop group

To configure a static route within a workload intent, configure at least one next-hop group containing a next hop, then configure one or more routes. Different routes can reference the same or different next-hop groups.

## Aggregate routes

A BGP aggregate route is a configured route that combines a set of routes into a single route.

### 6.5.6.1 Displaying the routing view for a node

#### About this task

Use this procedure to display the active routing protocols configured for a node. You can view BGP settings (global, next-hop groups, and neighbors), static routes, and aggregate routes configured for the node. This view is read-only.

#### Procedure

- Step 1.** From the main menu , select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Double-click a workload intent, then from the view drop-down list, select **Routing**.
- Step 4.** In the **Protocols** drop-down list, select **Active Protocols by Node**.
- Step 5.** Double-click a node to display active routing protocols for that node.  
You can filter the configured settings per router in the **Router Name** field. Routers that have been created and deployed are displayed in this field.

## 6.5.6.2 BGP parameters

Table 31: Global BGP parameters

Parameter	Description	Values
<b>Global BGP Name</b>	Specifies the name for this global BGP configuration.	String
<b>Local AS</b>	Specifies the global BGP local AS.	Integer
<b>Import Policy</b>	Specifies the name of a BGP policy to use as an import policy.	Name of an existing BGP policy
<b>Export Policy</b>	Specifies the name of a BGP policy to use as an export policy.	Name of an existing BGP policy
Node and router assignment parameters		
<b>Node Name</b>	Specifies the node on which to apply this global BGP configuration.	An existing node or the name of a node that does not yet exist
<b>Router Name</b>	Specifies the router on which to apply this global BGP configuration.	An existing router or the name of a router that does not yet exist
<b>Router ID</b>	Specifies the router ID for the router.	—

## BGP group configuration

Table 32: BGP basic group properties

Parameter	Description	Values/Range
<b>Group Name</b>	Specifies the name of the BGP group. The system creates a default group, <code>default-bgp-group</code> .	String
<b>BFD</b>	Enables or disables bidirectional forwarding on the BGP sessions established by neighbors that belong to this group.	Default: enabled
<b>Connect-Retry</b>	Specifies the duration of the connect-retry timer.	Default: 120
<b>Peer AS</b>	Specifies the peer AS to use for any neighbor that belongs to this group (does not override at the neighbor level).	Default: 1
<b>Local AS</b>	Specifies a local AS to use for any neighbor that belongs to this group. This setting (does not override at the neighbor level). By default, the global BGP configuration local AS is used by all peers that belong to this group.	Default: 1

Parameter	Description	Values/Range
<b>Prepend Global AS</b>	Specifies whether to prepend the global AS value to the AS path of inbound routes from each eBGP peer that belongs to the group.	Default: disabled
<b>Prepend Local AS</b>	Specifies whether to prepend the local AS value to the AS path of inbound routes from each eBGP peer that belongs to the group.	Default: disabled
<b>Toggle Max Hops</b>	Specifies the number of maximum hops. By default, eBGP sessions have a maximum hop of 1 configured.	1 to 255
<b>IPv4 Unicast</b>	Enables the router to advertise to and receive IPv4 unicast routes from neighbors that belong to this group.	Default: enabled
<b>IPv6 Unicast</b>	Enables the router to advertise to and receive IPv6 unicast routes from neighbors that belong to this group.	Default: enabled
<b>Minimum-Advertisement-Interval</b>	Specifies how long a BGP router waits before sending an advertisement for all neighbors in this group.	Default: 1
<b>Import Policy</b>	Specifies the BGP import policy.	Name of an existing BGP policy
<b>Export Policy</b>	Specifies the BGP export policy.	Name of an existing BGP policy

## BGP neighbor

Table 33: Basic BGP neighbor parameters

Parameter	Description	Values/Range
<b>Peer Address</b>	Specifies the IP address of the BGP peer.	A valid IPv4 or IPv6 address
<b>Local Address</b>	Specifies the local address to use for this peering session.	A valid IPv4 or IPv6 address
<b>Group Name</b>	Specifies the name of this neighbor group.	String
<b>Override Peer AS</b>	Specifies the peer AS to use for this peering session. This value overrides the default peer AS value configured in the main or group BGP configuration used by all peers that belong to this group.	Default: disabled
<b>Override Local AS</b>	Specifies the local AS to use for this peering session. This value overrides the local AS setting in the main BGP configuration is used by all peers that belong to this group. When this parameter is enabled, you can optionally prepend the global AS and the local AS.	Local AS

Parameter	Description	Values/Range
<b>Toggle Max Hops</b>	Specifies the maximum number of hops for a BGP session.	1 to 255
<b>Override IPv4 Unicast</b>	Specifies whether IPv4 unicast routes are advertised to and received from neighbors that belong to this group.	Default: disabled
<b>Override IPv6 Unicast</b>	Specifies whether IPv6 unicast routes are advertised to and received from neighbors that belong to this group. This setting overrides any configuration at the group or global level.	Default: disabled
<b>Import Policy</b>	Specifies the name of a BGP policy to use as an import policy.	Name of an existing BGP policy
<b>Export Policy</b>	Specifies the name of a BGP policy to use as an export policy.	Name of an existing BGP policy
Node and router assignment parameters		
<b>Node Name</b>	Specifies the node on which to apply the basic neighbor properties.	An existing node or the name of a node that does not yet exist
<b>Router Name</b>	Specifies the router on which to apply the basic neighbor properties.	An existing routers or the name of a router that does not yet exist
<b>Inherit</b>	Specifies that this neighbor inherits the basic settings from the template configured for the BGP neighbor group. To modify the basic property settings, disable this parameter.	Default: enabled

### 6.5.6.3 Configuring BGP

#### About this task

You configure BGP from within a workload intent. If you have not yet deployed the workload intent, you can freely modify its design. If the you have already deployed the workload VPN, create a new candidate version of the workload VPN.

At a high level, BGP configuration includes the following tasks:

1. Configure the global BGP settings.  
This initial configuration is the template that you can apply to a specific node and router instance. The node and router do not need to exist in the workload intent when you assign the global BGP template, which means that you can pre-provision nodes by reapplying this global template to different unique node and router instances.
2. Configure a BGP group.

When you create a workload intent, the system creates a default BGP group for the workload, `default-bgp-group`. You can create additional BGP groups as needed and apply the group to multiple nodes as needed.

### 3. Configure BGP neighbors.

Just as with the creation of global BGP settings, you configure basic settings to create a template for a BGP neighbor. Then, you can assign this template to different unique node and router instances. If you want to create additional neighbors, but want change some of the settings in the template, during node assignment, disable the **Inherit** flag.



#### Note:

Release 23.8.1 includes the following high-level changes to BGP configuration:

- In Release 23.4.1, BGP settings were configured per node; in Release 23.8.1, BGP global, group, and neighbor settings are configured per workload. This change allows for the pre-provisioning of nodes.
- In Release 23.4.1, a single router ID was applied to all routing instances; in Release 23.8.1, you can assign a routing ID for each routing instance.
- In the Release 23.8.1 Fabric Services System UI, global BGP routes created in Release 23.4.1 are identified in the **Routing** view, **Node Assignment** pane with an asterisk under the **Router Name** column. The router ID shown is applied to all routing instances. You can change this routing ID and the provide a router name as needed.
- The **BGP- Group** view of a workload intent includes the **Config-name** column which displays the name of a deployed route deployed for the node in a previous release.

## Procedure

**Step 1.** From the main menu , select **Workload VPN Intents**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.

**Step 3.** Double-click a workload intent and from the view drop-down list, select **Routing**.

**Step 4.** Configure global BGP settings.

a. In the **Protocols** drop-down list, click **BGP**, then select **BGP- Global**.

b. Click **+ CREATE GLOBAL**.

c. In the **Basic Properties** pane, provide the global BGP name, local AS number, and optional import and export policies.

Global import and export BGP policies are optional. You can also specify import and export policies at the BGP group or BGP neighbor level to override the settings at the global or group level. The system does not check the validity of the policy names that you specify; the BGP policies are assumed to be configured on the node using the global configuration override feature or some other mechanism.

d. In the **Node Assignment** pane, click **+ ASSIGN NODES**.

Set the node name, router name, and router ID.

e. Optional: Assign the global configuration to another node and routing instance.

The node name and router name combination must be unique.

f. Click **CREATE**.

g. Optional: Create another global BGP template.

Repeat steps 4.a through 4.f.

**Step 5.** Create a BGP group.

- a. In the **Protocols** drop-down list, click **BGP**, then select **BGP- Group**.
- b. Click **+ CREATE BGP GROUP**.
- c. Set parameters for the BGP group.  
Set the appropriate parameters for your deployment scenario.
- d. Click **CREATE**.

**Step 6.** Create one or more BGP neighbors.

- a. From the **Protocols** drop-down list, click **BGP**, then select **BGP- neighbor**.
- b. Click **+ CREATE BGP NEIGHBOR**.
- c. Configure the basic properties for a BGP neighbor.  
Set the following parameters as needed in the **Basic Properties** pane:
- d. Assign the basic properties of the BGP neighbor to a node and router instance.  
In the **Node Assignment** pane, click **+ ASSIGN NODES**.
  - Specify the node name and router name.
  - To modify any of the basic property settings for this neighbor, disable the **Inherit** parameter.
  - When you are finished, click **Save**.
- e. Assign the basic properties of the BGP neighbor to another node and router instance as needed.
- f. Click **CREATE**.

**Step 7.** Update the workload VPN intent.

- a. On the **Workload VPN Intents** page, click the view drop-down list and select **Workload Design**.
- b. Click **GENERATE WORKLOAD**.

#### Related topics

[BGP parameters](#)

[Viewing a workload VPN intent as code](#)

[Displaying the routing view for a node](#)

[Creating a new version of a workload VPN intent](#)

## 6.5.6.4 Static route parameters

### Next Hop Group instance settings

Table 34: Next-hop group parameters

Field Name	Description	Value
<b>Next Hop Group Name</b>	Specifies the name of the next-hop group.	String

Field Name	Description	Value
<b>Description</b>	Describes the next-hop group.	String
<b>Blackhole</b>	Specifies that next-hop cannot be created for this next-hop group.	Default: disabled
<b>Generate-icmp</b>	If the <b>Blackhole</b> parameter is enabled, specifies whether the router generates ICMP messages for dropped packets.	Default: disabled
<b>Resolve</b>	When this parameter is enabled, SR Linux can use a resolved next-hop instead of a directly connected next-hop. In SR Linux, this setting is configured per next-hop; in the Fabric Services System, this setting applies at a group level and is applied to every next hop within the group.	Default: disabled
<b>BFD</b>	Enables BFD on each of the next-hop instances within the group. In SR Linux, this setting is configured per next-hop; in the Fabric Services System, this setting applies at a group level and is applied to every next hop within the group.	Default: disabled
<b>BFD Local Address</b>	Specifies a BFD local address to use for the BFD session, If BFD is enabled. This setting applies to each next-hop instance within the next-hop group. If an IPv4 address is specified, it is applied to any next hop that is configured with an IPv4 address. Similarly, if an IPv6 address is specified, it is applied to any next hop that is configured with an IPv6 address.	IPv4 or IPv6 address, an existing gateway IP addresses within the 'router' (ip-vrf)
<p>The <b>Next Hops</b> pane contains the parameters that configure a next-hop instance for the next-hop group. Each next-hop instance is defined by the following of parameters.</p>		
<b>Index</b>	Specifies a number associated with this next-hop group.	Integer
<b>IP address</b>	Specifies the IP address for this next-hop group.	IPv4 or IPv6
<b>Admin State</b>	Specifies whether this next hop-group instance is enabled or disabled.	Default: disabled
<b>Inherit</b>	By default, the settings are inherited from the basic parameters. If disabled, specifies some settings can be modified.	Default: enabled

## Static Route Details parameters

Table 35: Basic properties

Parameter	Description	Value
<b>Name</b>	Specifies the name of the static route.	String
<b>Description</b>	Provides an optional description for the route.	String
<b>Prefix</b>	Specifies the prefix of a subnet for this static route.	IPv4 or IPv6 format
<b>Next Hop Group</b>	Specifies the name of an existing next-hop group.	String
<b>Preference</b>	Specifies the preference for this static route.	Default: 5

Table 36: Node assignment parameters

Parameter	Description	Value
<b>Node Name</b>	Specifies the node on which to apply this static route.	An existing node or the name of a node that does not yet exist
<b>Router Name</b>	Specifies the router on which to add the static route.	An existing router or the name of a router that does not yet exist
<b>Inherit</b>	By default, specifies that the settings for this route are inherited from the basic parameters configured for the template. To modify the basic property settings, disable this parameter.	Default: enabled

### Related topics

[Configuring static routes](#)

## 6.5.6.5 Configuring static routes

### About this task

Static routes are configured within a workload intent. If you have already deployed the workload VPN intent, create a new candidate version of the workload VPN intent before performing this procedure.

To configure a static route, configure at least one next-hop group. Then, create a static route that references that next-hop group. A next-hop group is not deployed unless it is referenced in a route.

### Procedure

**Step 1.** From the main menu , select **Workload VPN Intents**.

- 
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Double-click a workload intent, then from the view drop-down list, select **Routing**.
- Step 4.** Create one or more next-hop groups.
- In the **Protocols** drop-down list, highlight **Static Routing**, then click **Static Routing - Next Hop Groups**.
  - Click **+ CREATE NEXTHOPGROUP**.
  - Configure basic parameters for the next-hop group.  
This step configures that template for the next hops. Set the following parameters:
    - Next Hop Group Name**
    - Description**
    - Blackhole**
    - Generate -icmp**
    - Resolve**
    - If BFD is enabled, set a BFD local address
- Step 5.** Create the next hops for this next-hop group.
- In the **Next Hops** pane, click **+ CREATE**.
  - Set the index, IP address, and admin state for this next-hop instance.
  - To change some settings for this next-hop instance, disable the **Inherit** parameter.  
You can enable or disable BFD, specify a BFD local address. You can also configure the BFD Local Discriminator and BFD Remote Discriminator.
  - Click **SAVE**.
  - Optional: Create additional next hops as needed.  
Repeat steps [5.a](#) through [5.d](#).
- Step 6.** Configure a static route.
- In the **Protocols** drop-down list, highlight **Static Routing**, then click **Static Routing - Static Routes**.
  - Configure the basic properties for the static route.  
In the **Basic Properties** pane:
    - Provide a name and an optional description for the route.
    - Provide a prefix for the subnet.
    - Select an existing next-hop for this static route.
    - Set the preference for the route.
- Step 7.** Assign the static route to a node and router instance.
- In the **Node Assignment** pane, click **+ ASSIGN NODES**.
  - Specify the node name and router name.
  - To modify any of the basic settings for the static route, disable the **Inherit** parameter and make the changes as needed.

- d. When you are finished, click **Save**.
- e. Optional: Repeat sub-steps 7.a through 7.d to assign the static route to another node and router instance.
- f. Click **CREATE**.

**Step 8.** Update the workload VPN intent.

- a. On the **Workload VPN Intents** page, click the view drop-down list and select **Workload Design**.
- b. Click **GENERATE WORKLOAD**.

#### Related topics

[Viewing a workload VPN intent as code](#)

[Displaying the routing view for a node](#)

[Creating a new version of a workload VPN intent](#)

### 6.5.6.6 Aggregate route parameters

Table 37: Basic properties

Parameter	Description	Value
<b>Name</b>	Specifies the name of the aggregate route.	String
<b>Description</b>	Provides an optional description for the route.	String
<b>Prefix</b>	Specifies the prefix of a subnet for this route.	IPv4 or IPv6 format
<b>Aggregator Address</b>	Specifies the IP address of the aggregator route.	IPv4 or IPv6 format
<b>Aggregator AS Number</b>	Specifies the AS number for this route.	Integer
<b>Summary Only</b>	Specifies that the activation of an aggregate route automatically blocks the advertisement of all of its contributing routes by BGP.	Default: disabled
<b>Generate -icmp</b>	If enabled, specifies that the router generates ICMP unreachable messages for the dropped packets.	Default: disabled

Table 38: Node assignment parameters

Parameter	Description	Value
<b>Node Name</b>	Specifies the node on which to apply this aggregate route.	An existing node or the name of a node that does not yet exist

Parameter	Description	Value
<b>Router Name</b>	Specifies the router on which to add the aggregate route.	An existing router or the name of a router that does not yet exist
<b>Inherit</b>	Specifies that settings are inherited from the configured basic parameters template for the aggregate route. To modify the basic property settings, disable this parameter.	Default: enabled

#### Related topics

[Configuring aggregate routes](#)

### 6.5.6.7 Configuring aggregate routes

#### About this task

You configure aggregate routes within a workload intent. If you have already deployed the workload VPN intent, create a new candidate version of the workload VPN intent before performing this procedure. To configure aggregate routes, first configure basic parameters for the an aggregate route. This initial configuration is the template that you can apply to a specific node and router instance. By default, aggregate routes are applied to any node on with an attached. You can also apply the aggregate route only to a specific node or set of nodes.

#### Procedure

- Step 1.** From the main menu , select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Double-click a workload intent, then from the view drop-down list, then select **Routing**.
- Step 4.** Create an aggregate route.
  - a. In the **Protocols** drop-down list, highlight **Aggregate**.
  - b. Click **+ CREATE AGGREGATE**
  - c. Configure the basic properties for the aggregate route.
    - Provide a name and an optional description for the aggregate route.
    - Provide a prefix.
    - Provide an aggregator address and aggregator AS number.
    - Specify if the route is summary only.
    - Specify if the router generates ICMP unreachable messages.
  - d. Assign the aggregate route to a node and router instance.
 

In the **Node Assignment** pane, click **+ ASSIGN NODES**. Specify the node name and router name. To modify any of the basic property settings for this static route, disable the **Inherit** parameter and modify the settings as needed. When you are finished, click **Save**.
  - e. Optional: Assign the aggregate route settings to another node and routing instance.

Repeat steps 4.a through 4.d. The node name and router name combination must be unique within a workload intent.

**Step 5.** Click **CREATE**.

**Step 6.** Update the workload VPN intent.

- a. On the **Workload VPN Intents** page, click the view drop-down list and select **Workload Design**.
- b. Click **GENERATE WORKLOAD**.

#### Related topics

[Aggregate route parameters](#)

[Viewing a workload VPN intent as code](#)

[Displaying the routing view for a node](#)

[Creating a new version of a workload VPN intent](#)

## 6.6 Workload VPN intent deployment

Deploying a workload VPN intent creates a functioning instance of the workload VPN intent as an overlay to your fabric.

Before you can deploy your workload VPN intent, you must have saved the workload VPN intent and generated its configuration.

Deploying the workload VPN intent involves the following procedures:

1. [Adding a workload VPN intent to the deployment pipeline](#)
2. [Deploying a workload VPN intent from the deployment pipeline](#)

The system also supports the auto-deployment of workload intents. For example, you can use the Fabric Services System REST API to create workload intents that are automatically deployed, without having to add them to the deployment pipeline and then deploying them from the pipeline queue.

### System checks before workload intent deployment

The following circumstances can prevent you from deploying a workload VPN intent:

- If any of the nodes within the associated fabric intent are unavailable (that is, not in a Ready state), you cannot deploy the workload VPN intent.  
In such a case, you must correct the node state. When all nodes are back in a Ready state, you can proceed with the deployment of your workload VPN intent.
- If any of the nodes that belong to a workload VPN intent are already under deployment by another workload VPN intent, you must wait until the deployment of the previous workload VPN intent has completed.
- During the deployment of a workload intent, although the system checks whether the referenced interfaces are in the underlay intent, ensure that you do not remove interfaces that are being used in a workload intent from the underlay interfaces.

### Automatic deployment of global profiles referenced in a workload intent

For workload VPN intents that reference a global profile (ACL or QoS profile), after you generate the workload intent configuration and deploy the workload intent, the system:

- checks if a new version of the workload intent is required and, if needed, creates one
- adds the global profile associated with the workload intent to the deployment pipeline and deploys it
- adds the workload intent to the deployment pipeline

## 6.6.1 Adding a workload VPN intent to the deployment pipeline

### Prerequisites

You must have saved the workload VPN intent and generated its configuration.

### Procedure

- Step 1.** Click  to open the main menu and select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Find the workload VPN intent that you want to deploy from the displayed list and click  at the end of its row.
- Step 4.** Select **Open** from the drop-down list.
- Step 5.** Click  to deploy.
- Step 6.** Click **ADD TO PIPELINE**.

### Expected outcome

The system adds the workload VPN intent to the deployment queue for the region and updates the status of the workload VPN intent to Queued for deployment.

### Related topics

[Viewing a workload VPN intent](#)

## 6.6.2 Deploying a workload VPN intent from the deployment pipeline

### About this task

After you add a workload VPN intent to the deployment pipeline, it remains there until you tell the system to proceed with the deployment.

### Procedure

- Step 1.** Click  to open the main menu and select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Find the workload VPN intent that you want to deploy, then click  at the end of its row.
- Step 4.** Select **Deployment Pipeline** from the actions list.
- Step 5.** Find the workload VPN intent in the deployment pipeline list and click  at the end of its row.
- Step 6.** From the resulting actions list, select **Deploy**.  
You can view the progress of the software update deployment from the **Deployment Pipeline** page.

### Expected outcome

If deployment fails, the failure is reported as follows:

- queue status: reports Error with detailed status reason
- fabric intent: reports Deployed and shows a new entry in the Event log showing that the Workload deployment failed
- workload VPN intent: reports Failed

In the workload VPN intent Design view, the system also highlights deployment issues in the status bar by adding a red circle to the fabrics affected by the deployment error and with entries in the events log.

## 6.7 Workload VPN intent modification

If you have not yet deployed a workload VPN intent, you can freely modify its design.

If you have already deployed a workload VPN intent, you must create a new candidate version of the workload VPN intent before you can modify it. When you have updated the new design, you can deploy it to replace the previously deployed version.

You can also delete a workload VPN intent.

### Related topics

[Editing a workload VPN intent](#)

[Creating a new version of a workload VPN intent](#)

[Deleting a workload VPN intent](#)

### 6.7.1 Editing a workload VPN intent

#### Procedure

- Step 1.** Click the main menu  and select **Workload VPN intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.
- Step 3.** Double-click the saved workload VPN intent that you want to edit.
- Step 4.** From the view drop-down list, select the element that you want to edit.  
You can edit the following settings for a workload intent:
  - Basic parameters
  - Participating fabric intents
  - Subnets and their attributes
  - Sub-interfaces and their attributes
  - Routers and their attributes
  - Routing attributes for BGP, static and aggregate routes
- Step 5.** When you are finished editing parameters, click  to save the new configuration.

**Step 6.** Click  **GENERATE WORKLOAD.**

**Expected outcome**

The system generates updated configuration data for the nodes involved in the workload VPN intent and advances the workload State to Configuration Generated. The workload version increments.

## 6.7.2 Creating a new version of a workload VPN intent

### About this task

Follow this procedure to create a new candidate version of an already-deployed workload VPN intent.

### Procedure

**Step 1.** Click the main menu  and select **Workload VPN intents.**

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing a workload VPN intent.

**Step 3.** Double-click a saved workload VPN intent to open it.

**Step 4.** Click the  menu at the upper right of the page.

**Step 5.** Click **Create Candidate Version** from the drop-down list.

**Expected outcome**

The system creates a new, editable version of the current workload VPN intent, and the version number in the left panel is incremented to the next available number. You can now edit the workload VPN intent's parameters and save the result without affecting the original version.

**Step 6.** When you are satisfied with the new design, click **Generate** to generate a new workload VPN intent diagram and its associated configuration code.

**Step 7.** At any time before deploying the new version, you can discard it and revert to the previously deployed version by doing the following:

- a. Click the **More actions** icon (  ) in the **Workload Design** view.
- b. Select **Discard Changes.**
- c. Click **OK.**

**Expected outcome**

The system discards the changes and reverts the display to the previous version of the workload VPN intent. If you discarded changes for the initial version of a workload VPN intent that was never deployed, the workload VPN intent is deleted entirely.

## 6.7.3 Updating a group of sub-interfaces by label reference

### About this task

If you want to update information for a group of sub-interfaces in a workload VPN intent, you can make bulk updates to items assigned the same label. For example, when a series of edge link ports are identified with the same "Edge-Link" key type label, you can update the description field for each of the participating ports in one action, instead of updating the description field for each port individually.



**Note:** Do not delete auto-generated interfaces manually. Instead, use label manager to remove the label from the sub-interface; for instructions, see [Removing a label assigned to a specific sub-interface](#).

To update a group of sub-interfaces:

### Procedure

**Step 1.** Open the workload VPN intent in which the "Edge-Link" label is applied to the sub-interfaces.

**Step 2.** In the view drop-down list, click **Sub-interfaces**.

**Step 3.** Click  **EDIT BY LABEL**.

**Step 4.** In the form, select  to specify the label to edit. The **Label Picker** form opens.

**Step 5.** In the **Label Picker**, do the following:

- a. Locate the "Edge-Link" label you created to identify the set of edge link ports in question.
- b. Click the  from the left end of the row beside the label.
- c. Click **SELECT** to close the **Label Picker** form. You return to the **Edit Sub-Interfaces By Label** form.

**Step 6.** Select the VLAN ID to match the ID associated with the labels you intend to edit.

**Step 7.** Click **SELECT** to close the **Edit Sub-Interfaces By Label** form.

**Step 8.** In the **Basic Properties** header, enter new information in one or both of the following fields:

- Description
- ACL Profile

**Step 9.** Click **SAVE**.

#### Expected outcome

The edits are saved as a bulk update to the participating sub-interfaces. The **Edit Sub-Interfaces** overlay closes, returning you to the **Sub-Interfaces** page.

**Step 10.** Optional: To confirm the updates, do the following from the **Sub-Interfaces** page:

- a. For a specific row item that was subject to the bulk edit, click the  options menu at the right end of the row.
- b. Select **Open**.

#### Expected outcome

The **Edit Sub-Interface** overlay appears. The edits are visible in the **Description** and **ACL Profile** fields.

- c. Click **CANCEL** to exit without making any further edits.

## 6.8 Deleting a workload VPN intent

### Procedure

**Step 1.** From the main menu, select **Workload VPN Intents**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing the workload VPN intent.

**Step 3.** Find the workload VPN intent from the list, then, click  at the end of its row.

**Step 4.** Select **Delete** from the action list and click **OK** in the confirmation form.

**Expected outcome**

The workload moves to the Delete Config In Queue state. This means that the deletion workload has been added to the region's deployment pipeline, but has not yet taken place.

**Step 5.** Go to the region's deployment pipeline by clicking on the  icon and selecting **Deployment Pipeline** from the resulting menu.

**Step 6.** From the deployment pipeline, deploy the workload.

**Expected outcome**

If deployment of the workload deletion fails, you must redeploy the underlay intent. The failed attempt is reported as follows:

- The **Deployment Pipeline** screen reports Error and the **Status Reason** field displays the reason for the failure.
- In the **Event Log** screen, a message is reported similar to the following:

```
Message: Delete Workload ww1 Deployment Failed. Underlay Intent needs to be redeployed.
```

- In the **Workload VPN Intents** screen, the workload VPN intent for the deletion (created in step 4) is deleted.
- In the **Fabric intents** view, the **Status** field for the workload VPN intent for the deletion shows Deployed.

## 7 Configuration overrides

When you create a fabric intent or workload intent, the Fabric Services System prepares a set of configuration files for all of the affected nodes.

A configuration override allows you to specify a set of changes to the configuration files for one or more nodes. The Fabric Services System supports the following types of configuration overrides:

- Global configuration overrides, which modify one or more node configurations and are associated with a fabric intent
- Contextual configuration overrides, which alter one or more sub-interface or router configurations and are associated with a workload intent

Configuration overrides are strictly additive; they allow you to insert new or modify existing configuration data, but they cannot be used to remove any of the configuration data that the system creates based on the original fabric intent inputs.



**Note:** All configuration overrides, whether global or contextual overrides, are specific to the region in which they are created. An override that is created within one region is not visible within, or available to, other regions.

### Order of applying overrides

A node could be the subject of several different overrides: global configuration overrides, contextual configuration overrides, and deviation-associated overrides.

To ensure that the result of any combination of overrides is predictable, the Fabric Services System applies overrides in a specific sequence to arrive at the final configuration for a particular node:

1. The system begins with the normalized configuration for the node, based on the combination of a specific version of a fabric intent plus any workload intents that affect the node.
2. Then the system applies any global configuration overrides that affect the node.
3. Then the system applies any contextual configuration overrides that affect the node.
4. Finally the system applies any deviation-associated configuration overrides that affect the node.

#### Related topics

[Deviations](#)

[Global configuration overrides](#)

[Contextual configuration overrides](#)

### 7.1 Global configuration overrides

When you create a fabric intent, the Fabric Services System prepares a set of configuration files for all of the nodes participating in the fabric. The configuration for each of these nodes is based directly on the parameters you provide when creating the fabric intent. These are the "normalized" configurations for the participating nodes.

A global configuration override (GCO) allows you to specify a set of changes to the configuration files for one or more nodes within a fabric. GCOs allow you to:

- insert new configuration data
- modify existing configuration data
- remove existing configuration data (using the "Delete Paths" capability).

As part of the GCO, you must provide the JSON configuration data to be added (and/or delete paths to be deleted) to each configuration. You then specify the nodes within the fabric to which the overrides should be applied in one of the following ways:

- by selecting individual nodes from the inventory
- by applying a label to the nodes you want modified, and then selecting that label as the override target

The resulting change to the configuration is applied to the specified nodes the next time you deploy a fabric intent that includes any of those nodes. If you create a GCO during the initial fabric intent creation, you can proceed with deployment normally and the initial version of the fabric intent includes the overrides.

To apply the GCO to nodes within an existing fabric intent, you must first create a new candidate version of the fabric intent. You can then proceed with deployment of the new version normally, which includes any changes specified in the configuration override for the nodes within that fabric intent.

GCOs are not specific to individual fabrics; they can affect multiple nodes that have been either selected manually or identified by a shared label, and these nodes could belong to multiple fabrics. For the node configuration changes to take full effect, you must re-deploy any already-deployed fabrics that include the affected nodes.

GCOs themselves can have multiple versions, although only one version of a particular override can be active at one time.

### Applying multiple overrides

You can apply multiple GCOs to the same fabric intent, each with its own configuration changes and target nodes.

Each GCO you create includes a number indicating its execution order. When multiple GCOs are applied to a fabric intent, the overrides are applied to the participating nodes' configuration data based on the execution order value, from lowest to highest. This ensures that the overrides interact in a consistent and predictable way.

### Deploying fabric intents with overrides

Like any fabric intent deployment, the deployment of fabric intents that include GCOs proceeds as a single transaction. If the deployment of the modified fabric intent fails for any node, the entire deployment is rolled back and any modified nodes are restored to their previous states.

The reason for any failure is recorded in the Fabric Services System Events Log.

### Fabric intent displays and configuration overrides

When viewing a fabric intent in the Fabric Services System GUI, the system displays some parameters pertaining generally to the nodes within the fabric intent. These values always reflect the normalized configuration within the fabric intent, and do not reflect any configuration overrides that may have been applied to the nodes.

For example, the Fabric Intents design page displays values for FTP settings near the bottom of the left column. This reflects the FTP settings that are specified when creating the fabric intent, and so may continue to display FTP as disabled across the fabric even though a configuration override subsequently added data to enable FTP on some of the nodes.

### Deviation-related global configuration overrides

A deviation is a node configuration that originates outside the Fabric Services System, and so represents a change to the current configuration for that node that is stored in the related fabric intent. The Fabric Services System continues to monitor node configurations after fabric intents have been deployed; and when the node configuration is modified externally, the system reports this as a deviation.

You can accept or reject deviations that have been detected by the Fabric Services System. Deviations you reject are discarded, but deviations you accept are stored in the Fabric Services System as a special type of system-generated GCO.

Such deviation-related GCOs are automatically deployed to the affected nodes. The GCO parameters, and the modified configuration data that is part of the GCO, are set by the system and cannot be modified. Storing the deviation as a GCO ensures that the deviation-based configuration information is stored and managed in a manner consistent with other configuration exceptions that are created by the Fabric Services System user.

Deviation-related GCOs are always the last changes applied to the normalized node configuration (after standard global configuration overrides and contextual configuration overrides) to arrive at the final configuration for a node.

If you delete a node that is the subject of a deviation-related GCO, the GCO itself is deleted from the system.

### System global configuration overrides for unmanaged nodes

When a fabric consists of nodes that are not directly managed by the Fabric Services System, the complete set of configuration data for each node is stored within the Fabric Services System as a system-generated Global Configuration Override (system GCO).

Typically GCOs are used to store expected variations in a node's configuration. But for unmanaged nodes, a GCO is used to store the entire node configuration, where it is available for consultation by components of the Fabric Services System. For example, it is from this configuration data that the Fabric Services System creates initial configuration files as part of a maintenance intent.

System GCOs are read-only. However, the Fabric Services System supports the following capabilities to allow you to manage system GCOs and the configuration data they contain:

- you can duplicate a system GCO to a conventional GCO if required
- you can also delete a system GCO

Duplicating a read-only system GCO as a conventional GCO allows you to modify the configuration data as required. Generally you would then delete the original system GCO that was converted, to avoid duplicate configurations for a single node.



**Note:** Because system GCOs are critical for managing the data associated with unmanaged nodes, use caution when deleting system GCOs

#### Related topics

[Deviations](#)

[Viewing and managing configuration overrides](#)

## 7.1.1 Global configuration override parameters

Table 39: Global configuration override parameters

Parameter	Default value	Description
<b>Version</b>	See Description	Indicates the version number for this override. This value is assigned automatically. The first instance of any configuration override is Version 1.0. Subsequent versions of the same override increment this value.
<b>Active</b>	Enabled	Indicates which, if any, version of a configuration override should be used to modify the configuration data for the target nodes. A maximum of one version of any configuration override can be active at one time. It is possible to set all versions of an override to inactive.
<b>Automatic Deployment</b>	Disabled	Specifies whether the override should be automatically pushed down to the affected nodes. <ul style="list-style-type: none"> <li>• If enabled, the system automatically deploys the new configuration to the affected nodes.</li> <li>• If disabled, you must create and deploy a new version of the affected fabric intent to apply the new configuration to the affected nodes.</li> </ul>
<b>Name</b>	None	Identifies the configuration override and is used to identify it in the list of all overrides within the system. It must be unique (but is shared by all versions of the same override).
<b>Description</b>	None	Describes the purpose or impact of the override.
<b>Execution Order</b>	n+100	Indicates where this particular override should fall within the overall execution sequence when there are multiple overrides.  If two or more overrides target the same nodes, the execution order value dictates the order in which the configuration changes are merged into the configurations of the target nodes.  The override with the lowest Execution Order value is applied first.  The default value is the current highest value plus 100. As a result, the first override created has a default value of 100.
<b>Method for Selecting Nodes</b>	None	Indicates how to identify nodes to be affected by this override. This can be either of these values: <ul style="list-style-type: none"> <li>• By Label: you must later specify a label that has been applied to the intended target nodes.</li> <li>• By node: you must later select one or more nodes from the inventory to be the target of the override</li> </ul>
<b>Node Label Selector</b>	None	Identifies the label or labels that qualify a node as a candidate for this override.

Parameter	Default value	Description
		If the <b>Method for Selecting Nodes</b> parameter is set to By Label, this control is displayed and is active. You can use this control to specify one or more node labels. These labels must be "Node-Type" labels. When applied, the configuration data in this override is applied to all nodes within the fabric intent with at least one of these labels.
<b>Node Selector</b>	None	Identifies the specific nodes to which this override should be applied.  If the <b>Method for Selecting Nodes</b> parameter is set to By Node, this control is displayed and active. You can use this control to select one or more nodes in the system inventory. When applied, the configuration data in this override is applied to each of the selected nodes within the fabric intent.
<b>Configuration Type</b>	Global Override	This value cannot be modified.
<b>Target Configuration</b>	See Description	Identifies several configuration properties of any node that are expected if the node is a candidate for this override.  It consists of three sub-parameters: <ul style="list-style-type: none"> <li>• <b>Operating System:</b> SRLinux</li> <li>• <b>Software Version:</b> Can be: <ul style="list-style-type: none"> <li>– ALL</li> <li>– a major software version from the software catalog</li> <li>– or both</li> </ul> </li> <li>• <b>Data Model:</b> SRLinux</li> </ul>
<b>Configuration Data</b>	None	Contains the configuration data that is applied to each node that is subject to this configuration override.  Each configuration override must include a set of JSON configuration data to be merged into the configuration file for the target nodes. For example: <pre> {   "index": 1,   "ip-mtu": 5000,   "vlan": {     "encap": {       "single-tagged": {         "vlan-id": "100"       }     }   } } </pre>
<b>Delete Paths</b>	None	Enter any configuration data in this field that should be deleted from the existing configuration.  The delete path syntax follows gNMI path conventions. For example: /system/name/host-name.

Parameter	Default value	Description
		 <b>Note:</b> A single GCO can include JSON configuration data for addition/modification, or for deletion, or a mixture of both.

## 7.1.2 Creating a global configuration override

### Prerequisites

If you are going to specify the target nodes for this override by a "Node-Type" label they share in common, you should already have created and applied that label to the target nodes as described in [Labels](#).

### About this task

This procedure describes how to create the initial version of a global configuration override. At a high level, you create a configuration override by doing the following:

1. Provide a name and description to identify this override.
2. Set some configuration parameters for the override:
  - whether it is active
  - where it falls in the overall execution order for all active overrides
3. Identify one or more software configurations for the target nodes (the operating system, OS version, and data model).
4. Enter the specific node configuration data that should be applied to the target nodes. Each combination of OS, OS version, and data model can have its own unique configuration code to be applied as part of this override.
5. In the "Delete Paths" section of the override form, enter any configuration data that should be deleted from the target nodes.



**Note:** A single GCO can include JSON configuration data for addition/modification, or for deletion, or a mixture of both.

6. Identify the specific nodes to which this override applies by either:
  - selecting individual nodes
  - identifying a label common to the target nodes
7. Save the override.

While the override is active, it is applied to the targeted nodes when you generate configurations for any:

- in-progress fabric intents
- future fabric intents
- new versions of existing fabric intents

### Procedure

**Step 1.** Click the  menu.

**Step 2.** Select **Overrides** to open the **Overrides** page.

**Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the override.



**Note:** You cannot change the region selection after you begin creating the override. If you select a new region in the **Region Selector** while creating an override, the creation form closes and you are returned to the **Overrides** page.

**Step 4.** Click **+CREATE** to open the **Configuration Creation** page.

**Step 5.** Enter values for the following parameters as described in [Global configuration override parameters](#):

- **Active**
- **Automatic Deployment**
- **Name**
- **Description**
- **Execution Order**

**Step 6.** Identify a target configuration by doing the following:

- Click **+ ADD**.
- From the **Target Configuration** form, select the following:
  - **Operating System**
  - **Software Version**
  - **Data Model**
- Click **ADD**.

**Step 7.** In the **Configuration** panel, add the specific configuration data that you want to be merged into the configurations for the targeted nodes.



**Note:** You can enter independent instances of configuration code here for each combination of operating system, version, and data model you selected in step 6. Select each target configuration from the **Target** drop-down to see its unique window in which to enter configuration code.



**Note:** Instead of typing or pasting code in this window, you can use the **More...** button (  ) at the upper right of the form to select a file from which to import this code.

**Step 8.** In the **Delete Paths** panel, enter any configuration that should be deleted from the configurations for the target nodes:

- Click **+ADD** to open the **Add Delete Paths** form.
- Enter the configuration data to be deleted.



**Note:** The delete path syntax follows gNMI path conventions. For example: /system/name/host - name.

- Click **Close**.

#### **Expected outcome**

The configuration for deletion is added to the **Delete Paths** panel.

**Step 9.** For **Method for Selecting Nodes**, do one of the following:

- To specify the target nodes by a label they share, select **By Label** and go to step 10.
- To select one or more nodes directly from the inventory, select **By Node** and go to step 11.

**Step 10.** Select the **Edit** icon () above the **Node Label Selector** panel to open the **Label Picker** form and do the following:

- a. Ensure that you have created a label with the Label Key "Node-type" and applied that label to all of the nodes whose configurations you plan to override.



**Note:** Procedures for creating and applying labels to objects can be found in [Labels](#) and are beyond the scope of this procedure.

- b. Select the label with the Label Key "Node-type" from the displayed list of labels.
- c. Click the **+** icon beside the label name to add it to the panel at the top of the **Label Picker** form.
- d. Continue selecting labels and adding them to the top panel until you have selected all of the labels for nodes whose configuration you want to override.
- e. Click **SAVE**.

**Expected outcome**

The **Label Picker** form closes and you are returned to the **Create Override** page. The label or labels you selected are displayed in the **Node Label Selector** panel.

- f. Go to step 12.

**Step 11.** Select the edit icon () above the **Node Selector** panel to open the **Node Selector** and do the following:

- a. Check the box at the left edge of the row for each node whose configuration you plan to override.
- b. Click **ADD**.

**Expected outcome**

The **Node Selector** form closes and you are returned to the **Create Override** page. The node or nodes you selected are displayed in the **Node Selector** panel.

**Step 12.** Click the **Save** icon () to save the new override.

**What to do next**

If you created the configuration override in the midst of creating a fabric intent, you must generate that fabric intent again to incorporate the override data. You can then resume the creation and deployment of that fabric intent.

## 7.2 Contextual configuration overrides

Contextual configuration overrides (CCOs) alter the configurations for the following types of objects associated with a workload intent:

- sub-interfaces

- routers

When you create a workload intent, the Fabric Services System prepares a set of configuration files for all of the nodes participating in the workload intent. The configuration for each of these nodes is based directly on the parameters you provide when creating the workload intent. These are the "normalized configurations" for the participating nodes.

A CCO allows you to specify a set of changes to the configuration files for one or more nodes participating in a workload intent. CCOs are strictly additive; they allow you to insert new or modify existing configuration data, but cannot be used to remove any of the configuration data that the system creates based on the original workload intent inputs.

As part of the override, you must provide the JSON configuration data to be added to each configuration. You then specify the sub-interfaces or routers within the workload intent to which the overrides should be applied. Optionally, you can also specify the nodes on which the contextual configuration override should apply.



**Note:** Unlike global configuration overrides, the use of labels to identify target objects is not supported for CCOs.

When you generate a workload intent, the Fabric Services System checks for related, active CCOs. The additional configuration data from those overrides is applied to the configurations of the target objects if they are included in the current workload. This is true whether the workload you are generating is:

- a workload intent you are currently creating
- a new workload intent
- a new version of an existing workload intent

### Multiple versions of overrides

You can create additional versions of a contextual configuration override; only one of these can be set to Active at a time. You can set all versions of an override to Inactive if required.

### Deleting overrides

If you delete a CCO, or set all versions of an override to Inactive, existing workload intents (and the node configurations that support them) are unaffected.

However, the next time you generate a workload intent that previously used the deleted or inactive CCO, the former override modification does not take place.

### Deleting supporting elements

Each CCO involves one or more specific sub-interfaces or routers that you select when you create the override. If those sub-interfaces or routers are subsequently deleted, the system does not automatically update any CCO that refers to them. You must modify each affected CCO to correct the reference to the non-existent object.

However, if you try to save a CCO that refers to a non-existent object (for example, a router that existed when the CCO was first created but that has since been deleted), the system warns you that the result is an invalid workload configuration.

### Related topics

[Viewing and managing configuration overrides](#)

## 7.2.1 Contextual configuration override parameters

Table 40: Configuration override parameters

Parameter	Default value	Description
<b>Version</b>	See Description	Indicates the version number for this override. This value is assigned automatically. The first instance of any configuration override is Version 1.0. Subsequent versions of the same override increment this value.
<b>Active</b>	Enabled	Indicates which, if any, version of a configuration override should be used to modify the configuration data for the targets. A maximum of one version of any configuration override can be active at one time. It is possible to set all versions of an override to inactive.
<b>Automatic Deployment</b>	Disabled	This value cannot be altered.
<b>Name</b>	None	Identifies the configuration override and is used to identify it in the list of all overrides within the system. It must be unique (but is shared by all versions of the same override).
<b>Description</b>	None	Describes the purpose or impact of the override.
<b>Execution Order</b>	n+100	Indicates where this particular override should fall within the overall execution sequence when there are multiple overrides.  If two or more overrides target the same sub-interfaces or routers, the execution order value dictates the order in which the configuration changes are merged into the configurations data.  The override with the lowest Execution Order value is applied first.  The default value is the current highest value plus 100. As a result, the first override created has a default value of 100.
<b>Target Configuration</b>	See description	Identifies several configuration properties of any node that are expected if the node is a candidate for this override.  It consists of three sub-parameters: <ul style="list-style-type: none"> <li>• <b>Operating System:</b> SRLinux</li> <li>• <b>Software Version:</b> either <ul style="list-style-type: none"> <li>– ALL</li> <li>– a major software version from the software catalog</li> <li>– or both</li> </ul> </li> <li>• <b>Data Model:</b> SRLinux</li> </ul>
<b>Configuration Type</b>	Contextual Override	This value cannot be modified.

Parameter	Default value	Description
<b>Type</b>	sub-interface	Indicates the type of workload-related object whose configuration data is impacted by this override. Its value can be one of the following: <ul style="list-style-type: none"> <li>sub-interface</li> <li>router</li> </ul>
<b>Path</b>	/interface/ subinterface	Indicates where within the structure of a configuration file this override should be inserted. <ul style="list-style-type: none"> <li>if the override applies to sub-interfaces, the override text is inserted within the /interface/sub-interface section of the configuration file</li> <li>if the override applies to routers, the override text is inserted within the /network-instance section of the configuration file</li> </ul>
<b>Method Selector</b>	By Type	This value cannot be altered.
<b>Selector</b>	n/a	Indicates the list of sub-interfaces or routers that are affected by this override. The list of selected objects can span multiple workload intents.
<b>Node Selector</b>	None	Identifies specific nodes to which this override should be applied. You can optionally use this control to select one or more nodes in the system inventory. Regardless of how many nodes' configurations may be encompassed by the current list of sub-interface or router targets, the configuration override is applied only to these nodes.
<b>Configuration Data</b>	None	Contains the configuration data that is applied to each node that is subject to this configuration override. Each configuration override must include a set of JSON configuration data to be merged into the configuration data for the targets. For example: <pre> {"ip-mtu": 1501} </pre>

## 7.2.2 Creating a contextual configuration override

### Prerequisites

Create the specific sub-interfaces or routers whose configurations you are going to modify with this override. You must create these beforehand because you are prompted to select them during this procedure.

### About this task

This procedure describes how to create the initial version of a contextual configuration override. At a high level, you create a configuration override by doing the following:

1. Provide a name and description to identify this override.
2. Set some configuration parameters for the override:
  - whether it is active
  - where it falls in the overall execution order for all active overrides
3. Identify the software configuration of the target nodes (the operating system, OS version, and data model).
4. Enter the specific configuration data that should be applied to the target configurations.
5. Specify whether the configuration override applies to sub-interfaces or routers.



**Note:** Contextual configuration overrides for routers apply only to routed subnets or bridged subnets with IRB.

6. Identify the objects to which this override applies by selecting specific, previously-defined sub-interfaces or routers.
7. Optionally, restrict the override to only affect specific nodes.
8. Save the override.

While the override is active, it is applied to the targeted sub-interfaces and routers when you generate configurations for any:

- in-progress workload intents
- future workload intents
- new versions of existing workload intents

## Procedure

**Step 1.** Click the  menu.

**Step 2.** Select **Overrides** to open the **Overrides** page.

**Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the override.



**Note:** You cannot change the region selection after you begin creating the override. If you select a new region in the **Region Selector** while creating an override, the creation form closes and you are returned to the **Overrides** page.

**Step 4.** From the **View** drop-down, select Contextual Override.



**Note:** The View drop-down is set to Global Override by default.

**Step 5.** Click **+CREATE** to open the **Configuration Creation** page.

**Step 6.** Enter values for the following parameters as described in [Contextual configuration override parameters](#):

- **Active**
- **Name**
- **Description**

- **Execution Order**



**Note:** The **Automatic Deployment** property is disabled and cannot be altered for a contextual configuration override.

**Step 7.** Identify a target configuration by doing the following:

- Click **+ ADD**.
- From the Target Configuration form, select the following:
  - **Operating System**
  - **Software Version**
  - **Data Model**
- Click **ADD**.

**Step 8.** In the **Configuration** panel, add the specific configuration data that you want to be merged into the configurations for the targeted nodes.



**Note:** You can enter independent instances of configuration code here for each combination of operating system, version, and data model you selected in step 7. Select each target configuration from the **Target** drop-down list to see its unique window in which to enter configuration code.

**Step 9.** In the **Type** drop-down list, indicate whether this override affects the configuration of a sub-interface or a router.

**Step 10.** Select the edit icon (✎) above the **Selector** panel to open the **Selector** form (which lists all available sub-interfaces or routers), and do the following:

- Check the box at the left edge of the row for one or more displayed sub-interfaces or routers whose configuration you plan to override.
- Click **ADD**.

**Expected outcome**

The sub-interfaces or routers are added to the **Bin** panel to the right of the selection list.

- If you need to remove a sub-interface or router from the **Bin** panel, click the **More** icon (⋮) to the right of the sub-interface in the Bin panel and select **Delete** from the displayed action list.
- Repeat steps 10.a through 10.c until all sub-interfaces or routers you intend to target with this override have been added to the **Bin** panel.
- Click **SAVE**.

**Expected outcome**

The **Selector** form closes and you are returned to the **Create Override** page. The sub-interfaces or routers you selected are displayed in the **Selector** panel. The **Count** value indicates how many sub-interfaces or routers are in the **Selector** list.

**Step 11.** Optionally, use the **Node Selector** to restrict the scope of this override. Select one or more nodes that are managed by the Fabric Services System. If the set of sub-interface or router targets you selected for this override encompass multiple nodes, only the nodes on this list are affected by the CCO.



**Note:** If you do not select one or more nodes here, the contextual configuration override is applied to all nodes encompassed by the sub-interfaces or routers you selected.

**Step 12.** Click the **Save** icon () to save the new override.

### What to do next

If you created the configuration override in the midst of creating a workload intent, you must generate that workload intent again to incorporate the override data. You can then resume the creation and deployment of that workload intent.

## 7.3 Viewing and managing configuration overrides

### About this task

This procedure describes various ways to view and modify an existing global or contextual configuration override:

- viewing a list of configuration overrides
- deleting a configuration override



**Note:** You can delete system GCOs that are automatically created for managing deviations or unmanaged nodes. Since system GCOs have a higher precedence than other GCOs, the ability to delete them could be useful if the system GCO contains an undesirable configuration for some reason. Use caution when deleting system GCOs.

- viewing details about a single configuration override
- creating a new version of an existing configuration override
- deleting an existing version of a configuration override
- comparing two versions of a configuration override
- viewing a list of nodes affected by a particular configuration override
- duplicating a system GCO into a conventional GCO

### Procedure

**Step 1.** To view a list of previously configured overrides, do the following:

- a. Click the  menu.
- b. Select **Overrides**.

#### Expected outcome

The **Global Override** page displays, showing a list of all global configuration overrides already created within the currently selected region.

- c. To view contextual configuration overrides, use the **View** menu to select Contextual Override.

#### Expected outcome

The **Contextual Override** page displays, showing a list of all contextual configuration overrides already created within the currently selected region.

- d. If necessary, use the **Region Selector** at the top of the page to select the region containing the override you require.

**Step 2.** To delete a configuration override, do the following:

- a. Select a global or contextual configuration override in the list displayed after performing step 1.
- b. Click the **More** icon (  ) at the right end of the row and select **Delete** from the displayed list of actions.
- c. In the resulting confirmation dialog, click **OK**.

**Step 3.** To view details about a single override, do the following:

- a. Select a global or contextual configuration override in the list displayed after performing step 1.
- b. Click the **More** icon (  ) at the right end of the row and select **Open** from the displayed list of actions.

**Expected outcome**

The **Override Configuration** page displays, showing settings for the configuration override and the JSON code that it inserts into the node configurations.

**Step 4.** To create a new version of a configuration override, do the following:

- a. Open a single global or contextual configuration override as described in step 3.
- b. Click the **More** icon (  ) at the upper right of the page and select **New Version** from the displayed list of actions.

**Expected outcome**

The display updates, showing an incremented **Version** number for the configuration override.

- c. Update settings for the global or contextual configuration override and the JSON code that is inserted by the new version of the override.



**Note:** If you set this new version of the override to be Active, the other versions of the same override are automatically set to be Inactive because only one version can be Active at a time. If you leave this new version Inactive, the previously Active version of the same override remains Active.

Only the JSON code in the Active version of the configuration override is inserted into fabric intents.

- d. Click the **Save** icon (  ) to save the new version of the configuration override.



**Note:** To apply this new version of the configuration override, you must generate and deploy a new version of that fabric intent (for global overrides) or workload intent (for contextual overrides).

**Step 5.** To delete an existing version of a global or contextual configuration override, do the following:

- a. Open a single global or contextual configuration override as described in step 3.
- b. Click the **More** icon (  ) at the upper right of the page and select **Delete Version** from the displayed list of actions.



**Note:** You cannot delete an active version of an override.

### Expected outcome

The currently displayed version of the override is immediately deleted. There is no confirmation required.

**Step 6.** To compare the JSON configuration data in two different versions of a configuration override, do the following:

- a. Open the global or contextual configuration override as described in step 3.
- b. From the **Views** menu (currently set to Configuration), select **Comparison**.
- c. In the **Comparison** view, use the drop-down lists to select the following for two different versions of the current configuration override:

- **Version** number
- **Target Configuration**

### Expected outcome

The display updates to show the two versions side-by-side. Alterations are highlighted in red and green to show what has changed between the two selected versions.

- d. To close this view, select **Configuration** from the **Views** menu.

**Step 7.** To see a list of the nodes that are affected by any single global or contextual configuration override, do the following:

- a. Open the global or contextual configuration override as described in step 3.
- b. From the **Views** menu (currently set to Configuration), select Nodes List.

### Expected outcome

The display updates to show a list of nodes affected by the current configuration override. From this list, you can use the Actions menu to view the configuration of the individual node, or the Fabric Design for the fabric intent that includes that node.



**Note:** When you view the configuration for an individual node, you can select from two options:

- **Actual Configuration:** this shows the configuration data for that node including any insertions arising from configuration overrides.
- **Normalized Configuration:** this show the configuration data for that node as set exclusively by the fabric intent parameters. This configuration does not include any modifications arising from configuration overrides.

- c. To close this view, select Configuration from the **Views** menu.

**Step 8.** To duplicate a system GCO to a conventional or "normal" GCO, do the following:

- a. Open the global configuration override as described in step 3.
- b. Click the More icon (  ) and select **Duplicate as Normal GCO**.

## 8 Traffic mirroring

The Fabric Services System provides operators with the ability to mirror traffic from a source group (a set of interfaces or sub-interfaces) and send the traffic to a local or remote destination. Mirrors are typically used for troubleshooting or security.

You configure a mirroring instance to specify what traffic to mirror (the source group) and where to send the mirrored traffic (the destination). A mirroring instance can have many sources, but can only have one mirror destination.

Before you can configure a mirroring instance, you must first configure the mirror source groups and the mirror destination. A mirror destination cannot be reused in multiple mirroring instances. Within a mirroring instance, if an interface is configured as mirror source, a sub-interface within that interface cannot be added as another mirror source.

You can configure mirroring for traffic in a specific direction (ingress only or egress only) or bidirectional traffic (both ingress and egress).



**Note:** Traffic mirrors are specific to the region in which they are created. A traffic mirror that is created within one region is not visible within, or available to, other regions.

### Concurrent operations on the underlay

When you perform the following mirroring operations, you update the underlying fabric (underlay) and update the normalized configuration:

- create a new active mirroring instance
- delete an active mirror object
- enable an existing mirror object
- attach or delete a label that is applied to a node, interface, or sub-interface that is referenced in an active mirroring instance

If you are updating the mirroring configuration, if another user is updating the underlay (for example, adding a new node or importing a new manual topology), the system locks the underlay until that operation is complete. The changes to the mirroring configuration are merged into the normalized configuration when the lock is removed.

Conversely, if you are updating a mirror configuration, a lock is also applied to the underlay to prevent other updates to the underlay until the updates to the mirror configuration are merged.

### 8.1 Mirror source groups

You select the sources of the traffic that you want to mirror by creating a source group, which specifies the source interfaces or sub-interfaces. Only interfaces defined in the topology as either edge links or uplinks are available as options for sources of mirrored traffic.



**Note:** If an interface or sub-interface is part of a source group, before you delete it from the topology of a fabric intent, you must first remove it from the source group.

## 8.2 Mirror destinations

The mirror destination can be:

- a local destination (local mirroring)  
In a local mirroring configuration, both the mirror source and mirror destination reside on the same node. The local destination can be a switched port analyzer (SPAN).
- a remote destination  
In a remote mirroring configuration, the mirror source and mirror destination are on different nodes. The mirror source resides on one node and the mirrored packets are encapsulated into a tunnel toward the mirror destination. The remote destination can be an encapsulated remote switched port analyzer (ERSPAN).

## 8.3 Configuration parameters

### Mirroring instance parameters

Table 41: Mirroring instance parameter descriptions

Parameter	Description	Values
<b>Name</b>	Specifies the name of the mirroring instance.	String
<b>Labels</b>	Specifies the labels to apply to the mirror object.	Optional
<b>Description</b>	Describes this mirroring instance.	Optional
<b>Active</b>	Indicates whether the mirroring instance is active.	—
<b>Automatic deployment</b>	Specifies whether a mirroring instance is automatically deployed as soon as it is created.	—
<b>Source Group</b>	Specifies one or more sources previously configured sources for the mirroring instance. For more information, see <a href="#">Mirror source groups</a> .	—
<b>Destination</b>	Specifies a previously configured destination for the mirroring instance. For more information, see <a href="#">Mirror destinations</a> .	—

### Mirror source group settings

Table 42: Mirror source parameter descriptions

Field Name	Description	Values
<b>Name</b>	Specifies the name of the source group.	—

Field Name	Description	Values
<b>Labels</b>	Specifies the labels to assign to the source object.	Name of an existing label
<b>Description</b>	Provide a description for this source group.	String
<b>Source Group Type</b>	Specifies whether the sources of the mirror are interfaces or sub-interfaces.	<b>Interface</b> or <b>Sub - Interface</b>
<b>Node Selection Method</b>	Specifies whether you want to select the node by node name, by node label, or if you prefer not to specify a node by name or label and instead choose interfaces or sub-interfaces by label only.	<b>Node</b> , <b>Node Label</b> , or <b>Not Applicable</b>
<b>Nodes</b>	If node selection is by node name, specifies a node from which to select interfaces or sub-interfaces.	—
<b>Interface Selection Method</b>	If the source group type is interface, and the <b>Node Selection Method</b> parameter is set to: <ul style="list-style-type: none"> <li>• <b>Node</b>, this parameter is set to <b>Interface</b></li> <li>• <b>Node Label</b>, this parameter is set to <b>Interface Label</b></li> </ul>	This parameter cannot be modified.
<b>Interfaces</b>	Specifies one or more interfaces by interface name. If you selected a node, you can only select interfaces from that node. If you select multiple nodes, only the common interfaces among them are displayed for selection.	—
<b>Node Labels</b>	Specifies a node label. If you specify a node label, all nodes that match the selected label are used as mirror sources.	—
<b>Interface Labels</b>	If the node selection method is by label, specifies an interface label to filter the interfaces to use as source. <ul style="list-style-type: none"> <li>• If the <b>Node Selection Method</b> parameter is set to <b>Not Applicable</b>, the system selects interfaces that match the interface label from any node.</li> <li>• If you specified a node label, the system selects interfaces on the specified node that match the label.</li> </ul>	—
<b>Sub-Interfaces</b>	If the source group type is sub-interface, specifies one or more sub-interfaces by interface name. If you selected a node, you can only select sub-interfaces from that node.	—

Field Name	Description	Values
<b>Sub-Interface Labels</b>	Specifies a sub-interface label. If you do not specify a node label (by setting the <b>Node Selection Method</b> parameter to <b>Not Applicable</b> ), the system selects any sub-interface that matches the sub-interface label across any node. If you specify a node label, the system selects sub-interfaces that match the sub-interface label on the node.	—
<b>Traffic Direction</b>	Specify the direction of traffic to mirror.	<b>Ingress only, Egress only, or Ingress-egress</b>

### Mirror destination parameters

Table 43: Mirror destination parameter descriptions

Field Name	Description	Values
<b>Destination Name</b>	Specifies the name of the destination group object.	—
<b>Description</b>	Provides a description for the mirror destination.	—
<b>Labels</b>	Specifies the labels to assign to this mirror destination.	—
<b>Destination Type</b>	Specifies whether this mirror destination local or remote.	<b>Local or Remote</b>
<b>Interface</b>	If the destination is local, specifies the interface.	—
<b>VLAN ID</b>	If the destination is local, specifies the VLAN ID.	—
<b>Destination IP</b>	If the destination is remote, specifies the IP address of the destination.	IPv4 notation
<b>Source IP</b>	Specifies the source IP address; if no address is provided, the system IP address is used. This setting is required for unmanaged nodes.	IPv4 notation
<b>Encapsulation</b>	Specifies the encapsulation method.	The system supports L2oGRE encapsulation only. You cannot set this parameter.

#### Related topics

[Configuring mirroring sources](#)

[Configuring a mirror destination](#)

## Configuring a mirroring instance

# 8.4 Configuring mirroring sources

## Prerequisites

If you are selecting nodes, interfaces, or sub-interfaces by label, ensure that labels have already been assigned. Mirror sources can be ISL or downlinks.



**Note:** You cannot edit a mirror source after it has been created; you can only create or delete mirror sources.

## Procedure

- Step 1.** Click the main menu , then select **Traffic Mirroring**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the traffic mirroring object.
- Step 3.** From the drop-down list, select **Source Groups**.
- Step 4.** Click **+ CREATE SOURCE GROUP**.
- Step 5.** Configure basic settings for the mirror source.  
Under **Basic Properties**, provide a name for this mirror source and an optional description. You can also assign a label to this mirror source.
- Step 6.** Specify whether you are selecting the mirror source by interface or sub-interface.
  - If the source group type is an interface, set the **Source Group Type** parameter to **Interface**.
  - If the source group type is a sub-interface, set the **Source Group Type** parameter to **Sub - Interface**.
- Step 7.** Optional: Select the nodes.  
You can select nodes by node name or by node label. If you prefer not to specify a node, you can choose interfaces or sub-interfaces by label only.
  - Select a node by name.  
Set the **Node Selection Method** parameter to **Node**. Click **+ ADD** above the **Nodes** box, select the node, then click **ADD**. Then, go to Step 8.
  - Select nodes by node label.  
Set the **Node Selection Method** parameter to **Node Label**. Then, click  on top of the **Node Labels** box to open the **Label Picker** form and select a node label. Then, go to Step 9.
  - Do not select a node.  
Set the **Node Selection Method** parameter to **Not Applicable** to select interfaces or sub-interfaces by labels only. Then, go to Step 9.
- Step 8.** Select individual interfaces or sub-interfaces.
  - If the **Source Group Type** parameter is set to **Interface**, click  on top of the **Interfaces** box, then select the interfaces that you want to use as source. Then, go to Step 10.
  - If the **Source Group Type** parameter is set to **Sub-Interfaces**, click  on top of the **Sub-Interfaces**, then select sub-interfaces that you want to use as source. Then, go to Step 10.

**Step 9.** Select an interface label or sub-interface label.

- If the source group type is an interface, click  on top of the **Interface Labels** box to open the **Label Picker** form, then select a label.
- If the source group type is a sub-interface, click  on top of the **Sub-Interface Labels** box to open the **Label Picker** form and select a label.

**Step 10.** Specify the direction of traffic to monitor.

Set the **Traffic Direction** parameter to **Ingress Only**, **Egress Only**, or **Ingress-Egress**.

**Step 11.** Click **CREATE**.

### What to do next

If you have not configured a mirror destination, create the mirror destination. Then, create the mirror instance.

### Related topics

[Configuration parameters](#)

## 8.5 Configuring a mirror destination

### About this task

You can configure a local destination or a remote destination.

### Procedure

**Step 1.** Click the main menu , then select **Traffic Mirroring**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the traffic mirroring object.

**Step 3.** From the drop-down list, select **Destinations**.

**Step 4.** Click **+ CREATE A DESTINATION**.

**Step 5.** Configure basic settings for the mirror destination.

Under **Basic Properties**, set the destination name and provide an optional description. You can also add a label to this mirror destination.

**Step 6.** Configure the destination.

- If the destination is local, set the **Destination Type** parameter to **Local**, then go to Step 7.
- If the destination is remote, set the **Destination Type** parameter to **Remote**, then go to Step 8.

**Step 7.** If the destination is local, set the following parameters, then go to 9:

- **Interface**
- **VLAN ID**

**Step 8.** If the destination is remote, set the following parameters:

- **Destination IP:** specify an IPv4 address
- **Source IP:** specify an IPv4 address; if no address is provided, the system IP address is used

**Step 9.** Click **CREATE**.

## What to do next

If the sources for the mirroring instance have not been configured, create the mirror sources. Then, create the mirroring instance.

### Related topics

[Configuration parameters](#)

## 8.6 Configuring a mirroring instance

### Prerequisites

You must have already configured source groups and a destination for the mirror instance.

### Procedure

- Step 1.** Click the main menu , then select **Traffic Mirroring**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the traffic mirroring object.
- Step 3.** From the drop-down list, select **Mirrors**.
- Step 4.** Click **+ CREATE MIRROR**.
- Step 5.** Configure basic settings for the mirror source.  
Set the following parameters:
  - **Name**
  - **Labels**
  - **Description**
  - **Active**
  - **Automatic Deployment**
- Step 6.** In the **Source Groups** panel, click **+ ADD**, then select existing source groups.  
You can select more than one source.
- Step 7.** In the **Destination** panel, click **+ ADD**, then select an existing destination.
- Step 8.** Click **CREATE**.

### Expected outcome

The new mirroring instance is displayed in the **Mirrors** view in the Active state. You can also choose to create an inactive mirror instance.



**Note:** You can edit the mirror instance at any time.

## What to do next

Verify the configuration in Configuration Inspector.

### Related topics

[Configuration parameters](#)

## 9 Maintenance intents

A maintenance intent identifies a change you want to deploy to a node.

The Fabric Services System supports two types of maintenance intent:

- Software changes, which upgrade or downgrade the SR Linux software version running on one or more nodes within a single fabric.
- Node replacement, which updates the hardware association for an existing node within a fabric to a new piece of matching hardware with a different serial number. The intent also causes the system to download the necessary configuration files to the new node so that it can fully resume the role of the hardware it replaced.

When a deployed maintenance intent is in progress, whether for a software update or node replacement, the system locks the affected fabric intent. No other operations can be carried out on the locked fabric until the maintenance activity is finished.

Unlike fabric intents and workload VPN intents, a maintenance intent is a one-time operation. You cannot make new versions of a maintenance intent or otherwise maintain it successively over time. Each new maintenance operation requires a new maintenance intent.



**Note:** All maintenance intents are specific to the region in which they are created. A maintenance intent that is created within one region is not visible within, or available to, other regions.



**Note:** SR Linux Release 23.7 introduced a number of changes to the way certain capabilities are represented and configured on a node. When upgrading from a previous release to SR Linux 23.7, the Fabric Services System automatically updates these configurations to the format required by SR Linux 23.7. When downgrading from SR Linux 23.7 to a previous release, the system likewise reverts those configurations to those required by the previous format.



**Note:** Before performing any maintenance (such as a software upgrade or downgrade) on a node, ensure that the default network instance is present on the target node. If the default network instance is not present, the maintenance upgrade/downgrade process will fail at the initial "Deploy drain policies" step.



**Note:** The system does not support maintenance intents on the 7220 IXR-D1.

### 9.1 Viewing a maintenance intent

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** From the menu, select **Maintenance Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region containing a maintenance intent.
- Step 4.** To open a specific maintenance intent from the list, do one of the following:

- Double-click the row for that intent.
- Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.

**Expected outcome**

The system displays the selected maintenance intent in its **Design Intent** view.

### 9.1.1 Elements of the maintenance intent Design view

Figure 30: Maintenance intent view

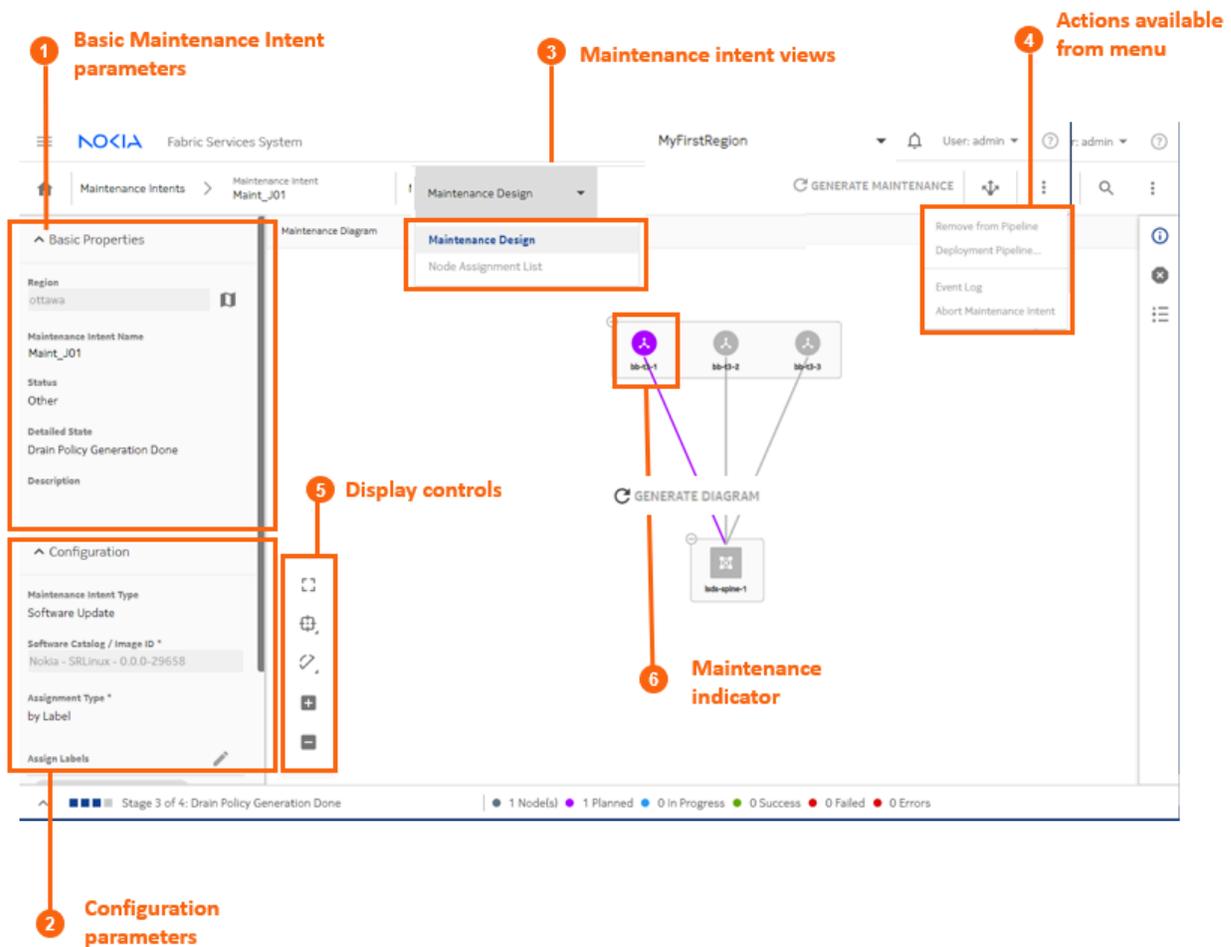


Table 44: Elements of the maintenance intent Design view

	Property	Description
1	Basic Maintenance Intent parameters	This panel shows basic parameters including the intent's name, description, and current state.

	Property	Description
2	Configuration parameters	This panel shows the details about the maintenance represented by this intent: <ul style="list-style-type: none"> <li>For software upgrades, this indicates the intended software version and the label that identifies affected nodes.</li> <li>For node replacement, this indicates the new serial number for the node being replaced, and the label that identifies the affected node.</li> </ul>
3	Maintenance Intent views	The <b>View</b> drop-down list provides access to the different views necessary to design a maintenance intent and view a list of the affected nodes.
4	Actions available from menu	From the actions menu you can manage deployment or view the intent's event log.
5	Display controls	These controls allow you to modify the way the system displays the maintenance intent.
6	Maintenance indicator	The purple shading indicates the presence of at least one node that is subject to maintenance.

### 9.1.2 Viewing affected nodes

#### About this task

From an open maintenance intent, you can view a list of all of the nodes that are affected by the current intent.

To view a list of affected nodes:

#### Procedure

**Step 1.** Open a maintenance intent.

**Step 2.** In the **Views** drop-down list, select **Node Assignment List**. The system opens a list of the nodes that will be altered by the current maintenance intent.

**Step 3.** To return to the **Design** view, select **Maintenance Design** from the **Views** drop-down list.

#### Related topics

[Viewing a maintenance intent](#)

### 9.1.3 Viewing the maintenance intent event log

#### About this task

The event log retains a detailed history of all registered events that have occurred with respect to the current maintenance intent. This can be useful when troubleshooting maintenance intent issues, or just to verify that the intent's history matches expectations.

The **More actions** (  ) menu at the upper right of the page provides access to the usual list management control.

To view the maintenance intent event log:

### Procedure

- Step 1.** Open a maintenance intent.
- Step 2.** Click the **More actions** icon (  ) to open the actions list.
- Step 3.** Click **Event Log** from the drop-down list. The system opens the **Event Log** overlay.
- Step 4.** Click  at the upper right of the overlay, or the **CLOSE** button, to return to the maintenance intent **Design** view.

### Related topics

[Viewing a maintenance intent](#)

[Lists](#)

## 9.2 Creating a maintenance label

### About this task

Before you create a maintenance intent, you must create and apply an identifying label to each of the objects intended to be the target of the intent. Objects are one or more nodes within a single fabric. Labeling objects is a powerful tool in the Fabric Services System. In the specific case of labels for a maintenance intent, you create and apply a label with the predefined key "Maintenance," and a value of your choosing.

To create a label you can use to identify a node for maintenance:

### Procedure

- Step 1.** From the main menu, click **Label Factory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the label.
- Step 3.** Click  **CREATE A LABEL**.
- Step 4.** In the **Label Key** drop-down list, click **Maintenance**.  
Maintenance is a predefined key that is specifically intended to flag objects as the targets of maintenance intents.
- Step 5.** In the **Value** field, enter text that identifies the maintenance activity.  
This value cannot include spaces or special characters.
- Step 6.** Optional: Enter information in the **Comments** field that helps uniquely identify this label.
- Step 7.** Click **CREATE**. The system adds your label to the library of available labels.

### Related topics

[Labels](#)

## 9.3 Labeling objects for maintenance

### About this task

Before you create a maintenance intent, you must apply an identifying label to each of the objects intended to be the target of the intent.



**Note:** Although maintenance is not supported on the 7220 IXR-D1, the system does prevent you from assigning maintenance label on it and creating maintenance intent.

### Procedure

- Step 1.** View a node that you intend to be a target of the maintenance operation and open the label assignment form using any of the following methods:
- Open the **Inventory** view and select the row showing the target node. Then, click the **More actions** icon (  ) icon at the right edge of the list and select **Edit Labels** from the resulting menu.
  - Open a fabric intent in the **Design and Emulation** view and expand the cluster containing the target node. Right-click the node icon and select **Edit Labels** from the resulting menu.
  - Open a fabric intent in the **Inventory** view and select the row showing the target node. Then, click the **More actions** icon (  ) at the right edge of the list and select **Edit Labels** from the resulting menu.
- Step 2.** In the **Label Picker** form, find the maintenance label you created earlier and click the  icon at the left edge of the row.

#### Expected outcome

The system adds the label to the panel at the top of the **Label Picker** form. If other labels were previously applied to the current node, those labels also appear at the top of the **Label Picker** form.

**Step 3.** Click **SAVE**. The maintenance label is now associated with the node.

**Step 4.** If your maintenance intent affects multiple nodes, repeat steps 1 through 3 for each of the remaining target nodes.

### Related topics

[Viewing the overall inventory](#)

## 9.4 Creating a maintenance intent

### Prerequisites

Ensure that:

- You have created and applied a maintenance label to the participating nodes. The system includes a "Maintenance" key that you can use to identify an object for maintenance. The value that you associate with that key can be whatever you like.
- If this maintenance intent is for a software update, ensure that the target SR Linux software load is present in the system's software catalog.

- If you are using an external DHCP server or file server for SR Linux Zero-touch provisioning (ZTP):
  - for software changes, update the `ztp_provision.py` file with the new software image details
  - for node replacement, update the `dhcpd.conf` file with the new serial number information  
Although the serial number of an unmanaged 7220 IXR-D1 is not needed for any of the Fabric Services System features, the system does not block a user from assigning it, which causes the 7220 IXR-D1 to have a `dhcp.conf` entry and may cause issues to the ZTP process as it should use the external DHCP server and not the internal DHCP server.
- If you are performing a software update, ensure that the target node is in a Ready or Version Mismatch state. Attempting the update on a node in any other state results in an error.
- If you are replacing a node, any state is acceptable but some states of the maintenance intent indicating its progress may be skipped depending on the original node state.
- The default network instance is present on the node. If the default network instance is not present, the maintenance upgrade or downgrade process will fail at the initial "Deploy drain policies" step.

### About this task

A maintenance intent indicates a type of change and the nodes this change affects.

### Procedure

**Step 1.** Click  to open the main menu, then select **Maintenance Intents**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the maintenance intent.



**Note:** You cannot change the region selection after you begin creating the maintenance intent. If you select a new region in the **Region Selector** while creating a maintenance intent, the creation form closes and you are returned to the **Maintenance Intents** page.

**Step 3.** Click **+ CREATE A MAINTENANCE INTENT**.

**Step 4.** Set the basic properties for the maintenance intent.

Provide a name for the maintenance intent and an optional description. Note that the region value is based on your selection in step 2 and cannot be changed without exiting this page.

**Step 5.** Specify whether the maintenance intent is for a software update or a node replacement.

To create a maintenance intent for:

- a software update, set the **Maintenance Intent Type** drop-down list to **Software Update**, then go to step 6.
- a node replacement, set the **Maintenance Intent Type** drop-down list to **Replace Node**, then go to step 7.

**Step 6.** Set the parameters for the software update.

- In the **Software Catalog/Image ID** field, select the target SR Linux software load.
- Update the default settings for the following timers:
  - **Traffic Drain Timer:** This timer starts when the SR Linux maintenance policy is applied to the node. The system reboots the node after the timer has expired. By default, the timer is set to 1 minute.

- **Convergence Timer:** This timer starts when the node has come back up and is in the Ready state. When the timer expires, the SR Linux maintenance policy is removed from the node. By default, the timer is set to 1 minute.
- In the **Node Label Selector** panel, click  and select the label corresponding to this maintenance action from the displayed list. You can select only one label as the target for the maintenance action.

When you are finished, go to step 8.

**Step 7.** Set parameters for the node replacement.

- a. In the **Node Label Selector** panel, click  and select the label that corresponds to this maintenance action from the displayed list.  
You can select only one label as the target for the maintenance action.  
The system updates the **Nodes List** panel to show the node bearing the selected label.
- b. Enter the replacement node's serial number in the **New Serial Number** field.
- c. Optional: Update the default settings for the following timers:
  - In the **Software Catalog/Image ID** field, select the target SR Linux software load.
  - Update the default settings for the following timers:
    - **Traffic Drain Timer:** This timer starts when the SR Linux maintenance policy is applied to the node. The system reboots the node after the timer has expired. By default, the timer is set to 1 minute.
    - **Convergence Timer:** This timer starts when the node has come back up and is in the Ready state. When the timer expires, the SR Linux maintenance policy is removed from the node. By default, the timer is set to 1 minute.
  - In the **Node Label Selector** panel, click  and select the label corresponding to this maintenance action from the displayed list. You can select only one label as the target for the maintenance action.

**Step 8.** Click  to save the maintenance intent.

**Step 9.** Click  **GENERATE MAINTENANCE**.

**Step 10.** Optional: Review the generated configuration code to confirm that it is satisfactory.

- a. Right-click the node that is the target of the maintenance intent and select **Inspect Configuration** from the pop-up menu.
- b. Review the code and verify that it is satisfactory.
- c. Click  at the upper right to close the **Configuration Inspector** window.

## What to do next

Deploy the maintenance intent.

### Related topics

[Labeling objects for maintenance](#)

[Software and image catalogs](#)

## 9.5 Duplicating a maintenance intent

### About this task

You can duplicate an existing maintenance intent, which can save you several steps if you need to apply the same or a similar maintenance action again (to the same or different target nodes).

### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** From the menu, select **Maintenance Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region containing the maintenance intent.
- Step 4.** Click a maintenance intent in the list to select it, and then do one of the following:
  - Double-click the row for that intent.
  - Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
  - Click the  menu at the upper right of the page and select **Abort Maintenance Intent**.  
Any further action on the maintenance intent is prevented, and the target node is restored to its configured state before the maintenance intent was applied.
- Step 5.** Click the **More actions** icon (  ) at the right edge of that row, and select **Duplicate Intent** from the displayed action list.

### Expected outcome

The **Maintenance Intents** page displays with the same data as the original intent, except for the **Maintenance Intent Name** field, which is blank.

- Step 6.** Enter a name for the new, duplicate maintenance intent.
- Step 7.** Update other values in the maintenance intent if required.
- Step 8.** Continue to save, generate, and deploy the maintenance intent.

### Related topics

[Creating a maintenance intent](#)

## 9.6 Maintenance intent deployment

Deploying a maintenance intent causes the system to deploy configuration files to the affected nodes, which ultimately results in the fabric changes embodied in the intent.

Before you can deploy your maintenance intent, you must have saved the maintenance intent and generated its configuration.

Deploying the maintenance intent involves two procedures:

- You must add the maintenance intent to the deployment pipeline, so the intent can take its place in the list of planned deployments of fabric, workload, and maintenance intents for the current region.

Adding a maintenance intent to the pipeline triggers preparatory actions that the system performs behind the scenes. These actions must finish before the system allows you to proceed with deployment.

For example, if your maintenance intent is replacing a node, adding the intent to the deployment pipeline causes the system to stop the DHCP client application on the existing node to release its current IP address. Only after this is complete can you deploy the maintenance intent from the deployment pipeline.

- From the deployment pipeline, you must then manually deploy the intent. This signals to the system that it can proceed with the deployment of configuration files to all participating nodes. This deployment may not occur immediately; the system resolves each active deployment in sequence, waiting until one is complete before proceeding to the next.

## 9.6.1 Adding a maintenance intent to the deployment pipeline

### About this task

To add the maintenance intent to the region's deployment pipeline, do the following:

### Procedure

- Step 1.** Open the list of maintenance intents.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing the maintenance intent.
- Step 3.** Select the maintenance intent you want to deploy from the displayed list.
- Step 4.** Click the **More actions** icon (  ) at the right edge of the maintenance intent's row, and select **Open** from the drop-down list.
- Step 5.** Click the **Add Intent to Deployment Pipeline** icon (  ).
- Step 6.** Click the **ADD TO PIPELINE** button.

### Related topics

[Viewing a maintenance intent](#)

## 9.6.2 Progress of a deployed maintenance intent

When a deployed maintenance intent is in progress, whether for a software update or node replacement, the system locks the affected fabric maintenance. No other operations can be carried out on the locked fabric until the maintenance activity is finished.

The deployed maintenance intent progresses through a series of steps, depending on the type of maintenance being performed. When a step completes successfully, the system automatically proceeds to the next steps. These steps are shown in the deployment pipeline under the **Source Name** column.

A software update progresses through the following steps:

- **Deploy drain policies: Step 1/4**  
The system starts the traffic drain timer and drains traffic (that is, diverts traffic away from the node) for the duration of the timer. When the traffic drain timer expires, the system reboots the node.  
If the maintenance intent fails at this stage, you must discard the fabric intent changes manually.
- **ZTP: Step 2/4**

After the maintenance intent's configuration change has been made on the node, the system rediscovers the updated node using Zero-touch Provisioning (ZTP). At the end of this step, the post-maintenance node returns to the Ready state.

- **Deploy drain policies post update: Step 3/4**

The system applies SR Linux drain policies on the node again for the duration of the convergence timer. The system waits for BGP convergence to complete at this step.

- **Deploy undrain policies: Step 4/4**

The traffic previously diverted away from the node is restored.

If the maintenance intent fails at this stage, you must re-deploy the affected fabric intent manually.

A node replacement goes through the following steps:

- **Deploy drain policies (1/4):**

The system diverts traffic away from the node in anticipation of the hardware replacement.

If this step is successful, the system automatically proceeds to step 2/3.

- **Releasing DHCP IP (2/4):**

The node's previous IP address is released from DHCP. If this step is successful, the system automatically proceeds to step 3/3.

- **Deploy drain policies post update: Step 3/4**

The system applies SR Linux drain policies on the node again for the duration of the convergence timer. The system waits for BGP convergence to complete at this step.

- **Deploy undrain policies (4/4):**

The traffic previously diverted away from the node is restored.

If the maintenance intent fails at this stage, you must redeploy the affected fabric intent manually.

The details of any failure can be found in the Maintenance Error/Event log.

If issues arise during deployment, they appear as alerts in the alert panel at the right side of the page. In the **Maintenance Design** view, the system also highlights them as deployment issues in the status bar by adding a red circle to the fabrics affected by the deployment error and with entries in the Events Log.

### 9.6.3 Deploying a maintenance intent from the deployment pipeline

#### About this task

The maintenance intent remains in the deployment pipeline until you tell the system to proceed with the deployment.

Adding a maintenance intent to the pipeline triggers preparatory actions that the system performs behind the scenes. These actions must finish before the system allows you to proceed with deployment.

When you are ready to proceed with deployment, do the following:

#### Procedure

**Step 1.** Click  to open the main menu, then select **Maintenance Intents**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing the maintenance intent.

**Step 3.** Find the maintenance intent you want to deploy, then click  at the end of its row.

**Step 4.** Click **Deployment Pipeline** from the actions list.

**Step 5.** Find the maintenance intent in the deployment pipeline list, then click  at the end of its row.

**Step 6.** From the resulting actions list, select **Deploy**.

You can view the progress of the software update deployment from the **Deployment Pipeline** page.

### Expected outcome

At the end of a successful deployment, the software version is incremented by 1.

### Related topics

[Progress of a deployed maintenance intent](#)

## 9.7 Removing a maintenance intent from the deployment pipeline

### About this task

To remove the maintenance intent from the region's deployment pipeline before you have deployed it, do the following:

### Procedure

**Step 1.** Open the list of maintenance intents.

**Step 2.** Use the **Region Selector** at the top of the page to select the region containing the maintenance intent.

**Step 3.** Select the maintenance intent that you want to remove from the deployment pipeline.

**Step 4.** Click the **More actions** icon (  ) at the right edge of the maintenance intent's row, and click **Open** from the drop-down list.

**Step 5.** Click the **More actions** icon (  ) at the upper right of the **Maintenance Design** page and select **Remove from Pipeline**.

### Related topics

[Viewing a maintenance intent](#)

## 9.8 Aborting a maintenance intent

### About this task

If a maintenance intent fails to complete properly, you can abort the deployed maintenance intent. When aborted, any further action on the maintenance intent is prevented, and the target node is restored to its configured state before the maintenance intent was applied.

You can only abort a maintenance intent after a ten-minute interval has passed after the maintenance intent adopted either the ZTP or Waiting for Dial Home state.

To abort a maintenance intent:

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** From the menu, select **Maintenance Intents**.

**Step 3.** Use the **Region Selector** at the top of the page to select the region containing the maintenance intent.

**Step 4.** To open a specific maintenance intent from the list, do one of the following:

- a. Double-click the row for that intent.
- b. Select a row, click the  icon at the right edge of that row, and select **Open** from the displayed action list.
- c. Click the  menu at the upper right of the page and select **Abort Maintenance Intent**.
- d. Click OK.

**Expected outcome**

If fewer than 10 minutes have passed, a form displays rejecting the Abort instruction. Otherwise, the system cancels the maintenance activity.

## 10 Labels

Labels are tags that help you group and organize fabric items according to specific criteria. You can assign the labels to various fabric items, such as fabric intents, workload VPN intents, maintenance intents, or to objects within each of these intents. For instance, within a fabric intent, you can label specific groupings of nodes. Within a workload VPN intent, you can label specific groupings of subnets and sub-interfaces.

Labels are a powerful tool for identifying a group of objects that can then be subject to collective actions. For example, as part of a maintenance intent, you can create a label and apply it to a group of items that are subject to a software upgrade. Then, you can apply the upgrade to all of the nodes with that label as a single action, instead of upgrading each node individually.

Labels can also be used for informational purposes. You can label specific fabric elements based on criteria such as their physical location. For instance, you can create a label to identify nodes located in a specific data center (DC), then apply this label to nodes physically located in that DC.



**Note:** Built-in system labels for node type and link type are available to all regions. But user-created labels are always specific to the region in which they are created, and are not visible within, or available to, other regions.

### Related topics

[Labeling objects for maintenance](#)

## 10.1 Label types

You can associate two types of labels: Nokia pre-defined labels and user-configured custom labels. Fabric Services System comes with a series of predefined labels that you can associate with fabric items. If these labels do not meet your specific tagging criteria, you can also create new labels in the Label Factory.

### 10.1.1 Nokia pre-defined labels

Nokia pre-defined labels are a basic set of grouping labels for common use. Some pre-defined grouping labels have a built-in priority to resolve when conflicting configuration profiles are applied to it.

Pre-defined fabric intent labels are key and value pairings that can be applied to fabric intents and workload VPN intents. Nokia has defined these labels for use in common tagging scenarios. [Table 45: Pre-defined labels](#) lists the Nokia pre-defined labels.

*Table 45: Pre-defined labels*

Label key	Value	Description
Node-type	T1_LEAF	Identifies all T1 nodes in a fabric. It is dynamic, so new T1 nodes can join this group.
	T2_SPINE	Identifies all T2 nodes in a fabric. It is dynamic, so new T2 can join this group.

Label key	Value	Description
	T3	Identifies all T3 nodes in a fabric. It is dynamic, so new T3 nodes can join this group.
	T4	Identifies all T4 nodes in a fabric. It is dynamic, so new T4 nodes can join this group.
	T5	Identifies all T5 nodes in a fabric. It is dynamic, so new T5 nodes can join this group.
Link-type	T1_ISL_T2	Identifies inter-switch links between T1 and T2 nodes. This label is fabric-scoped. It is dynamic so new ISLs can join this group.
	T2_ISL_T3	Identifies inter-switch links between T2 and T3 nodes. This label is fabric-scoped. It is dynamic so new ISLs can join this group.
	T3_ISL_T4	Identifies inter-switch links between T3 and T4 nodes. This label is fabric-scoped. It is dynamic so new ISLs can join this group.
	T4_ISL_T5	Identifies inter-switch links between T4 and T5 nodes. This label is fabric-scoped. It is dynamic so new ISLs can join this group.
	EDGE LINK	Identifies all edge links. This label is fabric-scoped. It is dynamic so new edge links can join this group.

### 10.1.2 User-configured custom labels

User-configured custom labels allow you to group entities on which configuration profiles can be applied. You can define labels that broadly identify fabric elements, such as by geographical region. You can also define labels to identify highly specific elements, such as a physical location within a DC.

Every custom label has two parts:

- a key, which identifies the type of information the label contains
- a value, which is a specific value appropriate to the label's type

When a label is created, the key and value must be assigned, along with a description to describe the purpose of the label. You may want to define a label to identify physical fabric elements found in a specific location. When creating the key-and-value pairing, you can identify the general location as the label key and a specific region indicator as the value.

Fabric Services System provides some pre-defined label keys that you can use when configuring a custom label. These pre-defined label keys identify common elements of network fabrics that you may want to label and can help to guide you when you create new labels. Consider using any of the label keys as a starting point when you configure a custom label. [Table 46: Pre-defined label keys](#) lists the pre-defined label keys. Nokia recommends using these label keys to ensure consistent naming conventions across fabric elements in your network.

Table 46: Pre-defined label keys

Label key	Description	Examples
ACL-GRP	Use to identify a group of switches, downlink ports, or a group of sub-interfaces used to apply ACL policies on the group.	ACL-GRP-1, ACL-GRP-WEB, ACL-GRP-APP, ACL-GRP-DB
AZ	Use to identify different availability zones (AZs) in a data center. By dividing the network infrastructure inside a DC into multiple fault boundaries, application teams can place compute across AZs and achieve higher availability to their distributed applications.	AZ-1, AZ-2
Building	Use to identify individual buildings that make up a data center. If the data center exists in a single building, use the DC label type instead.	Building-A, Building-B
DC	Use to identify specific data centers.	DC-1, DC-Europe, DC-Atlanta
Edge-Link	Use to identify downlink ports to be used in a workload design.	Edge-1, Edge-A
Link-type	Use to identify a group of links of a specified type.	ISL-1, ISL-A
Maintenance	Use to identify nodes to be part of a collective maintenance action; either upgrade or node replacement.	Maintenance-upgrade1, Maintenance-replacement1
Node-type	Use to identify a group of nodes of a specified type.	Node-1, Node-A
QOS-GRP	Use to identify a group of switches, downlink ports, or a group of sub-interfaces used to apply QoS policies on the group.	QOS-GRP-1, QOS-GRP-WORKLOAD1, QOS-GRP-MGMT, QOS-GRP-STORAGE
Region	Use to identify a group of resources on a per customer region. A region is a geographical area location, such as a city.	Region-A, Region-EAST, Region-Dallas
Room	Use to identify specific rooms within a data center.	Room-A, Room-B
Tenant	Use to identify a group of switches, downlink ports, or a group of sub-interfaces that describe a tenant in a DC.	Tenant-1, Tenant-ABC

For example, you can create a label to identify fabric elements located in particular Availability Zones (AZs). When entering the key-and-value pairing, you could enter *AZ* as the key, and *Zone1* as the value. The label can be assigned to all fabric elements in this specific zone. You can then create additional availability zone labels with the same *AZ* key, but define additional values to specify additional zones, such as *Zone2*, *Zone3*, and so on.

## 10.2 The Label Factory

The Label Factory shows an inventory of all the Nokia pre-defined labels as well as any user-configured custom labels. From the Label Factory, you create new labels and manage existing labels.

You can also query the Label Assignment list to find a specific subset of fabric items that are assigned a specific label. A label can have hundreds or even thousands of assignments to various fabric items throughout the system.

## 10.3 Viewing available labels

### About this task

Follow this procedure to view a list of all available labels in the Label Factory.

The Label Factory opens to the Available Labels category by default. If you navigate away from this list, you can select **Available Labels** from the drop-down list to return to the Available Labels category.

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** Select **Label Factory**.

#### Expected outcome

The **Label Factory** page opens, showing a list of both Nokia pre-defined and user-configured custom labels.

**Step 3.** Optionally, use the **Region Selector** at the top of the page to see the list of labels for a different region.

## 10.4 Creating a label

### About this task

You can create a custom label and assign it to one or more fabric elements. Follow this procedure to create a label in the Label Factory.

### Procedure

**Step 1.** Click the main menu , then select **Label Factory**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the label.

**Step 3.** Click **+ CREATE A LABEL**.

**Step 4.** In the **Label Key** drop-down list, select one of the pre-defined keys.

The label keys are described in [Table 46: Pre-defined label keys](#).

**Step 5.** In the **Value** field, enter text that identifies the purpose of the label.

Enter a value that you can easily recognize later. This value cannot include spaces or special characters.

**Step 6.** Optional: Enter information in the **Comments** field that helps uniquely identify this label.

**Step 7.** Click **CREATE**.

## 10.5 User-configured label manipulation

After you or another user creates a label, you can edit the description of that label, or delete the label from the available labels entirely.



**Note:** You cannot delete a Nokia pre-defined label, nor edit such labels other than to modify the Comments field.

### 10.5.1 Editing an existing label

#### About this task

You can edit an existing label to modify the comments that describe its purpose. Follow this procedure to edit an existing label in the Available Labels list.

#### Procedure

- Step 1.** Click the main menu , then select **Label Factory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose label you want to work with.
- Step 3.** Locate the label that you want to edit, then click the  options menu at the right end of its row.
- Step 4.** Click **Edit Label**.
- Step 5.** Edit the value in the **Comments** field.
- Step 6.** Click **SAVE**.

### 10.5.2 Deleting an existing label

#### About this task

You can only delete a user-configured label. Before deleting a label, you must remove any associations the label has with fabric objects.

Follow this procedure to delete a label.

#### Procedure

- Step 1.** Click the main menu , then select **Label Factory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose label you want to work with.
- Step 3.** Locate the label that you want to delete, then click the  options menu at the right end of its row.
- Step 4.** Select **Delete** from the action list.
- Step 5.** Click **OK**.

#### Related topics

[Removing a label assigned to a fabric intent](#)

[Removing a label assigned to a node in a fabric intent](#)

[Removing a label assigned to a workload VPN intent](#)

### *Removing a label assigned to a management profile*

## 10.6 Label assignments to fabric intent elements

You can assign labels from the Label Factory to various fabric elements. You can assign Nokia pre-defined labels, user-configured labels, or a combination of both. Labels can be assigned to a specific fabric intents, or to specific elements within each intent.

Multiple labels can be associated with a single fabric element. They can also be applied to real or planned nodes in a fabric.

After a label has been assigned to a fabric element, you can also remove it.

### 10.6.1 Assigning labels to a fabric intent

#### About this task

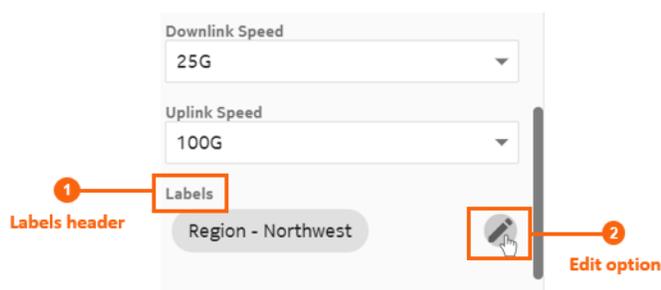
When you assign a label to an intent, the individual elements that make up the intent do not inherit the label. These elements can each be assigned their own labels separately.

Follow this procedure to assign labels found in the Label Factory to a specific fabric intent.

#### Procedure

- Step 1.** Click the main menu ☰, then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** On the **High Level Intent** panel, click  from under the Labels header.

*Figure 31: Assigning a label to a fabric intent*



- Step 5.** From the list of labels, choose one or more labels to assign to the fabric intent. Click **+** from the left end of the row for each label you want to add.
- Step 6.** Click **SAVE**.

## 10.6.2 Assigning a label to a specific node in a fabric intent

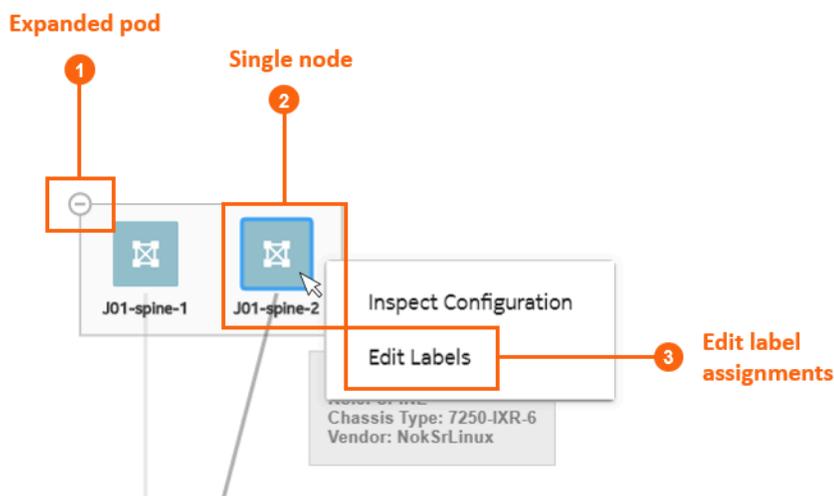
### About this task

Follow this procedure to assign labels found in the Label Factory to specific nodes in a fabric intent.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Right-click a node in the cable diagram.  
You may have to expand a cluster to find a specific node.
- Step 5.** Select **Edit Labels**.

Figure 32: Assigning a label to a specific node



- Step 6.** From the list of labels, choose one or more labels to assign to the fabric intent. Click the **+** from the left end of the row for each label you want to add.
- Step 7.** Click **SAVE**.

### Related topics

[Groups](#)

[Viewing a fabric intent](#)

## 10.6.3 Assigning labels to a fabric link

### About this task

Follow this procedure to assign labels from the Label Factory to specific fabric links within the intent.

## Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **Fabric Links** from the drop-down list.
- Step 5.** For a specific fabric link, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.
- Step 7.** From the list of labels, choose one or more labels to assign to the fabric link. Click  from the left end of the row for each label you want to add.
- Step 8.** Click **SAVE**.

### 10.6.4 Assigning labels to an edge link interface

#### About this task

You can assign labels from the Label Factory to specific edge link interfaces or LAGs within a fabric intent. Edge link interfaces must be labeled using the "Edge-Link" label key. You can create specific labels for edge link interfaces, apply the labels, then reference these labeled items when creating a workload VPN intent sub-interface via the label assignment.

To assign a label to an edge link interface, you must first deploy the fabric intent.

Follow this procedure to assign a label to an edge link interface:

#### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **Edge-Links** from the drop-down list.
- Step 5.** For a specific item in the list, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.
- Step 7.** From the list of labels, choose one or more labels to assign to the edge link port. Click  from the left end of the row for each label you want to add.  
Only labels with the "Edge-Link" label key are displayed.
- Step 8.** Click **SAVE**.

#### Related topics

[Adding sub-interfaces to the workload VPN intent](#)

## 10.6.5 Assigning labels to ISL interfaces

### About this task

You can assign labels from the Label Factory to ISL interfaces within a fabric intent. ISL interfaces must be labeled using the "ISL-Interface" label key.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **ISL Interfaces** from the drop-down list.
- Step 5.** Find the interface in the list and click  at the right end of its row.
- Step 6.** Select **Edit Labels**.
- Step 7.** From the list of labels, choose one or more labels to assign port. Click  at the left end of the row for each label you want to add.
- Step 8.** Click **SAVE**.

## 10.6.6 Removing a label assigned to a fabric intent

### About this task

Follow this procedure to remove labels assigned to a fabric intent.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** On the **High Level Intent** panel, click  from the **Labels** section.

### Expected outcome

The **Label Picker** form opens. The currently assigned labels are displayed in the top of the form.

- Step 5.** From the assigned labels displayed, click the  next to the label you want to remove from the fabric intent.  
You can do this for one or more labels.
- Step 6.** Click **SAVE**.

## 10.6.7 Removing a label assigned to a node in a fabric intent

### About this task

Follow this procedure to remove labels assigned to node in a fabric intent.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Right-click a node in the cable diagram.
- Step 5.** Select **Edit Labels**.

### Expected outcome

The **Label Picker** form opens. The assigned labels are displayed in the top of the form.

- Step 6.** From the assigned labels displayed, click the  next to the label you want to remove from the fabric intent.  
You can do this for one or more labels.
- Step 7.** Click **SAVE**.

## 10.6.8 Removing a label assigned to a fabric link

### About this task

Follow this procedure to remove labels assigned to a fabric link.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **Fabric Links** from the drop-down list.
- Step 5.** For a specific fabric link, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.

### Expected outcome

The **Label Picker** form opens. The currently assigned labels are displayed in the top of the form.

- Step 7.** From the assigned labels displayed, click the  next to one or more labels you want to remove from the fabric link.
- Step 8.** Click **SAVE**.

## 10.6.9 Removing a label assigned to an edge link interface

### About this task

Follow this procedure to remove labels assigned to an edge link interface.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **Edge-Links** from the drop-down list.
- Step 5.** For a specific item in the list, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.

### Expected outcome

The **Label Picker** form opens. The currently assigned labels are displayed in the top of the form.

- Step 7.** From the assigned labels displayed, click the  next to one or more labels you want to remove from the edge link port.
- Step 8.** Click **SAVE**.

## 10.6.10 Removing a label assigned to an ISL interface

### About this task

Follow this procedure to remove labels assigned to an ISL interface.



**Note:** If you delete a label applied to an ISL interface that is referenced in an active mirror instance, you can end up removing the last source in mirror instance and disable the mirror instance.

### Procedure

- Step 1.** Click the main menu , then select **Fabric Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose fabric intent you want to work with.
- Step 3.** Open a fabric intent.
- Step 4.** Select **ISL Interfaces** from the drop-down list.
- Step 5.** Find the interface in the list and click  at the right end of its row.
- Step 6.** Select **Edit Labels**.
- Step 7.** From the assigned labels displayed, click  next to one or more labels you want to remove from the ISL interface
- Step 8.** Click **SAVE**.

## 10.7 Label assignments to workload VPN intent elements

Labels from the Label Factory can be assigned to workload VPN intents, and to the elements contained within workload VPN intents. You can assign Nokia pre-defined labels, user-configured labels, or a combination of both. Labels can be assigned to a specific workload VPN intent, or to specific subnets or sub-interfaces within an intent.

Multiple labels can be associated with a single workload VPN intent or element. You can also remove labels from workload VPN intent elements after they have been assigned.

### 10.7.1 Assigning labels to a workload VPN intent

#### About this task

You can assign labels from the Label Factory to specific workload VPN intents. When you assign a label to an intent, the individual elements that make up the intent do not inherit the label. These elements can each be assigned their own labels separately.

Follow this procedure to assign a label to a workload VPN intent.

#### Procedure

- Step 1.** Click the main menu , then select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose workload VPN intent you want to work with.
- Step 3.** Open a workload VPN intent.
- Step 4.** In the **Workload Design** view, on the left panel, click  under the **Labels** header.
- Step 5.** From the list of labels, choose one or more labels to assign to the workload VPN intent. Click the  from the left end of the row for each label you want to add.
- Step 6.** Click **SAVE**.

### 10.7.2 Assigning a label to a specific sub-interface

#### About this task

You can assign labels found in the Label Factory to specific sub-interfaces within a workload VPN intent. You must have a sub-interface created before beginning this procedure. Follow this procedure to assign a label to the specific sub-interface.

#### Procedure

- Step 1.** Click the main menu , then select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose workload VPN intent you want to work with.
- Step 3.** Open a workload VPN intent.
- Step 4.** In the workload VPN intents drop-down list, switch to the **Sub-Interfaces** view.

- Step 5.** For a specific row item, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.
- Step 7.** From the list of labels, choose one or more labels to assign to the element. Click the  from the left end of the row for each label you want to add.
- Step 8.** Click **SAVE**.

### 10.7.3 Removing a label assigned to a workload VPN intent

#### About this task

Follow this procedure to remove labels assigned to a workload VPN intent.

#### Procedure

- Step 1.** Click the main menu , then select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose workload VPN intent you want to work with.
- Step 3.** Open a workload VPN intent.
- Step 4.** In the **Workload Design** view, on the left panel, click  from under the **Labels** header.

#### Expected outcome

The **Label Picker** form opens. The currently assigned labels are displayed in the top of the form.

- Step 5.** From the assigned labels displayed, click the  next to the label you want to remove from the workload VPN intent.  
You can do this for one or more labels.
- Step 6.** Click **SAVE**.

### 10.7.4 Removing a label assigned to a specific sub-interface

#### About this task

Follow this procedure to remove labels assigned to a specific sub-interface within a workload VPN intent.

#### Procedure

- Step 1.** Click the main menu , then select **Workload VPN Intents**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose workload VPN intent you want to work with.
- Step 3.** Open a workload VPN intent.  
You can double-click the workload VPN intent to open it.
- Step 4.** In the workload VPN intents drop-down list, switch to the **Sub-Interfaces** view.
- Step 5.** For a specific row item, click the  options menu at the right end of the row.
- Step 6.** Select **Edit Labels**.

**Expected outcome**

The **Label Picker** form opens. The currently assigned labels are displayed in the top of the form.

**Step 7.** From the assigned labels displayed, click the **X** next to the label you want to remove.  
You can do this for one or more labels.

**Step 8.** Click **SAVE**.

## 10.8 Label assignments to management profiles

You can assign labels from the Label Factory to management profiles. You can assign Nokia pre-defined labels, user-configured labels, or a combination of both.

You can assign a label to an SNMP or gNMI management profile.

**Related topics**

[Management profiles](#)

### 10.8.1 Assigning a label to a management profile

**About this task**

Use this procedure to assign a label to an existing SNMP or gNMI profile. Note that you can also assign a label to an SNMP management profile as you are creating it.

**Procedure**

**Step 1.** From the main menu select **Inventory** to open the Inventory view, and then select **Management Profiles** from the Views selector to open the Management Profiles view.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose management profile you want to work with.

**Step 3.** In the **Management Profiles** view, locate the management profile that you want to modify.

**Step 4.** Click the **Table row actions** icon  at the right edge of its row, then select **Open**.

**Step 5.** Click the edit icon  above the **Labels** box.

**Step 6.** Locate the label that you want to assign, then click **+** at the left its row.

**Step 7.** When you are finished assigning labels, click **SAVE**.

**Related topics**

[Creating a management profile](#)

[Editing a management profile](#)

### 10.8.2 Removing a label assigned to a management profile

**About this task**

Follow this procedure to remove labels assigned to a management profile.

## Procedure

- Step 1.** In the **Inventory** → **Management Profiles** view, locate the management profile that you want to modify.
- Step 2.** Click the **Table row actions** icon  at the right edge of its row, then select **Open**.
- Step 3.** From the assigned labels displayed, click the  next to the labels that you want to remove. You can do this for one or more labels.
- Step 4.** When you are finished, click **SAVE LABELS**.

## 10.9 Label assignment management

In addition to creating labels, you can use the Label Assignments view to understand how the inventory of labels is being used throughout the system. A label can have hundreds or even thousands of assignments, so the **Label Assignments** page can show you the fabric elements, either real or virtual, that a specific label is assigned to.

### 10.9.1 Viewing the assignments of a specific label

#### About this task

You can view a list of resources that each label is assigned to in the Label Factory. Follow this procedure to view the assignments of a specific label.

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **Label Factory**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region containing the label you want to view.
- Step 4.** For a specific row item, click the  options menu at the right end of the row.
- Step 5.** Select **Show Assignment**.

#### Expected outcome

The **Label Assignment** page opens to show a list of assignments for the selected label under the **Object Name** heading.

#### Related topics

[Viewing available labels](#)

### 10.9.2 Querying the label assignment list

#### About this task

After opening the label assignments of a specific label, you can refine the assignment list to find a specific subset of fabric items with a specific label or series of labels associated. Follow this procedure to query the Label Assignment list.

## Procedure

- Step 1.** Open the **Label Factory** view.
- Step 2.** Use the **Region Selector** at the top of the page to select the region containing the label assignments you want to view.
- Step 3.** Select the **Label Assignment** category filter.
- Step 4.** Enter a search string in the search query field.
- Step 5.** Click  **REFRESH LIST**.

### Expected outcome

The list is refined to display only fabric elements that match the query.

- Step 6.** Optional: Add additional details in the search query field to further refine the list of fabric elements.
- Step 7.** Optional: Click  **REFRESH LIST** to display the refined fabric elements list.

## What to do next

You can continue to refine the Label Assignment list by adding more values to the search query.

Click **CLEAR ALL** to clear the queries.

You can view more information about specific fabric elements that have been assigned the label by clicking an element in the list, then expanding the information panel on the right side of the page by clicking . The information panel shows more details about the currently selected fabric element.

## Related topics

[Viewing available labels](#)

[Label assignment queries](#)

## 10.9.3 Label assignment queries

The **Label Assignment** view shows a list of all assignments for a specific label. A label can have hundreds or even thousands of assignments.

You can refine the label assignment list to find a specific subset of fabric items that are assigned a specific label. Use the search query field to refine the list. In the search query field, enter an expression to include or exclude particular label values, using the following operators:

- OR, AND, and NOT  
 "<LabelName1>" = "<LabelValue1>" or "<LabelName2>" = "<LabelValue2>"
- IN and NOT IN  
 "<LabelName>" in ( "<LabelValue1>" or "<LabelValue2>" or "<LabelValue3>" )
- EXISTS and NOT EXISTS  
 exists ("<LabelName1>" or "<LabelName2>" or "<LabelName3>")

Entering a search query allows you to filter a series of fabric items that have specific multiple labels assigned, while excluding fabric items that also have specific other labels assigned.

For example, if you want to see a list of fabric elements assigned the Label\_A label, but exclude any fabric items that are also assigned the Label\_B label, you can enter the following search query:

INCLUDE (Label = "Label\_A") EXCLUDE (Label = "Label\_B")

Based on the search query, the label assignment list displays only fabric elements assigned Label\_A. The fabric elements in the list may also be assigned multiple other labels. You can continue to refine the search by adding additional criteria to the query.

You can filter the list of available labels using lists.

**Related topics**

[Lists](#)

# 11 Inventories

The Fabric Services System **Inventory** menu provides the following views:

- **Fabric Elements:** provides information about the fabric elements across all fabrics in Fabric Services System and options for associating nodes to hardware.
- **Management profiles:** provides information about the management profiles in the Fabric Services system and how to associate them.

## 11.1 Fabric elements inventory

The **Fabric Elements** view of the Fabric Services System inventory shows a complete list of all fabric elements across all fabrics in Fabric Services System. When you create a new fabric intent, you specify the number of nodes it should contain. These nodes are represented in the inventory. You can also add additional nodes to an existing fabric intent, which also appear in the inventory. The types nodes in the inventory list include:

<b>planned nodes</b>	created when you define a fabric intent, but do not yet have associated real-world hardware
<b>real-world hardware</b>	physical nodes that are present in the network, but are not associated with planned nodes
<b>associated nodes</b>	where a planned node and real-world hardware node are associated in the system, and configurations can be deployed to the physical node as part of a fabric intent

In addition to the overall inventory, each fabric intent has a dedicated inventory that is specific to that one fabric intent. The nodes in the fabric intent inventory list can include planned nodes and real-world hardware. This list is a subset of the full inventory maintained by the system; it shows only the nodes participating in the selected fabric intent.

From these inventories, you can view specific details about both the planned nodes and real hardware in your fabrics. The columns of the **Fabric Elements** page show information about each inventory item, such as status, serial numbers, MAC addresses, and so on.

For each element, you can access several methods to associate planned nodes to real-world hardware.



**Note:** All fabric elements are specific to one region. A fabric element that is managed in one region is not visible within, or available to, other regions.

### 11.1.1 Viewing the inventory of a single fabric intent

#### About this task

There are two types of inventories: the overall inventory and fabric intent-specific inventories. You can view either the overall inventory of nodes, which is a complete inventory of nodes known to the system, or a subset of these nodes which are contained in a specific fabric intent.

Follow this procedure to open the inventory associated with a single fabric intent.

### Procedure

**Step 1.** Open a fabric intent in the **Fabric Design** view.

**Step 2.** In the **View** drop-down list, click **Fabric Inventory**.

#### Expected outcome

A list of all nodes participating in the current fabric intent displays.

**Step 3.** To return to the **Fabric Design** view, select **Fabric Design** in the **View** drop-down list. You can filter and sort the fabric intent inventory list according to specific parameters.

### Related topics

[Viewing a fabric intent](#)

[Lists](#)

## 11.1.2 Viewing details about a node in the inventory

### About this task

Follow this procedure to view more information about a node in the inventory list.

### Procedure

**Step 1.** With the **Fabric Inventory** view open, select a node in the displayed list.

**Step 2.** Click  at the top of the right panel.

Figure 33: Node details

Info	
Device	
Name	5-node-srlinux-demo-spine-1
MAC Address	
Serial Number	fsafaf
Management Address	
State	Unassociated
Assigned Management Profile	defaultRealGnmiprofile
Role	T2 (Spine)
Description	
Operating System	SRLinux
Node Software Version	
Border Leaf	false
Intent Software Version	21.11.2-72
Chassis Type	7250 IXR-6
Labels	Node-type-T2_SPINE
Intent Name	Test
Intent Version	1.0
Region Name	Sunnyvale
Rack ID	0
Digital Sandbox	false

### 11.1.3 Node states

Nodes in the inventory appear in various possible states. With the **Fabric Inventory** inventory view open, the **State** column displays the states of each node in the inventory. The possible states are:

<b>Booted</b>	Real hardware booted in the Fabric Services System. The node is not yet associated with a planned node.
<b>In Discovery</b>	Association of an individual planned node to real hardware has started, and is waiting for the connection.

<b>Ready</b>	An individual planned node has associated successfully with real hardware.
<b>Not Ready</b>	<p>An individual planned node has not successfully associated with real hardware. When a node is in the Not Ready state, error codes describe the reason the association was unsuccessful. The following error codes are displayed in the <b>Detailed Status</b> column:</p> <ul style="list-style-type: none"><li>• <b>Unavailable</b>: the connection failed.</li><li>• <b>ChassisMismatch</b>: the planned node's chassis type is different from the real hardware.</li><li>• <b>VersionMismatch</b>: the planned node's image version is different from the real hardware.</li><li>• <b>ManagementProfileMismatch</b>: the error can be one of the following:<ul style="list-style-type: none"><li>– the management profile is missing for the association</li><li>– the real-world node is using a management profile with a name that is different from the name in the planned node request</li></ul></li></ul>
<b>Unassociated</b>	A planned node is not associated with any real hardware. The planned node may have made an association using a serial number, but the ZTP process has not completed.

## 11.1.4 Inventory manipulation

From the **Fabric Inventory** view, you can manage the nodes known to the system. You can modify the basic information about individual nodes, associate any planned nodes to real-world hardware, and manage items by importing and exporting information using a spreadsheet.

Each time you change the device associations for a specific node, the notifications menu indicates a Device Status Change update for each node that has been updated.

### 11.1.4.1 Editing node information

#### About this task

You can edit the serial number and description information of individual real nodes in the inventory. For Digital Sandbox nodes, you can only edit the description.

Follow this procedure to edit a node.

#### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** From the **Fabric Elements** view, find the node that you want to update and double-click it.
- Step 4.** Enter a serial number or edit the existing serial number.

To edit the serial number, double-click the **Serial Number** field. Optionally, you can edit the description.

For real-world hardware and previously associated nodes, you can also edit the **RackID**.

**Step 5.** Click **SAVE** to save your changes.

#### Related topics

[Viewing the overall inventory](#)

### 11.1.4.2 Viewing the configuration file for a single node

#### About this task

You can view the configuration file that the system has generated for each node in the inventory. Viewing the configuration file can be helpful for verifying the precise configuration that is planned for the node and possibly revising the configuration if needed.

Follow this procedure to view the current configuration planned for a single node.

#### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Find the node and click  at the end of its row.
- Step 4.** Select **Inspect Configuration** from the displayed actions list.
- Step 5.** Optional: Do any of the following:
  - To save the fabric configuration in a local file, click  and choose a location and name for the file.
  - To copy a portion of the fabric configuration, select the portion and click . The selection is added to your clipboard.
  - To find a particular string of text within the fabric configuration, click  and enter the text string. The first instance is highlighted; use the arrows to navigate forward or backward to additional instances, or click ALL to highlight all instances simultaneously.
- Step 6.** Click  at the upper right of the overlay to close the **Inspect Configuration** overlay.

### 11.1.4.3 Planned node and real-world hardware association

When designing a fabric intent, you can incorporate planned nodes into the design. You can use the **Inventory** view to associate individual, pending nodes with their real-world counterparts. For 100% fabric intent deployment, the planned node should be associated with real-world hardware. However, each planned node does not need to be associated with real-world hardware to deploy the 0% fabric intent.

For SR Linux deployments, when you associate planned nodes and real-world hardware and generate a fabric intent containing those nodes, the system performs the association between the planned and real devices via the node discovery and ZTP processes. A configuration file is generated containing the association details.

For WBX deployments, there is no ZTP. The Fabric Services System supports only the overlay. After you create the fabric intent using the imported manual topology, you associate the management profiles with the WBX nodes and add the IP addresses, then wait for the WBX nodes to transition to the Ready state.

If you provide the management address and serial number first, and there is already a real device ZTP with the same serial number, but with a different management address, the inventory subsystem updates the association with the latest values according to the latest ZTP message.

**Related topics**[Node discovery](#)[Creating a management profile](#)

### 11.1.4.3.1 Associating a planned node

**Prerequisites**

Before you can associate a fabric's planned nodes with hardware:

- you must have created and saved the fabric intent, and generated the fabric topology.
- the inventory must include entries for the real-world nodes that correspond to the planned nodes in the fabric intent.

**About this task**

Follow this procedure to associate a planned node with its real-world counterpart. The system can only download the necessary configuration data to a node within the fabric intent after this association has been made.

The inventory can gain entries for nodes in two ways:

- when the node hardware is installed, the system automatically discovers the hardware through the node discovery process
- before the node hardware is available, you can pre-load the inventory with information about the anticipated hardware

This procedure describes how to manually associate each node with hardware, one node at a time. You can identify the hardware before it is installed and available or you can select from installed hardware.

Alternatively, you can also bulk-associate a series of nodes using a spreadsheet and then upload the spreadsheet to the Fabric Services System.

**Procedure**

- Step 1.** Click  to open the main menu, then select **Fabric Elements**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Find the node that you want to associate.  
If necessary, use the controls at the top of each column to filter the list.
- Step 4.** Click the **Table row actions** icon  at the right edge of the row.
- Step 5.** Find the node that you want to associate, then click  at the end of its row.  
If necessary, use the controls at the top of each column to filter the list.
  - To identify the expected hardware for the node before the hardware is available, go to [Step 6](#).
  - To select from installed hardware, go to [Step 7](#).
- Step 6.** Identify the expected hardware for the node before the hardware is available.

Click **Open** from the displayed actions list. In the **Update Inventory** form, set the following parameters. When you are finished, click **SAVE**.

- **Serial Number** - enter the serial number of the real hardware.
  - For SR Linux deployments, this value is required.  
If you provide only a management address for a real device and the ZTP process has completed, the inventory subsystem automatically associates the planned node and real-world hardware.
  - For WBX deployments, leave this field blank if you do not know the serial number
- **Management Address** - required for WBX deployments and for unmanaged SR Linux nodes.
- **Rack ID** - optional
- **Description** - optional

**Step 7.** Select from installed hardware.

Click **Associate** from the displayed actions list. In the **Associate Real Device** form, select one of the displayed devices, then click **ASSOCIATE**.

### Expected outcome

The system associates the planned nodes with the real-world hardware. In the UI, the status of the node is set to In Discovery. When the process completes, the status of the node is set to Ready.

### Related topics

[Node discovery](#)

[Inventories](#)

[Uploads of inventory items](#)

[Lists](#)

[Viewing the overall inventory](#)

## 11.1.4.3.2 Associating a planned node using a serial number

### About this task

Follow this procedure to associate an individual planned node with its real-world counterpart using the **Serial Number** column. Use this simple method to quickly add a new serial number without opening a series of menus in the UI.

Alternatively, you can bulk-associate a series of nodes by uploading a spreadsheet that provides the association of inventory items.

### Procedure

**Step 1.** Open the inventory.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.

**Step 3.** Select a row corresponding to one of the planned nodes and do the following:

- a. Click in the **Serial Number** column for that row.
- b. Enter the serial number for the real-world node that corresponds to this planned node.
- c. Wait for the real-world node to adopt a Ready state.

### Expected outcome

Before the association, the planned node and real-world hardware each appeared on individual rows in the inventory. After the association completes, the planned and real-world hardware combine on the same row, as they are now functional nodes in the fabric.

### Related topics

[Uploads of inventory items](#)

[Viewing the overall inventory](#)

## 11.1.4.4 Disassociating planned nodes from real hardware

### About this task

If a planned node is associated with a real-world hardware device, and you want to change this association to a different real-world device, you must remove the existing association. If the node is not currently deployed in a fabric intent, the Inventory allows you to perform the disassociation.

Alternatively, you can bulk-disassociate a series of nodes by uploading a spreadsheet that contains the relevant information.

You cannot disassociate a planned node from real-world hardware while the node is in the In Discovery state; the **Disassociate** action is disabled.

When creating a candidate fabric intent, you can disassociate a planned node from real-world hardware while the node is in the Ready state; however, the node loses all established end service bindings with the real hardware.

If the associated node is deployed as part of a fabric intent, but has not been successfully connected to real-world hardware (that is, the node has never been in the Ready state) you can disassociate it. If you want to replace a node that is part of the deployed fabric intent, you must perform a maintenance operation, that is, create a node replacement maintenance intent.

Follow this procedure to disassociate a planned node from real hardware.

### Procedure

**Step 1.** From the main menu  **Inventory** view, locate the planned node and click  at the right edge of its row.  
If necessary, use the **Region selector** and the controls at the top of each column to manage the displayed list.

**Step 2.** Select **Disassociate** from the displayed actions list.

**Step 3.** Click **OK**.

### Expected outcome

The system disassociates the planned and real nodes.

### What to do next

After you disassociate a real-world hardware device from a planned node, the real-world hardware item remains in the inventory. It can be re-associated with the planned node. However, if you would like to associate the real-world hardware device to a different planned node, you must update the real-world hardware device's configuration.

When a fabric intent is deployed, if any previous versions of that deployed intent included unassociated real-world hardware items, those hardware items are removed. When a fabric intent is deleted, any associated real-world hardware in that intent is removed from the inventory.

**Related topics**

[Maintenance intents](#)

[Lists](#)

[Planned node and real-world hardware association](#)

[Changes to node associations](#)

[Uploads of inventory items](#)

### 11.1.4.5 Changes to node associations

When you disassociate a successfully connected planned node from a real-world hardware device (that is, the node has been in the Ready state), you can only continue to use the planned node and hardware device if the hardware device is again reassociated with the same planned node. During the initial association, a configuration file is generated by the system containing configuration information, and the node is discovered by the initial ZTP process. This configuration is unique to the real-world hardware; you cannot associate the planned node with a different real-world hardware device because the certificates do not match. In this scenario, to change the node association, you must manually trigger the ZTP process for the real-world hardware in the management stack so that it receives a new configuration and updated software image. You can then perform a new association.

For example, you cannot change the association of a planned node from one real-world hardware device (leaf-1) to a secondary real-world hardware device (leaf-2). This is true even if the hardware devices (leaf-1 and leaf-2) are the same type (that is, both could be 7220 IXR-D3 chassis). Each node has a unique serial number and system name, which is specified in the configuration file. If you generated a certificate on a node for an initial association, then disassociate the node, and try to re-associate with a different node, the action fails as the certificate provided was intended for the original node.

This scenario is true if you use either the Fabric Services System DHCP server or an external DHCP server.

For WBX associations, you can change association at any time, except when it is In Discovery state.

#### 11.1.4.5.1 Node associations in candidate fabric intents

If you create a new candidate version of a deployed fabric intent, you can configure new associations between planned nodes and real-world hardware that differ from the associations in the initial version. The system does not enforce any associations established in the initial version of the intent. Rather, in the candidate version, you can disassociate a previously successful connection between a deployed node and real hardware, resulting in loss of connection and services for the deployed nodes.

When creating a candidate fabric intent, the system assigns an incremented version number. Within the candidate version, you can revise the associations between planned nodes and real-world hardware.

Any revised node associations in the candidate version of the fabric intent appear in the dedicated fabric intent inventory and the overall inventory. When you associate a planned node, the association is versionless. For this reason, if you discard the candidate version of the intent, the node associations do not change.

**Related topics**

[Creating a new version of a fabric intent](#)

[Planned node and real-world hardware association](#)  
[Disassociating planned nodes from real hardware](#)

### 11.1.4.6 Updating the system name of a node in a fabric inventory

#### About this task

You can update the system name of a single node in a fabric intent inventory. You can also bulk-update a series of nodes by modifying the system name mapping data for a specific fabric intent. System names cannot be updated from the overall inventory. You must open a fabric intent and update the node's system name from the fabric intent inventory.

You cannot update the system name of a node if it is already associated with real-world hardware or is in the Deployed state.

Follow this procedure to update the system name of a node in a fabric inventory.

#### Procedure

- Step 1.** Open the fabric intent inventory.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** For the node you want to update, at the right edge of the row, click  to open the inventory menu.
- Step 4.** Select **Update SysName** from the displayed actions list.
- Step 5.** In the editable field, specify the new system name for the node.
- Step 6.** Click **SAVE**.

#### Related topics

[Modifying the system name of nodes in an existing fabric intent](#)  
[Viewing the overall inventory](#)

### 11.1.4.7 Viewing platform details for nodes in the inventory

#### About this task

In the **Inventory** view, you can view specific platform details about each node and the real-world hardware that it is associated with.

For deployed nodes, you can open the **Platform Details** form to display information about the chassis components, including fans, power supply units (PSUs), control processing modules (CPMs), and line cards. Specifically, you can view the operational state for each chassis component. When the system raises an alarm for any of these components, the platform details shows the precise inventory objects that are affected.

Follow this procedure to view specific platform details on a node in the inventory.

#### Procedure

- Step 1.** Open the fabric intent inventory.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.

**Step 3.** For the node you want to view, at the right edge of the row, click  to open the inventory menu.

**Step 4.** Select **Show Platform** from the displayed actions list.



**Note:** The **Show Platform** action is only selectable if the planned node is associated with real hardware and the node is deployed in a fabric intent.

**Step 5.** Observe the details in the form. You can view the operational state for the component.

**Step 6.** Click either the **CLOSE** button or the  at the top of the form to return to the list of nodes.

### What to do next

See [Alarms](#) for more information about how to create and define specific system alarms.

#### Related topics

[Viewing the overall inventory](#)

## 11.1.4.8 Inventory items in spreadsheet format

You can export the information from the inventory into a spreadsheet (.csv ) format. The spreadsheet lists either a complete or partial inventory of devices in the network, depending on the selections you make before you export. The exported fabric information shows a list of device names and the serial numbers associated with those devices. In a fabric intent inventory, you can also export a spreadsheet that lists the mapping of system names in the fabric.

The inventory can contain a combination of both actual equipment and pre-planned equipment that is not yet purchased or incorporated into the network.

A fabric inventory spreadsheet can be used when planning and implementing a network. It can be distributed to personnel who do not have access to Fabric Services System. For example, a spreadsheet can be used for device procurement, internal asset management, or can be distributed to data center contractors who perform physical cabling to build the network.

You can export fabric data from either the overall inventory of fabric items or the fabric intent inventory of a specific fabric intent.

### 11.1.4.8.1 Exporting an overall inventory spreadsheet

#### About this task

You can export a spreadsheet containing all of the information listed in the overall inventory of fabric items. The spreadsheet can be distributed to personnel who do not have access to the Fabric Services System UI.

Follow this procedure to export the overall inventory.

#### Procedure

**Step 1.** Open the overall inventory.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.

**Step 3.** From the inventory, click the  menu at the upper right of the page.

**Step 4.** Select **Export**, then **CSV export** from the resulting list.

**Step 5.** When the download completes, either open the CSV file or save it to your local system.

**Expected outcome**

The resulting spreadsheet displays all of the data shown on the **Inventory** screen.

**Related topics**

[Viewing the overall inventory](#)

## 11.1.4.8.2 Exporting association data from the overall inventory

### About this task

In the overall inventory of fabric items, you can download an associate mapping file to export the details about the inventory items.

Follow this procedure to export multiple items from the overall inventory to a spreadsheet.

### Procedure

**Step 1.** Open the inventory.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.

**Step 3.** Select two or more rows by checking the box at left side of each row.

**Step 4.** Click the **Download Associate Mapping File** icon .

**Step 5.** When the download completes, either open the CSV file or save it to your local system.

**Expected outcome**

The resulting spreadsheet displays information about each inventory item you selected; for related information, see [Spreadsheet information for associations and disassociations](#).

**Related topics**

[Viewing the overall inventory](#)

## 11.1.4.8.3 Exporting association data from a fabric intent inventory

### About this task

In the inventory of a specific fabric intent, you can download an associate mapping file or a system name mapping file to export the details on the inventory items contained within the fabric intent.

Follow this procedure to export multiple items from the fabric intent inventory of a specific fabric intent to a spreadsheet.

### Procedure

**Step 1.** Open the inventory.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.

**Step 3.** Select two or more rows by checking the box at left side of each row.

**Step 4.** Click either the **Download Associate Mapping File** icon  or **Download SysName Mapping File** icon .

**Step 5.** When the download completes, either open the .csv file or save it to your local system.

**Expected outcome**

The resulting spreadsheet displays information about the multiple selected items. If you downloaded the Associate Mapping File, the spreadsheet **name** column displays the name of the items and the **serialNumber** column displays the associated serial numbers (if associated). If you downloaded the SysName Mapping File, the spreadsheet **name** column displays the name of the nodes and the **newSystemName** column displays a blank field where you can add a new system name.

**Related topics**

[Viewing the overall inventory](#)

### 11.1.4.9 Uploads of inventory items

You can upload (import) information from a spreadsheet (CSV) format to the Fabric Services System. The spreadsheet contains information to complete the missing details of planned items in the inventory or update details on existing items. You can upload a spreadsheet of details related to one or more planned fabric items.

After you have exported an inventory spreadsheet and updated it with new information (such as serial numbers, management address, and management profiles to associate specific real-world hardware to each pending item in the inventory), you can import the spreadsheet back into the system to bulk-associate pending nodes with real-world counterparts. Import fabric data from either the overall inventory of fabric items or the fabric intent inventory of a specific intent.

You can also use a spreadsheet to disassociate inventory items from real-world hardware.

**Spreadsheet information for associations and disassociations**

- Associations

An entry for an association must include the following values:

- name
- serial number or IP address
- management profile

- Disassociations

The trigger for a disassociation is the absence of a serial number and IP address; therefore, the entries for disassociation should not include a serial number and IP address. An entry for a disassociation should contain only a name and optionally, a management profile.

If you provide a management profile, note that:

- if management profile has not changed, you can disassociate it from the real hardware and keep the management profile the same.
- If the management profile has changed, the planned node updates the new management profile.
- If the management profile is not present, the planned node deletes the current management profile.

**Related topics**

[Exporting association data from the overall inventory](#)

[Exporting association data from a fabric intent inventory](#)

### 11.1.4.9.1 Uploading association data to the overall inventory

#### About this task

Follow this procedure to upload data about new inventory items installed on the network.

#### Procedure

- Step 1.** Open the inventory.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Click **UPLOAD ASSOCIATE MAPPING FILE**.
- Step 4.** Select the spreadsheet from your local system.
- Step 5.** In the selection window, click **Open**.

#### Related topics

[Viewing the overall inventory](#)

### 11.1.4.9.2 Uploading association data to the fabric intent inventory

#### About this task

Follow this procedure to upload data about new inventory items for a specific fabric intent.

#### Procedure

- Step 1.** Open the fabric intent inventory.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Click **UPLOAD ASSOCIATE MAPPING FILE**.
- Step 4.** Select the spreadsheet from your local system.
- Step 5.** In the selection window, click **Open**.

#### Related topics

[Viewing the overall inventory](#)

### 11.1.4.9.3 Modifying the system name of nodes in an existing fabric intent

#### About this task

If you need to change the system name of items in an existing fabric intent, you can download the System Name Mapping file, update the information, then upload the file to add the node names to the fabric intent. The System Name Mapping file allows you to bulk-update the nodes the inventory instead of updating each individually.

Follow this procedure to modify system name mapping data for a specific fabric intent.

#### Procedure

- Step 1.** Open the fabric intent and download the System Name Mapping file (.csv) by following the procedure in [Exporting association data from a fabric intent inventory](#).

- Step 2.** Open the downloaded file on your local system.  
The spreadsheet displays two columns. The **name** column displays the current names of the nodes and the **newSystemName** column displays a blank field where you can add the new system names.
- Step 3.** In the **newSystemName** column, enter a new name for each node you would like to change.
- Step 4.** Save the updated spreadsheet on your local system.
- Step 5.** Click **UPLOAD SYSNAME MAPPING FILE** and select the updated spreadsheet from your local system to upload the changes.

**Expected outcome**

The names of the nodes in the inventory are updated as specified in the spreadsheet. You can verify the changes immediately in the **Fabric Inventory** view.

## 11.1.5 Viewing the overall inventory

### About this task

There are two types of fabric element inventories: the overall inventory and fabric intent-specific inventories. You can view either the overall inventory of nodes, which is a complete inventory of nodes known to the system, or a subset of these nodes which are contained in a specific fabric intent. Follow this procedure to open the overall inventory known to the system.

### Procedure

- Step 1.** Click  to open the main menu, and select **Inventory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose inventory you want to work with.
- Step 3.** Select **Fabric Elements**.  
You can filter and sort the inventory list according to specific parameters.

### Related topics

[Lists](#)

## 11.2 Management profiles

A management profile defines the connectivity parameters and details that the Fabric Services System uses to connect to a device in the inventory that it needs to manage or communicate with.

A management profile can be one of the following:

- SNMP management profile – used to manage devices via SNMP
- gNMI management profile – used to manage devices via the gRPC Network Management Interface (gNMI)

The system provides default gNMI management profiles. When the Fabric Services System starts, it initializes the following default gNMI profiles:

- **defaultDSGnmiprofile\_22\_6** – Digital Sandbox gNMI profile for SR Linux version earlier than Release 22.11

- **defaultRealGnmiprofile\_22\_6** – real fabric gNMI profile for SR Linux version earlier than Release 22.11
- **defaultDSGnmiprofile** – Digital Sandbox gNMI profile for SR Linux version Release 22.11 and later
- **defaultRealGnmiprofile** – real fabric gNMI profile for SR Linux version Release 22.11 and later

When the Fabric Services System generates the inventory list from an intent topology, the system assigns the default management profile to the nodes using the gNMI.

-  **Note:** The system-defined management profiles listed above are available to all regions. But user-created management profiles are always specific to the region in which they are created, and are not visible within, or available to, other regions.

## 11.2.1 Management profile parameters

Each type of management profile has its own set of parameters that define how the Fabric Services System connects to a device or a set of devices to which the management profile is applied.

The following table describes the parameters that are used to configure a gNMI profile.

Table 47: gNMI profile parameters

Parameter	Description	Values
<b>Name</b>	Specifies the name of the gNMI profile.	String
<b>Description</b>	This parameter describes the profile.	String
<b>Labels</b>	Specifies the labels to apply to this gNMI profile.	Existing label
<b>User</b>	Specifies the user account that the Fabric Services System uses to authenticate against the gNMI server.	String
<b>Password</b>	Specifies the password used to authenticate against the gNMI server.	String
<b>Port</b>	Specifies the port used for gNMI communication to the device. This field cannot be modified.	—

The following table describes the parameters that are used to configure an SNMP profile.

Table 48: SNMP management profile parameters

Parameter	Description	Values
<b>SNMP Version</b>	Specifies the SNMP version.	v3 (the default)
<b>Security Level</b>	Specifies the security level; it is required for SNMP version 3.	noAuthNoPriv, authNoPriv, or authPriv
<b>Privacy protocol</b>	Specifies the privacy protocol; it is required for SNMP version 3.	DES or AES-128-CFB

Parameter	Description	Values
<b>Privacy Key Password</b>	Specifies the privacy key password; it is required for SNMP version 3.	String
<b>User</b>	Specifies the SNMPv3 user configured on the devices; it is required for SNMP version 3.	String
<b>Authentication Key Password</b>	Specifies the authentication key password for this user.	String
<b>Authentication Protocol</b>	Specifies the authentication protocol.	<b>SHA or MD5</b>

## 11.2.2 Creating a management profile

### About this task

Use this procedure to manually create an SNMP or gNMI profile.

### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory** .
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose management profiles you want to work with.
- Step 3.** In the **Management Profiles** view, click **+ CREATE MANAGEMENT PROFILE**.
- Step 4.** Set the basic parameters for the management profile.
  - **Name**
  - **Description**
  - **Type**
- Step 5.** Optional: Assign a label to the management profile.
  - a. Click  above the **Labels** box.
  - b. In the **Label Picker** form, locate the labels that you want to assign, then click **+** at the beginning of its row.
  - c. When you are finished assigning labels, click **SAVE**.
- Step 6.** Set additional management profile parameters in the **Profile Definition** pane.



**Note:** Ensure that you provide the correct syntax for the credentials that you are providing for use by the real node; the Fabric Services System does not validate the syntax.

- Step 7.** When you are finished, click **CREATE**.

### Related topics

[Management profile parameters](#)

## 11.2.3 Editing a management profile

### About this task

You can modify a management profile regardless of whether it is assigned to a real device or not. If the management profile is already associated with a planned node and if real-world devices are using the management profile, then:

- if you attempt to update a management profile and the real-world device is in the In Discovery state, the system rejects the update
- if you attempt to update a management profile and the real-world device is not in the Ready state, the system re-associates the real-world device with the profile
- you cannot change the port value or the name assigned to the profile



**Note:** If you modify a management profile that it assigned to real device, the real device state can change to Unavailable.



**Note:** Changing the gNMI management profile password does not create the users on the devices, nor does it remove the default user. To remove the default username and password from SR Linux, you must do it outside of the Fabric Services System.

### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory** .
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose management profiles you want to work with.
- Step 3.** In the **Management Profiles** view, find the management profile that you want to modify and double-click it.
- Step 4.** In the **Edit Management Profile** window, update the fields that you want to modify. You can update the following fields:
  - **Description**
  - **Labels**
  - **User**
  - **Password**



**Note:** Ensure that you provide the correct syntax for the credentials that you are entering for use by the real node; the Fabric Services System does not validate the syntax.

- Step 5.** When you are finished, click **SAVE**.

### Related topics

[Assigning a label to a management profile](#)

[Removing a label assigned to a management profile](#)

## 11.2.4 Assigning a management profile to a node

### About this task

Use this procedure to associate a management profile to a node.

### Prerequisites

Before you can update a user-defined gNMI management profile in and SR Linux node, complete the following tasks:

1. Deploy the target node.
2. Change the user credential on the target node.



**Note:** Do not apply user-defined gNMI management profile while a fabric intent in candidate mode.

3. When the credential change appears as a deviation, accept the deviation as a read-only system GCO; see [Deviations](#).



**Note:** Do not change credentials on the target node while the fabric intent in candidate mode.

You can now change the management profile in the inventory.

### Procedure

- Step 1.** Click  to open the main menu, then select **Inventory**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region whose management profiles you want to work with.
- Step 3.** In the **Fabric Elements** view, find the node that you want to associate.
- Step 4.** Click  at the right edge of its row and select **Assign Management Profile**.
- Step 5.** Click the management profile that you want to assign to the node, then click **ASSIGN**.



**Note:** The original default gNMI profile created by the system is not deleted on real node. If needed, change credentials of the default gNMI on the real node.



**Note:** Deleting a gNMI profile on the SR Linux and then adding it back later triggers a Fabric Services System deviation, because the SR Linux password hashing is different. You must accept the deviation.

### What to do next

Repeat these steps for each node for which you want to assign a profile.

## 11.2.5 Deleting a management profile

### About this task

You can only delete a manually created profile.

- You cannot delete the Fabric Services System default management profiles.
- You cannot delete a management profile if it has already been associated.

**Procedure**

- Step 1.** In the **Inventory** → **Management Profiles** view, locate the management profile that you want to delete.
- Step 2.** Click  at the right edge of the row, then select **Delete**.
- Step 3.** In the **Delete Confirmation** form, click **OK**.

## 12 Alarms

In the Fabric Services System, alarms arise when a managed object enters an undesirable state; for example, if a managed node goes out of service, an associated alarm is raised.

The Fabric Services System supports the following alarm types, each with its own subtypes:

- Communication
- Configuration
- Environment
- Equipment
- Operational

The Fabric Services System includes the following tools that can help you manage alarms:

- the **Alarms** panel on the dashboard, which summarizes current alarms.
- the **Alarms List** page, which you can use to view and manage individual alarms
- the policy manager, which you can use to customize the severity level for specific types of alarm or to suppress alarms of a specific type entirely.

### Alarm states

In the Fabric Services System, an alarm can adopt the following states:

- **Acknowledged:** An acknowledged alarm still displays in the **Alarms List** page. When viewing details for the individual alarm, its state displays as Acknowledged and any note you added to the alarm while acknowledging it is displayed as well. You can use the Acknowledge state as the basis for filtering or sorting the alarm list.
- **Closed:** A Closed alarm still displays in the **Alarms List** page. This state can be the basis for filtering the alarms included in the list. Closing an alarm does not resolve the condition that caused the alarm to be raised in the first place.
- **Cleared:** An alarm is Cleared when the condition that raised the alarm has been resolved. Unlike Acknowledged and Closed, the Cleared state cannot be assigned manually by a Fabric Services System operator. Only the device or devices that raised the original alarm can determine and communicate its closure.

### Related topics

[The dashboard](#)

[Appendix A: Supported alarms](#)

## 12.1 Displaying alarms

### About this task

The **Alarms List** view displays a list of current alarms known to the Fabric Services System. From this page, you can view details about the state of each alarm and also acknowledge any alarm.

To view and manage alarms with the **Alarm List** page:

## Procedure

**Step 1.** From the main menu, select **Alarms List**.

### Expected outcome

The alarm list displays, showing all active alarms for the current region (where "active" refers to alarms that have not been cleared).



**Note:** Cleared alarms are not included in this list because the "Cleared" filter for this display is set to "False" by default. To view cleared alarms in this list, clear that filter.



**Note:** A set of default columns display in the **Alarms List** view:

- Severity
- Alarm type
- Node name
- Resource name
- Cleared
- Occurrence
- Last Raised

There are other columns available to show more information about each alarm. You can add or remove columns from any list.

**Step 2.** If required, use the **Region Selector** at the top of the page to select a different region whose alarms are displayed.

**Step 3.** To view details about an alarm and its state:

- Select an alarm in the list.
- At the right edge of the row, click  and select **State Details** from the displayed action list.
- Click the **ALARM STATE** tab to view details about the alarm's severity, a description of the alarm, and the time it was raised.
- Click the **OPERATOR STATE** tab to view the state assigned by the operator to address the alarm (either Acknowledged or Closed).
- When you are finished, click **CLOSE** to return to the **Alarms List** page.

**Step 4.** To acknowledge an alarm:

Acknowledging an alarm marks it as received, but does not clear the alarm from the alarm list.

- Select an alarm in the list.
- At the right edge of the row, click  and select **Acknowledge** from the displayed action list.
- Optionally, enter any comments about the acknowledgement in the **Additional Info** field.
- Click **SAVE**.

### Expected outcome

The alarm is marked as Acknowledged (but not Closed).

**Step 5.** To close an alarm:

Closing an alarm prevents the alarm from appearing in the **Alarms List** page, but does not resolve the condition that raised the alarm in the first place.

- a. Select an alarm from the list.
- b. At the right edge of the row, click  and select **Close** from the displayed action list.
- c. Optionally, enter any comments about the acknowledgement in the **Additional Info** field.



**Note:** The text you enter here is displayed in the **Additional Info** column of the **OPERATOR STATE** tab in the **Alarm Details** overlay.

- d. Click **SAVE**.

### Related topics

[Lists](#)

## 12.2 Customizing an alarm severity level

### About this task

Policies allow you to customize the severity level associated with individual supported alarms. A policy affects the alarm type of all future alarms raised; it does not retroactively modify existing alarms of the same type.

Each policy can include a start time and an end time; these are boundaries on the time of day during which the policy applies. An alarm raised outside these boundaries has its default severity instead of the severity level defined by the policy. If no start and end times are defined, the policy is always active.

You can also use a policy to suppress an alarm entirely while the policy is in effect.

The Fabric Services System supports the definition of a policy's scope in two mutually exclusive ways:

- by key value, which allows you to trigger a policy based on the name of the object (node, fabric, intent, or region) affected by the alarm
- by alarm category and type, to apply the policy regardless of the object affected.



**Note:** All policies are specific to the region in which they are created. A policy that is created within one region is not visible within, or available to, other regions.

To customize an alarm's severity:

### Procedure

- Step 1.** From the main menu, select **Policies**.
- Step 2.** Use the **Region Selector** at the top of the page to select the region in which to create the policy.
- Step 3.** Click **+ CREATE A POLICY**.
- Step 4.** Set the **Name** and **Description** fields for the policy.
- Step 5.** In the **Policy Definition** panel, set the **Start Time** and **End Time** fields.  
An alarm that would be affected by this policy uses the customized severity level only if it is raised during this period. If it is raised outside this period, it uses the default severity level.
- Step 6.** Do one of the following:

- To configure a policy based on the object it affects, click the **Key Value** toggle to enable it and go to step 7.
- To configure a policy based on alarm category and type, leave the **Key Value** toggle disabled and go to step 8.

**Step 7.** With the **Key Value** toggle enabled, do the following:

a. Click the **Key Value Objects** drop-down list and select one or more of the displayed key value candidates:

- Node Name
- Fabric Name
- Intent Name
- Region Name

**Expected outcome**

For each selected item, a field displays.

- b. Use the displayed field or fields to provide the unique name of each object type you selected. This name identifies the unique object of that type for which the system changes the alarm severity or suppresses alarms, depending on how you configure the remainder of the policy.
- c. Go to step 9.

**Step 8.** In the **Policy Definition** panel, do the following:

a. Click the **Alarm Category** drop-down list and select from the following values:

- Communication
- Equipment
- Operational
- FSS

b. Click the **Alarm Type** drop-down list and check the box beside one or more alarms types in the displayed list.

c. Select a value for the **Alarm Severity** field:

- Major
- Minor
- Critical
- Warning
- Default

**Step 9.** Configure the way this policy modifies alarms:

a. Click the **Priority** drop-down list and select a value from 0 to 9, with 0 being the highest priority.

b. Optionally, enable **Suppress Alarms** toggle.

Enabling this option means that alarms of this type are disabled and are not triggered while the policy is in effect.

c. Optionally, enable the **Deployed Intent Alarms Only** toggle.

This option applies to "Communication – Interface Down" alarms. If this option is enabled, alarms are raised only on interfaces that are part of the intent configuration.

**Step 10.** Click **CREATE**.

## 12.3 Third-party tool access to Fabric Services System alarms

You can configure the system to allow third-party tools to access Fabric System Services alarms to allow operators to use their operational tool sets to monitor and operate their network. The Fabric Services System exposes raised alarms to third-party tools through a Kafka message bus. The system publishes all generated alarms on a Kafka topic to which an external system can subscribe.

The Kafka broker used for this topic only exposes SSL connections to itself for external systems to use. An external client must authenticate before being able to subscribe to a topic. The Kafka broker allows the external client to only subscribe to the topic, but not publish to it.

### Alarm messages

The alarm messages that are published to the Kafka topic are in Protocol Buffer (protobuf) format. For an example, see [Appendix B: Protobuf file message format](#).

From the Fabric Services System, you can obtain this file using the following REST call:

**`https://fss.domain.tld/rest/alarmmgr/fss_alarmexternal.proto`**

### Configuration

The settings that enable third-party tools to access Fabric Services System alarms are configured during the Fabric Services System application installation; for more information, see "Editing the installation configuration file" in the *Fabric Services System Software Installation Guide*.

After the Fabric Services System application has been installed, you can update the settings as described in [Updating configuration for the external Kafka service](#).

### 12.3.1 Enabling Kafka alarms after software installation

#### Prerequisites

Perform this procedure only during a maintenance window.

#### About this task

Use this procedure to enable Kafka alarms after the Fabric Services System application has been installed. Perform this procedure only if the generation of Kafka alarms was not configured during initial installation or software upgrade.

#### Procedure

**Step 1.** Update the `sample-input.json` file with the Kafka alarm settings:

In the `fss` section of the `sample-input.json` file, add the following lines:

```
"kafkaconfig": {
  "port": "31000",
  "groupprefix": "fsskafka",
  "user": "fssalarms",
```

```
"password": "fssalarms",
"maxConnections": 2
```

- Step 2.** Update the system configuration.  
Execute the following commands:

```
/root/bin/fss-install.sh configure input.json
/root/bin/fss-upgrade.sh upgrade
/root/bin/update-kafka.sh
```

## 12.3.2 Updating configuration for the external Kafka service

### Prerequisites

- You must perform this procedure during a maintenance window.
- All external connections must be closed before executing this procedure; you can initiate the connections again after you have completed this procedure.

### Procedure

- Step 1.** Update the `sample-input.json` file.

The parameters are in `kafkaconfig` sub-section of the `fss` section.

```
"fss": {
  "dhcpnode": "fss-node01",
  "dhcpinterface": "192.0.2.11/24",
  "ztpaddress": "192.0.2.11",
  "httpsenabled": true,
  "certificate": "/root/certs/fss-tls.crt",
  "privatekey": "/root/certs/fss-tls.key",
  "domainhost": "myhost.mydomain.com",
  "kafkaconfig": {
    "port": "32425",
    "groupprefix": "mygrp",
    "user": "myuser",
    "password": "mypasswd",
    "maxConnections": 2
  }
}
```

Currently, you can only change the setting for the **maxConnections** parameter. This parameter specifies the maximum number of clients that can connect to the Kafka service; the maximum value for this parameter is 10.

### Example

```
[root@fss-deployer ~]# diff updated-input.json input.json
<     "maxConnections": 3
---
>     "maxConnections": 2
```

- Step 2.** Run the `fss-install.sh` script to update the system configuration.  
The `fss-install.sh` script is available in the `/root/bin` directory.

**Example**

```
[root@fss-deployer ~]# /root/bin/fss-install.sh configure updated-input.json
WARNING: truststore not configured
  Timesync service is running on 10.254.45.123 Time difference is 0 seconds
  Timesync service is running on 10.254.44.123 Time difference is 0 seconds
  Timesync service is running on 10.254.43.123 Time difference is -1 seconds
  Timesync service is running on 10.254.42.123 Time difference is 0 seconds
  Timesync service is running on 10.254.41.123 Time difference is 0 seconds
  Timesync service is running on 10.254.40.123 Time difference is 0 seconds
  Maximum time difference between nodes 1 seconds
WARNING: Storage related disks will be wiped clean during install, data will be lost.
Please verify that correct disks are referred in the input configuration.
```

**Step 3.** Update the Kafka service.**Example**

```
[root@fss-deployer ~]# /root/bin/update-kafka.sh
Kafka will be updated with the current config.
release "kafka" uninstalled
Using User certificates for the cluster
secret "kafka-fss-cluster-ca-cert" deleted
secret/kafka-fss-clients-ca-cert created
secret/kafka-fss-cluster-ca-cert created
secret "kafka-fss-cluster-ca" deleted
secret/kafka-fss-cluster-ca created
secret/kafka-fss-clients-ca created
secret/kafka-fss-cluster-ca-cert labeled
secret/kafka-fss-clients-ca-cert labeled
secret/kafka-fss-cluster-ca labeled
secret/kafka-fss-clients-ca labeled
secret/kafka-fss-cluster-ca-cert annotated
secret/kafka-fss-clients-ca-cert annotated
secret/kafka-fss-cluster-ca annotated
secret/kafka-fss-clients-ca annotated
NAME: kafka
LAST DEPLOYED: Fri Mar 31 05:03:05 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
Fri Mar 31 05:03:07 UTC 2023 Start: Checking Kafka pods status
Fri Mar 31 05:03:07 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:03:18 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:03:28 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:03:39 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:03:49 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:04:00 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:04:10 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:04:52 UTC 2023 wait 800s for kafka cluster to startup
Fri Mar 31 05:05:02 UTC 2023 Kafka Operator is up

NAME                                READY   STATUS    RESTARTS   AGE
kafka-fss-entity-operator-b6757b664-bvvpq   3/3     Running   0           37s
kafka-fss-kafka-0                          1/1     Running   0           71s
kafka-fss-kafka-1                          1/1     Running   0           71s
kafka-fss-kafka-2                          1/1     Running   0           71s
kafka-fss-zookeeper-0                     1/1     Running   0          115s
kafka-fss-zookeeper-1                     1/1     Running   0          115s
kafka-fss-zookeeper-2                     1/1     Running   0          115s
strimzi-cluster-operator-5bc66cb4f9-dnkcw   1/1     Running   0           12h

NAME                                CLUSTER   AUTHENTICATION   AUTHORIZATION   READY
```

fss-kafka-admin	kafka-fss	scram-sha-512	simple	True		
myuser	kafka-fss	scram-sha-512	simple	True		
NAME			TYPE			DATA
AGE						
default-token-tr6nz			kubernetes.io/service-account-token			3
12h						
fss-kafka-admin			Opaque			2
116s						
kafka-fss-clients-ca			Opaque			1
2m1s						
kafka-fss-clients-ca-cert			Opaque			3
2m3s						
kafka-fss-cluster-ca			Opaque			1
2m1s						
kafka-fss-cluster-ca-cert			Opaque			3
2m2s						
kafka-fss-cluster-operator-certs			Opaque			4
115s						
kafka-fss-entity-operator-token-zhz2r			kubernetes.io/service-account-token			3
37s						
kafka-fss-entity-topic-operator-certs			Opaque			4
37s						
kafka-fss-entity-user-operator-certs			Opaque			4
37s						
kafka-fss-kafka-brokers			Opaque			12
71s						
kafka-fss-kafka-token-52ddx			kubernetes.io/service-account-token			3
72s						
kafka-fss-zookeeper-nodes			Opaque			12
115s						
kafka-fss-zookeeper-token-xfvfb			kubernetes.io/service-account-token			3
115s						
myuser			Opaque			2
116s						
sh.helm.release.v1.kafkaop.v1			helm.sh/release.v1			1
12h						
strimzi-cluster-operator-token-q5kkt			kubernetes.io/service-account-token			3
12h						
NAME			TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
		AGE				
kafka-fss-kafka-0			NodePort	10.233.61.12	<none>	
9094:31239/TCP			72s			
kafka-fss-kafka-1			NodePort	10.233.5.58	<none>	
9094:30182/TCP			72s			
kafka-fss-kafka-2			NodePort	10.233.56.222	<none>	
9094:30026/TCP			72s			
kafka-fss-kafka-bootstrap			ClusterIP	10.233.34.56	<none>	9091/
TCP,9092/TCP,9093/TCP		72s				
kafka-fss-kafka-brokers			ClusterIP	None	<none>	9090/
TCP,9091/TCP,9092/TCP,9093/TCP		72s				
kafka-fss-kafka-external-bootstrap			NodePort	10.233.45.207	<none>	
9094:32425/TCP			72s			
kafka-fss-zookeeper-client			ClusterIP	10.233.21.201	<none>	2181/
TCP		115s				
kafka-fss-zookeeper-nodes			ClusterIP	None	<none>	2181/
TCP,2888/TCP,3888/TCP		115s				

**Step 4.** Wait for the Fabric Services System application to stabilize.  
Convergence may take some time. During this period, pods are known to fail and restart.

**Example**

The system is stable when all the pods are in Running state.

```
[root@fss-deployer ~]# export KUBECONFIG=/var/lib/fss/config.fss
[root@fss-deployer ~]# kubectl get pods
```

NAME	READY	STATUS	RESTARTS
AGE			
fss-logs-fluent-bit-56t99 12h	1/1	Running	0
fss-logs-fluent-bit-d94x2 12h	1/1	Running	0
fss-logs-fluent-bit-hbvzt 12h	1/1	Running	0
fss-logs-fluent-bit-q7f6g 12h	1/1	Running	0
fss-logs-fluent-bit-r5tr4 12h	1/1	Running	0
fss-logs-fluent-bit-tmldd 12h	1/1	Running	0
prod-ds-apiserver-88fcd7cd7-lhmhh 12h	1/1	Running	0
prod-ds-cli-7cfd7664db-6xhk5 12h	1/1	Running	0
prod-ds-docker-registry-5b467bbf67-4lh2z 12h	1/1	Running	0
prod-ds-imsvc-deploy-5f99648577-fjfdg 12h	1/1	Running	0
prod-fss-alarmmgr-78fd576464-2tfl9 12h	1/1	Running	1 (2m19s ago)
prod-fss-auth-6c99d44ccb-tnt8t 12h	1/1	Running	1 (3m20s ago)
prod-fss-catalog-54cb57645-s6mj7 12h	1/1	Running	1 (2m50s ago)
prod-fss-cfggen-6dfc6d8ccb-rjmx 12h	1/1	Running	1 (2m49s ago)
prod-fss-cfgsync-78df54976f-nqrcm 12h	1/1	Running	0
prod-fss-connect-58c98db7d4-x4w5g 12h	1/1	Running	1 (3m18s ago)
prod-fss-da-0 12h	1/1	Running	1 (2m20s ago)
prod-fss-da-1 12h	1/1	Running	1 (2m20s ago)
prod-fss-da-2 12h	1/1	Running	1 (2m20s ago)
prod-fss-da-3 12h	1/1	Running	1 (2m20s ago)
prod-fss-da-4 12h	1/1	Running	1 (2m18s ago)
prod-fss-da-5 12h	1/1	Running	1 (2m48s ago)
prod-fss-da-6 12h	1/1	Running	1 (2m18s ago)
prod-fss-da-7 12h	1/1	Running	1 (2m18s ago)
prod-fss-deviationmgr-acl-7d8d878d66-jc48z 12h	1/1	Running	0
prod-fss-deviationmgr-bfd-5f6bcf7d-xsq46 12h	1/1	Running	0
prod-fss-deviationmgr-interface-5f7fdcf6c-fpk48 12h	1/1	Running	0
prod-fss-deviationmgr-netinst-c7d5648d7-z9mdp 12h	1/1	Running	0

prod-fss-deviationmgr-platform-6d9c574bb9-l4cb7 12h	1/1	Running	0
prod-fss-deviationmgr-qos-5b99fcc7d9-977r6 12h	1/1	Running	0
prod-fss-deviationmgr-routingpolicy-775f49b66-qnrqj 12h	1/1	Running	0
prod-fss-deviationmgr-system-557bbbc75f-rjknq 12h	1/1	Running	0
prod-fss-dhcp-5bc95b6966-kzd2n 12h	1/1	Running	0
prod-fss-dhcp6-69d8785d64-l4qdk 12h	1/1	Running	0
prod-fss-digitalsandbox-5c44679f86-4bp8p 12h	1/1	Running	1 (2m50s ago)
prod-fss-filemgr-65c6799996-ggl27 12h	1/1	Running	0
prod-fss-imagmgr-fd97fc4fb-6w8t4 12h	1/1	Running	1 (2m50s ago)
prod-fss-intentmgr-64f97dc466-ftjgm 12h	1/1	Running	1 (2m20s ago)
prod-fss-inventory-6f84769f46-w8h97 12h	1/1	Running	1 (3m18s ago)
prod-fss-labelmgr-847575b8c6-4m8xj 12h	1/1	Running	1 (3m19s ago)
prod-fss-maintmgr-7f599dd5db-fqk29 12h	1/1	Running	1 (2m20s ago)
prod-fss-mgmtstack-79c67c585c-pk2nv 12h	1/1	Running	1 (2m20s ago)
prod-fss-oper-da-0 12h	1/1	Running	1 (2m20s ago)
prod-fss-oper-da-1 12h	1/1	Running	1 (2m20s ago)
prod-fss-oper-da-2 12h	1/1	Running	1 (2m20s ago)
prod-fss-oper-da-3 12h	1/1	Running	1 (2m20s ago)
prod-fss-oper-da-4 12h	1/1	Running	1 (2m19s ago)
prod-fss-oper-da-5 12h	1/1	Running	1 (2m18s ago)
prod-fss-oper-da-6 12h	1/1	Running	1 (2m18s ago)
prod-fss-oper-da-7 12h	1/1	Running	1 (2m18s ago)
prod-fss-oper-topomgr-6b848bbcf7-5z8c9 12h	1/1	Running	1 (2m19s ago)
prod-fss-protocolmgr-776bdf59c7-zvfl2 12h	1/1	Running	0
prod-fss-topomgr-5dd97997b8-jw8rk 12h	1/1	Running	1 (2m19s ago)
prod-fss-transaction-79bdb7d78d-lxwpp 12h	1/1	Running	1 (2m50s ago)
prod-fss-version-767b859c96-t2v5w 12h	1/1	Running	1 (2m20s ago)
prod-fss-web-5c94fd7455-l4sfz 12h	1/1	Running	1 (2m20s ago)
prod-fss-workloadmgr-7b8f44b95d-f8cv6 12h	1/1	Running	1 (3m19s ago)
prod-fss-ztp-86cbf5cdc-ctx9q 12h	1/1	Running	1 (2m49s ago)
prod-keycloak-0 12h	1/1	Running	0
prod-mongodb-arbiter-0 12h	1/1	Running	0

```
prod-mongodb-primary-0      1/1    Running  0
 12h
prod-mongodb-secondary-0   1/1    Running  0
 12h
prod-neo4j-core-0          1/1    Running  0
 12h
prod-postgresql-0          1/1    Running  0
 12h
prod-sftpserver-77cd8696d5-fxsw 1/1    Running  0
 12h
[root@6node-deployer-vm ~]#
```

**Step 5.** Initiate external connections from Kafka clients.

## 13 Operations views

To help you assess the overall health of your fabrics, the Fabric Services System includes some high-level perspectives on the current status of all managed regions and fabrics. These views are intended to help you assess at a glance what is working and what is not, what is configured or misconfigured, and to understand what other related objects could be affected by an unhealthy entity.

From these high-level views that begin at the regions level, you can drill down to see the health of individual fabrics, nodes, and links within a region.

Health is a reflection of the operational state of nodes, links, and any node-based alarms within the set of managed fabrics. Although health alarms are distinct from intent deviations and deployment status, they may be related.

You can access the main operational view by selecting **Operation and Health Insights** from the system's main menu. The resulting page includes two main overlays:

-  **Health**, which displays information about objects subject to alarms, and their operational state (Up, Down, or Missing Hardware)
-  **Operational Deviation**, which indicates where there are errors within the physical topology of a fabric.

In each case, this information is conveyed through a combination of graphical display for overall status, and the information panel on the right side of the page that provides more details.

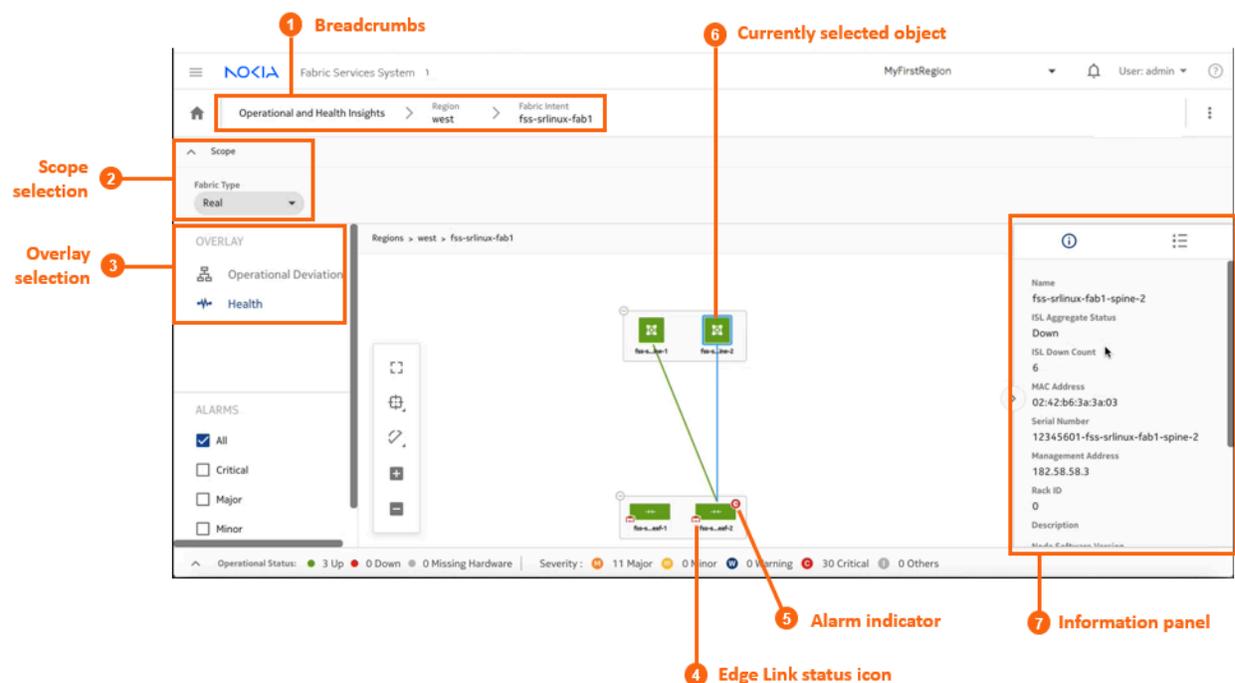


Table 49: Operational health display

#	Description
1	Breadcrumbs: as you drill down into a region, this displays the level of your current view and the sequence of objects between that view and the region level.
2	Scope selection: use this drop-down list to control whether the view displays Real objects, or simulated Digital Sandbox objects.
3	Overlay selection: use this panel to select one of the available views, either Health or Operational Deviations.
4	Edge Link status icon: this icon is only present on Leaf nodes, and indicates the presence (and status) of edge links leading to customer assets. Possible statuses are Up (green) or Down (red).
5	Alarm indicator: this icon indicates the aggregate alarm level for an object. If absent, there are no alarms on the object or its child objects. If present, it indicates (by its color and letter) the single highest-level severity of any alarms that are active on the object or any of its children.
6	Currently selected object: a light blue outline indicates the currently selected object within the view. The same light blue is superimposed on any ISLs associated with that object.
7	Information panel: open this panel by clicking the right of the page. The panel displays information about the currently selected object. Of particular interest are the aggregate status and count indicators: <ul style="list-style-type: none"> <li>• For a region: Node Aggregate Status and Node Down Count</li> <li>• For a fabric: Node Aggregate Status, Node Down Count, Interface Aggregate Status, Interface Down Count</li> <li>• For spine nodes: ISL Aggregate Status and ISL Down Count</li> <li>• For leaf nodes: ISL Aggregate Status, ISL Down Count, Edge Link Aggregate Status, and Edge Link Down Count.</li> </ul>

## The Health overlay

-  **Health**, which displays information about:
  - which objects that are subject to alarms
  - for parent objects, the single highest-severity alarm on any of its child objects
  - which nodes and links are down
  - for parent objects the proportion of immediate child objects that are up or down
-  **Operational Deviation**, which indicates where there are errors within the physical topology of a fabric. A deviation in this context represents:
  - a port that should be connected to another port, but is not
  - a port that should not be connected to another port, but is
  - a port that should be connected to a specific other port, but is connected to a different port instead

Displays on the Health overlay follow the conventions for other maps in the Fabric Services System UI. The possible states that can be displayed vary depending on whether you are looking at the Health page or the Operational Deviations page.

Table 50: Operational states and alarm severities

Information type	Display
Operational states (Health)	   Operational Up      Operational Down      Missing Hardware
Operational states (Deviations)	   Intent Matching      Operational Deviation      Missing Hardware
Alarm severities	     Major      Minor      Warning      Critical      Other (Information)

### The Operational Deviation overlay

The **Operational Deviation** overlay indicates where there are errors within the physical topology of a fabric. A deviation in this context represents:

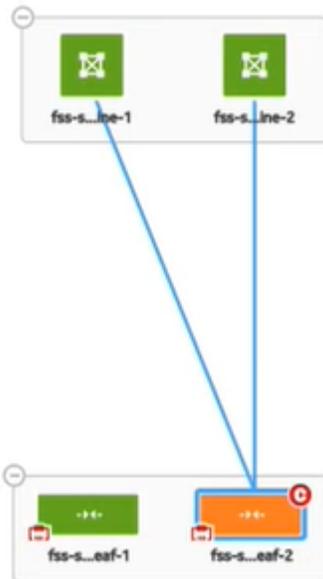
- a port that should be connected to another port, but is not
- a port that should not be connected to another port, but is
- a port that should be connected to a specific other port, but is connected to a different port instead

Any object that is the subject of deviations is shaded orange in the overlay topology.

Figure 34: Operational deviation on a fabric



Figure 35: Operational deviation on a leaf node



## 13.1 Viewing the operational topology

### Procedure

**Step 1.** From the main menu, select **Operational and Health Insights**.

#### Expected outcome

The **Operational and Health Insights** page displays, showing a map of all configured regions.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose data should display.

**Step 3.** In the **OVERLAY** drop-down list, select **Health**.

**Step 4.** Optional: Select the type of fabrics to display by making a selection in the **Fabric Type** drop-down list.

The drop-down list supports two fabric types:

- Real (default)
- Digital Sandbox

**Step 5.** To learn more about any region on the map, do the following:

**a.** Check the circle in the center of the region.

If the circle contains any red, that indicates the proportion of fabrics within the region that have operational issues. You can expand the region and view more information about its status by going to sub-step 5.b.

**b.** Click the region icon to select it, then click the **i** icon at the right of the page to open the information panel.

The information panel displays more information about the selected region, including the number of fabrics and number of links it contains.

**Step 6.** To learn more about the fabrics within the region, do the following:

a. Expand the region icon.

**Expected outcome**

The region icon expands to show all of the fabrics it contains. Fabrics that contain nodes with operational issues are shaded red.

b. Select a fabric and, if the information panel is not open, click the ⓘ icon at the right of the page to open the information panel.

The information panel displays more information about the selected fabric. Pay particular attention to the following values:

- **Node Aggregate Status:** summarizes the overall state of nodes within the fabric
- **Node Down Count:** the number of nodes within the fabric that are currently operationally Down
- **Interface Down Aggregate Status:** provides some information about the proportion of interfaces within with the fabric that are experiencing operational issues
- **Interface Down Count:** the number of interfaces within the fabric that are experiencing operational issues

**Step 7.** To navigate to another view to learn more about an individual fabric, right-click the fabric on the map and select one of the following from the resulting contextual menu:

- **Open Fabric Design** to open the fabric intent's Design view
- **Open Fabric Inventory** to open the fabric intent's Inventory view
- **Show Intent Alarms** to open the Alarms List

**Expected outcome**

The selected view displays, replacing the **Regions and Health** view. The new view adopts the context of the fabric whose row you selected.

**Step 8.** To learn more about the spine nodes within a single fabric, do the following:

a. Double-click the fabric icon on the **Operations and Health Insights** map.

**Expected outcome**

The map shows the topology of the selected fabric, including any backbone, spine, and leaf groups.

b. Select and expand a spine cluster.

**Expected outcome**

The expanded group shows the individual nodes within that cluster. Node icons are shaded green if Up, red if Down, and gray if Unassociated with hardware.

c. Select an individual spine node within the expanded group and, if the information panel is not open, click the ⓘ icon at the right of the page to open the information panel.

The information panel displays more information about the selected node. Pay particular attention to the following values:

- **ISL Aggregate Status:** provides some information about the proportion of ISLs associated with the selected node that are experiencing operational issues

- **ISL Down Count:** the precise number of ISLs associated with the selected node that are experiencing operational issues

**Step 9.** To learn more about a leaf nodes within a single fabric, do the following:

- a. Select and expand a leaf group.

**Expected outcome**

The expanded group shows the individual nodes within that cluster. Node icons are shaded green if Up, red if Down, and gray if Unassociated with hardware. Similarly, the Edge Link icon at the bottom right of the leaf node icon is shaded green if all edge links are Up, or red if at least some edge links are Down.

- b. Select an individual leaf node within the expanded group and, if the information panel is not open, click the ⓘ icon at the right of the page to open the information panel.

The information panel displays more information about the selected node. Pay particular attention to the following values:

- **Edge Link Aggregate Status:** provides some information about the proportion of edge links associated with the selected node that are experiencing operational issues
- **Edge Link Down Count:** the precise number of edge links associated with the selected node that are experiencing operational issues
- **ISL Aggregate Status:** provides some information about the proportion of ISLs associated with the selected node that are experiencing operational issues
- **ISL Down Count:** the precise number of ISLs associated with the selected node that are experiencing operational issues

## 13.2 Viewing operational insights

### Procedure

**Step 1.** From the main menu, select **Operational and Health Insights**.

**Step 2.** Use the **Region Selector** at the top of the page to select the region whose data should display.

**Step 3.** Optional: Select the type of fabrics to display by making a selection in the **Fabric Type** drop-down list.

The drop-down list supports two fabric types:

- Real (default)
- Digital Sandbox

**Step 4.** In the **OVERLAY** drop-down list, click  **Operational Deviation**.

**Step 5.** Expand the region icon.

**Expected outcome**

The region icon expands to show all of the fabrics it contains. Fabrics that contain nodes that are the subject of deviations are shaded orange.

**Step 6.** Double-click a fabric icon on the **Operations and Health Insights** map.

**Expected outcome**

The expanded fabric cluster shows the individual nodes within that cluster. Nodes that are the subject of deviations are shaded orange.

## 14 Network resources

The Network Resources page of the Fabric Services System UI allows you to create and manage the set of IP pools and Autonomous System numbers (ASNs) available for use by fabric intents within a region.

As part of region creation, you configure a single, default instance of each of the following IP address pools (or in the case of Autonomous Systems, value ranges) for use with any fabrics within the region:

- System IP pools
- Inter Switch Link (ISL) IP pools
- Out of Band Management IP pool



**Note:** The option to configure Out of Band Management IP pools is only available if you configured your region to use an internal DHCP server.

- Autonomous System pools

From the Network Resources page you can configure additional IP address pools. These additional pools are also associated with the region, and can be assigned to any fabric intent you create within the region.

From the same page you can view the set of available pools, including the default pool that is configured as part of the region. You can also modify and delete these pools, subject to some restrictions.



**Note:** All IP pools and ASN pools are specific to the region in which they are created. A pool that is created within one region is not visible within, or available to, other regions. Despite this, there cannot be any overlapping management IP CIDR blocks across the IP management pools of all regions.

### Overlapping IP pools and backbone fabrics

Every Management IP pool must be unique and must not overlap with any other Management IP pool. The Fabric Services System checks each Management IP pool that you create to ensure that it does not overlap with any other.

However, the Fabric Services System does not prevent you from creating different pools with overlapping IP addresses for the following pool types associated with the data plane:

- System IP pools
- ISL pools

Such overlapping pools can generally coexist without any issue. But when creating a backbone fabric, be sure to select only a non-overlapping System IP pool and a non-overlapping ISL pool.



**Note:** The overlaps described here are tolerated only for certain types of IP address pools; not for IP blocks within a pool. No IP address block within a pool of any type is permitted to overlap any other block in the same pool.

## 14.1 The Network Resources page

From the **Network Resources** page you can create and manage IP pools (or in the case of Autonomous System pools, ranges of numeric values):

- review already-configured IP pools or ASN value ranges
- add new IP pools or ASN value ranges
- modify existing IP pools or ASN value ranges (with some restrictions)
- delete existing IP pools or ASN value ranges (with some restrictions)

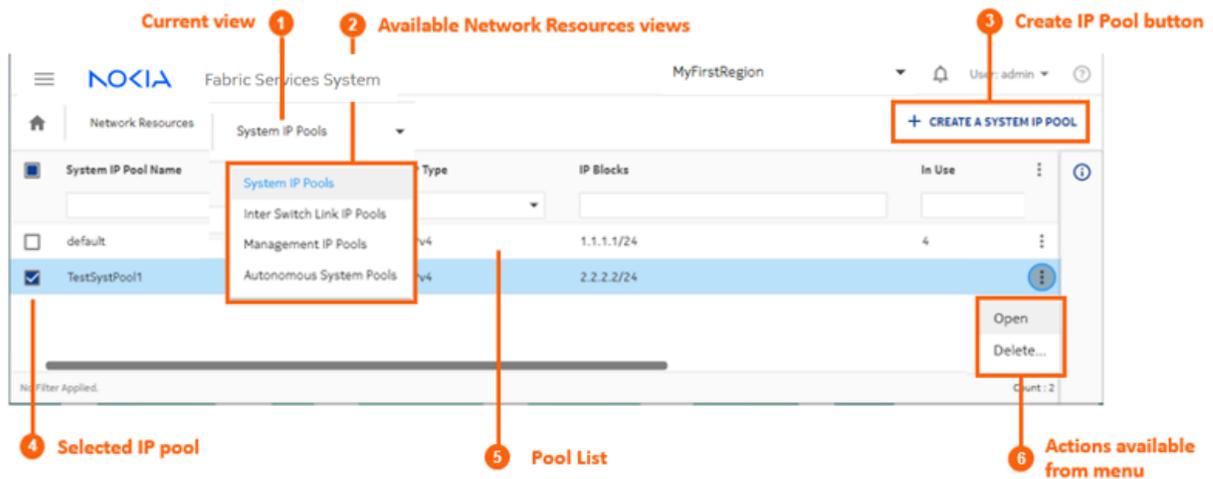


Table 51: Network Resources page elements

#	Description
1	Current view: of the available views from this page, this identifies the view currently displayed.
2	Available views: several different views available from this page allow you to interact with different IP pools: <ul style="list-style-type: none"> <li>• <b>System IP Pools</b>: the basic view showing a map of the fabric topology</li> <li>• <b>Inter Switch Link IP Pools</b></li> <li>• <b>Management IP Pools</b></li> <li>• <b>Autonomous System Pools</b></li> </ul>
3	The <b>Create IP Pool</b> action button allows you to create an IP pool of the type matching the current view.
4	Any selected IP pools of the current type display a check here.
5	The pool list displays information about each pool: <ul style="list-style-type: none"> <li>• <b>Name</b>: the name configured when creating the pool.</li> </ul>

#	Description
	<ul style="list-style-type: none"> <li>• <b>IP Type:</b> the type of IP address (IPv4 or IPv6) for the current pool. Does not apply to ASN pools.</li> <li>• <b>IP Blocks</b> or <b>ASN Range Blocks:</b> the blocks and their ranges currently configured within the pool.</li> <li>• <b>In Use:</b> Indicates the number of IP addresses or ASN values from the pool that are currently in use by fabrics managed by the system.</li> <li>• <b>Description:</b> the description configured when creating the pool.</li> <li>• <b>Intents Associated:</b> Displays the set of intents that use IP addresses from the pool.</li> </ul>
6	Actions available from menu: you can perform one of the available actions on the currently selected IP pool or pools.

## 14.2 Network Resource properties

The Network Resources page allows you to configure a set of IP pools (or in the case of AS pools, numeric values) available to all fabric intents created within the associated region.

The properties that are available depend on which view is currently selected:

- **System IP Pools**
- **Inter Switch Link IP Pools**
- **Management IP Pools**
- **Autonomous System Pools**

Table 52: Network resource properties

Property	Description
System IP Pool	<b>Pool name:</b> provides the name by which this pool will appear and be selected.
	<b>Description:</b> optionally, describes the nature of the pool.
	<b>IP Type:</b> set as "IPv4" and cannot be changed.
	<b>IP Blocks:</b> Contains one or more CIDR blocks representing IP addresses that can be used for the system IP address or router ID for fabric nodes. Enter these blocks using CIDR notation; for example, 192.0.2.0/24.  If you need more IP addresses for devices in your fabrics than the current CIDR blocks support, you can modify the pool to add more CIDR blocks to the System IP pool.
Inter Switch Link IP Pool	<b>Pool name:</b> specifies the name by which this pool will appear and be selected.
	<b>Description:</b> optionally, describes the pool.
	<b>IP Type:</b> set as "IPv4" and cannot be changed.

Property	Description
	<p><b>IP Blocks:</b> Contains one or more CIDR blocks representing IP addresses that can be assigned to inter-switch links between devices in fabric intents throughout this region. Enter these blocks using CIDR notation. For example: 192.0.2.0/24.</p> <p> <b>Note:</b> Each link within a fabric intent requires two IP addresses from this block; one for each endpoint. If you create a fabric intent that requires more links than are available with the current pool, fabric generation fails. The event log for the fabric intent indicates that there are insufficient IP addresses for the required links, and shows the number of addresses required versus the number available.</p> <p>If you need more IP addresses available to your fabrics than the current CIDR blocks support, you can modify the pool to add more CIDR blocks and thereby support additional links.</p>
Management IP Pool	<p><b>Pool name:</b> specifies the name by which this pool will appear and be selected.</p> <p><b>Description:</b> optionally, describes the pool.</p> <p><b>IP Type:</b> identifies the type of IP addresses included in the IP pool. For the Management IP pool, you can set this value to either IPv4 or IPv6.</p> <p> <b>Note:</b> If you are using IPv6 management pools, a Router Advertisement daemon must be present on the IPv6 network. This function is not handled by the Fabric Services System itself.</p> <p><b>CIDR Blocks:</b> Contains a CIDR block representing the IP addresses that will be assigned to the management interfaces of devices. Enter these blocks using CIDR notation for either IPv4 or IPv6, depending on your selection.</p> <p> <b>Note:</b> From the set of IP addresses within the specified CIDR block, two are reserved for use by the system to represent the network IP address and the broadcast address. These are unavailable for the out-of-band management IP pool. If you need more links in your fabrics than the CIDR blocks you specified here support, you can modify the region to add more CIDR blocks to this pool.</p> <p> <b>Note:</b> When you create individual CIDR blocks within a Management IP pool, the IsManaged property is enabled by default. This setting indicates that these IP addresses should be managed directly by the Fabric Services System. If you are creating a fabric consisting of nodes managed by some process external to the Fabric Services System ("unmanaged nodes"), you must create a CIDR block of IP addresses intended for use by those unmanaged nodes; and for that block, you must disable the IsManaged property.</p>

Property	Description
Autonomous System Pools	<p>Contains a set of Autonomous System Numbers (ASNs) consisting of multiple blocks of numbers.</p> <p>ASNs are used to uniquely identify a network with a unique routing policy. The ASN must be unique so that IP address blocks appear to originate from a unique location to which BGP can determine a route.</p> <p>The single pool of numbers is assigned the label "default", and this cannot be altered.</p> <p>Within the pool, you can define one or more blocks of contiguous ASNs by providing a start and end number for each block. ASNs can be any number from 0 to 4294967295.</p> <p>Blocks of numbers within the same pool cannot contain overlapping values.</p> <p>As one block of ASNs is exhausted, the system begins assigning values from whichever remaining, unexhausted block contains the highest available values. Even if some numbers from the first block become available, the system continues allocating numbers from this second block until the second block is exhausted. Then the system again looks for the block with the highest available values from which to allocate new ASNs, and so on.</p>
	<b>Pool name:</b> specifies the name by which this pool will appear and be selected.
	<b>Description:</b> optionally, describes the pool.
	<b>Start:</b> specifies the lowest permissible value in a range of a single block of ASNs.
	<b>End:</b> specifies the highest permissible value within a block of ASNs.

## 14.3 Creating IP and Autonomous System pools

### About this task

Follow this procedure to create a new, non-default pool of any of the following types:

- System IP pools
- Inter Switch Link (ISL) IP pools
- Out of Band Management IP pool
- Autonomous System pools

### Procedure

- Step 1.** Click the  menu.
- Step 2.** Select **Network Resources**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region whose IP pools you want to work with.
- Step 4.** From the **View** drop-down, select the type of IP pool you are going to create. The System IP Pool view is selected by default.

**Step 5.** Click the **+CREATE A <pool type> POOL** button, where <pool type> is the pool type for the currently selected view.



**Note:** The option to configure Out of Band Management IP pools is only available if you configured your region to use an internal DHCP server. If you did not, the **CREATE A MANAGEMENT IP POOL** link is inactive.

**Step 6.** Enter values for the following parameters as described in [Network Resource properties](#):

- **<pool type> Pool Name**
- **Description**
- **IP Type** (either IPv4 or IPv6; configurable for Management IP pools only)

**Step 7.** Choose one of the following:

- To create a System IP pool or ISL pool, go to step [8](#).
- To create a Management IP pool, go to step [11](#).
- If you are creating an ASN pool, go to step [14](#).

**Step 8.** Configure the System IP pool or ISL pool by creating one or more blocks of IP addresses:

- a. Click the **+ADD** button.
- b. Enter an IP address block in CIDR notation. For example: 192.0.2.0/24.
- c. Click **ADD**.

**Step 9.** Repeat step [8](#) until you have created all of the required IP address blocks.

**Step 10.** Go to step [16](#).

**Step 11.** Configure the Management IP pool by creating one or more blocks of IP addresses:

- a. Click the **+ADD** button.
- b. If you are creating a pool for use by unmanaged nodes (nodes that are not managed by the Fabric Services System), disable the **Is Managed** property. Otherwise, leave this property enabled.
- c. Enter an IP address block in CIDR notation. For example: 192.0.2.0/24.
- d. Enter a **Gateway** IP address.
- e. Click **ADD**.

**Step 12.** Repeat step [11](#) until you have created all of the required IP address blocks.

**Step 13.** Go to step [16](#).

**Step 14.** Configure one or more Autonomous System pools by doing the following:

- a. Click the **+ADD** button.
- b. Configure a **Start** value.
- c. Configure an **End** value.
- d. Click **SAVE**.

**Step 15.** Repeat step [14](#) until you have created all of the required ASN blocks.

**Step 16.** Click **CREATE**.

**Expected outcome**

The new pool is added to the system.

## 14.4 Managing Network Resources pools

### About this task

Follow these steps to view, modify, or delete a Network Resources pool or a block within the pool.



**Note:** The following restrictions apply when deleting a pool or a block within a pool that is in use:

- You cannot delete a pool if it is currently in use. To see whether a pool is in use, check the **In Use** column in the list of pools on the Network Resources page.
- You cannot delete a block within a pool if it is in use.



**Note:** When editing a block within a pool that is in use, you can increase the block size but you cannot decrease it.

### Procedure

**Step 1.** Begin by accessing the appropriate pool within the Network Resources page:

- a. Click the  menu.
- b. Select **Network Resources**.
- c. Use the **Region Selector** at the top of the page to select the region whose IP pools you want to work with.
- d. From the **View** drop-down, select the type of IP pool you are going to create. Choose from:
  - **System IP Pools** (selected by default)
  - **Inter Switch Link IP Pools**
  - **Management IP Pools**
  - **Autonomous System Pools**

**Step 2.** Choose the action you want to perform:

- To view details about an individual pool, go to step 3.
- To modify a pool, go to step 4.
- To delete a pool, go to step 5.

**Step 3.** To view details about an individual pool, do the following:

- a. Select a row from the list of configured pools of the selected type and click  at the end of the row.
- b. Select **Open**.

**Step 4.** To modify a pool, do the following:

- a. Select a row from the list of configured pools of the selected type and click  at the end of the row.
- b. Select **Open**.

c. Modify any of the following:

- Description
- Blocks (by adding a block, deleting a block, increasing or decreasing any block's size)



**Note:** If a block is in use, you can increase its size but you cannot decrease its size or delete it.

d. Click **SAVE**.

**Step 5.** To delete a pool, do the following:



**Note:** You cannot delete a pool if it is currently in use.

- a. Find the pool in the list and click  at the end of the row.
- b. Select **Delete...**
- c. Click **OK** in the confirmation form.

## 15 Digital Sandbox

The Fabric Services System Digital Sandbox is a network simulator that can emulate data center fabric designs ("underlays") and the workload constraints configured upon those fabrics ("overlays").

The Digital Sandbox normally runs on a three-node Kubernetes cluster. Each SR Linux node is emulated as its own virtual machine within the cluster, running its own copy of the SR Linux operating system like the real node it represents.

Before you can use the Fabric Services System Digital Sandbox, you must install its software components and perform any configuration steps described in the *Fabric Services System Software Installation Guide*. This ensures that the Digital Sandbox software is ready to simulate your fabrics and workloads, and is ready to communicate with the Fabric Services System to receive model data and send status updates.

In its current form, the Digital Sandbox can emulate a region, the structures of fabrics within that region, and the workload constraints that are configured upon those fabrics (including the edge links that are referred to by the workload). It does not yet simulate dynamic features like traffic flow between the simulated nodes and their endpoints.

The Digital Sandbox requires its own license, purchased separately from the license for the Fabric Services System itself.

### 15.1 Integration with the Fabric Services System

Most interaction with the Digital Sandbox is performed using CLI commands and API calls.

In integrated mode, the Digital Sandbox can communicate with the Fabric Services System, receiving configuration data for fabric and workload designs, and returning status updates for those intents. The Fabric Services System UI does not support all of the available Digital Sandbox operations; it is only used to design fabric intents and workload VPN intents, and to send those configurations to the Digital Sandbox for further action.

A technically proficient user who is familiar and experienced with the Digital Sandbox CLI could configure fabrics, workloads, and participating endpoints using only the Digital Sandbox CLI or REST API calls. But for most operators, taking advantage of integration with the Fabric Services System makes these operations much faster and easier.

#### 15.1.1 Digital Sandbox status display

When using the integrated mode, the Fabric Services System UI displays the status of the Digital Sandbox in the lower-left corner of the fabric intents geographical map. Possible statuses are:

- Unavailable: the Digital Sandbox has either:
  - not been installed
  - not been configured for integration with the Fabric Services System
  - been misconfigured, such that the Fabric Services System is pointing to the wrong location for the Digital Sandbox

- has been installed and configured for integration but has not been started
- Running: the Digital Sandbox is installed, started, and ready to receive data.
- Busy: the Digital Sandbox is installed and started, but is processing data recently sent from the Fabric Services System for incorporation into its simulation.

## 15.2 Creating a region in the Digital Sandbox

### About this task

When using the integrated mode, you must create a deployment region from the Fabric Service System's **Regions** page before you can create any fabric intents or workload VPN intents that are destined for the Digital Sandbox.

### Procedure

Create a region by following the procedure [Creating a region](#).

### Expected outcome

After you have created the region, the Digital Sandbox creates a set of internal structures ("pods") for use in its simulation.

As the Digital Sandbox creates these structures, its status advances from Unavailable to Busy to Running. When it reaches the Running state, you can create fabric intents and workload VPN intents that are destined for the Digital Sandbox.

### 15.2.1 Modifying a region in the Digital Sandbox

#### About this task

If you modify any of the properties of a region, including changes to the fabric and workload VPN intents in the Fabric Services System UI destined for digital sandbox, you must explicitly trigger a corresponding update in the Digital Sandbox.

To update the Digital Sandbox after modifying a region:

#### Procedure

**Step 1.** If the **Deployment Regions** page is not already open:

- Click  to open the main menu.
- In the main menu, select **Deployment Regions**.

#### Expected outcome

The **Deployment Regions** page opens, showing a graphical representation of regions already created.

**Step 2.** Right-click the region icon on the **Deployment Regions** page, and select **Update Digital Sandbox** from the displayed menu.

#### Expected outcome

In the lower left corner of the page:

- the Digital Sandbox status advances to the Busy state.

- the Digital Sandbox status returns to the Running state.

When the Digital Sandbox has returned to the Running state, the update to its model of the region is complete.



**Note:** If an error occurs during the update, an error indication appears on the lower left corner. Hover over the error indication to display the description of the error in the Digital Sandbox.

## 15.3 Fabric intents and the Digital Sandbox

Working with a fabric intent that is destined for the Digital Sandbox is just like working with a fabric intent destined for real hardware, with the following exceptions:

- When you create the fabric intent, set the **Fabric Type** field to **Digital Sandbox** instead of **Real**.
- The deployment threshold for a Digital Sandbox fabric intent is always 100% (whereas the threshold is always 0% for Real fabric intents).
- After you design a Digital Sandbox fabric intent, and before you deploy that intent, you must manually send the updated configuration to the Digital Sandbox using the Update Digital Sandbox command.

After you update the Digital Sandbox and deploy your fabric intent, the Digital Sandbox creates a corresponding set of simulated nodes, their configurations, and their relationships in its own data model.

The representation of the fabric at this stage is coarse; it is limited to the nodes themselves. Endpoints and other lower-level details are not added to the model until subsequently required by workload VPN intents.

### 15.3.1 Creating a fabric intent in the Digital Sandbox

#### Prerequisites

Before you create a fabric intent that is destined for the Digital Sandbox, ensure that a region has already been created.

#### About this task

The procedure to design a fabric intent destined for the Digital Sandbox is nearly identical to designing an intent for deployment to real hardware.

To create a fabric intent that is destined for the Digital Sandbox:

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** In the main menu, select **Fabric Intents**.
- Step 3.** Use the **Region Selector** at the top of the page to select the region in which to create the fabric intent.
- Step 4.** Click the **+ CREATE A FABRIC INTENT** button to open the Fabric Design page.
- Step 5.** Select or import the topology file on which this fabric is based:
  - To select an already-imported topology file, click in the Imported Topology From File field and select the topology from the displayed list.

- To import a new topology from a file, click the **Import** icon and select the topology file. The topology in that file is automatically selected as the basis for this fabric intent.
- Step 6.** In the **Fabric Type** drop-down list, click **Digital Sandbox**.
- Step 7.** On the left-side panel, enter or select the basic parameters that define your intended fabric as described in [Table 23: Basic parameters for manual topology fabric intent](#).
- Step 8.** Optionally, select an ASN pool and IP pools other than the default pools for the region as described in [Table: Optional parameters for fabric intents](#):
- **ASN Pool Name**
  - **Inter Switch Link Pool Name**
  - **Management Pool Name**
  - **System Pool Name**
- Step 9.** Specify the **DS Management Network IP family** (either IPv4 or IPv6) for this fabric The default value is IPv4.
- Step 10.** Click  to save the fabric intent. When you save the fabric intent, the system:
- updates the state of the fabric intent to Created.
  - updates the version number of the fabric intent to 1.0.
  - enables the  **GENERATE FABRIC** button.
- Step 11.** Click  **GENERATE FABRIC**.

## 15.3.2 Updating the Digital Sandbox

### About this task

After you design your fabric intent, but before deploying it, you must update the Digital Sandbox with information about the fabric intent. This causes the Digital Sandbox to create virtual nodes onto which the fabric can be deployed.

Any time you make subsequent changes to the fabric intent, you should follow these steps to again update the Digital Sandbox with the new configuration data.

To update the Digital Sandbox with information about your fabric intent, do the following:

### Procedure

- Step 1.** Open the fabric intent in the **Fabric Design** view, if it is not already open.
- Step 2.** At the upper right of the page click the **More actions** icon (  ) and select **Update Digital Sandbox** from the displayed menu. As a result:

#### Expected outcome

In the lower left corner of the page:

- the Digital Sandbox status advances to the Busy state.
- one by one, each virtual node in the fabric intent advances to the Ready state.
- the Digital Sandbox status returns to the Running state.

When all nodes are in a Ready state and the Digital Sandbox has returned to the Running state, you can deploy the fabric intent.



**Note:** If an error occurs during the update, an error indication appears on the lower left corner. Hover over the error indication to display the description of the error in digital sandbox.

Additional messages can also appear in the lower middle section of the page.

### 15.3.3 Deploying a fabric intent in the Digital Sandbox

#### About this task

The procedure to deploy a fabric intent to the Digital Sandbox is the same as that for a real fabric intent. For detailed steps, see [Fabric intent deployment](#).

#### Procedure

**Step 1.** Add the fabric intent to the deployment pipeline.

**Step 2.** From the deployment pipeline, select the fabric intent and click **Deploy**.

#### Expected outcome

When the fabric has attained the Deployed state, the result in the Digital Sandbox is a new Underlay 1 (UL1) construct, represented by a collection of Kubernetes pods.

At this stage only the nodes themselves are modeled in the Digital Sandbox data. Endpoints, and details about those endpoints such as IP addresses, are not yet present in the model.

#### Related topics

[Creating a region](#)

## 15.4 Workload VPN intents

There are no differences in workload VPN intent design or deployment when the target is the Digital Sandbox; the procedures are the same as those for workload VPN intents destined for real hardware.

When you design a Workload VPN intent that includes fabrics that were created for the Digital Sandbox, the Fabric Services System sends the workload VPN intent information to the Digital Sandbox for incorporation into its simulation.

### 15.4.1 Creating a workload VPN intent in the Digital Sandbox

#### About this task

Use the procedures for creating a workload VPN intent for real fabrics.

#### Procedure

Create your workload VPN intent as described in [Workload VPN intents](#).

#### Expected outcome

When you are finished, you are ready to deploy the workload VPN intent.

## 15.4.2 Deploying a workload VPN intent to the Digital Sandbox

### About this task

Use this procedure to deploy a workload VPN intent to the Digital Sandbox

### Procedure

**Step 1.** Add the workload VPN intent to the pipeline.

**Step 2.** Deploy the workload VPN intent from the pipeline.

### Expected outcome

When you deploy the workload VPN intent, the Digital Sandbox updates the configuration files of the participating, simulated nodes. In the Digital Sandbox, the simulated workload is classified as a Candidate Workload, but is not active; the participating nodes are identified, but endpoint data is not yet present.

If you update the workload VPN intent design in the Fabric Services System and re-deploy it, it overwrites the information for the candidate workload in the Digital Sandbox. This is true even if you update the workload information in the Digital Sandbox directly using the CLI after the last deployment; the re-deployment overwrites the workload VPN intent data and erases your changes.

## 15.4.3 Updating the Digital Sandbox

### About this task

After you deploy your workload VPN intent, you must explicitly update the Digital Sandbox with information about the workload VPN intent. This causes the Digital Sandbox to add endpoint data to the workload model, and the result is an Active Workload.

Any time you make subsequent changes to the workload VPN intent, you should follow these steps to again update the Digital Sandbox with the new configuration data.

To update the Digital Sandbox with information about your fabric intent, do the following:

### Procedure

**Step 1.** Click  to open the main menu.

**Step 2.** In the main menu, select **Deployment Regions**.

**Step 3.** Right-click the region object on the **Deployment Region** map.

**Step 4.** From the contextual menu, select **Update Digital Sandbox**.

**Step 5.** Click **OK**.

### Expected outcome

The Digital Sandbox updates its model based on the latest data in the Fabric Services System. While it does this, the Digital Sandbox status advances from the Running state to the Busy state, and then back to the Running state.

When the Digital Sandbox has returned to the Running state, its simulated model of the workload VPN intent includes all of the participating endpoints.

If you update the workload VPN intent design in the Fabric Services System, re-deploy it, and re-update the Digital Sandbox, it overwrites the Active Workload's information in the Digital Sandbox. This is true even if you had updated the workload information directly in the Digital

Sandbox using the CLI after the last update; re-updating overwrites the workload VPN intent data and erases your changes.

## 16 System administration

System administration encompasses a range of activities that configure or otherwise manage the Fabric Services System application itself.

Some of these operations are performed within the Fabric Services System user interface. Others are performed as command-line operations on the server that hosts the application.

### 16.1 Application settings

The Fabric Service System's **Settings** page provides access to several configuration settings that affect the way the system behaves.

- The **Image Management and Software Catalog** panels pertain to the software images available on the Fabric Services System's file server.
- The **Common Application Settings** page determines the background map image used with specific views, and whether you are using Kibana and Elasticsearch to view log data generated by the system's internal microservices.

#### 16.1.1 Viewing software and image catalogs

##### About this task

The Fabric Services System maintains a catalog of software images which are used to configure nodes within managed fabrics.

The software images in the catalog are used when:

- creating a fabric intent; you must select one version of the software which is deployed to all nodes participating in the fabric as part of their configuration.
- creating a maintenance intent for changes to a node's software version; you must select one version of the SR Linux software from the catalog which is deployed to all nodes that are subjects of the maintenance intent.



**Note:** The same software and image catalog are shared by all regions.

##### Procedure

**Step 1.** Click  to open the main menu, then select **Settings**.

**Step 2.** Click **Software Catalog**.

The current software catalog displays, identifying for each entry:

- the vendor: currently, only Nokia is supported
- the operating system
- the software version: the system is pre-loaded with selected software versions; this list grows as you load additional software versions over time

**Related topics**[Fabric intents](#)[Maintenance intents](#)

### 16.1.1.1 Adding a new software image

**About this task**

Follow this procedure to upload a new SR Linux image file and associate a description. Once uploaded, this image becomes available for selection in the software catalog.



**Note:** You can upload the file using HTTP, HTTPS, or FTP.

**Procedure**

**Step 1.** Click  to open the main menu.

**Step 2.** Select **Settings**.

**Step 3.** Click **Image Management**.

**Step 4.** Click **+ADD IMAGE**.

**Step 5.** Enter information in the following fields for the new software image:

- **Image URL:** a location from which the system can download the image file to the file server. The URL provided determines whether the download will use HTTPS or FTP.  
For example:  
`http://<IP address>//<path>//srlinux<version>.bin`  
`https://<IP address>//<path>//srlinux<version>.bin`  
`ftp://:<IP address>//<path>//srlinux<version>.binsrlinux-22.11.3-141.bin`
- **MD5 URL:** a location (on the same file server as the image URL) from which the system can download the MD5 file to the file server. The URL provided determines whether the download will use HTTPS or FTP.  
For example:  
`http://<IP address>//<path>//srlinux<version>.bin.md5`  
`https://<IP address>//<path>//srlinux<version>.bin.md5`  
`ftp://:<IP address>//<path>//srlinux<version>.bin.md5`
- **Username** and **Password:** credentials required to download files from the image URL (these are not required for HTTP)
- Optionally: **Description** for this software image

**Step 6.** Click **SAVE**.

**Expected outcome**

The system adds the image to the image catalog and downloads the specified image from the source you provided to its integrated file server.

In addition to the information you provided in step 5, the image catalog displays for each software image:

- **Image Name:** the name of the image file on the file server

- **MD5:** either True or False, indicating whether the MD5 file has been successfully added
- **Upload Status:** the image status, which can be any of the following:
  - Done
  - Failed
  - Adding
  - Overwriting
- **Path:** for a successfully obtained image, the path to the image and MD5 files on the integrated file sever

## 16.1.2 Common application settings

The Common Application Settings panel determines the background map image used with certain views.

### 16.1.2.1 Configuring Geomap Tile Server settings

#### About this task

The background image displayed for the region map is determined by the system's Geomap Tile Server setting.

To assign or change the background map:

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **Settings**.
- Step 3.** Click **Common Application Settings**.
- Step 4.** In the **Geomap** panel, set the values for:
  - **Tile Server:** enter a URL that points to a specific tile server and an image on that server to function as the background for the region map
  - **Attribution:** enter an attribution for the background image; this field is mandatory
- Step 5.** Click **SAVE** at the lower right of the page.

#### Expected outcome

The selected image displays the next time you open the region map.

## 16.2 Adding a new network operating system version to the software catalog

#### About this task

The software catalog contains a list of network operating system (NOS) images that the Fabric Services System can use to deploy on nodes. The Fabric Services System ships with the latest version of supported

NOS versions in the catalog at the time of release, but a operator can add a minor release to the software catalog.



**Note:** The same software and image catalog are shared by all regions.

Use this procedure to update the Fabric Services System software catalog. You may need to execute this procedure in the following scenarios:

- a new version the NOS is introduced for use with fabrics
- after restoring a Digital Sandbox fabric
- after a software upgrade



**Note:** Deleting an existing NOS that is in use causes a ZTP process failure. If you must delete an existing NOS image, ensure that no fabric is using it.

### Prerequisites

To guarantee that the functionality of Fabric Services System features are not affected, the addition of a new catalog and its SRL image must be approved and qualified by Nokia.

### Procedure

**Step 1.** From the main menu , select **Settings**.

**Step 2.** Select **Software Catalog**.

**Step 3.** Click **+ADD SOFTWARE CATALOG**.

**Step 4.** In the **Software Image Details** form that displays, set the following fields:

- **Vendor** – this field is always set to Nokia
- **Operating System** – select **SRLinux** or **SROS210WBX**
- **Software version** – enter a valid software version; the Fabric Services System performs semantic validation depending on the selected operating system

**Step 5.** Click **Save**.

### What to do next

You can now use the new software image in a fabric intent or a maintenance intent.

## 16.3 Uploading SR Linux container images for Digital Sandbox

### About this task

This procedure describes how to upload SR Linux container images to Fabric Services Systems deployed with a Digital Sandbox.

### Prerequisites

Ensure that the software catalog is updated with the latest SR Linux container image. To update the software catalog, see [Adding a new network operating system version to the software catalog](#).

## Procedure

**Step 1.** Select the option for your deployment scenario.

- If the deployer VM has access to the Internet, go to Step 2.
- If the deployer VM does not have access to the Internet, go to Step 3.

**Step 2.** Use the built-in Skopeo tool to upload the SR Linux container images.

**a.** Enter the following command:

```
skopeo copy --src-tls-verify=false --dest-tls-verify=false docker://ghcr.io/nokia/srlinux:<SRL Version> docker://localhost/registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:<SRL Version>
```

### Example

```
[root@fss-deployer-vm ~]# skopeo copy --src-tls-verify=false --dest-tls-verify=false docker://ghcr.io/nokia/srlinux:22.6.1-281 docker://localhost/registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:22.6.1-281
Getting image source signatures
Copying blob 499dd76466a8 done
Copying config 92168ca66d done
Writing manifest to image destination
Storing signatures
[root@blrsrlfsp12-deployer-vm ~]#
```

**b.** Go to Step 4.

**Step 3.** If the deployer VM does not have access to the Internet, use the Docker CLI to upload the container images.

You must have machine/VM (referred to as docker-host in this procedure) with Internet access and Docker installed. Run the below commands on the docker-host.

**a.** Pull the required SR Linux container image from the `ghcr.io` repository to the docker-host. Use the following command:

```
docker pull ghcr.io/nokia/srlinux:<SRL Version>
```

### Example

```
[usr@docker-host ~]$ docker pull ghcr.io/nokia/srlinux:22.6.4-90
22.6.4-90: Pulling from nokia/srlinux
72e3470bdea0: Pull complete
Digest: sha256:edcd5354061c91201361cf0e42cdb9e428b2f3d9847a3ecb58c896a3582a668d
Status: Downloaded newer image for ghcr.io/nokia/srlinux:22.6.4-90
ghcr.io/nokia/srlinux:22.6.4-90
```

**b.** Tag the image for the deployer VM.

Use the following command:

```
[usr@docker-host ~]$ docker tag ghcr.io/nokia/srlinux:<SRL Version> <deployer VM IP>/registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:<SRL Version>
```

**Example**

```
[usr@docker-host ~]$ docker tag ghcr.io/nokia/srlinux:22.6.4-90 10.10.10.123/registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:22.6.4-90
```

- c. Update the deployer VM IP address under the `insecure-registries` section of the `/etc/docker/daemon.json` file in the `docker-host`.

**Example**

```
[usr@docker-host ~]$ cat /etc/docker/daemon.json
{
  "registry-mirrors": [
    https://docker-registry-remote.artifactory-espoo1.int.net.abc.com,
    https://docker-registry-remote.artifactory-espoo2.int.net.abc.com,
    https://docker-registry-remote.artifactory-fpark1.int.net.abc.com,
    https://docker-registry-remote.artifactory-blrl.int.net.abc.com
  ],
  "insecure-registries": [
    "registry.gitlab.sr.nuq.ion.abc.net",
    "10.10.10.123"
  ]
}
```

- d. Push the container image to the deployer VM.

Use the following command:

```
[usr@docker-host ~]$ docker push <deployer VM IP>/registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:<SRL Version>
```

- e. Go to Step 4.

- Step 4.** Display the SR Linux container images in the deployer repository.

**Example**

```
[root@fss-deployer-vm ~]# docker_img_list.sh list localhost
registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux
20221213_161953 List
registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:22.11.1-184
registry.gitlab.sr.nuq.ion.nokia.net/sr/linux/fsp/srl-registry/srlinux:22.6.2-24
```

## 16.4 User and resource management

The user and resource management menu allows administrators to manage users and their access to resources in Fabric Services System.

Administrators can configure four types of elements related to users and resources:

<b>Roles</b>	Specifies which network resources users or associated user group members can access. You assign network resource access to roles through resource groups.
<b>Resource groups</b>	Specifies resources that users can access. When you create a role, you associate specific resource groups to it. Then, when you assign a role to a user group, the members of the user group gain access to the resource groups

associated with the role. For example, resource groups can represent the ability to configure fabrics or QoS policies.

### User groups

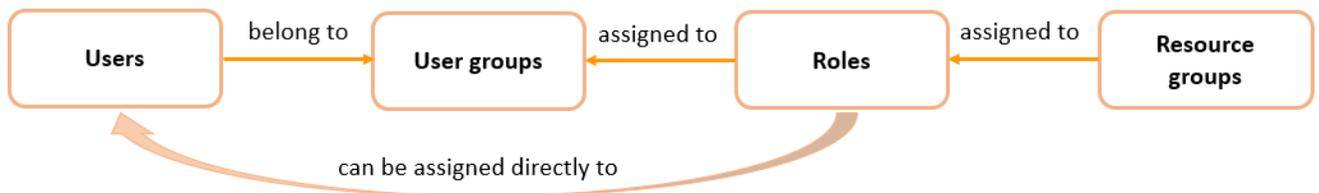
A collection of users organized according to the type of network activities they are meant to perform. You assign resource access rights to user groups through user roles. When you assign a role to a user group, all access rights defined in the role are inherited by the users of the group.

### Users

Individuals with access to the system. Each user has a user information profile to store information about them. You can assign users to user groups. When you assign a role directly to a user, all access rights defined in the role are inherited by the user.

Figure 36: User and resource management shows the user and resource management architecture.

Figure 36: User and resource management



## 16.4.1 Roles

Roles define the application access and resource permissions that can be assigned. You first create roles, then associate them to user groups according to the type of network activities the user group is meant to perform. Each member user of a user group can perform the roles specified for that group.

Optionally, you can also choose to assign a role directly to a user. When a user requires a specific set of permissions, you can bypass the use of user groups entirely.

Each role is mapped to a specific set of resource group access permissions. When a role is created, you can set the possible levels of permission for the associated resource groups to any the following:

### No Access

Users or user group members do not have access to this resource group. The No Access permission is set by default for each resource until you change it when defining the role.

### Read

Read permissions allow users or user group members to view specific resources, but they cannot make changes.

### Read / Write

Read/write permissions allow users or user group members to view and modify resources.

After a role is created, you can return to the role and modify the resource access permissions.

### 16.4.1.1 Viewing a list of existing roles

#### About this task

Follow this procedure to view a list of existing roles.

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **User and Resource Management**.
- Step 3.** Select **Roles** from the drop-down list.

### 16.4.1.2 Predefined roles

The following table shows the predefined system roles and describes the specific permissions each role allows users. These roles are defined with common resource access privileges that you can quickly assign to new users. Administrators can associate roles to a specific user or to all members of user groups.

Predefined roles cannot be modified. You can create customized roles for users that require specific permissions.

Table 53: Predefined roles

Role	Description
fabric-operator	Allows read/write access to all system resources except infra components (such as users, roles, and resource groups).
fabric-viewer	Allows read only access to system resources.
fss-admin	Allows admin privileges for all system resources in default namespaces.
geored	Allows access to geo-redundancy settings.
ztp	Allows access to node management resources.

#### Related topics

[Creating a role](#)

### 16.4.1.3 Creating a role

#### About this task

Follow this procedure to create a new role.

#### Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **Roles** from the drop-down list.
- Step 2.** Click **+ CREATE ROLE**.

**Step 3.** Under the **Role Info** heading, specify a role name and add an optional description to describe the purpose of the role.

**Step 4.** Specify the resource access permissions for the role. For a specific resource group permission, select one of the following options from the drop-down list.

- Read
- Read / Write

Do this for one or more resource groups.

The No Access permission is automatically selected for each resource until you change it.

You can also use the sort and filter columns to narrow the list of resource access options.

**Step 5.** Click **CREATE**.

#### Related topics

[Lists](#)

### 16.4.1.4 Modifying the resource access permissions of a role

#### About this task

After a role is created, you can modify its resource access permissions. You can specify the resources in the system that can be accessed by users or user group members with the role applied.

Follow this procedure to modify the application access permissions of a role.

#### Procedure

**Step 1.** From the main menu  → **User and Resource Management** page, select **Roles** from the drop-down list.

**Step 2.** Locate the role that you want to modify, click the  options menu at the right end of the row.

**Step 3.** Select **Open**.

**Step 4.** Under the **Resource Access** heading, for a specific resource group permission, select one of the following options from the drop-down list.

- No Access
- Read
- Read / Write

Do this for one or more resource groups.

You can also use the sort and filter columns to narrow the list of resource access options.

**Step 5.** Click **SAVE**.

#### Related topics

[Lists](#)

### 16.4.1.5 Deleting a role

#### About this task

Follow this procedure to delete a role.

## Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **Roles** from the drop-down list.
- Step 2.** Locate the role that you want to delete and click the  options menu at the right end of the row.
- Step 3.** Click **Delete**.  
If prompted, confirm that you want to delete the selected role.

## 16.4.2 Resource groups

Resource groups define the specific system resources that users can access in the UI. Resource groups are associated with roles. When administrators assign a role to a user group, the user group gains access to the network resources specified in the resource group. Resource groups represent the ability to configure specific types of things in the system, such as fabrics, workloads, or profiles.

Resources are organized into groups based on logical functional boundaries. Access to the specific functional areas can be associated with roles, and ultimately to user groups according to the type of network activities the user group is meant to perform.

The system comes with a set of predefined resource groups for common functional areas. Administrators cannot create new resources groups.

### 16.4.2.1 Viewing a list of resource groups

#### About this task

Follow this procedure to view a list of existing resource groups.

#### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **User and Resource Management**.
- Step 3.** Select **Resource Groups** from the drop-down list.  
Use the sort and filter columns to narrow the list.

#### Related topics

[Lists](#)

### 16.4.2.2 Predefined resource groups

[Table 54: Predefined resource groups](#) shows the predefined resource groups and describes the specific permissions that the resource groups allow for users. When creating a role, administrators can associate a specific resource group or a combination of multiple resource groups.

For example, the administrator may want to allow only a specific set of users to configure QoS policies. The administrator can create a role that includes the **RG-qos** resource group, with the permission set to **read / write**. Then, the administrator can create a user group with this particular role assigned. User members of the user group are allowed permission to view and modify QoS profiles.

Table 54: Predefined resource groups

Resource group	Resource type	Description
RG-AAA	AAA	User, user group, and resource group management
RG-alarmmgr	alarmmgr	Access to system alarm manager objects
RG-catalog	catalog	Access to catalogs
RG-connect	connect	Access to system connect objects
RG-devices	device	Configuring devices
RG-fabric	fabric	Configuring fabrics between the switches
RG-fabric-telemetry	fabric-telemetry	Access to all the statistics, events, and so on that are collected between the switches.
RG-fabric-uplinks	fabric-uplinks	Configuring the fabric uplinks of a fabric
RG-fabric-uplink-protocols	fabric-uplink-protocols	Configuring the uplinks of the fabric, which can include Layer-2, Layer-3, and protocols to interface with external routers connected to fabric uplinks
RG-images	image	Provides OS images of the device
RG-infra	infra	Access to infra settings
RG-label	label	Access to labels
RG-Layer-1	layer1	Configuration of Layer-1 aspects such as SFP, breakout, and so on
RG-Layer-2	layer2	Layer-2 configurations such as VLANs, LAG, and so on
RG-Layer-2-protocols	l2protocols	Layer-2 protocol configurations such as LLDP, LACP, and so on
RG-Layer-3	layer3	Layer-3 configurations such as sub-interfaces, static and dynamic routing policies, and so on
RG-Layer-3-protocols	l3protocols	Layer-3 protocol configurations such as BGP, OSPF, ISIS, and so on
RG-Maintenance	maintenance	Access to node maintenance intent
RG-Management	management	Configuring the management VRF and the relevant CoPP aspects
RG-mgmt-protocols	management-protocols	Configuring protocols used to manage SR Linux such as SSH, gNMI, NTP, FTP, and so on
RG-qos	qos	Configuration of QoS policies; can include CoPP
RG-region	region	Creating and configuring data center regions
RG-sandbox	sandbox	Access to sandbox environment

Resource group	Resource type	Description
RG-security	security	Configuring security policies for workloads and fabrics
RG-sync	sync	Provides access to sync objects when configuring geo-redundancy
RG-topology	topology	Topology access for fabrics
RG-workload	workload	Configures workload related policies, including ACL and QoS policy profiles, but cannot edit the ACL and QoS profiles
RG-workload-attachments	workload-attachments	Configure workload attachment points
RG-workload-telemetry	workload-telemetry	Access to all statistics, events, flows, and so on, collected on the downlinks

### 16.4.3 User groups

A user group associates multiple users with a role, enabling them to access network resources. Administrators can create user groups and assign a specific role to each group according to the type of network activities the user group is meant to perform. When a role is assigned to a user group, all users within the group have the same access to resources, as specified by the role.

You can assign multiple users to a group. Users can also be members of multiple groups.



**Note:** The set of Fabric Services System users global; that is, it is shared by all regions. However, each user account is granted separate access to each region by associating it with the specific operator group for that region. A region's operator group is automatically created whenever you configure a region in the Fabric Services System. Because the Fabric Services System UI only displays information about regions to which the current user has access, and because so many objects are region-specific, the region operator groups to which a user account belongs have a significant impact on what that user can see and do within the Fabric Services System user interface.

#### 16.4.3.1 Viewing a list of existing user groups

##### About this task

Follow this procedure to view a list of existing user groups.

##### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **User and Resource Management**.
- Step 3.** Select **User Groups** from the drop-down list.

### 16.4.3.2 Predefined user groups

The following table shows the predefined system user groups. Each member of the user group is assigned the corresponding role and gains the resource access privileges associated with the role. These user groups represent collections of users that have been assigned specific system predefined roles.

You can also create customized user groups for users that require specific permissions.

Table 55: Predefined user groups

User Group Name	Role	Description
fabric_operator_grp	fabric-operator	Allows group members read/write access to all system resources except infra components (such as users, roles, and resource groups).
fabric_viewer_grp	fabric-viewer	Allows group members read only access to system resources.
fss_admin_grp	fss-admin	Allows group members admin privileges for all system resources in default name spaces.
ztp_grp	ztp	Allows group members access to node management resources.
<region>_operator_grp0	fabric-operator	Allows group members to see and interact with objects associated with this region. A region operator group is automatically created whenever you create a new region, and has the name <region>_opreator_grp where <region> is the name of the region.

#### Related topics

[Predefined roles](#)

[Creating a user group](#)

### 16.4.3.3 Creating a user group

#### About this task

A user group can be configured initially without assigning users. If you want to add users to the group, you should also have some users configured.

Follow this procedure to create a new user group.

## Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **User Groups** from the drop-down list.
- Step 2.** Click **+ CREATE USER GROUP**.
- Step 3.** Under the **User Info** heading, specify a user group name and add an optional description to describe the purpose of the group.
- Step 4.** Under the **Assigned Users** heading, select users to add to the user group. Click the check box next to a user's name to add them to the group.  
Optionally, you can create a user group without users, then add users later.  
You can sort and filter the list of users.
- Step 5.** Click **CREATE**.

## Related topics

[Users](#)

[Lists](#)

### 16.4.3.4 Assigning the role of a user group

#### About this task

You can assign a role to a user group by editing an existing user group. A role can also be assigned when you create a group.

You can only assign one role per user group. If a user group has an assigned role, you can change the assignment.

Follow this procedure to assign a role to a user group.

## Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **User Groups** from the drop-down list.
- Step 2.** For a specific user group, click the  options menu at the right end of the row.
- Step 3.** Select **Open**.
- Step 4.** Select **Assigned User Roles** from the left navigation menu.  
You can sort and filter the list of user roles.
- Step 5.** Select one of the listed roles to assign to the user group.  
You can only select one role.
- Step 6.** Click **SAVE**.

## Related topics

[Lists](#)

### 16.4.3.5 Deleting a user group

#### About this task

Follow this procedure to delete a user group.

#### Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **User Groups** from the drop-down list.
- Step 2.** For a specific user group, click the  options menu at the right end of the row.
- Step 3.** Click **Delete**.  
If prompted, confirm that you want to delete the selected user group.

### 16.4.4 Users

Users are individuals with access to the system. Users gain access to application and network resources through the user groups to which they are assigned.

Individual users can also be assigned roles directly, without membership to a user group.



**Note:** The Fabric Services System uses Keycloak, a well-known and secure solution, for its identity and access management. Keycloak does not store passwords in raw text; instead, passwords are stored as hashed text, using the PBKDF2-HMAC-SHA512 message digest algorithm. Keycloak performs 210,000 hashing iterations, the number of iterations recommended by the security community. Out of security concerns, both Keycloak and the Fabric Services System do not provide any API or UI functionality to see a user's password, even hashed.

#### 16.4.4.1 Viewing a list of existing users

##### About this task

Follow this procedure to view a list of existing users.

##### Procedure

- Step 1.** Click  to open the main menu.
- Step 2.** Select **User and Resource Management**.
- Step 3.** Select **Users** from the drop-down list.

#### 16.4.4.2 Creating a new user

##### About this task

Follow this procedure to create a new user.

## Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **Users** from the drop-down list.
- Step 2.** Click **+ CREATE USER**.
- Step 3.** In the User Info section of the overlay, enter the required job information data for the new user.
  - a user name to identify the user
  - the user's e-mail address
  - a password the user uses to sign in to the system
  - confirm the password
  - optionally, enter the user's first and last name to identify them
- Step 4.** Confirm the **User Enabled** toggle is enabled.  
You can also disable the user by disabling the toggle. Only enabled users are permitted access to the system.
- Step 5.** Assign the user to one or more user groups. Select **Assigned User Groups** from the left navigation menu.  
The user must be assigned to at least one user group to gain resource permissions. Optionally, you can create a user without assigning the user to a user group, then add the user to a user group later.
- Step 6.** Click the check box next to one or more user groups to which you would like to assign the user.
- Step 7.** Click **CREATE**.

### 16.4.4.3 Assigning a user to a user group

#### Prerequisites

You must configure appropriate user groups before starting this procedure.

#### About this task

You can assign a user to one or more user groups. The user gains the resource permissions that are associated with the group.

Follow this procedure to assign a user to a group.

#### Procedure

- Step 1.** From the main menu  → **User and Resource Management** page, select **Users** from the drop-down list.
- Step 2.** For a specific user, click the  options menu at the right end of the row.
- Step 3.** Select **Open**.
- Step 4.** Select **Assigned User Groups** from the left navigation menu.  
You can sort and filter the list of user groups.
- Step 5.** Click the check box next to one or more user groups to which you would like to assign the user.
- Step 6.** Click **SAVE**.

## Related topics

[Lists](#)

### 16.4.4.4 Assigning a role to a user

#### About this task

You can assign a role directly to a user, bypassing user groups entirely. The user gains the resource permissions specified in the role. Only one role can be assigned to an individual user.

If the user requires customized permissions that are not present in any of the existing roles in the system, you can create a custom role for the user containing the specific permissions.

Follow this procedure to assign a role directly to a user.

#### Procedure

**Step 1.** From the main menu  → **User and Resource Management** page, select **Users** from the drop-down list.

If the user that you would like to assign the role to is not yet created in the system, create the user.

**Step 2.** For the specific user, click the  options menu at the right end of the row.

**Step 3.** Select **Open**.

**Step 4.** Select **Assigned User Roles** from the left navigation menu.

You can sort and filter the list of user roles.

**Step 5.** Select one of the listed roles to assign to the user.

You can only select one role for a user.

**Step 6.** Click **SAVE**.

#### What to do next

To confirm the assignment, you can open the role and view the list of assigned users.

#### Related topics

[Creating a new user](#)

[Lists](#)

### 16.4.4.5 Deleting a user

#### About this task

Follow this procedure to delete a user from the list of users.

#### Procedure

**Step 1.** From the main menu  → **User and Resource Management** page, select **Users** from the drop-down list.

**Step 2.** For a specific user, click the  options menu at the right end of the row.

**Step 3.** Click **Delete**.

If prompted, confirm that you want to delete the selected user.

**Expected outcome**

The user is deleted from the list of users. The user is also removed from any user group memberships.

## 16.5 Health monitoring

The Fabric Services System includes a health monitoring service that allows you to monitor the state of the system and its component software services. You can use this service to ensure that the system is functioning correctly, and to alert you to components that are encountering issues.

The Fabric Services System uses Prometheus to store these metrics, a common approach and a well-known format within the Kubernetes ecosystem. This standard format is intended to make it easier to find information, and to monitor and interpret the telemetry provided.

The health monitoring system currently retrieves the following metrics:

- Fabric Services System API metrics
  - fss\_request\_duration\_seconds
  - fss\_request\_duration\_seconds\_count
  - fss\_request-size\_bytes\_sum
  - fss\_request\_size\_bytes\_count
  - fss\_requests\_total
  - fss\_response\_size\_bytes\_sum
  - fss\_response\_size\_bytes\_count
- Fabric Services System Apps, Pods, and other Golang Apps metrics
  - go\_gc\_duration\_seconds
  - go\_gc\_duration\_seconds\_count
  - go\_gc\_duration\_seconds\_sum
  - go\_goroutines"  
go\_info
  - go\_memstats\_alloc\_bytes
  - go\_memstats\_alloc\_bytes\_total
  - go\_memstats\_buck\_hash\_sys\_bytes
  - go\_memstats\_frees\_total
  - go\_memstats\_gc\_cpu\_fraction
  - go\_memstats\_gc\_sys\_bytes
  - go\_memstats\_heap\_alloc\_bytes
  - go\_memstats\_heap\_idle\_bytes
  - go\_memstats\_heap\_inuse\_bytes
  - go\_memstats\_heap\_objects
  - go\_memstats\_heap\_released\_bytes

- go\_memstats\_heap\_sys\_bytes
- ago\_memstats\_last\_gc\_time\_seconds
- go\_memstats\_lookups\_total"
- go\_memstats\_mallocs\_total
- go\_memstats\_mcache\_inuse\_bytes
- go\_memstats\_mcache\_sys\_bytes
- go\_memstats\_mspan\_inuse\_bytes
- go\_memstats\_mspan\_sys\_bytes
- go\_memstats\_next\_gc\_bytes
- go\_memstats\_other\_sys\_bytes
- go\_memstats\_stack\_inuse\_bytes
- go\_memstats\_stack\_sys\_bytes
- go\_memstats\_sys\_bytes
- go\_threads

## Deployment and configuration

By default, the health monitoring and telemetry feature is disabled and the Prometheus server and the node exporters are not deployed.

A configuration file is available during deployment of the Fabric Services System which allows you to:

- Enable the health monitoring and telemetry
- Enable or disable worker node monitoring. These are disabled by default; but when enabled, Prometheus is configured for node metrics/telemetry.
- Configure a scrape interval, which determines how often the telemetry is gathered from all the endpoints and stored. The default value is one minute.
- Configure retention time. This is represented in a time notation with indication of hours (h). For instance "3h". The default retention time is one hour.
- Configure user authentication information. This allows you to manage the users and passwords of the Prometheus service which have Read access.



**Note:** Passwords must be provided as a bcrypt hash. Use any of the standard methods exist to generate bcrypt hashes from text.

The configuration is part of the `user_values.yaml` file that can be found in the `/root` folder of the Deployer VM. This file is used to configure some advanced features in the platform, and may contain additional parameters that impact other features.

For the Health Monitoring feature, the following example contains all the possible settings as described above with example values:

```
prometheus:
  enabled: true
  serverFiles:
    fssconfig.yaml:
      basic_auth_users:
```

```
fss-user: '$2a$12$d81J/Hadc/rb2ei0QYN0T.wCYvSi29RiQ2Q13JR9dcmUtt5L/39i.'
extrauser: '$2b$12$bjTt0toB5WVhC5KAmtrBt0LE2PB5HKqHfLWacytVGnAqLyWcSU1Ry'
server:
  global:
    scrape_interval: 30s
    retention: 6h
  prometheus-node-exporter:
    enabled: true
```



**Note:** The following constraints apply to the health monitoring service:

- Prometheus data is not backed up during a backup of the Fabric Services System.
- the system stores a maximum 2GB of data
- the minimum scrape interval is 30 seconds
- the maximum retention time is 3 hours
- the system supports a maximum of five users

### Enabling health monitoring during installation

After deploying the Deployer VM and deploying the different Fabric Services System node VMs, but before running the actual installation process, create the appropriate entries in the `/root/user_values.yaml` file as described in [Deployment and configuration](#).

The configuration is applied during the regular installation process.

### Accessing the Prometheus metrics

The metrics gathered by Prometheus should only be retrieved using the standard Prometheus API, and using any tool of choice that can use this API. Direct access to the Prometheus UI is not supported.

To access the health monitoring metrics, direct the tool to the `https://fss.domain.tld/prometheus` URI, where "fss.domain.tld" is replaced with the FQDN of the Fabric Services System deployment.

## 16.5.1 Updating the health monitoring configuration after installation

### About this task

You can change the configuration of the health monitoring feature after installation. This includes enabling or disabling aspects of the feature, and changing the passwords for users.

Follow these steps to change the health monitoring configuration:

### Procedure

**Step 1.** Update the `/root/user_values.yaml` file with the new configuration.

**Step 2.** Execute the following command to make sure the Deployer VM has the latest configuration information of your deployment. The command uses the input json file that was used during installation as well. The example shows the expected output as well.

```
# /root/bin/fss-upgrade.sh configure input.json
Timesync service is running on 192.0.2.201 Time difference is -1 seconds
Timesync service is running on 192.0.2.202 Time difference is -1 seconds
Timesync service is running on 192.0.2.203 Time difference is -1 seconds
Timesync service is running on 192.0.2.204 Time difference is -1 seconds
Timesync service is running on 192.0.2.205 Time difference is -1 seconds
```

```
Timesync service is running on 192.0.2.206 Time difference is -1 seconds
Maximum time difference between nodes 1 seconds
Successfully configured. Please run /root/bin/fss-upgrade.sh discover
```

- Step 3.** Execute the following command to update the configuration in the actual application. You must confirm that the new values must be configured (which is considered an ":upgrade"). The example shows the expected output as well.

```
# /root/bin/fss-upgrade.sh upgrade
NAME          STATUS    ROLES          AGE   VERSION
fss-node01    Ready    control-plane,master 12d   v1.23.1
fss-node02    Ready    control-plane,master 12d   v1.23.1
fss-node03    Ready    control-plane,master 12d   v1.23.1
fss-node04    Ready    <none>         12d   v1.23.1
fss-node05    Ready    <none>         12d   v1.23.1
fss-node06    Ready    <none>         12d   v1.23.1
FSS will be upgraded from fss-FSS_23_4_B1-charts-v23.4.1-18 to fss-FSS_23_4_B1-charts
v23.4.1-18 : Are you sure [YyNn]? Y
Upgrade in progress...
Upgrading fss-logs
fss-logs release discovered: fluent-bit-0.20.9 ; Deployer packages fss-logs release:
fluent-bit 0.20.9
fss-logs upgrade not required
Upgrading traefik and ingress routes
traefik release discovered: traefik-21.0.0 ; Deployer packages traefik release:
traefik 21.0.0
traefik upgrade not required
Upgrading kafka and kafkaop if required
kafkaop release discovered: strimzi-kafka-operator-0.31.0 ; Deployer packages kafkaop
release: strimzi-kafka-operator 0.31.0
kafka release discovered: fss-strimzi-kafka-0.1.8 ; Deployer packages kafka release:
fss-strimzi-kafka 0.1.8
kafka and kafkaop upgrade not required
Release "prod" has been upgraded. Happy Helming!
NAME: prod
LAST DEPLOYED: Wed May 24 22:04:36 2023
NAMESPACE: default
STATUS: deployed
REVISION: 4
NOTES:
  Checking for FSS pods
  All FSS pods are running
  Checking for FSS digitalsandbox pods
  FSS digital sandbox pods are running
  Checking for digitalsandbox pods
  Digital sandbox pods are running

  FSS is ready, you can access FSS using https://fss.domain.tld
```

## 16.6 Recovery after application node failure

When an application node (the Kubernetes master node) fails or reboots in a Fabric Services System cluster, the application is unavailable until the services running on that node can be recovered on the remaining application nodes.

This recovery process may require manual intervention as some services do not restart automatically on the remaining nodes.

## Distribution of applications and services across nodes

The different applications or services that are part of the Fabric Services System are deployed across the three application nodes. When a node is unavailable or reboots, Kubernetes reschedules some of these applications on the remaining nodes. By default, Kubernetes waits 5 minutes before it considers a node to be unavailable and starts rescheduling the applications.

Applications in the Fabric Services System deployment can be one of the following types:

- Deployments – stateless applications that Kubernetes automatically reschedules after the unreachable timeout
- Stateful Sets – stateful applications that Kubernetes does not automatically reschedule after the unreachable timeout  
Stateful Sets are applications that use persistent storage. Kubernetes does not automatically restart them because it cannot ensure that it can restart them.

The following procedures are used to instruct Kubernetes to restart the Stateful Set services on a different node in case they are unavailable.

When a node that only contains Deployments becomes unavailable for more than five minutes, the Fabric Services System automatically recovers, as Kubernetes restarts those services immediately on the remaining available nodes without intervention.

By default, the Fabric Services System deploys all the Stateful Set applications on a single node. This strategy lowers the risk of the need for a manual intervention; manual intervention is needed only when the node running the Stateful Sets is down for more than 5 minutes.



**Note:** Execute the following procedures only if the application does not automatically recover when a node is down. By default, it takes Kubernetes up to five minutes to detect that a node is down, and another couple of minutes to restart the stateless applications. Normally, manual intervention is needed only if the node containing the stateful applications is unavailable.

### 16.6.1 Recovering an application after node failure

#### About this task

When a node fails for a longer period and cannot immediately be recovered, the services are also offline for a longer time. In this scenario, Kubernetes must be instructed to restart these services on the remaining nodes so that the application is recovered as quickly as possible.

#### Prerequisites

Execute the commands from a system that has its kubeconfig environment configured to reach the Fabric Services System Kubernetes cluster.

#### Procedure

**Step 1.** Verify which node is offline.

##### Example

In this example, node03 is offline.

```
$ kubectl get nodes
NAME          STATUS    ROLES          AGE   VERSION
fss-node01   Ready    control-plane,master   26d   v1.23.1
fss-node02   Ready    control-plane,master   26d   v1.23.1
fss-node03   NotReady control-plane,master   26d   v1.23.1
```

```
fss-node04 Ready <none> 26d v1.23.1
fss-node05 Ready <none> 26d v1.23.1
fss-node06 Ready <none> 26d v1.23.1
```

**Step 2.** Determine which pods on this node have failed.

### Example

```
$ kubectl get pods -n default --field-selector spec.nodeName=fss-node03
NAME                                READY   STATUS    RESTARTS
AGE
prod-cp-kafka-2                     2/2    Terminating 0
26d
prod-cp-zookeeper-1                 2/2    Terminating 0
26d
prod-fss-cfggen-86859cfb5f-ctn54    1/1    Terminating 0
26d
prod-fss-da-78f9fd7c-cwsv9          1/1    Terminating 0
26d
prod-fss-da-78f9fd7c-rtrc6          1/1    Terminating 0
26d
prod-fss-da-78f9fd7c-wltzd          1/1    Terminating 0
26d
prod-fss-deviationmgr-netinst-7d7fc645bd-qbkf6 1/1    Terminating 0
26d
prod-fss-digital sandbox-7d86cc5fc4-7xfxn 1/1    Terminating 2 (26d ago)
26d
prod-fss-oper-da-67c6d6c6bb-2bzhx   1/1    Terminating 0
26d
prod-fss-oper-da-67c6d6c6bb-8r4w9   1/1    Terminating 0
26d
prod-fss-oper-topomgr-6548c8d6c4-vsttk 1/1    Terminating 1 (26d ago)
26d
prod-fss-topomgr-5f997b544d-4mfnk   1/1    Terminating 0
26d
prod-fss-version-f5b4d74f-9nhss     1/1    Terminating 0
26d
prod-fss-workloadmgr-64ffcf7547-rrvbc 1/1    Terminating 1 (26d ago)
26d
prod-fss-ztp-7bd78ccd9-x5vb7        1/1    Terminating 1 (26d ago)
26d
prod-mongodb-arbiter-0              1/1    Terminating 0
26d
prod-mongodb-secondary-0           1/1    Terminating 0
26d
prod-neo4j-core-0                   1/1    Terminating 0
26d
prod-postgresql-0                   1/1    Terminating 0
26d
```

**Step 3.** Wait for the pods to all show Terminating.

**Step 4.** Force delete all of the pods that are in the Terminating state, as shown in the output of Step 2. Enter the following command for each pod:

```
$ kubectl delete pods --grace-period=0 --force <pod-name>
```

### Example

```
$ kubectl delete pods --grace-period=0 --force prod-fss-cfggen-86859cfb5f-ctn54
Warning: Immediate deletion does not wait for confirmation that the running resource
has been terminated. The resource may continue to run on the cluster indefinitely.
```

```
pod "prod-fss-cfggen-86859cfb5f-ctn54" force deleted
```

**Step 5.** Wait for all pods in the default namespace to be in a Running state again.

This step can take a longer time if there are pods in a CrashLoopBackOff state, because they try to restart only with an increased delay between attempts.

**Step 6.** When all the kafka, zookeeper, postgresql, mongodb and neo4j pods are in a Running state, if pods continue to restart and enter another CrashLoopBackOff state, verify that all kafka pods are running.

To force another restart of the zookeeper pods, execute the following command:

```
$ kubectl -n default scale statefulset --replicas 0 prod-cp-zookeeper
statefulset.apps/prod-cp-zookeeper scaled
```

Wait until the prod-cp-zookeeper pod has scaled down, then scale up the prod-cp-zookeeper pod.

```
$ kubectl -n default scale statefulset --replicas 3 prod-cp-zookeeper
statefulset.apps/prod-cp-zookeeper scaled
```

## 16.6.2 Recovering an application after node reboot

### About this task

When a node reboots, services such as Kafka and Zookeeper try to recover after the reboot, but a known issue with Zookeeper may prevent Kafka and Zookeeper from recovering and recognizing that they are again in a working cluster.

Without a healthy Kafka cluster, the Fabric Services System microservices go into a failed state, as they require a stable Kafka cluster to function.

Complete the following steps to recover when the application is in a failed state and all Kubernetes nodes are available.

### Procedure

**Step 1.** Scale down, then scale up the zookeeper pod.

#### Example

```
$ kubectl -n default scale statefulset --replicas 0 prod-cp-zookeeper
statefulset.apps/prod-cp-zookeeper scaled
```

Wait for the prod-cp-zookeeper pod to scale down, then scale up the prod-cp-zookeeper pod.

```
$ kubectl -n default scale statefulset --replicas 3 prod-cp-zookeeper
statefulset.apps/prod-cp-zookeeper scaled
```

#### Expected outcome

The preceding commands force the Zookeeper service to restart and recover, which enables Kafka and other applications to recover.

**Step 2.** Optional: In some scenarios, you may also need to scale down and scale up the Kafka pod to recover the Kafka service.

**Example**

This command scales down the prod-cp-kafka pod.

```
$ kubectl -n default scale statefulset --replicas 0 prod-cp-kafka
statefulset.apps/prod-cp-kafka scaled
```

Wait for the prod-cp-kafka pod to scale down, then scale the prod-cp-kafka pod.

```
$ kubectl -n default scale statefulset --replicas 3 prod-cp-kafka
statefulset.apps/prod-cp-kafka scaled
```

## 16.7 Repairing a Fabric Services System cluster

After the initial Fabric Services System installation, in the event of node failure (such as when a node is unreachable), you can repair the node cluster by removing the failed cluster and replacing it with another node without having to reinstall or redeploy the system.

To remove a failed node and replace it with another node, use the `/root/bin/setup-k8s.sh` utility in the deployer VM.

Repairing a Fabric Services System cluster involves the following tasks:

1. Removing the failed node from the cluster
2. Adding a replacement node to the cluster, which includes:
  - Updating the `workernodes` section of the `input.json` file and initiating the setup of the Fabric Services System environment
  - Running the `setup-k8s.sh` utility

### 16.7.1 Updating the deployer installation file

#### About this task

The `workernodes` section in the `input.json` file includes the parameters that configure the nodes that you intend to add. The details you provide instruct the deployer how to proceed when setting up the nodes in the Fabric Services System.

#### Procedure

**Step 1.** Update the `workernodes` section in the `input.json`

To add a node, add another section with the parameters below. To delete a node, remove the section for the node that you want to delete.

Table 56: Parameters in the `workernodes` section

Parameter	Description
<code>hostip</code>	Specifies the IP address of the specific worker node.
<code>ip6</code>	Specifies the IPv6 address of the worker node; required if the <code>enable_dual_stack_networks</code> parameter is set to True.

Parameter	Description
<b>hostname</b>	Specifies the hostname of the worker node.
<b>role</b>	Specifies the role of the node. Digital Sandbox nodes: set to <code>digitalsandbox</code> For Kubernetes master nodes: set to <code>master</code> .

**Step 2.** Initiate the setup of the Fabric Services System environment.

#### Example

```
[root@fss-deployer ~]$ fss-install.sh configure input.json
```

## 16.7.2 Removing a node from a cluster

### About this task

Use the following command to remove a node:

```
/root/bin/setup-k8s.sh remove <nodename> [--local-only]
```

Use the `--local-only` argument if the node you are removing cannot be reached.

### Procedure

**Step 1.** Display the nodes in the cluster.

#### Example

```
[root@fss-deployer ~]# kubectl get nodes -owide
NAME                STATUS    ROLES    AGE     VERSION    INTERNAL-IP
EXTERNAL-IP        OS-IMAGE   KERNEL-VERSION CONTAINER-RUNTIME
fss-computendrplc-01 Ready    control-plane,master  78m    v1.23.1    192.168.102.119 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-02 Ready    control-plane,master  77m    v1.23.1    192.168.102.158 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-03 Ready    control-plane,master  77m    v1.23.1    192.168.102.155 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-04 Ready    <none>          76m    v1.23.1    192.168.102.124 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-05 Ready    <none>          27m    v1.23.1    192.168.102.129 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-06 Ready    control-plane,master  7m46s v1.23.1    192.168.102.195 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
```

**Step 2.** Remove a node from the cluster.

**Example**

This example removes node `fss-computendrplc-05` from the cluster.

```
[root@fss-deployer ~]# /root/bin/setup-k8s.sh remove fss-computendrplc-05 [--local-only]
fss-computendrplc-05 is not reachable at 192.168.102.124
SUCCESS: Removed node fss-computendrplc-05
```

**Step 3.** Display the nodes in the cluster**Example**

```
[root@fss-deployer ~]# kubectl get nodes -owide
NAME                STATUS    ROLES    AGE     VERSION    INTERNAL-IP
EXTERNAL-IP    OS-IMAGE    CONTAINER-RUNTIME
fss-computendrplc-01 Ready    control-plane,master    78m    v1.23.1    192.168.102.119 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-02 Ready    control-plane,master    77m    v1.23.1    192.168.102.158 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-03 Ready    control-plane,master    77m    v1.23.1    192.168.102.155 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-04 Ready    <none>    76m    v1.23.1    192.168.102.124 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
fss-computendrplc-06 Ready    control-plane,master    7m46s  v1.23.1    192.168.102.195 <none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64 containerd://1.6.15
```

**Expected outcome**

You can now replace the failed node with the same type of node (that is, replace a failed master node with another master node or replace a failed worker node with another worker node).

**Related topics**

[Adding a worker node to a cluster](#)

[Adding a master node to a cluster](#)

## 16.7.3 Adding a worker node to a cluster

**About this task**

Use the following command to add a worker node. The worker node can also be a Digital Sandbox.

```
/root/bin/setup-k8s.sh addworker <workernode_name>
```

**Note:**

- You cannot add a new storage node; you can only replace an existing one.
- You can only replace with a new node that matches the exact configuration. For example, the storage disk names must be the same.

## Prerequisites

Ensure that the new node is reachable from the deployer VM. As root user, you can connect to the new node with the root login using a passwordless SSH connection with the SSH key in the deployer. This requirement is similar to the requirement for installing the Fabric Services System.

## Procedure

**Step 1.** Update the deployer installation file.

For instructions, see [Updating the deployer installation file](#).

### Example

This procedure adds a Digital Sandbox node. In the `workernode` section of the `input.json` file, the added section for this node should be similar to the following:

```
{
  "hostip": "192.0.2.129",
  "ip6": "",
  "hostname": "fss-computendrplc-05",
  "role": "digitalsandbox"
}
```

**Step 2.** Display the existing configuration.

### Example

```
[root@fss~]# kubectl get nodes -owide
NAME                                STATUS    ROLES    AGE   VERSION   INTERNAL-IP
EXTERNAL-IP  OS-IMAGE          CONTAINER-RUNTIME
fss-computendrplc-01  Ready    control-plane,master  45m   v1.23.1   192.0.2.119
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-02  Ready    control-plane,master  44m   v1.23.1   192.0.2.158
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-03  Ready    control-plane,master  44m   v1.23.1   192.0.2.155
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-04  Ready    <none>          43m   v1.23.1   192.0.2.124
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
```

**Step 3.** Add a new worker node.

### Example

In this example, you are adding compute node `fss-computendrplc-05`.

```
[root@fss]# /root/bin/setup-k8s.sh addworker fss-computendrplc-05
Start adding worker node fss-computendrplc-05
SUCCESS: Added worker node fss-computendrplc-05
```

**Step 4.** Display system configuration:

### Example

```
[root@fss ~]# kubectl get nodes -owide
NAME                                STATUS    ROLES    AGE   VERSION   INTERNAL-IP
EXTERNAL-IP  OS-IMAGE          CONTAINER-RUNTIME
fss-computendrplc-01  Ready    control-plane,master  45m   v1.23.1   192.0.2.119
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-02  Ready    control-plane,master  44m   v1.23.1   192.0.2.158
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-03  Ready    control-plane,master  44m   v1.23.1   192.0.2.155
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-04  Ready    <none>          43m   v1.23.1   192.0.2.124
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-05  Ready    <none>          43m   v1.23.1   192.0.2.129
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
```

```

fss-computendrplc-01 Ready control-plane,master 53m v1.23.1 192.0.2.119
<none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-02 Ready control-plane,master 53m v1.23.1 192.0.2.158
<none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-03 Ready control-plane,master 53m v1.23.1 192.0.2.155
<none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-04 Ready <none> 52m v1.23.1 192.0.2.124
<none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-05 Ready <none> 2m32s v1.23.1 192.0.2.129
<none> Rocky Linux 8.7 (Green Obsidian) 4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15

```

**Related topics**[Removing a node from a cluster](#)

## 16.7.4 Adding a master node to a cluster

**About this task**

The system does not prevent you from adding another master node even if doing so does not conform to the recommended number of master nodes.

Use the following command to add a master node:

```
/root/bin/setup-k8s.sh addmaster
```

**Prerequisites**

Ensure that the new node is reachable from the deployer VM. As root user, you can connect to the new node with the root login using a passwordless SSH connection with the SSH key in the deployer. This requirement is similar to the requirement for installing the Fabric Services System.

**Procedure**

**Step 1.** Update the deployer installation file.

For instructions, see [Updating the deployer installation file](#).

**Example**

In the `workernode` section of the `input.json` file, the added section for the master node should be similar to the following:

```

{
  "hostip": "192.0.2.195",
  "ip6": "",
  "hostname": "fss-computendrplc-06",
  "role": "master"
}

```

**Step 2.** Display the existing configuration.

**Example**

```

[root@fss-deployer ~]# kubectl get nodes -owide
[root@fss-deployerndrplc userdata]# kubectl get nodes -owide

```

NAME	EXTERNAL-IP	STATUS	ROLES	AGE	VERSION	INTERNAL-IP
		OS-IMAGE		KERNEL-VERSION		
CONTAINER-RUNTIME						
fss-computendrplc-01		Ready	control-plane,master	57m	v1.23.1	192.0.2.119
<none>		Rocky Linux 8.7 (Green Obsidian)		4.18.0-425.13.1.el8_7.x86_64		
containerd://1.6.15						
fss-computendrplc-02		Ready	control-plane,master	57m	v1.23.1	192.0.2.158
<none>		Rocky Linux 8.7 (Green Obsidian)		4.18.0-425.13.1.el8_7.x86_64		
containerd://1.6.15						
fss-computendrplc-03		Ready	control-plane,master	57m	v1.23.1	192.0.2.155
<none>		Rocky Linux 8.7 (Green Obsidian)		4.18.0-425.13.1.el8_7.x86_64		
containerd://1.6.15						
fss-computendrplc-04		Ready	<none>	56m	v1.23.1	192.0.2.124
<none>		Rocky Linux 8.7 (Green Obsidian)		4.18.0-425.13.1.el8_7.x86_64		
containerd://1.6.15						
fss-computendrplc-05		Ready	<none>	6m32s	v1.23.1	192.0.2.129
<none>		Rocky Linux 8.7 (Green Obsidian)		4.18.0-425.13.1.el8_7.x86_64		
containerd://1.6.15						

**Step 3.** Add a new master node.

#### Example

In this example, you are adding compute node `fss-computendrplc-06`.

```
[root@fss-deployer ~]# /root/bin/setup-k8s.sh addmaster
Start adding master node/s
SUCCESS: Added master node
```

**Step 4.** Verify the addition of the master node.

#### Example

```
[root@fss-deployer ~]# kubectl get nodes -owide
NAME          STATUS    ROLES    AGE     VERSION    INTERNAL-IP
EXTERNAL-IP  OS-IMAGE                                KERNEL-VERSION
CONTAINER-RUNTIME
fss-computendrplc-01  Ready    control-plane,master  73m    v1.23.1    192.0.2.119
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-02  Ready    control-plane,master  73m    v1.23.1    192.0.2.158
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-03  Ready    control-plane,master  73m    v1.23.1    192.0.2.155
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-04  Ready    <none>    72m    v1.23.1    192.0.2.124
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-05  Ready    <none>    22m    v1.23.1    192.0.2.129
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
fss-computendrplc-06  Ready    control-plane,master  3m7s   v1.23.1    192.0.2.195
<none>          Rocky Linux 8.7 (Green Obsidian)  4.18.0-425.13.1.el8_7.x86_64
containerd://1.6.15
```

#### Related topics

[Removing a node from a cluster](#)

## 16.8 Backup and restore

The Fabric Services System uses Kubernetes' ability to take a point-in-time snapshot to implement backup and restore capabilities. A snapshot represents a point-in-time copy of a volume. A snapshot can be used to provision a new volume or to restore the existing volume to the previous state captured in the snapshot.

To take these point-in-time snapshots, the container storage interface (CSI) standard is used. This standard allows the creation and deletion of volume snapshots via the Kubernetes API and the creation of new volumes pre-populated with the data from a snapshot via Kubernetes dynamic volume provisioning.

The Fabric Services System uses the BorgBackup program to move the point-in-time snapshot to a remote location. For more information about the BorgBackup program, see [BorgBackup documentation](#). Through BorgBackup, the backup solution supports:

- deletion of duplicated data (data deduplication) to lower the amount of storage used
- data encryption to secure the backup
- (optional) remote storage of the backup over SSH on a remote server (using SSHFS)

### Best practices and considerations

Nokia recommends that you follow the best practices listed below when backing up the Fabric Services System:

- Store the backup data on an external system for extra safety. If you store the backup on the deployer VM and the deployer VM is lost, you have no access to your backups to restore.
- Schedule the backup to be executed on a regular basis, for instance every night, so that you have regular backups, in case there is a need to restore.

Be aware of the following considerations when deploying the Fabric Services System:

- The backup process brings down the application for 5-10 minutes to create a consistent backup, as described in [Backup consistency in a microservice architecture](#). Take this down time into account when planning the regular backup window.
- While the application is down, no changes can be made to the environment. New alarms are raised when the application has started again. Alarms that happen during the downtime and are resolved within the same downtime are not raised, as they were already resolved.

### Backup consistency in a microservice architecture

A backup must be consistent across all the microservices of an application to guarantee the restore capabilities. To achieve consistency, before the snapshots can be taken, ensure that all the microservices have finished their write activity and that no further changes are accepted or incoming to the system. The best way to achieve this is by blocking access to the services temporarily.

The backup script has an option (**-s**) to specifically guarantee this consistency by scaling down all the microservices for a short duration (the time to take the snapshots). Using this option prevents any changes from being made while the snapshot is taken. Nokia strongly recommends the use of this option; it is a requirement in production environments. If you omit this option, the restore functionality cannot be guaranteed.

Additionally, execute a backup only when no background tasks are running, such as configuration generation or deployment of a fabric or workload VPN intent.

The deployer VM is used for the backup procedure and ensures that the Kubernetes cluster has access to the registry with the image of all services, in case they are needed.

### Supported restore scenarios

The restore operation from a backup always requires a fresh installation of the entire Fabric Services System deployment. The restore takes place as part of the installation of a fresh environment. The restore process has the following requirements:

- The same version of the Fabric Services System must be used; for example, if a backup is taken from a deployment with version 22.8.1, the restore must use that same version.
- The Fabric Services System nodes on which the restore is executed must have the same IP addresses and FQDNs as the environment from which the backup was taken.
- The `input.json` file that was used by the deployer VM for the installation of the environment from which the backup was taken must be used for the restore procedure.

### Backup and restore scripts

Execute backup and restore commands from the deployer VM. The deployer VM must have passwordless (key-based) SSH access to all other Kubernetes nodes as the root user. If a remote server is used through the `FSS_BACKUP_REPOS` environment variable, passwordless (key-based) SSH is also required to the remote server.

Use the **fss-backup.py** script for the backup operation.

```
fss-backup.py [-h] -b backup-name [-s true] [-i] [-l [location]] [-c]
```

Use the **fss-restore.py** script for the restore operation.

```
fss-restore.py [-h] -b backup-name [-l [location]] [-c]
```

The backup and restore scripts are located in the `/root/bin/backup-restore` directory of the deployer VM.

The following table describes the options for the preceding commands.

Table 57: Parameter descriptions

Option	Specifies
<b>-h</b>	Display the help screen and describes the usage of the script and CLI arguments.
<b>-b backup-name</b>	Name of the backup file, consisting of lower-case alphanumeric characters, '-' or '.', and must start and end with an alphanumeric character.
<b>-s true</b>	<p>Enable automatic scale down of the Fabric Services System deployment before taking the snapshots. This option guarantees that all services are temporarily down and a consistent backup of the entire environment is taken. This option is strongly recommended.</p> <p> <b>Note:</b> If you run the <b>fss-backup.py</b> command with the <b>-s true</b> option, expect deviations in the fabric intents for about 1 to 2 minutes after the pods are scaled up.</p>

Option	Specifies
<b>-i</b>	Include the logs volume in the backup.
<b>-l location</b>	Full path to the folder where the backup file is stored, for example /data/backups. This option overwrites and takes precedence over the FSS_BACKUP_REPOS environment variable. This folder is local on the Kubernetes node where the backup command is executed but can be mounted from a remote location like an NFS storage. Use the following format: <ul style="list-style-type: none"> <li>• if the backup file is stored in a local directory: /&lt;full path&gt;/&lt;directory&gt;, for example, /data/backups</li> <li>• if the backup file is stored in a remote location: user@&lt;ip address hostname&gt;:/&lt;full path&gt;/&lt;directory&gt;</li> </ul>
<b>-c</b>	Clean up a failed backup.

The scripts read the following environment variables:

- **BORG\_PASSPHRASE** – a required environment variable; this password string is required by the Borg Backup utility to encrypt or decrypt backup files  
Keep this pass phrase safe and secure; if you lose it, access to the backup data is lost as well.
- **FSS\_BACKUP\_REPOS** – specifies the location where the backup files are stored.  
The directory can be local or remote. You can use the **-l** option to override the setting for the **FSS\_BACKUP\_REPOS** variable.
- **BACKUP\_RESTORE\_LOGS** – specifies the log location for the backup and restore operation  
If this variable is not configured, logs are stored in the `logs` directory where scripts are being executed.

In the following example, `compute6` is used as the backup repository.

```
export BORG_PASSPHRASE='XYZ2022_n0kIaNeTw0rKs_FsS1!&&123'
export FSS_BACKUP_REPOS=root@compute6:backup-repo
export BACKUP_RESTORE_LOGS=/root/bin/backup-restore/logs
```

You can set environment variables in the `~/ .env` directory in the deployer VM:

```
root@fss-deployer backup-restore]# cat ~/.env
KUBECONFIG=/var/lib/fss/config.fss
BORG_PASSPHRASE='XYZ2022_B0RgPasSheRE_FsS1!&&123'
FSS_BACKUP_REPOS=root@compute6:backup-repo
BACKUP_RESTORE_LOGS=/root/bin/backup-restore/logs
root@fss-deployer backup-restore]#
```

## 16.8.1 Backing up

### About this task

The resulting backup is a full backup.

## Procedure

- Step 1.** Run the backup script on the deployer VM of the Fabric Services System deployment that you are backing up.

```
/root/bin/backup-restore/fss-backup.py -b <backup_name> [-s true] [-i] [-l <location>]
```

### Example

```
/root/bin/backup-restore/fss-backup.py -b 0810-fss -s true
```



**Note:** If you run the **fss-backup.py** command with the **-s true** option, expect deviations in the fabric intents for about 1 to 2 minutes after the pods are scaled up.

- Step 2.** Display the status of the backup operation.

### Example

You can view the status of the backup in the `status.json` file in the logs directory.

```
[root@fss-deployer backup-restore]# cat logs/0810-fss-logs/status.json{
  "fss_version": "v22.8.0-5",
  "status": "backup_completed",
  "completed_pvcs": [
    "data-prod-postgresql-0",
    "datadir-prod-mongodb-primary-0",
    "datadir-prod-mongodb-secondary-0",
    "datadir-prod-neo4j-core-0",
    "prod-fss-dhcpd",
    "prod-fss-dhcpd-lease",
    "prod-fss-image-mgmtstack",
    "prod-fss-node-checkpoints",
    "prod-fss-nodecfg"
  ],
  "failed_pvcs": [],
  "failure_msg": "",
  "last_updated_at": "2022-08-10T17:51:23.521569+00:00"
}
```

- Step 3.** Verify the backup files in the repository.

### Example

```
[root@compute1 backup-repo]# borg list 0810-fss --short
manifests
data-prod-postgresql-0
datadir-prod-mongodb-primary-0
datadir-prod-mongodb-secondary-0
datadir-prod-neo4j-core-0
prod-fss-dhcpd
prod-fss-dhcpd-lease
prod-fss-image-mgmtstack
prod-fss-node-checkpoints
prod-fss-nodecfg
```

## 16.8.2 Restore a backup and install the Fabric Services System application

### Restore process

The restore process consists of the following high-level tasks:

1. Setting up new Fabric Services System nodes with the same IP addresses and FQDNs.
2. Installing the Kubernetes cluster only (not the application) using the same `input.json` that was used for the original environment.
3. Restoring the backup data from the deployer node.
4. Installing the Fabric Services System application components using the same `input.json` that was used for the original environment.
5. For Digital Sandbox only, restoring Digital Sandbox fabrics.

### Restoring a geo-redundant deployment

In a geo-redundant deployment, restore the active system from the backup using the process described above. When the active site is up and running, bring up the standby site from the `input.json` and the SSH files. When the standby is up, just follow the <Geo reundance configuration on UG> to configure GR between the active and the standby FSS.

### 16.8.2.1 Setting up new Fabric Services System nodes

#### About this task

The restore of the Fabric Services System backup has to be executed in a clean environment that uses the same IP addresses and FQDNs of the original environment. You can use the procedures from the *Fabric Services System Software Installation Guide* to set up such an environment.

#### Procedure

- Step 1.** Optional: Complete the procedure "Deploying and configuring the Fabric Services System deployer VM".



**Note:** Complete this procedure only if the deployment VM is also lost or needs to be reinstalled. Ensure that you are using the appropriate version, that is, the same version that was used to install original environment.

- Step 2.** Create and configure the Fabric Services System virtual machine nodes.  
You must use the Fabric Services System base OS image. For instructions, see *Preparing the Fabric Services System virtual machine nodes*.

#### Expected outcome



**Note:** When the nodes have been installed, do not execute "Installing Fabric Services System" in the *Fabric Services System Software Installation Guide*.

## 16.8.2.2 Installing the Kubernetes cluster

### About this task

Execute this procedure on the deployer VM using the same `input.json` that you used for the installation of the original deployment.

### Procedure

**Step 1.** Initiate the setup.

#### Example

```
[root@fss-deployer ~]$ /root/bin/fss-install.sh configure sample-input.json
```



**Note:** Do not execute the `fss-install.sh` script as suggested by the output of the preceding command.

**Step 2.** Start the installation of Kubernetes.

#### Example

```
[root@fss-deployer ~]$ /root/bin/setup-k8s.sh
```

The installation time varies depending on the capacity of your system.

## 16.8.2.3 Restoring a backup

### About this task

Use this procedure to restore a previously saved backup.

### Prerequisites

- Ensure that the backup folder created by the backup script is available to the `restore` command on the deployer VM on which you are executing this procedure. You can configure this set up by:
  - ensuring that the `FSS_BACKUP_REPOS` environment variable points to a remote server using SSH or SSHFS
  - putting the backup folder on the Kubernetes node itself and using the `-l` option, similar to how the backup was taken
- Ensure that the `BORG_PASSPHRASE` environment variable is set to the correct passphrase, otherwise, the restore fails.

### Procedure

**Step 1.** Restore the backup from the deployer VM.

```
/root/bin/backup-restore/fss-restore.py -b backup_name [-l location]
```

#### Example

```
./root/bin/backup-restore/fss-restore.py -b 0810-fss
```

The script does following:

- a. checks if the specified backup filename exists in the specified location in the Borg repository; if the file is not present, the script terminates
- b. mounts the Borg repository locally to read the backup content
- c. creates PVCs from the YAML files in the manifest archive in the Borg repository that contains the PVC definition for all the original PVCs; also creates PVs of the same size as the original PVs in the source cluster
- d. creates a mount pod for each PVC on the node where the restore script is being executed, then mounts newly created empty PVCs
- e. copies the volume content of the mounted backup to the corresponding PVC volume
- f. deletes the PVC mount pod; the PVCs and PVs are still present and point to the new data volume where data is copied

**Step 2.** Check the status of the restore operation in the `status.json` file in the `logs/<backup-name>-restore-logs` directory.

#### Example

```
[root@fss-deployer backup-restore]# cat logs/0501-3-fssbackup-restore-logs/status.json
{
  "fss_version": "v22.8.0-5",
  "status": "restore_completed",
  "completed_pvcs": [
    "data-prod-postgresql-0",
    "datadir-prod-mongodb-primary-0",
    "datadir-prod-mongodb-secondary-0",
    "datadir-prod-neo4j-core-0",
    "prod-fss-dhcpd",
    "prod-fss-dhcpd-lease",
    "prod-fss-image-mgmtstack",
    "prod-fss-node-checkpoints",
    "prod-fss-nodecfg"
  ],
  "failed_pvcs": [],
  "failure_msg": "",
  "last_updated_at": "2022-08-10T17:51:23.521569+00:00"
}
```

**Step 3.** Verify the restored files.

Execute the `kubectl get pvc` command. The output should match the output of the `borg list` command in step 3 of [Backing up](#).

#### Expected outcome

When the output shows that the PVC have been installed, you can install the Fabric Services System application.

## 16.8.2.4 Installing the Fabric Services System application

### About this task

Now that the backup data has been restored, you can install the Fabric Services System in the Kubernetes cluster. Use the deployer VM from which you executed the commands to install the Kubernetes cluster.

## Procedure

**Step 1.** Initiate the setup.

### Example

```
[root@fss-deployer ~]$ /root/bin/fss-install.sh configure sample-input.json
```



**Notice:** Do not execute the **fss-install.sh** script as suggested by the output of the preceding command.

**Step 2.** Start the installation of the Fabric Services System application.

### Example

The installation time varies depending on the capacity of your system.

```
[root@fss-deployer ~] $ /root/bin/fss-app-install.sh
```

## Expected outcome

At the end of this procedure, the Fabric Services System application has been restored from your backup and is ready for use.

## 16.8.2.5 Restoring Digital Sandbox fabrics

### About this task

Digital Sandbox fabrics are not automatically restarted after a deployment has been restored from a backup. To recover all the Digital Sandbox Fabrics, follow these steps:

### Procedure

**Step 1.** Complete the procedure [Updating the Digital Sandbox](#).

#### Expected outcome

This action triggers a full redeployment of all Digital Sandbox components for the region.

**Step 2.** From the **Fabric intents** overview page, wait until all nodes show up as blue.

This state indicates all nodes are associated and are up.

**Step 3.** Complete the procedure [Deploying a fabric intent in the Digital Sandbox](#).

**Step 4.** Repeat Steps 2 and 3 for each Digital Sandbox fabric.

#### Expected outcome

At the end of procedure, all Digital Sandbox fabrics should be running in the same state as before the backup.

**Step 5.** Restore previously added software catalog images.

If you previously added SR Linux images to the software catalog, you must modify the software catalog, then upload the SR Linux images. For instructions, see [Adding a new network operating system version to the software catalog](#) and [Uploading SR Linux container images for Digital Sandbox](#).

### Related topics

[Fabric intent deployment](#)

[Adding a fabric intent to the deployment pipeline](#)  
[Deploying a fabric intent from the deployment pipeline](#)

## 16.9 Geo-redundancy

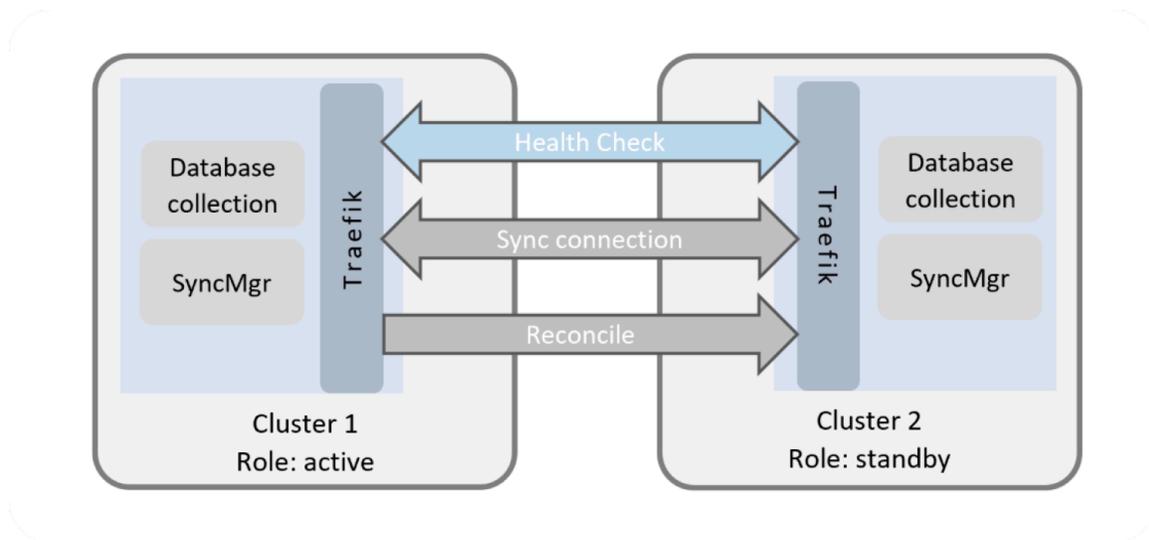
Geo-redundancy is the practice of replicating data and applications across multiple geographic locations. If the system fails in one location, a system in another location can continue to provide services without interruption.

The Fabric Services System is deployed to manage one or more data centers, which can require changes in the fabric on short notice or be dynamically integrated with OSS and CMS platforms. In such an environment, the availability of the Fabric Services System is crucial; if the system is unavailable, no changes can be made to the fabric and the network and applications are at risk. Typically, failures are not within the application or single rack infrastructure, but rather from power outages or network outages in the data center where the system is deployed. For this reason, disaster recovery plans are in place so infrastructure and applications can recover quickly.

As part of disaster recovery plans, the Fabric Services System supports geo-redundancy, in which a backup deployment of Fabric Services System is dormant and waiting to be activated in case a disaster happens on the active site. This standby site has the necessary data and synchronizes all data from the active site that is needed to quickly recover fabric management functionality if so instructed.

In the Fabric Services System, geo-redundancy is configured between two independent clusters. One system is configured as the active cluster and the other the standby.

Figure 37: Geo-redundancy in the Fabric Services System



After geo-redundancy has been configured on both the active and standby instances and the sync connection is active, changes to the active instance are replicated on the standby instance. A health check between the active and standby instances detects anomalies and, if present, generates alarms.

The active site manages the fabric and accepts API and configuration changes. The active site ensures that the standby site is in sync with the active site and pushes all required data to the standby site.

The standby site has continuously synchronizes all data from the active site so that the standby site can quickly recover fabric management functionality if so instructed.

The standby site does not actively manage or monitor any fabric; it operates in read-only mode. The UI loads with the **Geo-Redundancy** page as its home page and only the **Alarms** and **Geo-Redundancy** pages are available on the main menu. The API is disabled, except for GET calls for alarms and geo-redundancy.

You can configure geo-redundancy using the UI or REST API.

## 16.9.1 Geo-redundancy for systems integrations with Connect plugins

For systems integrated with a Connect plugin, see the applicable topic in the *Fabric Services System Connect Guide*:

- If the geo-redundant Fabric Services System setup is integrated with an OpenShift/Kubernetes cluster using Fabric Services System Connect, see *Geo-Redundancy* in *The OpenShift and Kubernetes plugin*.
- If the geo-redundant Fabric Services System setup is integrated with an OpenStack cluster using Fabric Services System Connect, see *Geo-Redundancy* in *The OpenStack plugin*.
- If the geo-redundant Fabric Services System setup is integrated with a VMware cluster using Fabric Services System Connect, see *Geo-Redundancy* in *VMware plugin*.

## 16.9.2 Geo-redundancy operations

### Synchronizing (sync)

During geo-redundancy configuration, the sync connection is established between the active and standby sites. The active and standby clusters are considered synchronized after a successful heartbeat exchange between active and standby sites. When the sync connection is active, configuration data is copied from the active to the standby site, which includes all the data needed by the standby system to manage the fabric as needed.



#### Note:

- Data that can be relearned or regenerated within a reasonable time (minutes) if it was not fully synced.
- Software images are not synced because they are large files. If a standby cluster becomes the new active cluster, it uploads images from the local image source. Software images are not transferred from an old active cluster to new active cluster.
- Performance metrics or platform health metrics (Prometheus data) are not synchronized.

The options to restart and stop synchronizing are available only on the active site. When the active system stops synchronizing, it goes into the Sync Stopped state and becomes read-only. When the standby system stops synchronizing, it goes to a Sync Aborted state.

### Audits

If the sync connection is down and the active system is reachable and is operational, you can perform sync and reconcile operations to retain the active site. If you want to failover to the standby system and make it active, you must run an audit on a standby system before you can make it active.

During an audit, each service in the cluster verifies its own data, in sequence, to ensure that no data in its database is in an inconsistent state. An audit is not a verification of data between the clusters in a geo-redundant system. An audit is performed on the standby system. The system blocks any changes to system configuration while an audit is in progress.

You can run an audit only when the sync connection is down between the active and standby cluster, and the standby system is in the Sync Aborted state. After an audit completes successfully, you can:

- initiate failover: make the original standby system active and the original active system the standby  
For instructions, see [Initiating failover: switching between the active and standby clusters](#).
- initiate standalone operations for the standby system  
For instructions, see [Making the standby system active - standalone operation](#).

You can generate an audit report to display, for each app, what that app tried to do to correct the inconsistencies of the respective data collections. If the data is already consistent, the report may not contain much information.

In the rare event of an audit failure (that is, the state moves to Audit Fail), you can recover by reinstalling the configuration from backup. For instructions, see [Backup and restore](#).

## Reconciling

The reconcile operation initiates the replication of the data set from the active to the standby cluster. This action replaces the data set in the standby system with the data set from the active system.

This operation is needed when the data between the active and standby sites is not in sync, such as in the following scenarios:

- When the sync is first established (such as during the initial geo-redundancy configuration)
- When the sync connection recovers after an unintentional disruption in the sync connection



### Note:

- This operation is available from the active system.
- The sync connection must be active before you can initiate the reconcile operation.
- There should be no pending workload jobs, deployments or any operations which could potentially modify the database in the background.

### 16.9.2.1 REST API geo-redundancy operations

The operations allowed on the REST API varies depending on current geo-redundant status of the site.

*Table 58: Allowed REST operations based on current state*

State	Active site	Standby site
STANDALONE	Read/Write	Read/Write
ACTIVE_SYNCING	Read/Write	Not applicable
SYNC_STOPPED	Read/Write	Not applicable
SYNC_ABORTED	Read-only	Read-only
AUDIT	Not applicable	Read-only

State	Active site	Standby site
AUDIT_DONE	Not applicable	Read-only
AUDIT_FAILED	Not applicable	Read-only
STANBY_SYNCING	Not applicable	Read-only
RECONCILE	Read-only	Read-only

### 16.9.3 Geo-redundancy configuration

#### Deployment considerations

- Geo-redundancy works for one, three, or six node deployments, but both clusters must have the same type of node deployment.
- The active and standby deployments must have the same number of nodes and the same resource configuration.
- The active and standby deployments must have the same version installed.
- Signing certificates must be aligned.  
For instructions, see [Realigning certificates](#).

#### Networking considerations

Geo-redundancy has a few requirements and considerations from a networking perspective and connectivity between the active and standby site:

- Synchronization uses the API service and should use the OAM network. When configuring geo-redundancy, make sure to use the FQDN or VIP of the other cluster on the OAM network.
- Connectivity between the active and standby cluster can be through a stretched L2 subnet between the sites, or routed with two different L2 subnets.
- The active and standby site must use different IP and VIP addresses.
- Synchronisation is supported over IPv4 and IPv6.
- The maximum allowed RTT latency between the active and standby sites is 100ms, but a maximum of 50ms is highly recommended. The lower the RTT latency, the better.
- The connection speed between the active and standby site must be a minimum of 1Gbps.

#### ZTP and DHCP handling after active failure

When the active site fails and the standby site has not been activated yet, the DHCP and ZTP capabilities of the platform are unavailable. At that time, SR Linux nodes cannot be rebooted, bootstrapped, or upgraded.

After the standby site has been made active, the standby site now runs the DHCP service and supports the ZTP process of SR Linux nodes.

#### Geo-redundancy configuration tasks

Following are the high-level tasks that you need complete to configure a geo-redundant system.

1. Deploy the deployer VM on the active and standby sites.

For instructions, see “The Fabric Services System deployer VM” in the *Fabric Services System Installation Guide*.

2. Deploy the Fabric Services System on the active and the standby sites, using the installation procedures provided in the *Fabric Services System Installation Guide*.



**Note:** You can also upgrade an existing standalone deployment first, then set up the standby site.

3. [Configuring geo-redundancy information in deployer VMs](#)
4. [Verifying that the setup is ready for geo-redundancy using the deployer VMs](#)
5. [Realigning certificates](#)
6. [Configuring geo-redundancy](#)

### 16.9.3.1 Configuring geo-redundancy information in deployer VMs

#### About this task

Use this procedure to configure the deployer VMs with the remote site details. The steps in this procedure help you view the status of both the local and remote Fabric Services System clusters and determine whether both the sites are configured to allow for geo-redundancy to work correctly.

This section is optional as it does not affect the actual geo-redundancy functionality of the platform. Configuring the deployer VMs on the active and standby site to know about each other, does help in potential troubleshooting and inspecting the infrastructure for discrepancies.

#### Procedure

- Step 1.** Configure passwordless SSH access locally on both the active and standby deployer VMs. Enter the following command on both the active and standby deployer VMs:

```
cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys
```

- Step 2.** Configure passwordless SSH access from the local deployer VM to the remote deployer VM and vice-versa.

Copy the contents of the `/root/.ssh/id_rsa.pub` file on the remote deployer and update the `/root/.ssh/authorized_keys` file of the local deployer.

- Step 3.** Add the necessary details from the remote site by copying the `input.json` file from the remote site.

Enter the following command:

```
fss-install.sh add-remote-deployer <input_json_of_remote_deployer>
```



**Note:** For this command to work, on the `input.json` file of the active site, the `deployernode.role` field must be set to `active` and on the `input.json` file of the standby site, the `deployernode.role` field must be set to `standby`. This setting is needed so each deployer VM is aware which site is considered active and which is standby by default.

**Step 4.** Repeat step 3 from the remote deployer.

**Step 5.** Verify the configuration by displaying the contents of the `sites.json` file on both deployer VMs.

### Example

```
[root@fss-deployersite01 ~]# cat /var/lib/fss/sites/sites.json
{
  "local": {
    "name": "site01",
    "ipv4": "10.x.x.1",
    "ipv6": "",
    "accessip": "10.x.x.1",
    "role": "active"
  },
  "remote": [
    {
      "name": "site02",
      "ipv4": "10.x.x.11",
      "ipv6": "",
      "accessip": "10.x.x.11",
      "role": "standby"
    }
  ]
}
```

### 16.9.3.2 Verifying that the setup is ready for geo-redundancy using the deployer VMs

The deployer VM provides the following tools that you can use to verify and display information about geo-redundancy configuration and status:

- `cat /var/lib/fss/sites/sites.json`: displays the local and remote clusters, the access IP address and IP addresses, and role of each cluster
- `/root/bin/fss-install.sh status-georedundancy`: displays basic geo-redundancy status
- `/root/bin/fss-install.sh status-georedundancy -v`: displays detailed geo-redundancy status
- `/root/bin/fss-install.sh status-georedundancy -t site01`: displays the details of the Fabric Services System cluster, certificates, and applications

#### Example: Status of geo-redundancy, basic output

In the output, Active(`fd56:1:91:2::21`) vs Standby(`fd56:1:91:2::6a`) reports the IP addresses used to connect to the active and standby sites.

```
[root@fss-deployersite01 ~]# /root/bin/fss-install.sh status-georedundancy
=====
Sites Overview
=====
+-----+-----+-----+-----+
|  NAME  |  ROLE  |  STATUS  |  CONSISTENCY  |
+-----+-----+-----+-----+
| site01(self) | active | GOOD | N/A |
| site02  | standby | GOOD | ERROR |
+-----+-----+-----+-----+
=====
Active(fd56:1:91:2::21) vs Standby(fd56:1:91:2::6a)
=====
+-----+-----+
|  NAME  |  STATUS  |
+-----+-----+
```

NODES	GOOD
PASSWORDS	GOOD
CERTIFICATES	MISMATCH
VERSION	GOOD



**Note:** If the system displays ERROR or MISMATCH in the output, align both sites so they have the same configuration before you start the procedure [Configuring geo-redundancy](#).

You can also display the output in YAML format:

```
[root@fss-deployersite01 ~]# /root/bin/fss-install.sh status-georedundancy -o yaml
Overview:
- NAME: site01(self)
  ROLE: active
  STATUS: GOOD
  CONSISTENCY: N/A
- NAME: site02
  ROLE: standby
  STATUS: GOOD
  CONSISTENCY: ERROR
standby-site02:
- NAME: NODES
  STATUS: GOOD
- NAME: PASSWORDS
  STATUS: GOOD
- NAME: CERTIFICATES
  STATUS: MISMATCH
- NAME: VERSION
  STATUS: GOOD
[root@fss-deployersite01 ~]#
```

If the CONSISTENCY column reports an error, use the **-v** option with the **/root/bin/fss-install.sh status-georedundancy** command to display more information. To display detailed information about a particular site, use the **fss-install.sh status-georedundancy -t <site name>** command.

## Detailed geo-redundancy information

Use the **/root/bin/fss-install.sh status-georedundancy -v** command to display details about consistency errors. In the output, the Sites Overview section shows a consistency error. The subsequent sections indicate the area with the consistency error. In the example below, the details about the error are shown in Details about CERTIFICATES section. The serial number mismatch is not a severe issue, but the different node CAs between in site01 and site02 should be addressed.

### Example

```
[root@fss-deployersite01 ~]# /root/bin/fss-install.sh status-georedundancy -v
=====
Sites Overview
=====
+-----+-----+-----+-----+
|  NAME  |  ROLE  | STATUS | CONSISTENCY |
+-----+-----+-----+-----+
| site01(self) | active | GOOD  | N/A         |
|  site02  | standby | GOOD  | ERROR       |
+-----+-----+-----+-----+
=====
Active(fd56:1:91:2::21) vs Standby(fd56:1:91:2::6a)
=====
```

```
-----
Details about fss VERSION
-----
```

NAME	CHARTVERSION	STATUS
cert-manager	GOOD	GOOD
fss-logs	GOOD	GOOD
kafka	GOOD	GOOD
kafkaop	GOOD	GOOD
metallb	GOOD	GOOD
prod	GOOD	GOOD
rook-ceph	GOOD	GOOD
rook-ceph-cluster	GOOD	GOOD
traefik	GOOD	GOOD

```
-----
Details about CERTIFICATES
-----
```

CERTSOURCE	ISSUER	SUBJECT	SERIAL-NUMBER	VALIDTO
fss gui/rest	GOOD	GOOD	MISMATCH	GOOD
kafka	GOOD	GOOD	MISMATCH	GOOD
node CA	ERROR	ERROR	ERROR	ERROR

```
-----
Details about NODES
-----
```

NAME	CONSISTENCY
master_cnt	GOOD
total_cnt	GOOD

```
-----
Details about PASSWORDS
-----
```

APP	USER	CONSISTENCY
mongodb	root	GOOD
mongodb	fss_user	GOOD
neo4j	root	GOOD
keycloak	master	GOOD
keycloak	fss	GOOD
keycloak	ztp	GOOD
postgresql	root	GOOD
postgresql	keycloak	GOOD
kafka	fss-kafka-admin	GOOD

You can also display the output in YAML format:

```
[root@fss-deployersite01 ~]# /root/bin/fss_geo_redundancy.py status -v -o yaml
Overview:
- NAME: site01(self)
  ROLE: active
  STATUS: GOOD
  CONSISTENCY: N/A
- NAME: site02
  ROLE: standby
  STATUS: GOOD
```

```
CONSISTENCY: ERROR
standby-site02:
HELMVERSION:
- NAME: cert-manager
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: fss-logs
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: kafka
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: kafkaop
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: metallb
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: prod
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: rook-ceph
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: rook-ceph-cluster
  CHARTVERSION: GOOD
  STATUS: GOOD
- NAME: traefik
  CHARTVERSION: GOOD
  STATUS: GOOD
CERTIFICATES:
- CERTSOURCE: fss gui/rest
  ISSUER: GOOD
  SUBJECT: GOOD
  SERIAL-NUMBER: MISMATCH
  VALIDTO: GOOD
- CERTSOURCE: kafka
  ISSUER: GOOD
  SUBJECT: GOOD
  SERIAL-NUMBER: MISMATCH
  VALIDTO: GOOD
- CERTSOURCE: node CA
  ISSUER: ERROR
  SUBJECT: ERROR
  SERIAL-NUMBER: ERROR
  VALIDTO: ERROR
NODES:
- NAME: master_cnt
  CONSISTENCY: GOOD
- NAME: total_cnt
  CONSISTENCY: GOOD
PASSWORDS:
- APP: mongodb
  USER: root
  CONSISTENCY: GOOD
- APP: mongodb
  USER: fsp_user
  CONSISTENCY: GOOD
- APP: neo4j
  USER: root
  CONSISTENCY: GOOD
- APP: keycloak
  USER: master
  CONSISTENCY: GOOD
```

```

- APP: keycloak
  USER: fss
  CONSISTENCY: GOOD
- APP: keycloak
  USER: ztp
  CONSISTENCY: GOOD
- APP: postgresql
  USER: root
  CONSISTENCY: GOOD
- APP: postgresql
  USER: keycloak
  CONSISTENCY: GOOD
- APP: kafka
  USER: fss-kafka-admin
  CONSISTENCY: GOOD

```

### 16.9.3.3 Realigning certificates

#### About this task

When the independent clusters of a geo-redundant system are first installed, each site has a set of default self-installed certificates; the node-signing CA is unique for each cluster. In a geo-redundant system, the node-signing certificate must be the same for the active and standby cluster.



**Note:** Before configuring geo-redundancy, the node signing certificates must be aligned. In addition, if custom signing certificates are in use for northbound, UI, and Kafka interfaces, these certificate files must also be aligned.

Use the **fss-certificate.sh export [-d directory]** utility to export the signing certificate files installed in the intended active system to a local directory. If you do not specify a directory, the certificates are exported to the local `/root/userdata/certificates` directory. Then, copy the needed certificate files to the intended standby and deploy them.

#### Procedure

**Step 1.** From the deployer of the intended active system, execute the **fss-certificate.sh export** command.

#### Example

The following output shows that default signing certificates are present.

```

[root@fss-deployer ~]# /root/bin/fss-certificate.sh export
Certificates will be exported to /root/userdata/certificates
Default install generated signing certificates are in use for generating node
certificates
Default install generated signing certificates are in use for nbi/gui/kafka
Server Certificates are generated and renewed using signing certificates for nbi/gui/
kafka

```

#### Example

In this example, custom signing certificates are in use for the northbound, GUI, and Kafka interfaces.

```

Certificates will be exported to /root/userdata/certificates
Default install generated signing certificates are in use for generating node
certificates
Custom signing certificates are in use for nbi/gui/kafka

```

Server Certificates are generated and renewed using signing certificates for nbi/gui/kafka

**Step 2.** View the exported certificate files.

### Example

```
[root@fss-deployer ~]# ls -ltr /root/userdata/certificates/
total 28
-r----- 1 root root 1675 Dec 13 03:59 current-nodesigning__rootCA.key
-r----- 1 root root 1269 Dec 13 03:59 current-nodesigning__rootCA.pem
-r----- 1 root root 1679 Dec 13 03:59 current-nbi__tls.key
-r----- 1 root root 1501 Dec 13 03:59 current-nbi__tls.crt
-r----- 1 root root 1874 Dec 13 03:59 current-nbi__ca.crt
-r----- 1 root root 3272 Dec 13 03:59 current-nbisigning__tls.key
-r----- 1 root root 1874 Dec 13 03:59 current-nbisigning__tls.crt
```

The following files were exported to the /root/userdata/certificates directory.

- `current-nodesigning__rootCA.key`: the signing certificate used by Cert-Manager to sign and generate certificates for managed nodes.
- `current-nodesigning__rootCA.pem`: the self-signed root certificate for managed nodes.
- `current-nbi__tls.crt`: the signing certificate used by Cert-Manager to sign and generate other certificates.
- `current-nbi__tls.key`: the private key.
- `current-nbi__ca.crt`: the signed certificate.
- `current-nbisigning__tls.key`: the private key for the signing certificate.
- `current-nbisigning__tls.crt`: the signed certificate.



**Note:** `current-nbi__ca.crt` and `current-nbisigning__tls.crt` are the same files.

**Step 3.** Copy the contents of the `current-nodesigning__rootCA.key` and `current-nodesigning__rootCA.pem` files to a directory in the intended standby system.

### Example

The content of the `current-nodesigning__rootCA.key` file resembles the example below. Copy the entire contents shown to a file in the intended standby.

```
-----BEGIN RSA PRIVATE KEY-----
MIIJKAIBAACAQEAuMg5L2oizpf+g77atvmtuv6Y4xBok27DbUDLYMBgkmy8Lj2
uoLLD+WGLEODCrPcn+88IMG+xiHyoumu0vqMVf2UxJZD8K0AHrhRv6uDPXPr+D1e
SHj3MfntkQEcCHH0Bakk7sc0FhgvgvWNJWRXz+g/QI24BAhJx/lvEDtwrwnLg4Sg
ydTjd2D+a+XtcxoMvyWGxQdkqse/qVY1zibzBtmQKJ+3dXj0c6UHVvyrxP5fgWn2
ebw1hxG6rQdJ7HkFpwH3p/rYUHjrGXSxhgm7YEPNLXuuhxzW+maFxZ3VpyHwL/LE
vrGzMhTsBXogm+Jj0fZdbiGF4khJwNp60aUhgHM37rabWCzMxki8uNR1pXkFdgHf
b9Ph5e0bfTix8L+keUmCSyfQdp404eKEsMmc3JFruH6oJU/9bdNESyHTZ2eK+F4g
+roe2Fu9TB1p64QUUtQv8k2s77qFiuqvaRL1hDNV4sNuIeNmKcu1n8dU+vRiGL2T
z95xqGyJYNx6SeNC/WCLBodyVAjPAayFRTB5y5K28x81Ip00zjz7+XdFFSV8am0a
```

**Step 4.** From the deployer on the intended standby system, deploy the required certificate files.

**Example**

In the standby system, assume that you copied the key to the `standby-nodesigning__rootCA.key` file and the root CA to the `standby-nodesigning__rootCA.pem` file. Enter the following command to deploy the certificates:

```
[root@fss-standby-deployer ~]# /root/bin/fss-certificate.sh deploy-node-ca-certs --
certificate
    /root/directory/standby-nodesigning__rootCA.key --pem
    /root/directory/standby-nodesigning__rootCA.pem
```



**Note:** If the deployer is already configured with the node signing certificates, use the `--force` option.

**Step 5.** Verify that the deployed certificates in the intended standby system are correct.

**Example**

```
[root@fss-standby-deployer ~]# /root/bin/fss-certificate.sh export -d directory
Certificates will be exported to /root/userdata/directory
Custom signing certificates are in use for generating node certificates
Custom server certificates are in use for nbi/gui/kafka
```

**Step 6.** Repeat steps 3 and 4 as needed for other certificates.

In step 1, if the output shows that custom signing certificates are in use for northbound, UI, and Kafka interfaces, copy the needed files to the intended standby and deploy them.

**What to do next**

[Configuring geo-redundancy](#)

**Related topics**

[Verifying that the setup is ready for geo-redundancy using the deployer VMs](#)

[Certificate management](#)

**16.9.3.4 Geo-redundancy parameters**

Table 59: Parameter descriptions

Parameter	Description	Value
Local parameters: configures the active system		
<b>Name</b>	Specifies the name of the local site. The local site is assigned the active role.	String
<b>URL</b>	Specifies the URL of the local system.	—
<b>User and Password</b>	Specifies the credentials that the remote system uses to log in to this local system. You can only configure geo-redundancy using the geored user account and you must provide the default geored password. You can change the password for the geored user as needed.	String

Parameter	Description	Value
<b>Active</b>	Specifies whether the local cluster is active.	Enable on the active cluster
<b>Verify Remote CA</b>	Checks whether the certificates on the standby cluster are valid. If enabled, enter the Root CA for the standby site.	—
Remote parameters: configures the standby system		
<b>Name</b>	Specifies the name of the remote site; the remote is the standby site.	String
<b>URL</b>	Specifies the URL of the remote system.	IP notation
<b>User and Password</b>	Specifies what credentials to use to log in to the remote system. You can only configure geo-redundancy using the geored user account. You can change the password for the geored user as needed.	String
<b>Active</b>	Specifies that the remote cluster is the standby; must be disabled for the standby.	—
<b>Verify Remote CA</b>	Checks whether the certificates on the active cluster are valid. If enabled, enter the Root CA for the active cluster.	
Sync queue length		
<b>Sync Queue Length</b>	Specifies the number of messages that can be buffered in the queue.   <b>Note:</b> Available only over the API.	Integer Default: 25000

### 16.9.3.5 Configuring geo-redundancy

#### Prerequisites

- Perform this procedure during a maintenance window.
- The intended active and standby systems must be running the same Fabric Services System software version.
- The intended active and standby systems must be reachable.
- The standby system should not be running Digital Sandbox workloads.
- Update the password for the geored user. For instructions, see [Resetting internal passwords](#).
- Be prepared to provide the following information for the active and standby Fabric Services System instances:
  - names for the active and standby systems
  - the URLs for the active and standby systems

- the password for the geored user

### About this task

Configure geo-redundancy locally from the intended active system. In the **Geo-Redundancy Configuration** form, the **Local Sites** section configures the active system; the **Remote Sites** section configures the standby system.

### Procedure

**Step 1.** Realign the certificates between the intended active and standby systems.

For instructions, see [Realigning certificates](#).

**Step 2.** From the main menu  of the intended active system, select **Geo-Redundancy**.

a. Click **CONFIGURE**.

In the **LOCAL SITE** section, configure the following settings for the active system.

- Provide a name for the active system.
- Provide the URL of the active system.
- Provide the login credentials (username and password) for the active system.
- Ensure that the **Active** parameter is enabled.

b. In the **Remote Site** section, configure the standby system.

- Provide a name for the standby system.
- Provide the URL of the standby system.
- Provide the login credentials (username and password) for the standby system.
- Ensure that the **Active** parameter is disabled.



**Note:** If a remote site is already configured as a standby for another system, this new configuration takes precedence and overwrites the previous configuration.

c. Test the connection between the endpoints of the geo-redundant system.

Click **TEST CONNECTION**.

d. Optional: Update the setting for the sync queue length.

e. Click **SAVE**.



**WARNING:** This action restarts multiple services on both the active and standby sites; it can take 5 to 10 minutes before all services are up and running again. In this time, the API and UI may behave unpredictably. Monitor the services in the **Geo-Redundancy** page and ensure that all services are up before proceeding.

### Expected outcome

The systems should automatically come up in the Syncing state. If the systems do not come up in the Syncing state, from the **Geo-Redundancy** page of the active cluster, click **SyncStart** to initiate the sync connection.

The **Geo-Redundancy** page displays:

- the names of the local and remote sites
- the role of each site, either active or standby

No services are displayed at this point.

**WARNING:**

Before proceeding to the next step, ensure that there are no pending workload jobs, deployments, or any operations that could potentially modify the database in the background.

**Step 3.** Reconcile data from the active to the standby.

From the active system **Geo-Redundancy** page, click select **Reconcile**.

**Expected outcome**

The Sync Status shows Reconcile, then moves to Syncing.

This action replaces the data collection set in the standby system with the data collection set from the active system.

**Expected outcome**

The **Geo-Redundancy** page displays:

- the names and roles (active or standby) of the clusters in the geo-redundant system  
The status of the active cluster is Active syncing; the status of the standby cluster is Standby syncing.
- the Fabric Services System services and the status of each service



**Note:** On the standby site, the UI loads with the **Geo-Redundancy** page as its home page. Only the **Alarms** and **Geo-Redundancy** options are available from the main menu. The API is disabled, except for **GET** calls for alarms and geo-redundancy.

**Related topics**

[Verifying that the setup is ready for geo-redundancy using the deployer VMs](#)  
[Geo-redundancy parameters](#)

## 16.9.4 Sync failure and recovery

### Sync failure

If the sync connection fails between the active and standby systems, the standby attempts to sync with the active instance for 90 seconds. The connection between the active instance and standby instance is considered lost after three missed heartbeats (approximately 90 seconds).

### Recovery

In a geo-redundant system, if the sync connection fails, the active cluster becomes read-only (the standby is always ready-only). An administrator can decide the next steps:

- attempt to recover from a temporary sync failure  
For instructions, see [Recovering from sync failure](#).
- initiate the switchover of roles between the two clusters, if the system with the active role is still reachable  
For instructions, see [Initiating failover: switching between the active and standby clusters](#).
- initiate standalone operation on the standby  
For instructions, see [Making the standby system active - standalone operation](#).

### 16.9.4.1 Recovering from sync failure

#### About this task

If the sync failure is because of a temporary glitch, and both active and standby clusters are still available, use the following procedure to recover the sync connection and initiate a force start on the active cluster.

#### Procedure

**Step 1.** From the main menu  of the active system, select **Geo-Redundancy**.

**Step 2.** On the upper-right, click **Start Syncing**.

#### Expected outcome



#### WARNING:

Before proceeding to the next step, ensure that there are no pending workload jobs, deployments, or any operations that could potentially modify the database in the background.

**Step 3.** Click **Reconcile**.

### 16.9.4.2 Initiating failover: switching between the active and standby clusters

#### Prerequisites

- Do not make configuration changes until the system is stable and geo-redundancy has been restored. If you must make a configuration change, use only the active site, not the standby.
- If both clusters are still active, ensure that both clusters are in maintenance mode. Do not restart SR Linux nodes.

#### About this task

In a geo-redundant system, assume cluster 1 is active and cluster 2 is standby. Use this procedure to switch the roles assigned to the cluster 1 and cluster 2 for the following scenarios:

- Maintenance: if you need to perform maintenance activities on the active system



**Note:** You need to disable the sync connection first, as shown in step 1 of this procedure.

- Disaster recovery: if cluster 1 becomes unreachable and you want to make cluster 2 the active cluster

#### Procedure

**Step 1.** If you are performing maintenance on the active cluster, stop the sync connection from the active cluster. Skip this step if you are switching over for disaster recovery reasons.

On the **Geo-Redundancy** page of the active system, click **Stop Sync**.

#### Expected outcome

The standby site automatically enters the Sync Aborted state when the active cluster has stopped syncing. Wait until the status of the standby system is Sync Aborted, then continue to step 2.

**Step 2.** From the **Geo-Redundancy** page on the standby system, run an audit.

From the standby **Audit** drop-down menu, click **AuditStart**.

#### Expected outcome

The status of the audit is shown on the **Geo-Redundancy** page.



**Note:** The auth pod can go into a crash loop for a few restarts.

When the audit completes, continue with the next step.

- Step 3.** From the upper-right drop-down list, select **ForceStandalone**.
- Step 4.** Optional: If you are performing this procedure for maintenance, click **CONFIGURE**; otherwise, skip this step.
  - a.** In the **Local Site** section, enable the **Active** field.
  - b.** In the **Remote Site** section, disable the **Active** field.
- Step 5.** Optional: If you are performing this procedure for maintenance, click **SAVE**; otherwise, skip this step.
- Step 6.** Upload software images to the standby system.
  - a.** Upload SR Linux images to the standby system.  
For instructions, see [Adding a new software image](#).
  - b.** After you have uploaded the software images to the standby system, from the **Image Catalog** page, click **REGEN PROVISION SCRIPTS**.  
This action re-initiates ZTP provisioning scripts in case you a need to bootstrap or upgrade SR Linux nodes.
- Step 7.** Align the deployer configuration between the active and standby systems.
  - a.** Determine the name of the active and standby sites.  
**[root@fss-deployer ~]# /root/bin/fss-install.sh status-georedundancy**
  - b.** While logged in to the deployer VM in cluster 1, enter the following command to make the deployer in cluster 2 the primary deployer.

```
[root@fss-deployer ~]# /root/bin/fss-install.sh set-active-deployer -t <name>
```



**Note:** If both deployer VMs are still available, repeat this step on the deployer VM in cluster 2.

## 16.9.5 Converting a geo-redundant system to a standalone system

### Prerequisites

Do not make configuration changes until the system is stable and geo-redundancy has been restored.

### About this task

Use this procedure to disable the geo-redundancy configuration and decommission a standby cluster. This procedure deletes the geo-redundancy configuration from both active and standby systems.

## Procedure

- Step 1.** From the main menu  of the active system, select **Geo-Redundancy**.
- Step 2.** On the upper right of the page, click **Stop Syncing**.  
You do not need to wait for the standby to be in Sync Aborted state to continue with the next step.
- Step 3.** On the active site, click **Force Standalone** on the upper right of the page.



**Note:** Do not change the standby site to operate in standalone mode; the standby site remains in read-only state.

- Step 4.** In the **Geo-redundancy** configuration page, click **Delete Configuration**.

## 16.9.6 Geo-redundancy status and statistics

After geo-redundancy has been configured, the **Geo-Redundancy** page displays the following information:

- the names and status of the active and standby systems
- the Fabric Services System services, including the following information for each service:
  - the sync status: the sync connection is UP or DOWN
  - if an audit or reconcile operation is initiated, the status of the operation

### Detailed audit and reconcile statistics

You can display detailed audit statistics or the JSON view of reconcile statistics by clicking the menu button at the end of each service.

### REST API status check

The Fabric Services System provides the following REST API that you can use to display health of each cluster in a geo-redundant system including details for each of the applications:

<https://fss.domain.tld/syncmgr/api/{version}/status>.

Following is sample output:

```
{
  "local": "cl3",
  "restStatus": "READ_WRITE",
  "remotes": [
    {
      "appStatus": {
        "alarmmgr": {
          "appStatus": "UP",
          "syncStatus": "OK",
          "reconcileStatus": "RECONCILE_NONE"
        },
        "auth": {
          "appStatus": "UP",
          "syncStatus": "OK",
          "reconcileStatus": "RECONCILE_NONE"
        },
        "catalog": {
          "appStatus": "UP",
```

```

        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "connect": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "imagemgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "intentmgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "inventory": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "labelmgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "mgmtstack": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "opertopomgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "topomgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    },
    "workloadmgr": {
        "appStatus": "UP",
        "syncStatus": "OK",
        "reconcileStatus": "RECONCILE_NONE"
    }
},
"lastHealthCheckTime": "2024-04-25T06:10:49.896879306Z",
"site": "cl4",
"status": "ACTIVE_SYNCING",
"configRole": "SITE_ROLE_ACTIVE"
}
]
}

```

For more information about the <https://fss.domain.tld/syncmgr/api/{version}/status> resource, see the Swagger UI. For information more information about how to use Swagger, see the *Fabric Services System API Integration Guide*.

## 16.10 The technical support script

If the Fabric Services System application suffers a failure that you would like to report to Nokia technical support, you can use the technical support script that is bundled with the system to generate information about the Fabric Services System application status. You can then share this information with Nokia support teams to assist in troubleshooting the problem.

The technical support script is included with the Fabric Services System, provided you deployed the system using the standard deployer virtual machine (VM).

You can execute the script from the installer VM or from any Kubernetes (K8s) master node (that is, the compute node that is marked as master explicitly, or the first node in the input configuration file). For details about the installer VM and the nodes participating in the Fabric Services System cluster, see the *Fabric Services System Software Installation Guide*.

The script creates a file containing status information about the Fabric Services System microservices and CPU statistics that can be useful when troubleshooting issues.

### 16.10.1 How to run the technical support script

The technical support script can be found on the installer VM or any K8s master node, and is named one of the following:

- on the installer VM: `deployer-tech-support.sh`  
Running the script on the installer VM is particularly useful if an error arises during installation of the Fabric Services System.
- on the K8s master node: `techsupport.sh`

For the current release, the script supports the following options:

**help**

displays a list of supported option and their descriptions

**minimal**

generates a `.tar` file containing basic system status output.

**complete**

generates a `.tar` file containing comprehensive system status output.

**db mongo**

generates a folder containing a backup of the mongo database.

**filter-criteria**

provides a sub-string to select pod names (for example, entering "fss-da" causes the script to gather data only for those pods whose name includes "fss-da").

**encryption**

encrypts the file so that the information is secure until accessed by Nokia technical support using its key.

For example, when running the script from the k8s master node:

```
bash techsupport.sh minimal
```

```
bash techsupport.sh db mongo
```

The script also creates a folder named `debug-dump`, into which it places each of the files generated by the data collection process.

When the script is complete, it generates a `.tar` file in the `debug-dump` folder whose name is the current date and time. For example: `Mon-Jul-26-18-08-31-IST-2021.tar.gz`.

You can optionally encrypt the file so that only the Nokia technical support team can unlock it with their key. For example:

```
bash tech-support.sh encrypt Mon-Jul-26-18-08-31-IST-2021.tar.gz
```

This generates a file named `Mon-Jul-26-18-08-31-IST-2021.tar.gz.enc`, which you can then share with the Nokia support team.

## 16.11 Node discovery

For the Fabric Services System to serve its function of building and managing a data-center fabric, it must be able to discover, communicate with, and configure any SR Linux nodes that are intended to be members of that fabric. The system accomplishes this using a variation of the standard SR Linux Zero-touch Provisioning (ZTP) process.

As described in [Fabric intents](#), node discovery is a key prerequisite before the system can deploy a fabric intent. Each node you include within a fabric intent must first be discovered and have reported itself to be in a Ready state before the system can deploy the additional configuration details to make the node adopt its assigned role in the fabric. This section describes how node discovery occurs, and how the node's Ready state is achieved.

This section summarizes the SR Linux ZTP process and highlights the elements that are new for the Fabric Services System. For additional details about the SR Linux ZTP process, see the *Nokia Service Router Linux Software Installation Guide*.

### 16.11.1 Participants

Several entities participate in the ZTP process when used in conjunction with the Fabric Services System.

- The Fabric Services System: this process assumes that the system is up and running and able to communicate across the network. Based on a user's input when designing a fabric intent, the system creates a set of configuration files that are used to configure each node so that it can perform its intended role in the fabric.  
Two subsystems integrated within the Fabric Services System also participate in the ZTP process:
  - a DHCP server: the system's integrated DHCP server is capable of receiving a newly booted SR Linux node's DHCP request and replying with an assigned IPv4 address, provided this SR Linux serial number is assigned to a fabric.  
You can use an external DHCP server if you prefer.
  - an HTTP file server: the system's integrated HTTP file server stages and transfers SR Linux image files, scripts, and other node configuration files after communication with an SR Linux node is established.  
You can use an external HTTP file server if you prefer.
- The SR Linux node: when first booted up, an SR Linux node continuously sends DHCP discovery messages requesting an IP address. After it receives an address and subsequent data from a DHCP server, ZTP software built into the Fabric Services System can process this information and allows the

node to configure itself in accordance with information sent by a management system (in this case, the Fabric Services System).

## 16.11.2 The discovery and configuration process

The following sequence describes the series of events from the initial fabric intent design, to the communication with a newly installed SR Linux node, to the configuration of that node so it can function in its assigned role in the planned fabric.

Steps 2 through 7 in this process represent the standard SR Linux ZTP process.

Step 8 configures support for ZTP with the Fabric Services System. This step specifies the system server, its role as the fabric manager, and the FTP and HTTPS server's IP address and authentication data. These settings allows the node to communicate directly with the system and to be configured as required to participate in the planned fabric design.

1. In the Fabric Services System, an operator:
  - a. adds any required SR Linux image intended for use on a fabric's nodes to the file server
  - b. creates a fabric, specifying an SR Linux image
  - c. associates individual nodes in the fabric intent with planned hardware, based on the hardware serial number

The system waits for the required nodes to communicate with the DHCP server.

2. A newly-booted SR Linux node comes online and sends out IPv4 DHCP discovery requests. For IPv6 DHCP soliciting, a hard reboot of SR Linux is required. In addition, the Fabric Services System expects a route advertisement (RA) to be available on the network.
3. The system's internal DHCP server receives the request, replies with the assigned IP address, and points the SR Linux node to a ZTP configuration file (*provisioning.py*) on the integrated file server.
4. The node requests the ZTP configuration file from the system's file server and receives the file in reply. The configuration file identifies a particular SR Linux image for the node to use, and an associated MD5 file.
5. The node requests the MD5 file from the system's file server and receives the file in reply.
6. The node requests the binary file containing the correct SR Linux image from the file server, and receives the file in reply.
7. The node compares the received SR Linux image version to its current SR Linux version. If necessary, the node upgrades or downgrades itself to the new version of SR Linux contained in the image file.
8. The node requests from the file server the *initconfig.json* file identified in the *provisioning.py* file, and receives the file in reply. This file identifies the Fabric Services System as the fabric manager and includes the IP address and authentication information that the node can use to communicate with the system.
9. The node "phones home," telling the Fabric Services System server that the node is configured and ready for management.
10. Internally, the system updates the node's status to Ready.
11. After all nodes are in the Ready state, the operator must:
  - a. add the fabric to region's deployment pipeline

b. deploy the fabric from the deployment pipeline

The system deploys additional configuration data to the node, so that the node can configure itself for participation in the fabric as intended by the fabric design from step 1.

### 16.11.3 Using an external DHCP server

#### About this task

You can use an external DHCP server instead of the one integrated into the Fabric Services System to discover and configure the nodes to participate in a fabric intent.



**Note:** When using an external DHCP server, set the region's **Use internal DHCP** property to Disabled.

For unmanaged fabrics, if a deployment is using an internal (for maintenance) and external DHCP servers at the same time, when the system is performing a maintenance intent on particular node, ensure that the internal DHCP is the same as external and disable the external server.

In this case, the process to configure an SR Linux node follows the usual Zero Touch Provisioning process documented in the *Nokia Service Router Linux Software Installation Guide*. However, the configuration that your DHCP server loads onto the new nodes must be the initial configuration that was generated by the Fabric Services System when you designed the fabric intent that contains it.

You use these node configurations as part of the node discovery and configuration process conducted using your DHCP server and otherwise following the Zero Touch Provisioning process documented in the *Nokia Service Router Linux Software Installation Guide*.

You also need to configure the nodes so that when they boot up, they "call home" to the Fabric Services System for subsequent management. For guidance on configuring nodes in this way when using an external DHCP server, please contact Nokia technical support.

To obtain the initial configuration for all of the nodes participating in a fabric intent, do the following:

#### Procedure

**Step 1.** Open the fabric intent.

**Step 2.** From the opened fabric intent, click the  menu at the upper right of the page.

**Step 3.** Select **Download Initial Node Configuration** from the list.

#### Expected outcome

The system immediately downloads the file "initialNodeConfigs" to your Downloads folder.

**Step 4.** To view the configuration, open the initialNodeConfigs file in a text editor.

#### Related topics

[Viewing a fabric intent](#)

# 17 Security

This section describes security features of the Fabric Services System such as user authentication and audit trail logging, and password control for internal services.

## 17.1 Platform password management

The Fabric Services System uses several internal passwords to communicate between its internal services. These passwords are set securely by default. From a security standpoint, Nokia recommends that you update these passwords after installation of the platform. You can change the internal passwords for the following services and their users:

*Table 60: Internal services and users*

Service	Users
MongoDB	root fsp_user
Neo4J	root
Keycloak	master fss ztp
Postgresq1	root keycloak
Kafka	—

### 17.1.1 Changing passwords for internal services

#### Prerequisites

- Perform this procedure only during a maintenance window. Changing the password for an internal service causes the service to be unavailable for some time.
- If you are only changing the password for a non-root or non-master user, you must provide the password for the root or master user for the service in the `sample-password-values.json` file.

#### About this task

Use the `fss-change-passwords.sh` command on the deployer VM to change the application passwords. The command then changes the internal passwords for the services and restarts any affected service.

The following is usage information for the `fss-change-passwords.sh` command:

```
# /root/bin/fss-change-passwords.sh -h

Usage: /root/bin/fss-change-passwords.sh configure <passwords-json-file> - Reads passwords
      json file and configures new passwords.
      /root/bin/fss-change-passwords.sh [help | -h] - Prints usage
```

Passwords have the following requirements:

- Passwords can consists of the following characters:
  - alphabetical: a - z, A - Z
  - numerical: 0 - 9
  - special characters: @#\$\$%^&\*()\_+ -= [] {} | .



**Note:** & is not supported for any of the keycloak user passwords.

- Passwords must consist of at least:
  - eight characters
  - two upper-case characters
  - two lower-case characters
  - one numerical character
  - one special character: @#\$\$%^&\*()\_+ -= [] {} | .



**Note:** Passwords must be provided in clear text. Ensure that the configuration file is secure; do not leave it unprotected.

## Procedure

**Step 1.** Create a JSON password configuration file.

In the JSON password configuration file, for each application, provide the supported users and the current and new password for each user.

### Example

The following example shows the contents of the `/root/sample-password-values.json` configuration file that is present on the deployer VM:

```
[root@fss-deployer ~]# cat /root/sample-password-values.json
{
  "fss": {
    "passwords": {
      "mongodb": {
        "fsp_user": {
          "current": "cleartext",
          "new": "cleartext_new"
        },
        "root": {
          "current": "cleartext",
          "new": "cleartext_new"
        }
      }
    },
    "neo4j": {
```

```

    "root": {
      "current": "cleartext",
      "new": "cleartext_new"
    }
  },
  "keycloak": {
    "master": {
      "current": "cleartext",
      "new": "cleartext_new"
    },
    "fss": {
      "current": "cleartext",
      "new": "cleartext_new_complexpass"
    }
  },
  "postgresql": {
    "root": {
      "current": "cleartext",
      "new": "cleartext_new"
    },
    "keycloak": {
      "current": "cleartext",
      "new": "cleartext_new"
    },
    "ztp": {
      "current": "cleartext",
      "new": "cleartext_new"
    }
  }
},
"kafka": {
  "passwords": {
    "current": "cleartext",
    "new": "cleartext_new"
  }
}
}

```

**Step 2.** Execute the command to change the internal passwords after updating current passwords and adding new passwords.

### Example

The following example shows a successful password change.

```

[root@fss-deployer ~]# ./bin/fss-change-passwords.sh configure password-values.json
SUCCESS kafka : fss-kafka-admin
SUCCESS mongodb : fsp_user
SUCCESS mongodb : root
SUCCESS neo4j : root
SUCCESS postgresql : root
SUCCESS postgresql : keycloak
Waiting for all the pods to come up...
All pods are Running!

```

### Example

In case of a failure, the script returns an error message. Depending on the error, the script may still continue with the non-errored password change requests. In the following example, the

password change failed for the MongoDB passwords because the current passwords does not match the configured passwords. The tool still updates the passwords for the other users.

```
[root@fss-deployer ~]# ./bin/fss-change-passwords.sh configure sample-password-values.json SUCCESS kafka : fss-kafka-admin
ERROR occured for mongodb fsp_user , please make sure you have provided correct credentials
ERROR occured for mongodb root , please make sure you have provided correct credentials
SUCCESS neo4j : root
SUCCESS keycloak : master
SUCCESS keycloak : ztp
SUCCESS keycloak : admin
SUCCESS postgresql : root
SUCCESS postgresql : keycloak
Waiting for all the pods to come up...
All pods are Running!
```

## 17.2 User password management

The system enforces a default system-wide user password policy for users. The default password policy includes password aging, password complexity rules, password history, and user lockout rules.

Password aging, password history, and complexity rules apply only to local users; they do not apply to LDAP users. Lockout policies apply to both local (including admin) and LDAP users.

The default policies are described below. An admin user can update these default policy settings as needed. The default policy also applies to the admin user.

### Password aging

By default, a user's password expires after 365 days.

### Password complexity rules

By default, passwords must consist of at least:

- eight characters
- two upper-case characters
- two lower-case characters
- one numerical character
- one special character

### Password history

By default, the system rejects the use of the last three previous passwords.

### Concurrent user sessions

By default, a user can be logged in to a maximum of three concurrent active sessions.

## Lockout policy

By default, a user is locked out after five failed attempts within specified period of time (the default is 5 minutes). The account used during those attempts is locked out for a preconfigured lockout period (the default is 10 minutes). After the lockout period, the account is unlocked without the intervention of an admin user.

An admin user can also configure users to be permanently locked out of their accounts after failing to login after a specified number of failed attempts. An admin user must unlock a permanently locked account.

### 17.2.1 Password policy parameters

Table 61: Password validity

Parameter	Description	Values
<b>Duration</b>	Specifies the how long user passwords are valid.   <b>Note:</b> Not applicable to LDAP users.	Default: 365 days

Table 62: Password complexity requirements

Parameter	Description	Values
<b>Minimum Length</b>	Specifies the minimum number of characters for the user password.   <b>Note:</b> Not applicable to LDAP users.	Default: 8
<b>Minimum Number of Uppercase Characters</b>	Specifies the minimum number of uppercase characters for the user password.   <b>Note:</b> Not applicable to LDAP users.	Default: 2
<b>Minimum Number of Lowercase Characters</b>	Specifies the minimum number of lowercase characters for the user password.   <b>Note:</b> Not applicable to LDAP users.	Default: 2
<b>Minimum Number of Numerical Characters</b>	Specifies the minimum number of numerical characters for the user password.	Default: 2

Parameter	Description	Values
	 <b>Note:</b> Not applicable to LDAP users.	
<b>Minimum Number of Symbols</b>	Specifies the minimum number of symbols for the user password.   <b>Note:</b> Not applicable to LDAP users.	Default: 2

Table 63: Lockout conditions

Parameter	Description	Values
<b>Maximum Number of Login Failures</b>	Specifies the maximum number failed log in attempts before a user is locked out.	Default: 5
<b>Maximum Number of User Sessions</b>	Specifies the maximum number of active log in session for a local user.	Default: 3
<b>Number of Old Passwords to Reject</b>	Specifies the number of unique new passwords assigned to a user before an old password can be reused.	Default: 3
<b>Permanent Lockout</b>	If enabled, specifies that a user is automatically locked out after exceeding the maximum number of log in failures. The account remains locked until an administrator unlocks it. If this parameter is enabled, the <b>Wait Increment</b> , <b>Maximum Wait Time</b> , and <b>Failure Reset Time</b> parameters do not apply.	Default: disabled
<b>Wait Increment</b>	Specifies how long a user must wait after a failed attempt before logging in again.   <b>Note:</b> This parameter is not applicable if the <b>Permanent Lockout</b> parameter is enabled.	Default: 60 seconds
<b>Maximum Wait Time</b>	Specifies how long an account remains locked out before it is automatically unlocked.	Default: 3600 seconds

Parameter	Description	Values
	 <b>Note:</b> This parameter is not applicable if the <b>Permanent Lockout</b> parameter is enabled.	
<b>Failure Reset Time</b>	Specifies how long after a failed log in attempt before the counter for the <b>Maximum Number of Login Failures</b> is reset to 0. This counter is also reset after a successful log in.   <b>Note:</b> This parameter is not applicable if the <b>Permanent Lockout</b> parameter is enabled.	Default: 30 seconds
<b>Quick Login Check</b>	Specifies how long the system checks if there are any concurrent login attempts made with an incorrect password for a user.	Default: 1000 milliseconds
<b>Quick Login Wait</b>	Specifies how long a user is locked out after being locked out from a quick login check.	Default: 60 seconds

## 17.2.2 Managing user password policies

### About this task

An admin user can update the policy for user passwords in the system default password policy **FSS-Default-Password**.

### Procedure

- Step 1.** From the main menu , select **Policies**.
- Step 2.** From the drop-down list, select **Passwords**.
- Step 3.** Double-click the default password policy, **FSS-Default-Password** to open it.
- Step 4.** In the **Password Expiry** pane, update the duration for the validity of user passwords as needed.
- Step 5.** In the **Password Requirements** pane, update password requirements as needed.
- Step 6.** In the **Lockout Conditions** pane, update the following settings as needed:
  - lockout policies, including enabling permanent lockout rules
  - maximum number of user sessions
  - password history

## 17.2.3 Managing a user profile and password

### About this task

Users can log in to the system GUI or API and view their own information from the **User: <username>** menu on the upper right of the page and selecting **User Profile**. The profile page displays the following information:

- username
- first name
- last name
- e-mail address
- assigned groups (group membership)
- assigned roles
- whether the user is local user or LDAP based user

While logged in to the UI, users can change their own passwords. The system does not keep a history of a user's password changes.

### Procedure

**Step 1.** Click the **User: <username>** drop-down list and select **Change Password**.

**Step 2.** To change your password, enter your old password, new password, and confirm the change. Password requirements are described in [Password complexity rules](#).

**Step 3.** When you are finished, click **Update**.

## 17.2.4 Resetting internal passwords

### About this task

If you need to change the password for the admin, connect, ztp, or geored user because of imminent password expiry or because the account is permanently locked after unsuccessful logins attempts, use the `fss-change-passwords.sh` utility on the deployer VM to change the password and unlock the account.

### Prerequisites

Perform this procedure only during a maintenance window. You can change the password for any of the admin, connect, ztp, or geored user accounts or all of them.

### Procedure

**Step 1.** Create a JSON password configuration file that contains the new passwords for internal users.

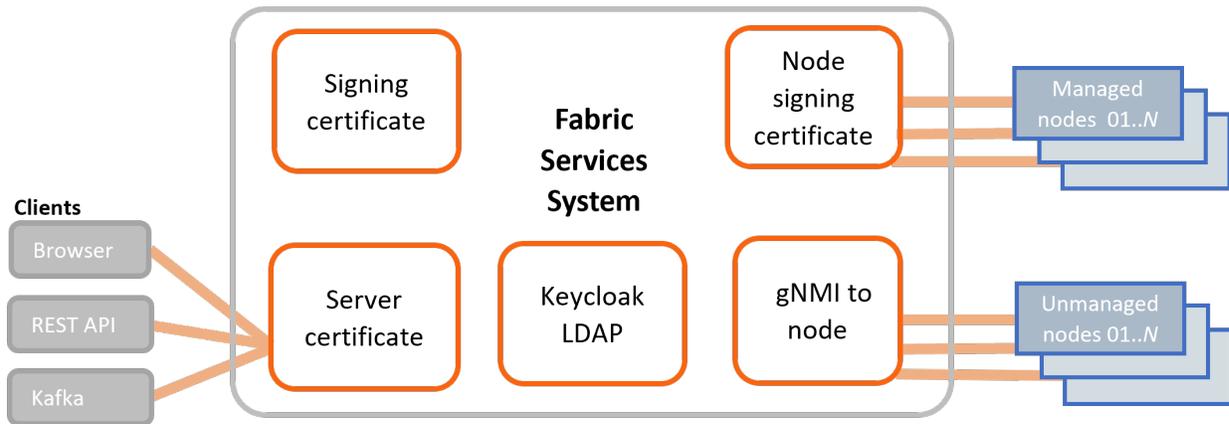
#### Example

The following example shows the contents of a `/root/sample-password-values.json` file in the deployer VM; the file contains only the new passwords for the admin, connect, geored, and ztp users:

```
{
  "fss": {
    "passwords": {
```



Figure 38: Certificates in the Fabric Services System



The certificates managed by the system include following:

- signing certificate: the issuer certificate provided to Cert-Manager and is used by Cert-Manager (an X.509 certificate controller) to sign any certificate request for the northbound interfaces certificate
- northbound server certificates: used to secure communication between the Fabric Services System and clients, such as a browser, the REST API, and Kafka
- node signing certificate: used to sign the gNMI server certificate for each fully managed SR Linux node during the bootstrap of that node. These gNMI server certificates are used to secure communication between the Fabric Services System and managed nodes
- gNMI-to-node certificates: used to secure communication between the gNMI interface of unmanaged nodes and the Fabric Services System  
These certificates are uploaded to the Fabric Services System trust store. Typically, each node has a certificate that was installed by the customer; the certificates are likely generated by the customer's internal certificate-generating service. Only the root certificate that signed all the certificates in each node is uploaded to the Fabric Services System trust store. For related information, see [Uploading a customer-generated root CA to the trust store](#).
- certificate for LDAP service - these certificates are needed for the Keycloak service to communicate with the LDAP service  
For related information, see [Configuring LDAP server details](#). During initial installation, the relevant parameters are **truststoreFilename** and **truststorePassword**, as described in "Editing the installation configuration file" in the *Fabric Services System Software Installation Guide*.

### Certificate management during initial installation

During the initial Fabric Services System installation, users do not need to provide certificates. During the installation process, the system:

- generates a self-signed root CA for signing the server certificates
- generates a self-signed root CA for signing the managed SR Linux node gNMI server certificates
- generates server certificates for all services in Kubernetes that need one

These certificates generated by the system are unique for every installation. By default, the root CAs are valid for 10 years; the northbound server certificates are valid for 365 days. The northbound server certificates are automatically renewed.

For managed nodes, the gNMI server certificates are generated with a 10 year validity and signed by the node signing certificate.

### Certificate management after the software installation

The Fabric Services system provides the `fss-certificate.sh` utility to manage certificates after the initial Fabric Services System software installation.

After initial installation, users can:

- for managed nodes, provide their own certificates to replace the certificates generated by the system during initial installation
- for the northbound interface and UI, provide their own CA (root CA or subCA) signing certificate which is used to generate the server certificates
- provide a server certificate for northbound services, including the UI, REST API, and Kafka
- provide their own CA (root CA or subCA) for signing the certificates for managed nodes
- generate a certificate signing request (CSR) for the northbound server certificate or either of the subCA certificates
- trigger the system to replace the root CA and dependent certificates that were created during installation

## 17.3.1 Managing certificates

After the initial Fabric Services System software installation, use the **fss-certificate.sh** utility to perform the following tasks:

- Deploy a user-provided CA signing certificate (root CA or subCA) for the signing of the northbound server certificates.
- Deploy a user-provided server certificate for northbound services, including the UI, REST API, and Kafka.
- Deploy a user-provided CA signing certificate (root CA or subCA) for generating the SR Linux node certificates.
- Generate a certificate signing request (CSR ) for the northbound services server certificate or either of the signing certificates.
- Generate a self-signed root CA key and CRT files and store them in the `/root/userdata/certificates` directory.
- Display the all certificates used by the system.

### 17.3.1.1 Deploying a user-provided CA certificate

#### Prerequisites

- Perform this procedure while logged in to the deployer VM.
- The customer-provided CA must be root CA or subCA.

## About this task

Use the following command to deploy a user-provided CA certificate to replace the CA certificates for internal servers that were generated during installation:

```
fss-certificate.sh deploy-fss-ca-certs --certificate <path> --key <path>
```

where:

- `--certificate <path>` is the path to certificate file, including the certificate chain and the trusted signing agency, in PEM format
- `--key <path>` is the path to the private key file, in PEM format

## Procedure

**Step 1.** Deploy a user-provided CA certificate.

### Example

```
# /root/bin/fss-certificate.sh deploy-fss-ca-certs --certificate /root/userdata/signingca-valid5years.crt --key /root/userdata/signingca-valid5years.key
Certificate is valid for 1825 days more till 2028-07-11 08:07:03
FSS updated successfully
Updating Kafka certs, this may take upto 10 minutes
SUCCESS: Certificates deployed!
```

**Step 2.** Optional: Verify the new server certificate.

You can log in to the UI and inspect the server certificate or perform the procedure [Displaying certificates](#).

## 17.3.1.2 Deploying a user-provided server certificate for northbound services

### Prerequisites

Perform this procedure while logged in to the deployer VM.

### About this task

Use the following command to deploy a user-provided certificate for northbound services (such as the UI, REST API, and Kafka) to replace the certificate that was generated during installation:

```
fss-certificate.sh deploy-fss-server-certs --certificate <path> --key <path>
```

where

`--certificate <path>` is the path to the certificate file to be used the northbound services in PEM format

`--key <path>` is the path to the private key file in PEM format

## Procedure

**Step 1.** Deploy the user-provided server certificate for northbound services.

### Example

```
# /root/bin/fss-certificate.sh deploy-fss-server-certs --certificate /root/userdata/mysrvr-valid29days.crt --key /root/userdata/mysrvr-valid29days.key
```

```
WARNING: Certificate validity is less than 30 days and expires 2023-08-10 08:07:04
FSS updated successfully
Updating Kafka server certs, this may take upto 10 minutes
```

**Step 2.** Optional: Verify the new server certificate.

You can log in to the UI and inspect the server certificate or perform the procedure [Displaying certificates](#).

### 17.3.1.3 Deploying a user-provided node CA certificate

#### Prerequisites

- Perform this procedure while logged in to the deployer VM.
- The customer-provided CA must be root CA or subCA.
- The CA must be valid for at least 10 years.

#### About this task

Use the following command to deploy the signing certificate CA is used to generate certificates for managed nodes.

```
fss-certificate.sh deploy-node-ca-certs --certificate <path> --key <path>
```

where

--certificate <path> is the path to the certificate file, in PEM format

--key <path> is the path to the private key file, in PEM format



#### Note:

Only nodes that are bootstrapped after the change of CA receive a gNMI server certificate signed by the new CA. Existing managed node gNMI server certificates are renewed or replaced with new server certificates signed by the newly provided CA.

#### Procedure

Deploy the customer-provided CA.

#### Example

```
# /root/bin/fss-certificate.sh deploy-node-ca-certs --certificate /root/userdata/nodesigningca-valid10yrs.crt --key /root/userdata/nodesigningca-valid10yrs.key
Certificate is valid for 3651 days more till 2033-07-11 08:07:05
FSS updated successfully
```

### 17.3.1.4 Generating a CSR

#### Prerequisites

Perform this procedure while logged in to the deployer VM.

#### About this task

Use this procedure to generate a CSR for the northbound interface certificate. The Fabric Services System generates a private key and creates a CSR which the user sends to a CA for signing. After the CSR is

signed, the customer provides the certificate and chain back to the system, which has stored the private key.

Use the following command:

```
fss-certificate.sh generate-csr --country <country> --province <province> --location <location>
--org <organization> --org-unit <organizational unit> --days <num of days> --input-file <path>
```

where:

- country <country> is the two-letter of the country for the certificate subject
- province <province> is the province or state (in full) for the certificate subject
- location <location> is the location name (typically city) for the certificate subject
- org <organization> is the organization or company name for the certificate subject
- org-unit <organizational unit> is the organizational unit or team for the certificate subject
- days <num of days> is the number of days the certificate is valid
- input-file <path> is the name of the input JSON configuration file

### Procedure

Generate a CSR for a user.

### Example

```
fss-certificate.sh generate-csr --country US --province CA --location Sunnyvale --org Nokia --
org-unit ION --days 3650 --input-file input.json
```

## 17.3.1.5 Generate a self-signed root CA certificate

### Prerequisites

Perform this procedure while logged in to the deployer VM.

### About this task

Use the following command to generate a self-signed root CA certificate and private key. The system generates a new self-signed root CA key and stores it the /root/userdata/certificates directory. This certificate can then be used with the other procedures as needed.

```
fss-certificate.sh create-certs --country <country> --province <province> --location <location>
--org <organization> --org-unit <organizational unit> --days <num of days> --input-file <path>
```

where:

- country <country> is the two-letter of the country for the certificate subject
- province <province> is the province or state (in full) for the certificate subject
- location <location> is the location name (typically city) for the certificate subject
- org <organization> is the organization or company name for the certificate subject
- org-unit <organizational unit> is the organizational unit or team for the certificate subject
- days <num of days> is the number of days the certificate is valid
- input-file <path> is the name of the input JSON configuration file

## Procedure

**Step 1.** Create a new root CA.

### Example

```
# fss-certificate.sh create-certs --country US --province CA --location Sunnyvale --
org Nokia --org-unit ION --days 3650 --input-file input.json
```

**Step 2.** Verify the generated certificate.

Use the following command:

```
openssl x509 -noout -text -in /root/userdata/certificates/fss-issuer.crt
```

## 17.3.1.6 Displaying certificates

### Prerequisites

Perform this procedure while logged in to the deployer VM.

### About this task

Use the following command to display certificates:

```
fss-certificate.sh show-certs [-v]
```

### Example: Displaying the summary of all certificates

```
# /root/bin/fss-certificate.sh show-certs
===== FSS GUI/REST CERTS =====
Showing certificate with connection to 192.168.102.106:443
Only showing issuer and subject
subject=C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
mysrvrvalid29days.jagi.nokia.net
issuer=C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
independentsigningcallyrs.jagi.nokia.net
===== KAFKA CERTS =====
Showing certificate with connection to 192.168.102.106:32100
Only showing issuer and subject
subject=C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
mysrvrvalid29days.jagi.nokia.net
issuer=C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
independentsigningcallyrs.jagi.nokia.net
```

### Example: Displaying details about all certificates

```
# /root/bin/fss-certificate.sh show-certs -v
===== FSS GUI/REST CERTS =====
Showing certificate with connection to 192.168.102.106:443
```

```
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      60:8d:ba:1a:d8:41:bf:38:81:c0:c9:4e:84:47:65:d9:55:4f:eb:2b
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
independentsigningcallrs.jagi.nokia.net
    Validity
      Not Before: Jul 12 08:07:04 2023 GMT
      Not After : Aug 10 08:07:04 2023 GMT
    Subject: C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =
mysrvrvalid29days.jagi.nokia.net
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public-Key: (4096 bit)
      Modulus:
        00:b7:9d:ef:af:f1:8e:89:89:58:63:f4:9c:58:74:
        44:9b:14:83:06:e7:69:e3:8e:46:b9:53:cd:95:49:
        9e:57:f0:24:d1:21:f4:77:08:4f:a2:38:6f:cd:92:
        9a:5d:d7:24:e0:8e:91:e6:6b:da:fd:28:b5:5a:c6:
        5a:fa:05:11:6e:36:66:c3:5d:67:ab:bd:04:ed:c8:
        a9:ef:80:f2:27:ba:c0:1c:0a:77:02:94:bd:79:aa:
        20:af:31:9c:53:2b:c8:c7:1b:b2:fd:8c:11:73:01:
        32:9a:e4:d3:2e:00:71:5d:58:6e:b7:82:65:34:48:
        cf:86:3f:ca:7e:6b:39:95:c4:97:42:c3:8c:9d:a6:
        1f:58:a2:fd:5d:e5:e4:9f:60:3b:60:a6:14:2c:2b:
        67:be:5c:97:4b:ce:ef:68:be:88:cb:91:f5:97:e2:
        ad:a9:a2:03:b1:e8:09:97:63:35:b3:0f:3d:ae:5d:
        49:a2:f5:2c:2c:d6:22:6b:67:33:49:ad:8f:57:70:
        0d:0b:12:69:ca:72:76:54:ab:0e:34:21:7a:ad:49:
        55:43:9d:a4:0a:fb:12:31:4b:f0:82:86:cf:40:c2:
        cc:8a:21:7f:c5:8b:35:b2:f6:84:2a:31:a1:bd:37:
        5e:9c:fc:43:99:55:8d:a3:5b:f3:13:95:2d:a7:71:
        7f:29:9f:6b:19:d0:f1:5c:14:f3:40:28:4c:f0:84:
        c9:af:94:bf:7c:26:e9:ad:dc:a9:bc:fc:c4:c6:bf:
        43:55:53:22:a8:eb:fa:3f:b0:99:37:f7:37:a6:2e:
        86:33:d2:1c:61:37:7b:89:fe:d2:9b:ba:db:9f:3e:
        2f:e4:9d:6c:e1:d4:96:a2:a5:e9:ed:b6:81:71:a3:
        94:b7:12:03:5b:c8:58:ac:de:16:01:30:eb:c4:38:
        52:d7:73:e7:04:89:dd:03:3e:a3:dd:bb:f4:6b:fc:
        75:b5:d7:82:a4:28:70:ab:b5:3e:c2:a2:22:80:dc:
        23:04:41:5b:d4:be:41:1c:78:51:05:cd:a1:55:f6:
        12:3e:03:0b:b1:22:8e:6e:31:ed:68:d3:17:ca:d9:
        29:c2:54:8f:f1:4e:08:a6:8e:65:4d:50:d9:6a:68:
        6b:ee:54:d8:fa:0f:c3:54:cd:57:b5:f9:48:cf:59:
        14:91:37:a3:3a:29:e3:d5:f8:32:04:d8:eb:50:1a:
        1d:2b:48:ca:68:ff:5d:68:17:f3:29:a0:72:01:08:
        36:fb:5d:d4:9f:fb:46:e1:66:c6:97:43:7f:cb:b0:
        7d:11:ab:24:f8:91:c3:a0:bc:3d:a7:79:05:eb:04:
        53:c5:36:84:d2:3e:8d:e8:30:9b:55:cd:89:e4:11:
        ac:e1:a7
      Exponent: 65537 (0x10001)
    X509v3 extensions:
      X509v3 Authority Key Identifier:
        keyid:4C:A4:FF:0C:07:77:99:29:35:BF:B8:3E:B4:2D:C1:81:68:50:CF:5F

      X509v3 Basic Constraints:
        CA:FALSE
      X509v3 Subject Alternative Name:
        DNS:mysrvr01.jagi.net, DNS:cutomersrvr01.customerdomain.net, DNS:localhost
    Signature Algorithm: sha256WithRSAEncryption
      54:57:03:a4:cc:f3:8c:73:a2:6f:67:67:e8:47:36:e1:8f:d6:
```

```

41:48:f4:c7:fb:26:10:83:f3:09:e0:f9:97:5f:3a:50:c6:8c:
de:46:22:54:f9:2d:58:1d:01:d6:43:e7:45:22:36:82:6d:1a:
a1:07:ed:4d:76:78:c7:5d:6c:c9:c3:96:99:b8:4b:74:cd:4f:
0e:0b:91:b4:bf:37:8c:d7:60:2e:00:32:1e:f3:c5:c1:19:51:
00:81:a2:1e:83:82:f6:cd:3f:50:c2:60:9f:51:9a:44:ff:e0:
50:ac:be:e7:1f:82:67:fd:fe:ec:8d:64:8e:70:b6:2e:ad:23:
7c:64:95:20:c7:8a:40:f4:51:5c:77:1a:78:a1:20:6d:ec:8d:
b0:2b:af:59:c6:03:54:cf:7b:2e:1b:b7:47:7f:51:0c:b5:9d:
5e:4f:52:6a:a8:5e:1a:6c:82:eb:6f:06:3c:06:37:df:98:75:
dc:64:fb:bd:cb:d0:2e:bd:b7:a7:93:db:c9:86:47:64:5f:74:
5c:90:e2:19:2d:d8:33:f0:92:52:8e:7c:9d:49:06:37:9e:7c:
3e:67:85:43:41:a1:dc:bd:cb:70:9b:03:c4:95:4a:72:d0:8c:
35:84:22:ee:2f:fc:4c:58:a6:43:71:d8:61:90:59:fc:15:fd:
a6:9d:df:b2:50:95:91:f5:35:ac:fa:1c:60:e1:f3:03:23:46:
34:96:78:9e:79:39:dc:63:fd:99:88:53:dc:d3:0b:e7:49:ae:
90:8c:89:09:c3:5b:c0:21:4d:0a:ce:81:0b:dc:4b:76:30:b9:
49:05:f6:59:03:dd:00:e2:4c:06:3d:f0:4d:d0:98:17:9c:0a:
c1:a8:e1:13:c9:0e:60:46:2a:d5:fc:d5:ad:1b:43:c1:77:8f:
3a:ae:ea:c8:f2:09:23:3a:86:27:c7:60:05:bc:ca:38:fd:60:
c8:1d:b9:8f:c2:13:47:e1:fd:45:b3:72:14:a4:ee:8a:83:6e:
12:61:65:45:40:dc:ab:97:87:e8:9d:76:29:60:94:43:e2:30:
d2:bd:74:04:34:7a:06:f6:57:80:87:d0:45:84:01:f5:3a:e3:
e2:4a:17:59:f7:0c:2a:f2:b1:9d:81:72:af:ae:0f:bb:1a:0d:
45:64:89:6f:36:bf:63:ff:bd:23:d0:62:b5:0d:6a:ac:7b:21:
f0:c3:7f:5a:03:6c:70:4f:f1:ca:27:61:b0:ea:b7:10:6f:f4:
99:e3:5c:30:29:3a:82:4b:eb:a4:6a:a0:c0:d0:c5:a2:8a:32:
12:c2:3f:fc:a9:8e:dc:ec:8a:25:23:4c:e9:f8:c3:af:f9:4a:
dc:c6:42:14:c4:0a:11:7e

```

===== KAFKA CERTS =====

Showing certificate with connection to 192.168.102.106:32100

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

60:8d:ba:1a:d8:41:bf:38:81:c0:c9:4e:84:47:65:d9:55:4f:eb:2b

Signature Algorithm: sha256WithRSAEncryption

Issuer: C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =

independentsigningcallrs.jagi.nokia.net

Validity

Not Before: Jul 12 08:07:04 2023 GMT

Not After : Aug 10 08:07:04 2023 GMT

Subject: C = IN, ST = Karnataka, L = Bengaluru, O = Nokia, CN =

mysrvrvalid29days.jagi.nokia.net

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public-Key: (4096 bit)

Modulus:

```

00:b7:9d:ef:af:f1:8e:89:89:58:63:f4:9c:58:74:
44:9b:14:83:06:e7:69:e3:8e:46:b9:53:cd:95:49:
9e:57:f0:24:d1:21:f4:77:08:4f:a2:38:6f:cd:92:
9a:5d:d7:24:e0:8e:91:e6:6b:da:fd:28:b5:5a:c6:
5a:fa:05:11:6e:36:66:c3:5d:67:ab:bd:04:ed:c8:
a9:ef:80:f2:27:ba:c0:1c:0a:77:02:94:bd:79:aa:
20:af:31:9c:53:2b:c8:c7:1b:b2:fd:8c:11:73:01:
32:9a:e4:d3:2e:00:71:5d:58:6e:b7:82:65:34:48:
cf:86:3f:ca:7e:6b:39:95:c4:97:42:c3:8c:9d:a6:
1f:58:a2:fd:5d:e5:e4:9f:60:3b:60:a6:14:2c:2b:
67:be:5c:97:4b:ce:ef:68:be:88:cb:91:f5:97:e2:
ad:a9:a2:03:b1:e8:09:97:63:35:b3:0f:3d:ae:5d:
49:a2:f5:2c:2c:d6:22:6b:67:33:49:ad:8f:57:70:

```

```

0d:0b:12:69:ca:72:76:54:ab:0e:34:21:7a:ad:49:
55:43:9d:a4:0a:fb:12:31:4b:f0:82:86:cf:40:c2:
cc:8a:21:7f:c5:8b:35:b2:f6:84:2a:31:a1:bd:37:
5e:9c:fc:43:99:55:8d:a3:5b:f3:13:95:2d:a7:71:
7f:29:9f:6b:19:d0:f1:5c:14:f3:40:28:4c:f0:84:
c9:af:94:bf:7c:26:e9:ad:dc:a9:bc:fc:c4:c6:bf:
43:55:53:22:a8:eb:fa:3f:b0:99:37:f7:37:a6:2e:
86:33:d2:1c:61:37:7b:89:fe:d2:9b:ba:db:9f:3e:
2f:e4:9d:6c:e1:d4:96:a2:a5:e9:ed:b6:81:71:a3:
94:b7:12:03:5b:c8:58:ac:de:16:01:30:eb:c4:38:
52:d7:73:e7:04:89:dd:03:3e:a3:dd:bb:f4:6b:fc:
75:b5:d7:82:a4:28:70:ab:b5:3e:c2:a2:22:80:dc:
23:04:41:5b:d4:be:41:1c:78:51:05:cd:a1:55:f6:
12:3e:03:0b:b1:22:8e:6e:31:ed:68:d3:17:ca:d9:
29:c2:54:8f:f1:4e:08:a6:8e:65:4d:50:d9:6a:68:
6b:ee:54:d8:fa:0f:c3:54:cd:57:b5:f9:48:cf:59:
14:91:37:a3:3a:29:e3:d5:f8:32:04:d8:eb:50:1a:
1d:2b:48:ca:68:ff:5d:68:17:f3:29:a0:72:01:08:
36:fb:5d:d4:9f:fb:46:e1:66:c6:97:43:7f:cb:b0:
7d:11:ab:24:f8:91:c3:a0:bc:3d:a7:79:05:eb:04:
53:c5:36:84:d2:3e:8d:e8:30:9b:55:cd:89:e4:11:
ac:e1:a7
Exponent: 65537 (0x10001)
X509v3 extensions:
  X509v3 Authority Key Identifier:
    keyid:4C:A4:FF:0C:07:77:99:29:35:BF:B8:3E:B4:2D:C1:81:68:50:CF:5F

  X509v3 Basic Constraints:
    CA:FALSE
  X509v3 Subject Alternative Name:
    DNS:mysrvr01.jagi.net, DNS:cutomersrvr01.customerdomain.net, DNS:localhost
Signature Algorithm: sha256WithRSAEncryption
54:57:03:a4:cc:f3:8c:73:a2:6f:67:67:e8:47:36:e1:8f:d6:
41:48:f4:c7:fb:26:10:83:f3:09:e0:f9:97:5f:3a:50:c6:8c:
de:46:22:54:f9:2d:58:1d:01:d6:43:e7:45:22:36:82:6d:1a:
a1:07:ed:4d:76:78:c7:5d:6c:c9:c3:96:99:b8:4b:74:cd:4f:
0e:0b:91:b4:bf:37:8c:d7:60:2e:00:32:1e:f3:c5:c1:19:51:
00:81:a2:1e:83:82:f6:cd:3f:50:c2:60:9f:51:9a:44:ff:e0:
50:ac:be:e7:1f:82:67:fd:fe:ec:8d:64:8e:70:b6:2e:ad:23:
7c:64:95:20:c7:8a:40:f4:51:5c:77:1a:78:a1:20:6d:ec:8d:
b0:2b:af:59:c6:03:54:cf:7b:2e:1b:b7:47:7f:51:0c:b5:9d:
5e:4f:52:6a:a8:5e:1a:6c:82:eb:6f:06:3c:06:37:df:98:75:
dc:64:fb:bd:cb:d0:2e:bd:b7:a7:93:db:c9:86:47:64:5f:74:
5c:90:e2:19:2d:d8:33:f0:92:52:8e:7c:9d:49:06:37:9e:7c:
3e:67:85:43:41:a1:dc:bd:cb:70:9b:03:c4:95:4a:72:d0:8c:
35:84:22:ee:2f:fc:4c:58:a6:43:71:d8:61:90:59:fc:15:fd:
a6:9d:df:b2:50:95:91:f5:35:ac:fa:1c:60:e1:f3:03:23:46:
34:96:78:9e:79:39:dc:63:fd:99:88:53:dc:d3:0b:e7:49:ae:
90:8c:89:09:c3:5b:c0:21:4d:0a:ce:81:0b:dc:4b:76:30:b9:
49:05:f6:59:03:dd:00:e2:4c:06:3d:f0:4d:d0:98:17:9c:0a:
c1:a8:e1:13:c9:0e:60:46:2a:d5:fc:d5:ad:1b:43:c1:77:8f:
3a:ae:ea:c8:f2:09:23:3a:86:27:c7:60:05:bc:ca:38:fd:60:
c8:1d:b9:8f:c2:13:47:e1:fd:45:b3:72:14:a4:ee:8a:83:6e:
12:61:65:45:40:dc:ab:97:87:e8:9d:76:29:60:94:43:e2:30:
d2:bd:74:04:34:7a:06:f6:57:80:87:d0:45:84:01:f5:3a:e3:
e2:4a:17:59:f7:0c:2a:f2:b1:9d:81:72:af:ae:0f:bb:1a:0d:
45:64:89:6f:36:bf:63:ff:bd:23:d0:62:b5:0d:6a:ac:7b:21:
f0:c3:7f:5a:03:6c:70:4f:f1:ca:27:61:b0:ea:b7:10:6f:f4:
99:e3:5c:30:29:3a:82:4b:eb:a4:6a:a0:c0:d0:c5:a2:8a:32:
12:c2:3f:fc:a9:8e:dc:ec:8a:25:23:4c:e9:f8:c3:af:f9:4a:
dc:c6:42:14:c4:0a:11:7e

```

## 17.3.2 Uploading a customer-generated root CA to the trust store

### About this task

A CA certificate is required for unmanaged nodes to communicate with the gNMI port. The certificate can be a root CA or certificate chain. The certificate must be in Privacy-Enhanced Mail (PEM) format.

### Procedure

**Step 1.** Click the main menu, then select **Settings**.

**Step 2.** From the menu on the left, click **Certificate Management**, then click **+ UPLOAD CERTIFICATE**.

**Step 3.** Find the certificates locally on your computer, click **Open**.

### What to do next

[Deploying a user-provided CA certificate](#)

## 17.4 LDAP server integration

The Fabric Services System supports the use of an optional Lightweight Directory Access Protocol (LDAP) server that the system can use to verify the authentication of users who were not created on the system.

You integrate an LDAP server by creating a Federation Provider instance on the Fabric Services System. After creating the Federation Provider instance, you can synchronize users from the LDAP server at any time.

The LDAP server is used as a read-only resource for authentication. When you integrate an LDAP server, you can continue to create users in Fabric Services System, but these newly created users are not pushed to the LDAP server when you synchronize.



### Note:

- You can only configure one instance of a Federation Provider.
- User permissions and group memberships are managed from the Fabric Services System; they are not learned from the LDAP server.
- The Fabric Services System supports up to 500 synchronized users.
- Deleting a Federation Provider instance deletes all users imported into the Fabric Services System from the configured LDAP server.

### 17.4.1 Federation Provider parameters

Table 64: General parameters

Parameter	Description	Values
Name	Specifies the name of the Federation Provider instance. This value cannot be edited after the instance has been created.	String

Parameter	Description	Values
<b>Enabled</b>	Specifies whether the Federation Provider is supported.	Default: enabled
<b>Import Users</b>	Specifies whether users are synchronized from the LDAP server.	Default: enabled
<b>Vendor</b>	Specifies the LDAP vendor type.	<ul style="list-style-type: none"> <li>• <b>Active Directory</b>: for Active Directory LDAP servers</li> <li>• <b>Other</b>: for other LDAP servers, such as OpenLDAP</li> </ul>

Table 65: LDAP server settings

Parameter	Description	Values
<b>Connection URL</b>	Specifies IP address of the LDAP server and the port on which it is running.	—
<b>Use TLS</b>	Enables the use of StartTLS when using regular LDAP (not LDAPS). This flag can only be enabled for the regular LDAP protocol, as it only applies in that case. If this parameter is enabled with LDAPS, the connection to the LDAPS server fails.	Default: disabled
<b>Bind Type</b>	Specifies how a user authenticates.	<ul style="list-style-type: none"> <li>• <b>simple</b>: a user authenticates with the values for <b>Bind DN</b> and <b>Bind Credential</b></li> <li>• <b>none</b>: use anonymous connections to LDAP</li> </ul>
<b>Bind DN</b>	Specifies the distinguished name (DN) of an LDAP admin user to connect to LDAP.	string
<b>Bind Credential</b>	Specifies the admin password.	If the <b>Bind Type</b> parameter is set to <b>simple</b> , this password is used to authenticate
<b>User DN</b>	Specifies the full DN of the LDAP tree where the users can be found in the LDAP server.	Fully qualified domain name
<b>Username LDAP attribute</b>	Specifies the name of the attribute that must be used as the username within the Fabric Services System.	Dynamically filled based on the value of the <b>Vendor</b> parameter, but is editable: <ul style="list-style-type: none"> <li>• <b>Active Directory</b>: cn</li> <li>• <b>Other</b>: uid</li> </ul> Usually the user ID, uid

Parameter	Description	Values
<b>RDN LDAP Attribute</b>	Specifies name of the LDAP attribute used for the relative distinguished name of a typical user DN.	Dynamically filled based on the value of the <b>Vendor</b> parameter, but is editable: <ul style="list-style-type: none"> <li>• <b>Active Directory:</b> cn</li> <li>• <b>Other:</b> uid</li> </ul> Usually the user ID, uid
<b>UUID LDAP Attribute</b>	Shows the name of the LDAP attribute that is used as a unique identifier for objects in LDAP.	Dynamically filled based on the value of the <b>Vendor</b> parameter, but is editable: <ul style="list-style-type: none"> <li>• <b>Active Directory:</b> objectGUID</li> <li>• <b>Other:</b> entryUUID</li> </ul> Usually the user ID, uid
<b>User Object Classes</b>	Specify a comma-separated list of user object classes used by LDAP to identify a user. Users can only be found if they have these object classes.	Dynamically filled based on the value of the <b>Vendor</b> parameter, but is editable: <ul style="list-style-type: none"> <li>• <b>Active Directory:</b> person, organizationalPerson, user</li> <li>• <b>Other:</b> inetOrgPerson, organizationalPerson</li> </ul>
<b>Custom User LDAP Filter</b>	Specify the filter to select the users that should be synchronized.	Filter string, enclosed in parentheses ( )
<b>Search Scope</b>	Specifies the type of search.	<b>One Level</b> or <b>Subtree</b>

Table 66: Advanced settings

Parameter	Description	Values
<b>Connection Timeout</b>	Specifies the LDAP server connection timeout, in milliseconds.	Default: 0
<b>Read Timeout</b>	Specifies the LDAP read timeout, in milliseconds.	Default: 0
<b>Pagination</b>	Specifies whether the Federation Provider supports pagination when fetching users.	Default: enabled
<b>Sync Batch Size</b>	Specifies the number of users to synchronize from the LDAP server in a single transaction.	Default: 1000

## 17.4.2 Integrating an LDAP server

### About this task

Use this procedure to integrate an LDAP server by configuring a Federation Provider instance on the Fabric Services System.

### Procedure

**Step 1.** From the main menu , select **User and Resource Management** → **Federation Providers**.

**Step 2.** Click **+CREATE FEDERATION PROVIDER**.

**Step 3.** Configure general settings for the Federation Provider instance.

Set the following parameters:

- **Name**
- **Enabled**
- **Import Users**
- **Vendor**

**Step 4.** Configure LDAP server settings.

Set the following parameters:

- **Connection URL**
- **Use TLS**
- **Bind Type**
- **Bind DN**
- **Bind Credential**
- **User DN**
- **Username LDAP Attribute**
- **RDN LDAP Attribute**
- **UUID LDAP Attribute**
- **User Object Classes**
- **Custom User LDAP Filter**
- **Search Settings**

**Step 5.** Configure advanced settings.

Set the following parameters:

- **Connection Timeout**
- **Read Timeout**
- **Pagination**
- **Sync Batch Size**

**Step 6.** Verify the settings.

- To verify the connection to the LDAP server, that is, the setting of the **Connection URL** parameter, click **TEST CONNECTION**.

- To verify the authentication with the LDAP server, click **TEST AUTHENTICATION**.

**Step 7.** When you are finished, click **CREATE**.

### What to do next

You can now synchronize users with the LDAP server. For instructions, see [Synchronizing with the LDAP server](#).

You can edit the settings for the Federation Provider instance or delete the Federation Provider instance entirely. For instructions, see [Managing the Federation Provider](#).

You also can perform the following procedures as needed:

- [Viewing a list of existing users](#)
- [Assigning a user to a user group](#)
- [Assigning a role to a user](#)

### Related topics

[Federation Provider parameters](#)

## 17.4.3 Synchronizing with the LDAP server

### About this task

You can synchronize with the LDAP server at any time after you have created a Federation provider instance.

### Procedure

- Step 1.** From the **Federation Providers** view, locate the Federation Provider instance and click  at the end of its row.
- Step 2.** Synchronize with the LDAP server.
- **Full Sync** — imports all changes from the LDAP server
  - **Changed User Sync** — imports only the changes in the LDAP server made after you last synchronized

## 17.4.4 Managing the Federation Provider

### About this task

You can edit the Federation Provider settings or delete the instance entirely.

### Procedure

- Step 1.** From the main menu , select **User and Resource Management** → **Federation Providers**.
- Step 2.** Locate the Federation Provider instance and right click  at the edge of its row.
- Step 3.** You can edit the settings for the Federation provider or delete the instance.
- Select **Details**, then update parameters as needed.
  - Select **Delete** to delete the Federation Provider.



**Note:** Deleting a Federation Provider deletes users imported from the LDAP server.

## 17.4.5 Configuring LDAP server details

### About this task

Use this procedure to enable LDAP capabilities after the Fabric Services System application has been installed or upgraded. You can also use this procedure to update or delete LDAP configuration.

### Procedure

**Step 1.** Update the `sample-input.json` file with the LDAP settings.

In the `fss` section of the `sample-input.json` file, set the following parameters:

- **truststoreFilename:** the location of the truststore filename with the absolute path information. The JKS file must be generated to access the LDAP server from the Fabric Services System instance. The alternate names in the certificate should match the name and IP address configured for the federation provider (using the Fabric Services System UI or REST API).
- **truststorePassword:** the password used to access the truststore

**Step 2.** Update the system configuration.

Execute the following commands:

```
# fss-install.sh configure <input.json>
# fss-upgrade.sh upgrade
```

### What to do next

Update federation provider settings as needed.

## 17.5 Forwarding user audit logs to a remote server

Audit logs contain a recording of all user activities. You can configure the Fabric Services System to send audit logs to a remote syslog server. The system uses the Fluent Bit utility to collect audit logs from the Digital Sandbox and Fabric Services System and forward the audit logs to a remote syslog server.

The settings for the remote syslog server are part of the `sample-input.json` file. You can configure a remote syslog server:

- during the Fabric Services System application installation; for more information, see “Editing the installation configuration file” in the *Fabric Services System Software Installation Guide*
- after the Fabric Services System application has been installed, as described in [Configuring a remote syslog server for user audit logs](#)



**Note:** The system currently supports one remote syslog server.

## 17.5.1 Configuring a remote syslog server for user audit logs

### About this task

Use this procedure to configure the remote syslog service on a remote server after the Fabric Services System application has been installed.

### Procedure

**Step 1.** Update the `sample-input.json` file.

In the `rsyslog` section, set the following parameters:

- **host** — the IP address or FQDN of the remote syslog server
- **port** — the port that the rsyslog utility uses for network connectivity
- **proto** — the protocol used for syslog traffic, either TCP or UDP

### Example

```
"fss": {
  ...
},
"rsyslog": {
  "host": "192.0.2.149",
  "port": 514,
  "proto": "udp"
},
```

**Step 2.** Run the `fss-fluent.sh` script to update the system configuration.

The `fss-fluent.sh` file is available in the `/root/bin` directory.

### Example

```
[root@fss-deployer ~]# ./bin/fss-fluentbit.sh install updated-input-kvm-fss-
deployer.json
source /var/lib/fss/config/fssEnv.sh
input_file: updated-input-kvm-fss-deployer.json
fss-logs      default      1      2023-03-14 06:24:22.306303226 +0000
  UTC deployed      fluent-bit-0.20.9      1.9.9
fss-logs chart already installed, SKIP_K8S=false
Do you want to restart fss-logs with new values : Are you sure [YyNn]? y
rsyslog_host: 192.0.2.149
rsyslog_port: 51400
rsyslog_proto: udp
Creating fss-logs-pvc
persistentvolumeclaim/fss-logs-pvc unchanged

Starting fss-logs
helm upgrade -i fss-logs /var/lib/rancher/k3s/storage/pvc-repo/download/charts/fluent-
bit-0.20.9.tgz -f /var/lib/fss/config/fluentbit-values.yaml
Release "fss-logs" has been upgraded. Happy Helming!
NAME: fss-logs
LAST DEPLOYED: Tue Mar 14 06:35:51 2023
NAMESPACE: default
STATUS: deployed
REVISION: 2
NOTES:
Get Fluent Bit build information by running these commands:
```

```
export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/
name=fluent-bit,app.kubernetes.io/instance=fss-logs" -o jsonpath=
"{.items[0].metadata.name}")
kubectl --namespace default port-forward $POD_NAME 2020:2020
curl http://127.0.0.1:2020
```

**Step 3.** Check the remote syslog server to ensure that logs are being forwarded.

## Appendix A: Supported alarms

### Equipment alarms

Among equipment alarms, there are several groups of transceiver-related environment alarms with escalating thresholds:

- low warning: a Warning alarm indicating that a minor low threshold has been crossed.
- low alarm: a Critical alarm indicating that a major low threshold has been crossed.
- high warning: a Warning alarm indicating that a minor high threshold has been crossed.
- high alarm: a Critical alarm indicating that a major high threshold has been crossed.

*Table 67: 1001 Fan tray fault*

Alarm ID:	1001
Alarm name:	Fan Tray Fault
Description:	This alarm is raised when the associated fan-tray is in down/empty/failed/degraded/low-power operational state. The system may have cooling issues.
Severity:	Major
Probable cause:	Equipment malfunction
Remedial action:	The failed fan unit should be replaced.

*Table 68: 1003 Power supply fault*

Alarm ID:	1003
Alarm name:	Power Supply Fault
Description:	The alarm is raised when the associated power supply is not operationally Up. The specified power supply can no longer supply power to the system.
Severity:	Critical
Probable cause:	Power problem
Remedial action:	Check the status of the power supply.

*Table 69: 1004 Chassis fault*

Alarm ID:	1004
Alarm name:	Chassis Fault
Description:	The alarm is raised when chassis is operationally down.
Severity:	Critical
Probable cause:	Equipment malfunction
Remedial action:	Chassis Down

*Table 70: 1005 CPM fault*

Alarm ID:	1005
Alarm name:	CPM Fault
Description:	This alarm is generated when the control module is in an operationally down/empty/failed/degraded/low-power state.
Severity:	Critical
Probable cause:	Equipment malfunction
Remedial action:	Remove the card and reset it. If this does not clear the alarm then please contact your Nokia support representative for assistance.

*Table 71: 1006 SFM fault*

Alarm ID:	1006
Alarm name:	SFM Fault
Description:	The alarm is raised when the associated SFM module is in operationally down/empty/failed/degraded/low-power state. Traffic could be impacted.
Severity:	Critical
Probable cause:	Equipment malfunction
Remedial action:	The active CPM is at risk of failing to initialize after node reboot because it cannot access the SFM. Contact Nokia customer support.

*Table 72: 1007 Line card fault*

Alarm ID:	1007
Alarm name:	Line Card Fault
Description:	The alarm is raised when the specified line card is in an operationally down/empty/failed/degraded/low-power state. Traffic is no longer being transmitted from this line card.
Severity:	Major
Probable cause:	Equipment malfunction
Remedial action:	Ensure that the line card is operationally up. Line card may need to be replaced.

*Table 73: 1008 Interface transceiver down*

Alarm ID:	1008
Alarm name:	Interface Transceiver Down
Description:	The alarm is raised when a transceiver goes into the operational Down state as a result of one of the following possible failures: <ul style="list-style-type: none"> <li>• read</li> </ul>

	<ul style="list-style-type: none"> <li>• checksum</li> <li>• unknown</li> <li>• tx-laser</li> <li>• connector</li> </ul>
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Ensure that the transceiver is in good operating condition and is compatible with the associated interface. The transceiver may need to be replaced.

*Table 74: 1009 Transceiver channel high input power warning*

Alarm ID:	1009
Alarm name:	Transceiver Channel High Input Power Warning
Description:	The alarm is raised when a transceiver's input power exceeds the configured Warning threshold for high input power.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 75: 1010 Transceiver channel high input power alarm*

Alarm ID:	1010
Alarm name:	Transceiver Channel High Input Power Alarm
Description:	The alarm is raised when a transceiver's input power exceeds the configured Alarm threshold for high input power.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 76: 1011 Transceiver channel low input power warning*

Alarm ID:	1011
Alarm name:	Transceiver Channel Low Input Power Warning
Description:	The alarm is raised when a transceiver's input power is below the configured Warning threshold for low input power.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 77: 1012 Transceiver channel low input power alarm*

Alarm ID:	1012
Alarm name:	Transceiver Channel Low Input Power Alarm
Description:	The alarm is raised when a transceiver's input power is below the configured Alarm threshold for low input power.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 78: 1013 Transceiver channel high laser bias current alarm*

Alarm ID:	1013
Alarm name:	Transceiver Channel High Laser Bias Current Alarm
Description:	The alarm is raised when a transceiver's high laser bias current exceeds the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 79: 1014 Transceiver channel high laser bias current warning*

Alarm ID:	1014
Alarm name:	Transceiver Channel High Laser Bias Current Warning
Description:	The alarm is raised when a transceiver's high laser bias current exceeds the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 80: 1015 Transceiver channel high output power warning*

Alarm ID:	1015
Alarm name:	Transceiver Channel High Output Power Warning
Description:	The alarm is raised when a transceiver's output power exceeds the configured Warning threshold for high output power.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 81: 1016 Transceiver channel high output power alarm*

Alarm ID:	1016
Alarm name:	Transceiver Channel High Output Power Alarm
Description:	The alarm is raised when a transceiver's output power exceeds the configured Alarm threshold for high output power.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 82: 1017 Transceiver channel low output power warning*

Alarm ID:	1017
Alarm name:	Transceiver Channel Low Output Power Warning
Description:	The alarm is raised when a transceiver's output is below the configured Warning threshold for low output power.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 83: 1018 Transceiver channel low output power alarm*

Alarm ID:	1018
Alarm name:	Transceiver Channel Low Output Power Alarm
Description:	The alarm is raised when a transceiver's output power is below configured Alarm threshold for low output power.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 84: 1019 Transceiver channel low laser bias current alarm*

Alarm ID:	1019
Alarm name:	Transceiver Channel Low Laser Bias Current Alarm
Description:	The alarm is raised when a transceiver's low laser bias current is below the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 85: 1020 Transceiver channel low laser bias current warning*

Alarm ID:	1020
Alarm name:	Transceiver Channel Low Laser Bias Current Warning
Description:	The alarm is raised when a transceiver's low laser bias current is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 86: 1021 Transceiver low laser bias current warning*

Alarm ID:	1021
Alarm name:	Transceiver Low Laser Bias Current Warning
Description:	The alarm is raised when a transceiver's laser bias current is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 87: 1022 Transceiver low laser bias current alarm*

Alarm ID:	1022
Alarm name:	Transceiver Low Laser Bias Current Alarm
Description:	The alarm is raised when a transceiver's laser bias current is below the configured Alarm threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 88: 1023 Transceiver high laser bias current warning*

Alarm ID:	1023
Alarm name:	Transceiver High Laser Bias Current Warning
Description:	The alarm is raised when a transceiver's laser bias current is above the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 89: 1024 Transceiver high laser bias current alarm*

Alarm ID:	1024
Alarm name:	Transceiver High Laser Bias Current Alarm
Description:	The alarm is raised when a transceiver's laser bias current is above the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 90: 1025 Transceiver high input power alarm*

Alarm ID:	1025
Alarm name:	Transceiver High Input Power Alarm
Description:	The alarm is raised when a transceiver's input power is above the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 91: 1026 Transceiver high input power warning*

Alarm ID:	1026
Alarm name:	Transceiver High Input Power Warning
Description:	The alarm is raised when a transceiver's input power is above the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 92: 1027 Transceiver low input power warning*

Alarm ID:	1027
Alarm name:	Transceiver Low Input Power Warning
Description:	The alarm is raised when a transceiver's input power is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 93: 1028 Transceiver low input power alarm*

Alarm ID:	1028
Alarm name:	Transceiver Low Input Power Alarm
Description:	The alarm is raised when a transceiver's input power is below the configured Critical threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 94: 1029 Transceiver high output power alarm*

Alarm ID:	1029
Alarm name:	Transceiver High Output Power Alarm
Description:	The alarm is raised when a transceiver's output power above the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 95: 1030 Transceiver high output power warning*

Alarm ID:	1030
Alarm name:	Transceiver High Output Power Warning
Description:	The alarm is raised when a transceiver's output power is above the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 96: 1031 Transceiver low output power warning*

Alarm ID:	1031
Alarm name:	Transceiver Low Output Power Warning
Description:	The alarm is raised when a transceiver's output power is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 97: 1032 Transceiver low output power alarm*

Alarm ID:	1032
Alarm name:	Transceiver Low Output Power Alarm
Description:	The alarm is raised when a transceiver's output power is below the configured Critical threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 98: 1033 Transceiver high voltage alarm*

Alarm ID:	1033
Alarm name:	Transceiver High Voltage Alarm
Description:	The alarm is raised when a transceiver's voltage is above the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 99: 1034 Transceiver high voltage warning*

Alarm ID:	1034
Alarm name:	Transceiver High Voltage Warning
Description:	The alarm is raised when a transceiver's voltage is above the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 100: 1035 Transceiver low voltage warning*

Alarm ID:	1035
Alarm name:	Transceiver Low Voltage Warning
Description:	The alarm is raised when a transceiver's voltage is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 101: 1036 Transceiver low voltage alarm*

Alarm ID:	1036
Alarm name:	Transceiver Low Voltage Alarm
Description:	The alarm is raised when a transceiver's voltage is below the configured Critical threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that correct cables and transceivers are used on both ends of the link.

*Table 102: 1037 Transceiver high temperature alarm*

Alarm ID:	1037
Alarm name:	Transceiver High Temperature Alarm
Description:	The alarm is raised when a transceiver's temperature rises above the configured Alarm threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that the correct cables and transceivers are used on both ends of the link. Additionally, verify that fans are operating correctly.

*Table 103: 1038 Transceiver high temperature warning*

Alarm ID:	1038
Alarm name:	Transceiver High Temperature Warning
Description:	The alarm is raised when a transceiver's temperature is above the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that the correct cables and transceivers are used on both ends of the link. Additionally, verify that fans are operating correctly.

*Table 104: 1039 Transceiver low temperature warning*

Alarm ID:	1039
Alarm name:	Transceiver Low Temperature Warning
Description:	The alarm is raised when a transceiver's temperature is below the configured Warning threshold.
Severity:	Warning
Probable cause:	DTE DCE TRANSCEIVER ERROR

Remedial action:	Verify that correct cables and transceivers are used on both ends of the link. Additionally, verify fans are operating correctly.
------------------	---

*Table 105: 1040 Transceiver low temperature alarm*

Alarm ID:	1040
Alarm name:	Transceiver Low Temperature Alarm
Description:	The alarm is raised when a transceiver's temperature is below the configured Critical threshold.
Severity:	Critical
Probable cause:	DTE DCE TRANSCEIVER ERROR
Remedial action:	Verify that the correct cables and transceivers are used on both ends of the link. Additionally, verify that fans are operating correctly.

## Communication alarms

*Table 106: 4001 LLDP adjacency down*

Alarm ID:	4001
Alarm name:	LLDP Adjacency Down
Description:	The alarm is raised when the Operational State of an LLDP adjacency is down. This is because the operational state of local interface is in a down state.
Severity:	Major
Probable cause:	DTE DCE Interface Error
Remedial action:	The operational state of the interface must be up in order for the selected adjacency to be up.

*Table 107: 4002 Interface down*

Alarm ID:	4002
Alarm name:	Interface Down
Description:	The alarm is raised when the operational state of interface is down.
Severity:	Critical
Probable cause:	DTE DCE Interface Error
Remedial action:	The condition exists because the physical interface is down either because it is administratively disabled, faulty or a cabling fault has occurred. Ensure that the interface is administratively up. Check for a poor cable connection to the port or for a faulty cable/fiber. If neither appears to be the problem run diagnostics on the port to determine if it is faulty.

*Table 108: 4003 Subinterface down*

Alarm ID:	4003
Alarm name:	Subinterface Down
Description:	The alarm is raised when the operational state of subinterface is down.
Severity:	Critical
Probable cause:	DTE DCE Interface Error
Remedial action:	The condition exists because the subinterface is down either because it is administratively disabled, faulty or a cabling fault has occurred. Ensure that the subinterface is administratively up. Check for a poor cable connection to the port or for a faulty cable/fiber. If neither appears to be the problem run diagnostics on the port to determine if it is faulty.

*Table 109: 4004 BGP adjacency down*

Alarm ID:	4004
Alarm name:	BGP Adjacency Down
Description:	The alarm is raised when the BGP neighbor state transitions out of the Established state.
Severity:	Critical
Probable cause:	DTE DCE BGP Error
Remedial action:	Verify reachability and BGP parameters match between BGP neighbors.

*Table 110: 4005 BFD session down*

Alarm ID:	4005
Alarm name:	BFD Session Down
Description:	The alarm is raised when the BFD session is operationally down.
Severity:	Critical
Probable cause:	DTE DCE BFD Error
Remedial action:	Verify reachability between BFD neighbors.

*Table 111: 4006 Network instance down*

Alarm ID:	4006
Alarm name:	Network Instance Down
Description:	The alarm is raised when a network-instance is down.
Severity:	Critical
Probable cause:	DTE DCE NET INST DOWN

Remedial action:	Verify the configuration of the network-instance.
------------------	---

*Table 112: 4007 Interface LAG member down*

Alarm ID:	4007
Alarm name:	Interface LAG Member Down
Description:	The alarm is raised when a member of a LAG goes into the operational down state.
Severity:	Warning
Probable cause:	DTE DCE INT LAG DOWN
Remedial action:	<p>The condition exists because the physical interface belonging to a LAG is down. The interface could be down because it is administratively disabled, faulty or a cabling fault has occurred.</p> <p>Do the following:</p> <ol style="list-style-type: none"> <li>1. Ensure that the interface is administratively up.</li> <li>2. Check for a poor cable connection to the port or for a faulty cable/fiber.</li> <li>3. If neither appears to be the problem run diagnostics on the port to determine if it is faulty.</li> </ol>

*Table 113: 4040 Network instance interface down*

Alarm ID:	4040
Alarm name:	Network Instance Interface Down
Description:	The alarm is raised when an interface configured within a network-instance is down.
Severity:	Critical
Probable cause:	DTE DCE NET INST INT DOWN
Remedial action:	Verify the operational state of the network instance interface.

*Table 114: 4041 Network instance VXLAN interface down*

Alarm ID:	4041
Alarm name:	Network Instance VXLAN Interface Down
Description:	The alarm is raised when a VXLAN interface configured within a network instance is down.
Severity:	Critical
Probable cause:	DTE DCE NET INST VXLAN INT DOWN
Remedial action:	Verify the operational state of the network instance VXLAN interface.

*Table 115: 4042 BGP down*

Alarm ID:	4042
-----------	------

Alarm name:	BGP Down
Description:	The alarm is raised when the BGP operational state transitions to the Down state.
Severity:	Critical
Probable cause:	DTE DCE BGP ERROR
Remedial action:	Verify the configuration of BGP on the affected device.

*Table 116: 4043 BGP EVPN instance down*

Alarm ID:	4043
Alarm name:	BGP EVPN Instance Down
Description:	The alarm is raised when the BGP operational state transitions to the Down state.
Severity:	Critical
Probable cause:	DTE DCE BGP ERROR
Remedial action:	Verify configuration of BGP on device.

*Table 117: 4044 BGP IPv4 neighbor down*

Alarm ID:	4044
Alarm name:	BGP IPv4 Neighbor Down
Description:	The alarm is raised whenever an IPv4 Unicast BGP family has not been negotiated correctly between two BGP neighbors.
Severity:	Major
Probable cause:	DTE DCE BGP ERROR
Remedial action:	Ensure that both neighbors are exchanging the same BGP families.

*Table 118: 4045 BGP IPv6 neighbor down*

Alarm ID:	4045
Alarm name:	BGP IPv6 Neighbor Down
Description:	The alarm is raised whenever an IPv6 Unicast BGP family has not been negotiated correctly between two BGP neighbors.
Severity:	Major
Probable cause:	DTE DCE BGP ERROR
Remedial action:	Ensure that both neighbors are exchanging the same BGP families.

*Table 119: 4046 BGP EVPN neighbor down*

Alarm ID:	4046
Alarm name:	BGP EVPN Neighbor Down

Description:	The alarm is raised whenever an EVPN BGP family has not been negotiated correctly between two BGP neighbors.
Severity:	Major
Probable cause:	DTE DCE BGP ERROR
Remedial action:	Ensure that both neighbors are exchanging the same BGP families.

## Operational alarms

*Table 120: 5001 GNMI connection fault*

Alarm ID:	5001
Alarm name:	GNMI Connection Fault
Description:	GNMI connection to the network element has been lost.
Severity:	Major
Probable cause:	DTE DCE Interface Error
Remedial action:	Check network connectivity to restore GNMI connection.

*Table 121: 5002 Memory usage warning*

Alarm ID:	5002
Alarm name:	Memory Usage Warning
Description:	The alarm is raised when a device's memory utilization exceeds 75%.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Check system process memory utilization.
	<b>Note:</b> This alarm indicates that memory utilization exceeds 75%. This alarm persists even if the Major alarm for utilization greater than 95% is triggered.

*Table 122: 5003 Memory usage major*

Alarm ID:	5003
Alarm name:	Memory Usage Major
Description:	The alarm is raised when a device's memory utilization exceeds 95%.
Severity:	Major
Probable cause:	SYSTEM WARNING
Remedial action:	Check system process memory utilization.



**Note:** This alarm indicates that memory utilization exceeds 95%. Its triggering does not affect the preceding Warning indicating 75% usage.

*Table 123: 5004 AAA server down*

Alarm ID:	5004
Alarm name:	AAA Server Down
Description:	The alarm is raised when a configured AAA server goes into the operationally down state.
Severity:	Major
Probable cause:	DTE DCE AAA DOWN
Remedial action:	Verify the device configuration and the ability to reach the AAA server from the device.

## Fabric Services System alarms

*Table 124: 6001 Connect Fabric Services System configuration failed*

Alarm ID:	6001
Alarm name:	Connect Fabric Services System Configuration Failed
Description:	The alarm is raised when changes on Plugin API cannot be configured on the Fabric Services System.
Severity:	Critical
Probable cause:	Configuration or customization error
Remedial action:	The condition exists because Connect cannot provision the Fabric Services System with the intended configuration on its Plugin API. Sanitize the Fabric Services System to resolve this error and perform audit on Connect.

*Table 125: 6002 Connect Fabric Services System workload intent deploy failed*

Alarm ID:	6002
Alarm name:	Connect Fabric Services System Workload Intent Deploy Failed
Description:	The alarm is raised when generating configurations and deploying a workload intent is not possible.
Severity:	Critical
Probable cause:	Configuration or customization error
Remedial action:	The condition exists because Connect cannot deploy the workload intent on the Fabric Services System. Please make sure the workload intent is in a deployable state and perform an audit on Connect.

*Table 126: 6003 Connect Fabric Services System authentication failed*

Alarm ID:	6003
Alarm name:	Connect Fabric Services System Authentication Failed
Description:	The alarm is raised when Connect cannot authenticate with the Fabric Services System.
Severity:	Critical
Probable cause:	Authentication failure
Remedial action:	The condition exists because Connect cannot authenticate with the Fabric Services System. Make sure the Connect configuration is correct and perform an audit on Connect.

*Table 127: 6004 Connect plugin heartbeat lost*

Alarm ID:	6005
Alarm name:	Connect Plugin Heartbeat Lost
Description:	The alarm is raised when Connect no longer detects heartbeat messages from one of its plugins.
Severity:	Critical
Probable cause:	CONNECT PLUGIN HEARTBEAT LOST
Remedial action:	The condition exists because Connect cannot detect the presence of one of its plugins. Make sure the plugin is running and actively issuing heartbeat messages.

*Table 128: 6005 Connect plugin CMS authentication failure*

Alarm ID:	6006
Alarm name:	Connect Plugin CMS Authentication Failure
Description:	The alarm is raised when a plugin fails to authenticate with the CMS for a given deployment.
Severity:	Critical
Probable cause:	CONNECT PLUGIN CMS AUTHENTICATION FAILURE
Remedial action:	The condition exists because a plugin cannot authenticate with the CMS. Make sure the deployment configuration is correct.

*Table 129: 6007 Connect plugin Connect out of sync with CMS*

Alarm ID:	6007
Alarm name:	Connect Plugin Connect Out of Sync with CMS
Description:	The alarm is raised when a plugin re-establishes communication with Connect, after having registered changes to its deployments during the period of connection loss.

Severity:	Critical
Probable cause:	CONNECT PLUGIN CONNECT OUT OF SYNC WITH CMS
Remedial action:	The condition exists because the plugin lost connectivity with Connect for some time. Perform an audit if Connect is out of sync with the CMS.

*Table 130: 6008 Connect plugin CMS connectivity failure*

Alarm ID:	6008
Alarm name:	Connect Plugin CMS Connectivity Failure
Description:	The alarm is raised when a plugin has lost communication with the CMS, or if there was no connectivity at all for a given deployment.
Severity:	Critical
Probable cause:	CONNECT PLUGIN CMS CONNECTIVITY FAILURE
Remedial action:	The condition exists because plugin cannot connect to the CMS. Please make sure you have connectivity between the plugin and the CMS.

*Table 131: 6009 Connect resource out of sync*

Alarm ID:	6009
Alarm name:	Connect Resource Out Of Sync
Description:	The alarm is raised when a resource in connect is out of sync.
Severity:	Critical
Probable cause:	INCORRECT CONFIGURATION
Remedial action:	The condition exists because a resource in Connect is out of sync. Run an audit on the deployment this resource belongs to.

*Table 132: 6010 Connect plugin CMS certificate verification failure*

Alarm ID:	6010
Alarm name:	Connect Plugin CMS Certificate Verification Failure
Description:	The alarm is raised when plugin failed to authenticate with CMS because of the given deployment certificate.
Severity:	Critical
Probable cause:	CONNECT PLUGIN CMS CERTIFICATE VERIFICATION FAILURE
Remedial action:	The condition exists because a plugin cannot authenticate with CMS because of the given deployment certificate. Make sure the deployment certificate is correct.

*Table 133: 6011 Connect plugin CMS resource misconfigured*

Alarm ID:	6011
Alarm name:	Connect Plugin CMS Resource Misconfigured

Description:	The alarm is raised when plugin failed to create resources because of a misconfigured CMS resource.
Severity:	Critical
Probable cause:	INCORRECT CONFIGURATION
Remedial action:	Configure the CMS resource correctly. If the alarm persists after reconfiguration, audit the deployment.

*Table 134: 9009 Instance down*

Alarm ID:	9009
Alarm name:	Instance Down
Description:	The alarm is raised when the pod instance has been down for more than 5 minutes.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 135: 9010 Kubernetes pod crash looping*

Alarm ID:	9010
Alarm name:	Kubernetes Pod Crash Looping
Description:	The alarm is raised when the pod instance is consistently crashing upon restart.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 136: 9011 Kubernetes pod not healthy*

Alarm ID:	9011
Alarm name:	Kubernetes Pod Not Healthy
Description:	The alarm is raised when the pod instance has been in a non-ready state for longer than five minutes
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 137: 9012 Kubernetes container OOM killer*

Alarm ID:	9012
Alarm name:	Kubernetes Container OOM Killer

Description:	The alarm is raised when the container in pod has been OOMKilled multiple times in the last 10 minutes.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 138: 9013 Kubernetes StatefulSet down*

Alarm ID:	9013
Alarm name:	Kubernetes Stateful Set Down
Description:	The alarm is raised when the Kubernetes StatefulSet is down.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 139: 9014 Kubernetes StatefulSet replicas mismatch*

Alarm ID:	9014
Alarm name:	Kubernetes Statefulset Replicas Mismatch
Description:	The alarm is raised when the StatefulSet does not match the expected number of replicas.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 140: 9015 Kubernetes deployment down*

Alarm ID:	9015
Alarm name:	Kubernetes Deployment Down
Description:	The alarm is raised when the deployment is in a down state.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 141: 9016 Kubernetes deployment replicas mismatch*

Alarm ID:	9016
Alarm name:	Kubernetes Deployment Replicas Mismatch
Description:	The alarm is raised when the deployment does not have the expected number of replicas.

Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 142: 9017 Kubernetes daemonset rollout stuck*

Alarm ID:	9017
Alarm name:	Kubernetes Daemonset Rollout Stuck
Description:	The alarm is raised when the pod instance DaemonSet rollout is stuck. Some pods of DaemonSet are not scheduled or are not ready.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.

*Table 143: 9030 Kubernetes node not ready*

Alarm ID:	9030
Alarm name:	Kubernetes Node Not Ready
Description:	The alarm is raised when the Kubernetes node is not ready.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 144: 9031 Kubernetes node out of memory*

Alarm ID:	9031
Alarm name:	Kubernetes Node Out Of Memory
Description:	The alarm is raised when the Kubernetes node has high memory utilization (> 90%).
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 145: 9032 Kubernetes node high CPU load*

Alarm ID:	9032
Alarm name:	Kubernetes Node High CPU Load
Description:	The alarm is raised when Kubernetes node has a high CPU load (> 80%)
Severity:	Warning
Probable cause:	SYSTEM WARNING

Remedial action:	Inspect the logs of the pod and Kubernetes to check the reason for its state.
------------------	---

*Table 146: 9033 Kubernetes node CPU high I/O wait*

Alarm ID:	9033
Alarm name:	Kubernetes Node Cpu High IO wait
Description:	The alarm is raised when the Kubernetes node has a high CPU I/O wait (> 10%).
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 147: 9034 Kubernetes node out of disk space*

Alarm ID:	9034
Alarm name:	Kubernetes Node Out Of Disk Space
Description:	The alarm is raised when the Kubernetes node is out of disk space.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 148: 9035 Kubernetes node clock not synchronizing*

Alarm ID:	9035
Alarm name:	Kubernetes Node Clock Not Synchronizing
Description:	The alarm is raised when the clock on the Kubernetes node is not synchronizing. Ensure that NTP is configured on this host.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state and make sure NTP is configured, enabled and working.

*Table 149: 9036 Kubernetes node out of capacity*

Alarm ID:	9036
Alarm name:	Kubernetes Node Out Of Capacity
Description:	The alarm is raised when the Kubernetes node is out of capacity and cannot support more workloads.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 150: 9037 Kubernetes volume out of disk space*

Alarm ID:	9037
Alarm name:	Kubernetes Volume Out Of Disk Space
Description:	The alarm is raised when the Kubernetes volume has a high usage (> 90%).
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the logs of the VM and Kubernetes to check the reason for its state.

*Table 151: 9050 Ceph state unhealthy*

Alarm ID:	9050
Alarm name:	Ceph State Unhealthy
Description:	The alarm is raised when the Ceph instance is in an unhealthy state
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph.

*Table 152: 9051 Ceph monitor low space*

Alarm ID:	9051
Alarm name:	Ceph Monitor Low Space
Description:	The alarm is raised when the Ceph monitor has low disk space.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph.

*Table 153: 9052 Ceph OSD down*

Alarm ID:	9052
Alarm name:	Ceph OSD Down
Description:	The alarm is raised when the Ceph OSD (Object Storage Daemon) is not in a healthy state.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph.

*Table 154: 9053 Ceph OSD high latency*

Alarm ID:	9053
Alarm name:	Ceph OSD High Latency
Description:	The alarm is raised when the Ceph OSD (Object Storage Daemon) has high latency.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph.

*Table 155: 9054 Ceph OSD low space*

Alarm ID:	9054
Alarm name:	Ceph OSD Low Space
Description:	The alarm is raised when the Ceph OSD (Object Storage Daemon) has low disk space.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph and add more storage if needed.

*Table 156: 9055 Ceph PG down*

Alarm ID:	9055
Alarm name:	Ceph PG Down
Description:	The alarm is raised when the Ceph PG (Placement Group) is down. A PG with fewer than the minimum replicas is marked as down.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph and check if all data is available.

*Table 157: 9056 Ceph PG incomplete*

Alarm ID:	9056
Alarm name:	Ceph PG Incomplete
Description:	The alarm is raised when the Ceph PG (Placement Group) is missing information about writes that might have occurred, or does not have any healthy copies.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph and check if all data is available.

*Table 158: 9057 Ceph PG inconsistent*

Alarm ID:	9057
Alarm name:	Ceph PG Inconsistent
Description:	The alarm is raised when the Ceph PG (Placement Group) has objects that have an incorrect size or are missing from one replica.
Severity:	Warning
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph and check if all data is available.

*Table 159: 9058 Ceph PG unavailable*

Alarm ID:	9058
Alarm name:	Ceph PG Unavailable
Description:	The alarm is raised when the Ceph PG (Placement Group) is in an unavailable state.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of Ceph and check if all data is available.

*Table 160: 9100 GeoRedundancy failed*

Alarm ID:	9100
Alarm name:	GeoRedundancy Failed
Description:	The alarm is raised when the Geo Redundancy fails.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of remote and local cluster and take appropriate action.

*Table 161: 9102 GeoRedundancy reconcile failed*

Alarm ID:	9102
Alarm name:	GeoRedundancy Reconcile Failed
Description:	The alarm is raised when a Geo Redundancy Reconcile fails.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of remote and local cluster services and take appropriate action.

*Table 162: 9103 GeoRedundancy audit failed*

Alarm ID:	9103
Alarm name:	GeoRedundancy Audit Failed
Description:	The alarm is raised when a Geo Redundancy Audit fails.
Severity:	Critical
Probable cause:	SYSTEM WARNING
Remedial action:	Inspect the state of remote and local cluster services and take appropriate action.

*Table 163: 9200 Image download failed*

Alarm ID:	9200
Alarm name:	Image Download Failed
Description:	The alarm is raised when image downloading fails.
Severity:	Major
Probable cause:	INCORRECT CONFIGURATION
Remedial action:	Inspect and correct the URL or the credentials. Also check if there is any connectivity issue.

## Appendix B: Protobuf file message format

### Example

```
syntax = "proto3";

package external;

message AlarmResponseKey {
  string alarmType = 1;
  string nodeId = 2;
  string resourceName = 3;
}

message FssAlarm {
  message History {
    message OperatorHistory {
      string additionalInfo = 1;
      string alarmUuid = 2;
      string endReason = 3;
      string toTimestamp = 4;
    }
    message OperatorHistoryKey {
      OperatorState operatorState = 1;
      string operatorName = 2;
      string fromTimestamp = 3;
      OperatorHistory operator_history = 4;
    }
    message StateHistory {
      string alarmPolicy = 1;
      string alarmText = 2;
      string toTimestamp = 3;
    }
    message StateHistoryKey {
      AlarmSeverity severity = 1;
      string fromTimestamp = 2;
      StateHistory state_history = 3;
    }
    string alarmPolicy = 1;
    AlarmSeverity currentSeverity = 2;
    repeated OperatorHistoryKey operator_history = 3;
    repeated StateHistoryKey state_history = 4;
  }
  message Resource {
    string fabricId = 1;
    string fabricName = 2;
    string intentId = 3;
    string intentName = 4;
    FabricOptions intentType = 5;
    string name = 6;
    string nodeId = 7;
    string nodeName = 8;
    string nodeType = 9;
    string objectId = 10;
    string objectName = 11;
    string objectType = 12;
    string objectURI = 13;
    string regionId = 14;
    string regionName = 15;
    string resourceName = 16;
    string serviceName = 17;
  }
}
```

```

    string uuid = 18;
  }
  string SuppressReason = 1;
  string alarmDescription = 2;
  string alarmType = 3;
  string createdAt = 4;
  string defaultProbableCause = 5;
  AlarmSeverity defaultSeverity = 6;
  uint64 faultCode = 7;
  GroupType group = 8;
  History history = 9;
  bool isCleared = 10;
  bool isRootCause = 11;
  bool isSuppressed = 12;
  string lastChanged = 13;
  string lastRaised = 14;
  uint32 occurrence = 15;
  string probableCause = 16;
  string remedialAction = 17;
  Resource resource = 18;
  string uuid = 19;
}

enum AlarmActionType {
  FssAlarmAlarmActionType_UNSET = 0;
  GenerateAlarm = 1;
  Email = 2;
  SMS = 3;
  Pager = 4;
}

enum AlarmCommunication {
  Communication_Unset = 0;
  AllCommunicationType = 1;
  LLDPAdjacencyDown = 3;
  InterfaceDown = 4;
  BGPAdjacencyDown = 5;
  BFDSessionDown = 6;
  SubinterfaceDown = 12;
  NetworkInstanceDown = 13;
  InterfaceLagMemberDown = 14;
  NetworkInstanceInterfaceDown = 56;
}

enum AlarmConfiguration {
  Configuration_Unset = 0;
  AllConfigurationType = 1;
}

enum AlarmEnvironment {
  Environment_Unset = 0;
  AllEnvironmentType = 1;
}

enum AlarmEquipment {
  Equipment_Unset = 0;
  AllEquipmentType = 1;
  FanTrayFault = 2;
  LineCardFault = 3;
  PowerSupplyFault = 4;
  ChassisFault = 11;
  CPMFault = 12;
  SFMFault = 13;
}

```

```

InterfaceTransceiverDown = 17;
TransceiverChannelHighInputPowerWarning = 18;
TransceiverChannelHighInputPowerAlarm = 19;
TransceiverChannelLowInputPowerWarning = 20;
TransceiverChannelLowInputPowerAlarm = 21;
TransceiverChannelHighLaserBiasCurrentAlarm = 22;
TransceiverChannelHighLaserBiasCurrentWarning = 23;
TransceiverChannelHighOutputPowerWarning = 24;
TransceiverChannelHighOutputPowerAlarm = 25;
TransceiverChannelLowOutputPowerWarning = 26;
TransceiverChannelLowOutputPowerAlarm = 27;
TransceiverChannelLowLaserBiasCurrentAlarm = 28;
TransceiverChannelLowLaserBiasCurrentWarning = 29;
TransceiverLowLaserBiasCurrentWarning = 30;
TransceiverLowLaserBiasCurrentAlarm = 31;
TransceiverHighLaserBiasCurrentWarning = 32;
TransceiverHighLaserBiasCurrentAlarm = 33;
TransceiverHighInputPowerAlarm = 34;
TransceiverHighInputPowerWarning = 35;
TransceiverLowInputPowerWarning = 36;
TransceiverLowInputPowerAlarm = 37;
TransceiverHighOutputPowerAlarm = 38;
TransceiverHighOutputPowerWarning = 39;
TransceiverLowOutputPowerWarning = 40;
TransceiverLowOutputPowerAlarm = 41;
TransceiverHighVoltageAlarm = 42;
TransceiverHighVoltageWarning = 43;
TransceiverLowVoltageWarning = 44;
TransceiverLowVoltageAlarm = 45;
TransceiverHighTemperatureAlarm = 46;
TransceiverHighTemperatureWarning = 47;
TransceiverLowTemperatureWarning = 48;
TransceiverLowTemperatureAlarm = 49;
}

enum AlarmFss {
  Fss_Unset = 0;
  AllFssType = 1;
  FssModule = 2;
  ExternalModule = 3;
  ConnectFSSConfigurationFailed = 4;
  ConnectFSSWorkloadIntentDeployFailed = 5;
  ConnectFSSAuthenticationFailed = 6;
  ConnectPluginHeartbeatLost = 8;
  ConnectPluginCmsAuthenticationFailure = 53;
  ConnectPluginConnectOutOfSyncWithCms = 54;
  ConnectPluginCmsConnectivityFailure = 55;
  ConnectResourceOutOfSync = 57;
  ConnectPluginCmsCertificateVerificationFailure = 58;
}

enum AlarmOperational {
  Operational_Unset = 0;
  AllOperationalType = 1;
  GnmiConnectionFault = 11;
  MemoryUsageWarning = 50;
  MemoryUsageMajor = 51;
  AaaServerDown = 52;
}

enum AlarmSeverity {
  AlarmSeverity_UNSET = 0;
  Critical = 1;
  Major = 2;
}

```

```
Warning = 3;
Minor = 4;
}

enum ComparisonOperatorType {
  ComparisonOperatorType_UNSET = 0;
  EqualTo = 1;
  NotEqualTo = 2;
  Contains = 3;
  NotContains = 4;
  GreaterThan = 5;
  LessThan = 6;
  GreaterThanEqualTo = 7;
  LessThanEqualTo = 8;
}

enum FabricOptions {
  FssAlarmFabricOptions_UNSET = 0;
  Real = 1;
  DigitalSandbox = 2;
}

enum GroupType {
  GroupType_UNSET = 0;
  Equipment = 1;
  Configuration = 2;
  Environment = 3;
  Communication = 4;
  Operational = 5;
  Fss = 6;
  All = 7;
}

enum LogicalOperatorType {
  LogicalOperatorType_UNSET = 0;
  And = 1;
  Or = 2;
}

enum OperatorState {
  OperatorState_UNSET = 0;
  Ack = 1;
  Closed = 2;
}

enum ProbableCause {
  ProbableCause_UNSET = 0;
  PROBCAUSE_INDETERMINATE = 1;
  OTHER = 2;
  ADAPTERERROR = 3;
  APPLICATIONSUBSYSTEMFAILURE = 4;
  BANDWIDTHREDUCED = 5;
  CALLESTABLISHMENTERROR = 6;
  COMMUNICATIONSPROTOCOLERROR = 7;
  COMMUNICATIONSSUBSYSTEMFAILURE = 8;
  CONFIGURATIONORCUSTOMIZATIONERROR = 9;
  CONGESTION = 10;
  CORRUPTDATA = 11;
  CPUCYCLESLIMITEXCEEDED = 12;
  DATASETORMODEMERROR = 13;
  DEGRADEDSIGNAL = 14;
  DTEDCEINTERFACEERROR = 15;
}
```

```
ENCLOSUREDOOROPEN = 16;
EQUIPMENTMALFUNCTION = 17;
EXCESSIVEVIBRATION = 18;
FILEERROR = 19;
FIREDETECTED = 20;
FLOODDETECTED = 21;
FRAMINGERROR = 22;
HEATINGVENTCOOLINGSYSTEMPROBLEM = 23;
HUMIDITYUNACCEPTABLE = 24;
INPUTOUTPUTDEVICEERROR = 25;
INPUTDEVICEERROR = 26;
LANERROR = 27;
LEAKDETECTED = 28;
LOCALNODETRANSMISSIONERROR = 29;
LOSSOFFRAME = 30;
LOSSOFSIGNAL = 31;
MATERIALSUPPLYEXHAUSTED = 32;
MULTIPLEXERPROBLEM = 33;
OUTOFMEMORY = 34;
OUPUTDEVICEERROR = 35;
PERFORMANCEDEGRADED = 36;
POWERPROBLEM = 37;
PRESSUREUNACCEPTABLE = 38;
PROCESSORPROBLEM = 39;
PUMPFailure = 40;
QUEUE SIZE EXCEEDED = 41;
RECEIVEFAILURE = 42;
RECEIVERFAILURE = 43;
REMOTENODETRANSMISSIONERROR = 44;
RESOURCEATORNEARINGCAPACITY = 45;
RESPONSE TIME EXCESSIVE = 46;
RETRANSMISSIONRATE EXCESSIVE = 47;
SOFTWAREERROR = 48;
SOFTWAREPROGRAMABNORMALLYTERMINATED = 49;
SOFTWAREPROGRAMERROR = 50;
STORAGECAPACITYPROBLEM = 51;
TEMPERATUREUNACCEPTABLE = 52;
THRESHOLD CROSSED = 53;
TIMINGPROBLEM = 54;
TOXICLEAKDETECTED = 55;
TRANSMITFAILURE = 56;
TRANSMITTERFAILURE = 57;
UNDERLYINGRESOURCEUNAVAILABLE = 58;
VERSIONMISMATCH = 59;
AUTHENTICATIONFAILURE = 60;
BREACHOFCONFIDENTIALITY = 61;
CABLETAMPER = 62;
DELAYEDINFORMATION = 63;
DENIALOFSERVICE = 64;
DUPLICATEINFORMATION = 65;
INFORMATIONMISSING = 66;
INFORMATIONMODIFICATIONDETECTED = 67;
INFORMATIONOUTOFSEQUENCE = 68;
INTRUSIONDETECTION = 69;
KEYEXPIRED = 70;
NONREPUDIATIONFAILURE = 71;
OUTOFHOURSACTIVITY = 72;
OUTOFSERVICE = 73;
PROCEDURALERROR = 74;
UNAUTHORIZEDACCESSATTEMPT = 75;
UNEXPECTEDINFORMATION = 76;
NODEREBOOT = 77;
DTEDCESUBINTERFACEERROR = 78;
DTEDCEBGPERROR = 79;
```

```
DTEDCEBFDERROR = 80;  
DTEDCENETINSTDOWN = 81;  
CONNECTPLUGINHEARTBEATLOST = 82;  
DTEDCEINTLAGDOWN = 83;  
DTEDCETRANSCEIVERERROR = 84;  
SYSTEMWARNING = 85;  
DTEDCEAAADOWN = 86;  
CONNECTPLUGINCMSAUTHENTICATIONFAILURE = 87;  
CONNECTPLUGINCONNECTOUTOFSYNCWITHCMS = 88;  
CONNECTPLUGINCMSCONNECTIVITYFAILURE = 89;  
DTEDCENETINSTINTDOWN = 90;  
INCORRECTCONFIGURATION = 91;  
CONNECTPLUGINMSCERTIFICATEVERIFICATIONFAILURE = 92;  
}
```

## Appendix C: Acronyms

### 0-9

**7220 IXR**                      Nokia 7220 Interconnect Router

### A

**ACL**                              access control list

**AES-128-CFB**                  Advanced Encryption Standard-128-Cipher Feedback

**API**                                application programming interface

**ARP**                                Address Resolution Protocol

**ASN**                                autonomous system numbers

### B

**BFD**                                bidirectional forwarding detection

**BGP**                                Border Gateway Protocol

### C

**CA**                                 certificate authority

**CE**                                 customer edge

**CIDR**                              classless inter-domain routing

**CoPP**                              control plane policing

**CPM**                                control processing module

**CSI**                                container storage interface

**CSR**                                certificate signing request

### D

**DC**                                 data center

**DCE**                                data circuit terminal equipment

**DES**                                data encryption standard

**DN**                                In LDAP configuration, the distinguished name

**DSCP**                              differentiated services code point

**DHCP**                              Dynamic Host Configuration Protocol

**DTE**                                data termination equipment

**E**

<b>EVI</b>	EVPN instance
<b>EVPN</b>	Ethernet virtual private network

**G****gNMI****I**

<b>ICMP</b>	Internet Control Message Protocol
<b>IRB</b>	integrated routing and bridging
<b>ISIS</b>	intermediate system to intermediate system
<b>ISL</b>	inter-switch link

**J**

<b>JSON</b>	Java Script Object Notation
-------------	-----------------------------

**L**

<b>LACP</b>	Link Aggregation Control Protocol
<b>LAG</b>	link aggregation group
<b>LDAP</b>	Lightweight Directory Access Protocol
<b>LLDP</b>	Link Layer Discovery Protocol

**M**

<b>MAC</b>	media access controller
<b>MH-LAG</b>	multi-home LAG
<b>MTU</b>	maximum transmission unit

**N**

<b>NFS</b>	network file server
<b>NTP</b>	Network Time Protocol

**O**

<b>OSPF</b>	open shortest path first
-------------	--------------------------

**P**

<b>PE</b>	provider edge
-----------	---------------

---

<b>PSU</b>	power supply unit
<b>PVC</b>	persistent volume claim
<b>R</b>	
<b>RDN</b>	In LDAP configuration, the relative distinguished name
<b>REST</b>	representational state transfer
<b>RT</b>	route target
<b>S</b>	
<b>SFM</b>	switch fabric module
<b>SHA</b>	secure hashing algorithm
<b>SSHFS</b>	secure shell file system
<b>SNMP</b>	Simple Network Management Protocol
<b>SRL</b>	Nokia Service Router Linux
<b>T</b>	
<b>TLS</b>	transport layer security
<b>V</b>	
<b>VPN</b>	virtual private network
<b>VNI</b>	virtual network identifier
<b>VRF</b>	virtual routing and forwarding
<b>VXLAN</b>	virtual extensible local area network
<b>Z</b>	
<b>ZTP</b>	SR Linux zero-touch provisioning

# Customer document and product support



## **Customer documentation**

[Customer documentation welcome page](#)



## **Technical support**

[Product support portal](#)



## **Documentation feedback**

[Customer documentation feedback](#)