Configuration Guide for Cisco NCS 1004, IOS XR Release 24.3.x

First Published: 2024-09-04

Last Modified: 2026-08-24

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

© 2025 Cisco Systems, Inc. All rights reserved.

CONTENTS

CHAPTER 1 **Cisco NCS 1004 Overview** 1

Cisco NCS 1004 1

Interoperability constraints and timing requirements for Cisco NCS 1001 and Cisco NCS 1004 2

Supported line cards 2

CHAPTER 2 **Configuring the Card Mode** 7

1.2T and 1.2TL Line Cards 7

Card modes 7

Sub 50G configuration 8

Configure Split Client Port Mapping 10

Supported Data Rates 12

Configuring the Card Mode 14

Regeneration Mode 19

Configuring the Card in Regen Mode 19

Verifying the Regen Mode 20

Configuring the BPS 20

Configuring the Trunk Rate for BPSK 21

Viewing the BPSK Trunk Rate Ranges 21

OTN-XP Card 23

LC Mode on OTN-XP Card 23

Configuring the LC Mode 25

Regeneration Mode on OTN-XP Card 31

Configuring Regen Mode on OTN-XP Card 31

FC-MXP Mode on OTN-XP Card 32

Configuring the OTN-XP Card in 16G FC Muxponder Mode 34

Configuring the OTN-XP Card in 32G FC Muxponder Mode 36

Supported Pluggables for OTN-XP Card 38
Muxponder Configuration on OTN-XP Card 39
Configuring the Muxponder Mode for 10G Grey Muxponder 47
Configuring DAC Rate for 400G Muxponder Modes 48
Configuring the Muxponder Mode for 4x100G MXP 50
Configuring the Muxponder Mode for 400G TXP 51
Static TPN and TS allocation for TXP-MXP-Grey muxponder modes 52
Configuring the Muxponder Mode for 40x10G Muxponder 53
Configuring the Muxponder Mode for 30x10G 56
Configuring Hybrid Modes Using 40x10G-4x100G-MXP 58
Configuring the Muxponder Mode for 200G on OTN-XP Card 69
Configuring the Muxponder Mode for 300G on OTN-XP Card 73
Configuring the Muxponder Mode for 4x100GE-MXP-DD 77
Configuring the Muxponder Mode for 2x100GE-MXP-DD 80
Configuring the Muxponder Mode for 3x100GE-MXP-DD 83
Configuring the Transponder Mode for 400GE-TXP-DD 86
Configuring Inverse Muxponder on OTN-XP Card for 400GE Client 89
2x100GE-TXP-MXP on OTN-XP Card 93
2-QDD-C Line Card 98
Limitations for 2-QDD-C 98
Unsupported Features for 2-QDD-C in R731 98
2-QDD-C Card Modes 98
Sub 50G Configuration 99
Supported Data Rates for 2-QDD-C Card 100
Configuring the Card Mode for 2-QDD-C Card 102
Configuring Mixed Client Traffic Mode 103
QXP Card 107

CHAPTER 3 **Configuring Controllers 117**
AINS 118
AINS States 118
Soak Time Period 118
Configuring AINS 118
Disabling AINS 119

Displaying the AINS Configuration 120
Configuring AINS on OTN-XP Card 123
FEC 130
FEC States for Ethernet Controller 131
Configuring FEC on the Ethernet Controller 133
FEC States for CoherentDSP Controller 134
Q-Margin Support 134
Configuring FEC on CoherentDSP Controllers 135
Verifying FEC on CoherentDSP Controllers 135
Configuring FEC on OTN-XP Card 136
Verifying FEC on OTN-XP Card 136
Configuring FEC on OTN-XP Card – QDD-400G-ZRP 137
Laser Squelching 143
Protection Switching Use Cases 144
Configuring Laser Squelching on OTN-XP Card 146
Idle Insertion 150
Enabling Idle Insertion on OTN-XP Card 152
Enable Idle Insertion on QXP Card 154
Idle Insertion for Ethernet Controllers 157
Recommended Topology for Link Verification 157
Configuring Idle Insertion for Ethernet Controllers 158
Verifying Idle Insertion Configuration for Ethernet Controllers 159
LLDP Drop 159
Configuring LLDP Drop 161
Verifying the Status of LLDP Drop 161
Link Layer Discovery Protocol (LLDP) Support on Management Interface 163
Daisy Chain Support on Management Ports 167
Configure Daisy Chain Support on Management Ports 167
Verify Daisy Chain 168
Enable Storm Control on TOR Switch 168
Disable DAD on Management Port 168
DHCP Client 169
DHCP Client Options 169
Enabling DHCP Client on Management Ethernet Interface 170

Verifying DHCP Client on Management Ethernet Interface 171
MAC Address Snooping on Client Ports 172
Configuring MAC Address Snooping on Client Ports 172
Viewing Neighbor MAC Address 173
Transmit Shutdown 174
Configuring Transmit Shutdown on Trunk Optics Controller 174
Verifying Transmit Shutdown on Trunk Optics Controller 175
Loopback 178
Restore Factory Settings 195
Headless Mode 197
Trail Trace Identifier 197
Configure TTI on OTN-XP Card 199
Configure TTI on QXP Card 205
Chromatic dispersion 206
Transmit Power 208
Laser Bias Current High Threshold 211
Differential Group Delay Threshold 213
Optical Signal to Noise Ratio 215
Chromatic Dispersion Threshold 217
Receive Power Threshold 219
Transmit Power Threshold 221
Frequency 223
Pseudo Random Binary Sequence 223
Configuring Pseudo Random Binary Sequence 225
Verifying PRBS 226
Viewing PRBS Performance Monitoring Parameters 226
Configuring PRBS on OTN-XP Card 227
Verifying PRBS on OTN-XP Card 229
Clearing Bit Errors and Lock Time for PRBS 230
FlexO GID and IID 231
Configuring FlexO GID and IID 231
Verifying FlexO GID and IID 231
Flexo Parameter Update on Inverse Muxponder Configuration on the OTN-XP Card 232
FPD 236

Automatic Protection Switching (APS) on OTN XP Card 237

CHAPTER 4 **Performance Monitoring 247**

Performance monitoring parameters 247

PM history persistence features 247

Instantaneous Q-margins 248

Configure PM parameters 249

EC-BITS threshold values for NCS1K4-1.2T-K9 line card 251

View PM parameters 252

Performance monitoring parameter outputs for Ethernet, FEC, optics, and Fibre Channel controllers 254

Clear PM parameters 262

CHAPTER 5 **IP Access Lists 265**

IP access control lists 265

Best practices and requirements for configuring IP access control lists 266

Configure an IP ACL 266

Verify access control lists 267

CHAPTER 6 **Layer 1 Encryption 269**

IKEv2 Overview 271

OTNSec Encryption Overview 273

Prerequisites 274

Limitations 275

Configuration Workflow 275

Configuring an IKEv2 Proposal 277

Configuring an IKEv2 Policy 278

Configuring a Keyring 279

Configuring a IKEv2 Profile 280

Configuring an OTNSec Policy 280

Configuring the GCC Interface 281

Configuring OTNSec on ODU4 Controllers 282

Configuring OTNSec on ODUC4 Controllers for OTN-XP Card 283

Configuration Example 284

Verification 287
Troubleshooting 288
IKEv2 Certificate-Based Authentication 288
Configuring IKEv2 Certificate-Based Authentication 288
You May Be Interested In 293

CHAPTER 7 Quantum-Safe Encryption Using Postquantum Preshared Keys 295
Postquantum preshared keys 295
Dynamic postquantum preshared keys 297
Cisco Secure Key Integration Protocol 297
Workflow for dynamic postquantum preshared keys 298
Configuring Dynamic PPK using SKS SKIP 299
Manual postquantum preshared keys 300
Configure a manual PPK in an IKEv2 keyring 300
Type 6 password support for preshared keys in IKEv2 authentication 302
Enable Type 6 password 303
Verify the PPK configuration 305

CHAPTER 8 GMPLS UNI for Packet and Optical Integration 311
GMPLS UNI 311
GMPLS UNI tunnels 313
Licensing and configuration prerequisites for GMPLS UNI tunnels 313
Limitations for a GMPLS UNI tunnel 314
Configuration workflow for a GMPLS UNI tunnel 314
Configure an LMP link and Alien Wavelength on an NCS 2000 node using CTC for signaled numbered circuit 314
Configure an LMP link with Alien Wavelength on NCS 2000 using CTC for signaled unnumbered circuits 317
Retrieve ifindex from the NCS 2000 node 319
Configure LMP on Cisco NCS 1004 node 320
Configure the MPLS tunnel on the NCS 1004 node 322
Configure RSVP on the NCS 1004 node 323
GMPLS UNI tunnel, RSVP, and LMP configuration commands and sample outputs 324
GMPLS UNI tunnel troubleshooting commands and resolutions 331

Related information for GMPLS UNI 332

CHAPTER 9 **Remote Node Management** 333

Remote node management using GCC 333

Supported and unsupported features of the GCC interface 334

Supported protocols 335

Enable the GCC interface 335

Configure the GCC interface 336

Configure static routes over the GCC interface 338

Configure OSPF routes over the GCC interface 338

iBGP supports over GCC interfaces 340

Caution: Follow restrictions for iBGP support using GCC 340

Enabling the GCC Interface 340

Configuring the Management Interface 341

Configuring the Loopback Interface 341

Configuring the GCC interface 341

Verifying iBGP Support Using GCC 343

iBGP configuration parameters for GCC interfaces 343

CHAPTER 10 **Smart Licensing** 347

Smart Licensing 347

Smart licensing benefits for Cisco NCS 1004 350

Smart Licensing in Cisco NCS 1004 351

Software entitlements of Cisco NCS 1004 351

Generic smart licensing 353

Create a token 357

Configure smart licensing 357

Verify smart licensing configuration 359

Criteria for license consumption of subsea networks 363

Criteria for license consumption of long-haul networks 365

Register a smart license on Cisco NCS 1004 366

Supported transport modes for Smart Licensing 367

Configure Call Home for Smart Licensing 367

Configure Smart Transport 368

Configure CSLU transport mode 369
Configure offline license mode 371
Smart licensing for OTN-XP line card 371
License usage count for Cisco NCS 1004 OTN-XP line card 372
Smart licensing for OTN-XP card in regen mode 374
Check the license count for OTN-XP line card in regen mode 374
Smart license model for QXP line card 376

CHAPTER 11 USB Device Automount 379
USB automounts 379
Mount a USB device in virtual machine environments 379
Unmount a USB device 380

CHAPTER 12 Fault Profiles 383
Fault profiles 383
Best practice for configuring fault profiles 384
Configure a fault profile 384

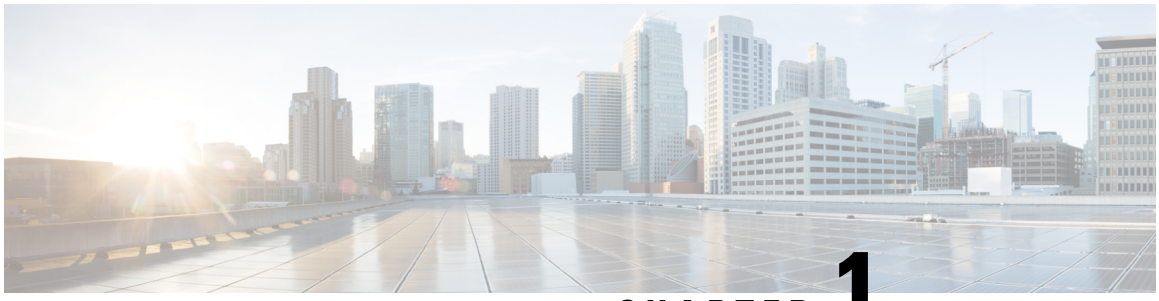
CHAPTER 13 OC Support for AAA User 385
OC Support for AAA User 385
AAA services 386
Admin access methods for NETCONF and gRPC 387
User profile mappings between XR VM and System Admin VM 387
Administration data access rules 389

CHAPTER 14 Implementing Host Services and Applications 391
HTTP client application 391
TCP 391

CHAPTER 15 Implementing Certification Authority Interoperability 393
Certification authority interoperability 393
Prerequisites for CA interoperability 393
Maximum supported CA key size for interoperability on Cisco NCS 1004 394

Enroll RSA key pairs for device certificates 394
Configure certificate enrollment using cut-and-paste 395
Public key pair capabilities in XR Config mode 396
How to implement CA interoperability 399
Configure the hostname and IP domain name on Cisco NCS 1004 400
Generate an RSA key pair on Cisco NCS 1004 400
Import a public key to Cisco NCS 1004 402
Declare a certification authority and configure a trusted point 403
Authenticate certification authority 405
Multi-tier certificate authority for trustpoint authentication 405
Using multi-tier CA for trustpoint authentication 406

APPENDIX A **Supported SNMP MIBs on NCS 1004** 409



CHAPTER 1

Cisco NCS 1004 Overview

This chapter provides an overview of the Cisco Network Convergence Series (NCS) 1004.

- Cisco NCS 1004 , on page 1

Cisco NCS 1004

A Cisco NCS 1004 unit is an optical transport platform that

- supports up to 4.8 Tbps traffic in a compact two-RU form factor,
- includes two redundant field-replaceable AC and DC power supply units; three redundant field-replaceable fans; a field-replaceable controller card; and SSD disks both on board the chassis and on the controller card. This configuration provides enhanced resiliency, and
- features four line card slots per chassis, each capable of hosting a line card for scalable network expansion.

See Supported Line Cards, for more information.

Key features and benefits of Cisco NCS 1004

The NCS 1004 delivers these benefits:

- Transport of any trunk rate from 150 200 to 600 Gbps wavelengths in 50 Gbps increments on the same platform through software provisioning.
- Support of granular control of baud-rate and modulation format to maximize spectral efficiency.
- One universal transponder optimized for metro, long-haul, and submarine applications.
- Support for up to 350,000 ps/nm of residual chromatic dispersion compensation.
- Ability to transport of 100GE and OTU4 client rates on the same platform through software provisioning.
- 600G DWDM, providing unparalleled scale and density—64 channels of 600G at 75 GHz, delivering 38.4 Tbps in 16 RU.
- State-of-the-art AES-256 encryption at scale—4.8 Tbps of encrypted trunk capacity per 2 RU.

Interoperability constraints and timing requirements for Cisco NCS 1001 and Cisco NCS 1004

Understand the interoperability constraints and timing requirements between Cisco NCS 1001 and Cisco NCS 1004.

Interoperability constraints and requirements

- Traffic loss may occur under certain conditions when Cisco NCS 1001 with Protection Switching Module (PSM) is configured as non-revertive and operates with Cisco NCS 1004.
- If traffic switches from the working path to the protect path, users must wait 120 seconds before performing a manual switch to prevent traffic loss.
- A manual switch performed before the interval, followed by a protect path failure, can result in up to 13 seconds of traffic loss.

Timing limitations

The PSM switching time for the ONS-CFP2D-400G-C and DP04CFP2-M20-191 modules may exceed 50 milliseconds.

Note PSM switching with x50G is not supported as the switching time is more than 50 ms.

Supported line cards

A supported line card is a hardware module for the Cisco NCS 1004 platform that

- provides specific types and numbers of high-speed client interfaces,
- supports variable trunk port configurations for flexible bandwidth and transmission rates, and
- offers distinctive modulation, frequency, and error-correction capabilities tailored to deployment needs.

These line cards are supported on Cisco NCS 1004.

NCS1K4-1.2T-K9 C-band line card

The NCS1K4-1.2T-K9 (or 1.2 Tbps) C-band line card contains 12 QSFP-28 clients and two DWDM trunk ports. These trunk ports support multiple line rates and offer precise control of the modulation format, baud rate, and forward error correction. The trunk ports can be configured using software. The line card supports both module and slice configurations.

Note "1.2TC" refers to the NCS1K4-1.2T-K9 C-band line card.

The features of the 1.2T line card are:

- The card provides up to 12 100G or OTU4 client ports.

- The baud rate can be controlled between 28 Gbd/s and 72 Gbd/s.
- The frequency range is 191.25 to 196.1 THz with a default value of 193.1 THz.
- The modulation formats can be QPSK, 8 QAM, 16 QAM, 32 QAM, or 64 QAM.
- Configure hybrid modulation formats using 1/128 bits per symbol granularity.
- Forward Error Correction (FEC) of 27% and 15% overhead across line rates (only 15% for 600G).
- In Release 7.1.1, the trunk line rate can be configured from 150G to 600G in 50G increments.
- In Release 7.2.1 and later releases, the trunk line rate can be configured from 50G to 600G in 50G increments.

NCS1K4-1.2TL-K9 L-band line card

The NCS1K4-1.2TL-K9 (or 800 Gbps) L-band line card has 12 QSFP-28 based clients and two DWDM trunk ports. The trunk ports are capable of several line rates with fine control of modulation formats, baud rate, and forward error correction and are software configurable. The line card supports module and slice configurations.

Note "1.2TL" refers to the NCS1K4-1.2TL-K9 L-band line card.

Note There is no support for GMPLS, remote management using GCC, and smart licensing.

The features of the 1.2TL line card are:

- The card provides up to eight 100G or OTU4 client ports.
- The client ports map to two trunk ports that operate on any rate between 200G and 400G with 50G increments.
- The modulation formats can be controlled between QPSK, 8 QAM, and 16 QAM.
- The baud rate can be controlled between 31.5Gbd/s and 72Gbd/s.
- The frequency range is 186.10 to 190.85 THz with a default value of 188.50 THz. Only 100 MHz spacing is supported.
- Hybrid modulations formats can be configured through 1/128 bits/symbol granularity.
- Forward Error Correction (FEC) supports 15% and 27% overhead.

NCS1K4-OTN-XP line card

From R7.2.1, NCS 1004 supports the NCS1K4-OTN-XP card with 100G grey-optics support.

Note "OTN-XP" refers to the NCS1K4-OTN-XP line card.

The OTN-XP card contains:

- Eight QSFP 28 ports
- Four QSFP-DD ports
- Two CFP2 ports

The OTN-XP card supports up to 1.6Tbps of OTN aggregation switching functionality to optimize the available bandwidth. A single line card supports 8x100GE muxponder or 2x400 GE transponder applications. The OTN-XP card supports 400GE/OTUC4, 100GE/OTU4, 10GE/OTU2/OTU2e, 16G FC, and 32G FC client rates.

For more information on the mode configuration, see Muxponder Configuration on OTN-XP Card, on page 39.

NCS1K4-2-QDD-C-K9 C-band line card

From Release 7.3.1, NCS 1004 supports the NCS1K4-2-QDD-C-K9 (or 2-QDD-C) C-Band line card. The card has eight client ports (QSFP28 and QSFP-DD) and two DWDM dual sub-channel module trunk ports. The FR4 and AOC are the two optics supported on the 400GE client ports. Each trunk port is capable of 200, 300, and 400 Gbps line rate with fine control of modulation formats, baud rate, and forward error correction. The trunk ports are software configurable. The line card supports module and slice configurations.

Note "2-QDD-C" refers to the NCS1K4-2-QDD-C-K9 C-band line card.

The features of the 2-QDD-C line card are:

- The card provides up to eight 100 GE or two 400 GE client ports.
- The trunk line rate can be configured from 200G to 400G in 100G increments.

Note The trunk line rates of 250G and 350G are not supported in R7.3.1.

- The client to trunk port mapping is based on type of configuration and the line rate.
- The modulation formats can be controlled between QPSK, 8 QAM, and 16 QAM including hybrid modulation.
- Hybrid modulations formats can be configured through 1/128 bits/symbol granularity.
- Forward Error Correction (FEC) of Soft Decision FEC 27% and Soft Decision FEC 15%.
- The baud rate can be controlled between 28 Gbd/s and 72 Gbd/s.
- The frequency range is 191.25 to 196.1 THz with a default value of 193.1 THz.

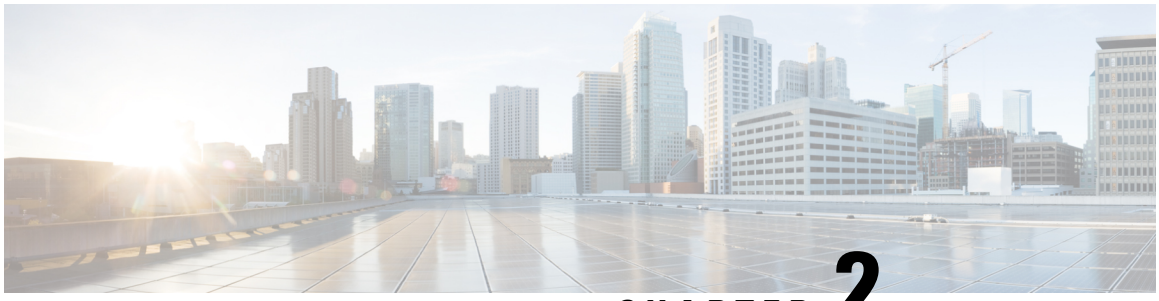
NCS1K4-QXP-K9 3.2T QSFP-DD DCO transponder line card

From Release 7.7.1, NCS 1004 supports the NCS1K4-QXP-K9 (or QXP) Line Card. The card has eight client ports (QSFP-DD) and eight trunk ports (QSFP-DD ZR+).

Note "QXP" refers to the NCS1K4-QXP-K9 C-band line card.

The features of the QXP line card are:

- Each line card supports up to 3.2 Tbps traffic.
- A single line card supports 6 slices of:
  - 4x100GE (breakout) clients in muxponder mode.
  - 400GE clients in transponder mode.
  - 100GE clients in transponder mode.
- The client rates that are supported are 400GE, 4x100GE, and 100GE Ethernet only.
- The modulation formats supported are 16 QAM for 400GE Txp/4x100GE Mxp and QPSK for 100GE Txp.



CHAPTER 2

Configuring the Card Mode

This chapter lists the supported configurations and the procedures to configure the card mode on the line cards.

Note Unless otherwise specified, “line cards” refers to 1.2T and 1.2TL line cards.

- 1.2T and 1.2TL Line Cards, on page 7
- OTN-XP Card, on page 23
- 2-QDD-C Line Card, on page 98
- QXP Card, on page 107

1.2T and 1.2TL Line Cards

The following section describes the supported configurations and procedures to configure the card modes on the line cards.

Card modes

A card mode is a configuration option for line cards that

- determines which trunk and client ports operate together,
- controls whether trunk rates are set globally or independently, and
- defines how client ports are mapped to trunk ports.

Each line card includes two trunk ports (0 and 1) and twelve client ports (2 through 13). You can configure each line card in one of two modes.

Muxponder mode

Both trunk ports (0 and 1) are configured with the same trunk rate. Client-to-trunk mapping is sequential and enables all client ports to use the combined bandwidth of the trunks efficiently.

Muxponder slice mode

Each trunk port (0 and 1) is configured independently, allowing different trunk rates for each. The client to trunk mapping is fixed. Client ports 2 through 7 map to trunk 0, and client ports 8 through 13 map to trunk 1.

Sub 50G configuration

A Sub 50G configuration is a line card setup that

- enables data transmission rates below 50 Gbps or coupled mode on muxponder line cards,
- allows flexible port assignments and mapping for various supported data rates, and
- imposes specific requirements and operational restrictions when configured.

Standard port configurations for Sub 50G data rates

You can configure Sub 50G or coupled mode on the line card only when operating in muxponder mode. This table shows the supported port configurations for various data rates.

Table 1: Port configuration for supported data rates

Trunk data rate (per trunk)	Total configured data rate	Card support	Trunk ports	Client ports for trunk 0 (100G)	Shared client port (50G per trunk)	Client ports for trunk 1 (100G)
50G	100G	1.2T, 1.2TL	0, 1	-	2	-
150G	300G	1.2T, 1.2TL	0, 1	2	3	4
250G	500G	1.2T , 1.2TL	0, 1	2, 3	4	5, 6
350G	700G	1.2T , 1.2TL	0, 1	2, 3, 4	5	6, 7, 8
450G	900G	1.2T	0, 1	2, 3, 4, 5	6	7, 8, 9, 10
550G	1.1T	1.2T	0, 1	2, 3, 4, 5, 6	7	8, 9, 10, 11, 12

Alternate port configurations (split client port mapping)

From Release 7.5.2, 1.2T and 1.2TL line cards support an alternate port configuration for Sub 50G (split client port mapping), which you configure using CLI. This table shows the alternate port mapping for the supported data rates:

Table 2: Alternate port configuration for Sub 50G (Release 7.5.2 and later)

Trunk data rate (per trunk)	Total configured data rate	Card support	Trunk ports	Client ports for trunk 0 (100G)	Shared client port (50G per trunk)	Client ports for trunk 1 (100G)
50G	100G	1.2T, 1.2TL	0, 1	-	7	-

Trunk data rate (per trunk)	Total configured data rate	Card support	Trunk ports	Client ports for trunk 0 (100G)	Shared client port (50G per trunk)	Client ports for trunk 1 (100G)
150G	300G	1.2T, 1.2TL	0, 1	2	7	8
250G	500G	1.2T , 1.2TL	0, 1	2, 3	7	8, 9
350G	700G	1.2T , 1.2TL	0, 1	2, 3, 4	7	8, 9, 10
450G	900G	1.2T	0, 1	2, 3, 4, 5	7	8, 9, 10, 11
550G	1.1T	1.2T	0, 1	2, 3, 4, 5, 6	7	8, 9, 10, 11, 12

Operational considerations

In all x50G configurations, client traffic on the middle port is affected by ODUK-BDI and LF alarms after a **power cycle** or **link flap** on the trunk side. This issue occurs when two network lanes operate in coupled mode and move from low to high power. To resolve this, create a new frame at the near-end or far-end by performing a **shut** or **no shut** of the trunk ports.

Restrictions for coupled mode configurations

These restrictions apply to coupled mode configuration:

- Both trunk ports must be configured with the same bits-per-symbol or baud rate and must be sent over the same fiber and direction.
- Chromatic dispersion values must be configured identically for both trunk ports.
- When trunk internal loopback is configured, it must be set for both trunk ports. Configuring internal loopback on only one trunk results in traffic loss.
- A fault on a trunk port of a coupled pair may cause errors on all clients, including those running only on the unaffected trunk port.

.

Configure Split Client Port Mapping

Table 3: Feature History

Feature Name	Release Information	Description
Split Client Port Mapping	Cisco IOS XR Release 7.5.2	A new trunk port to client port mapping for sub 50G configurations is now available on the 1.2T C band, 1.2T L band, and 800G QSFP-DD Transponder line cards. In this mapping, the same shared client port is used for all Sub 50G trunk data rates, eliminating recabling while changing the data rates.

You can configure the trunk port to client port mapping for sub 50G data ratesin the default mode or in the split client port mapping mode.

In the default mode, consecutive client ports carry the information. For example, on a 2-QDD-C card, if the trunk data rate per trunk is 150G, client ports 2, 3, and 4 carry the data and client port 3 is the shared client port. For a trunk data rate of 250G, client ports 2, 3, 4, 5, and 6 carry the data and client port 4 is the shared client port. However, if you configure split client port mapping, trunk port to client port mapping is fixed. The shared client port is client port 5 for 2-QDD-C card and client port 7 for 1.2T and 1.2TL cards.

To configure the split client port mapping, use the following commands.

configure

hw-module location *location* **mxponder**

split-client-port-mapping

commit

The following is a sample in which split-client-port-mapping is configured with a 450G trunk payload.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#split-client-port-mapping
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#end
```

To remove the split client port-mapping configuration and configure default client port mapping, use the following commands.

configure

hw-module location *location* **mxponder**

no split-client-port-mapping

commit

The following is a sample in which split client port-mapping configuration is removed.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#no split-client-port-mapping
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#end
```

Verifying the Port Mapping Configuration

The following is a sample ouput of the split client port-mapping.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder

Location:             0/1
Client Bitrate:       100GE
Trunk  Bitrate:       450G
Status:               Provisioning In Progress
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/0
CoherentDSP0/1/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/2         ODU40/1/0/0/1                         100
       0
HundredGigECtrlr0/1/0/3         ODU40/1/0/0/2                         100
       0
HundredGigECtrlr0/1/0/4         ODU40/1/0/0/3                         100
       0
HundredGigECtrlr0/1/0/5         ODU40/1/0/0/4                         100
       0
HundredGigECtrlr0/1/0/7         ODU40/1/0/0/5                          50
      50
HundredGigECtrlr0/1/0/8         ODU40/1/0/1/1                           0
     100
HundredGigECtrlr0/1/0/9         ODU40/1/0/1/2                           0
     100
HundredGigECtrlr0/1/0/10        ODU40/1/0/1/3                           0
     100
HundredGigECtrlr0/1/0/11        ODU40/1/0/1/4                           0
     100
```

The following is a sample ouput of the default client port mapping.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder

Location:             0/1
Client Bitrate:       100GE
Trunk  Bitrate:       450G
Status:               Provisioning In Progress
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/0
CoherentDSP0/1/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/2         ODU40/1/0/0/1                         100
       0
HundredGigECtrlr0/1/0/3         ODU40/1/0/0/2                         100
       0
HundredGigECtrlr0/1/0/4         ODU40/1/0/0/3                         100
       0
HundredGigECtrlr0/1/0/5         ODU40/1/0/0/4                         100
       0
HundredGigECtrlr0/1/0/6         ODU40/1/0/0/5                          50
      50
HundredGigECtrlr0/1/0/7         ODU40/1/0/1/1                           0
     100
```

```
HundredGigECtrlr0/1/0/8          ODU40/1/0/1/2                            0
     100
HundredGigECtrlr0/1/0/9          ODU40/1/0/1/3                            0
     100
HundredGigECtrlr0/1/0/10         ODU40/1/0/1/4                            0
     100
```

Supported Data Rates

The following data rates are supported on the line card.

In R7.0.1, you can configure the client port to OTU4 only in the muxponder mode. In R7.1.1 and later releases, you can configure the client port to OTU4 in both the muxponder and muxponder slice modes. In muxponder slice mode, both the slices must be configured with either OTU4 or 100GE Ethernet client rates in R7.1.1. In R7.2.0, a mixed configuration of OTU4 and 100GE is supported in the muxponder slice mode. LLDP drop, L1 encryption, and AINS are not supported on the OTU4 configuration.

The following table displays the client and trunk ports that are enabled for the muxponder configuration.

Trunk Data Rate	Card Support	Client Data Rate (100GE, OTU4)	Trunk Ports	Client Ports
100	1.2T, 1.2TL	100GE, OTU4	0, 1	2, 3
200	1.2T, 1.2TL	100GE, OTU4	0, 1	2, 3, 4, 5
300	1.2T, 1.2TL	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7
400	1.2T, 1.2TL	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9
500	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11
600	1.2T	100GE, OTU4	0, 1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

The following table displays the client and trunk ports that are enabled for the muxponder slice 0 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	1.2T, 1.2TL	100GE, OTU4	0	2
200	1.2T, 1.2TL	100GE, OTU4	0	2, 3
300	1.2T, 1.2TL	100GE, OTU4	0	2, 3, 4
400	1.2T, 1.2TL	100GE, OTU4	0	2, 3, 4, 5
500	1.2T	100GE, OTU4	0	2, 3, 4, 5, 6
600	1.2T	100GE, OTU4	0	2, 3, 4, 5, 6, 7

The following table displays the client and trunk ports that are enabled for the muxponder slice 1 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	1.2T, 1.2TL	100GE, OTU4	1	8
200	1.2T, 1.2TL	100GE, OTU4	1	8, 9
300	1.2T, 1.2TL	100GE, OTU4	1	8, 9, 10
400	1.2T, 1.2TL	100GE, OTU4	1	8, 9, 10, 11
500	1.2T	100GE, OTU4	1	8, 9, 10, 11, 12
600	1.2T	100GE, OTU4	1	8, 9, 10, 11, 12, 13

All configurations can be accomplished by using appropriate values for client bitrate and trunk bitrate parameters of the **hw-module** command.

The following table displays the trunk parameter ranges for the 1.2T card.

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
50G	15%	1	1.3125	24.0207911	31.5272884
50G	27%	1	1.4453125	24.0207911	34.7175497
100G	15%	1	2.625	24.0207911	63.0545768
100G	27%	1	2.890625	24.0207911	69.4350994
150G	15%	1.3203125	3.9375	24.0207911	71.6359689
150G	27%	1.453125	4.3359375	24.0207911	71.6749413
200G	15%	1.7578125	5.25	24.0207911	71.7420962
200G	27%	2	4.40625	31.51	69.43
250G	15%	2.1953125	6	26.2727403	71.8059237
250G	27%	2.4140625	6	28.9312914	71.9068991
300G	15%	2.6328125	6	31.5272884	71.8485385
300G	27%	2.8984375	6	34.7175497	71.8681352
350G	15%	3.0703125	6	36.7818364	71.8790086
350G	27%	3.3828125	6	40.503808	71.8404724
400G	15%	3.5078125	6	42.0363845	71.9018782
400G	27%	3.8671875	6	46.2900663	71.8197392
450G	15%	3.9453125	6	47.2909326	71.9196757
450G	27%	4.34375	6	52.0763245	71.9327648

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
500G	15%	4.3828125	6	52.5454806	71.93392
500G	27%	4.8281250	6	57.8625828	71.9068991
550G	15%	4.8203125	6	57.8000287	71.9455787
550G	27%	5.3125	6	63.6488411	71.88575
600G	15%	5.2578125	-	-	71.9552971

The following table displays the trunk parameter ranges for the 1.2TL card.

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
100G	15%	1	2.625	24.0207911	63.0545768
100G	27%	1	2.890625	24.0207911	69.4350994
150G	15%	1.3203125	3.9375	24.0207911	71.6359689
150G	27%	1.453125	4.3359375	24.0207911	71.6749413
200G	15%	2	4	31.5272884	63.0545768
200G	27%	2	4.40625	31.51664088	69.43509943
250G	15%	2.1953125	4.5	35.0303204	71.8059237
250G	27%	2.4140625	4.5	38.5750552	71.9068991
300G	15%	2.6328125	4.5	42.0363845	71.8485385
300G	27%	2.8984375	4.5	46.2900662857142	71.86813526
350G	15%	3.0703125	4.5	49.0424486	71.8790086
350G	27%	3.3828125	4.5	54.0050773	71.8404724
400G	15%	3.5078125	4.5	56.0485127	71.9018782
400G	27%	3.8671875	4.5	61.72008838	71.81973921

To configure the BPS, see Configuring the BPS, on page 20.

Configuring the Card Mode

You can configure the line card in the module (muxponder) or slice configuration (muxponder slice).

To configure the card in the muxponder mode, use the following commands.

configure

hw-module location *location* **mxponder client-rate {100GE | OTU4}**

hw-module location *location* **mxponder trunk-rate** {**50G** | **100G150G** | **200G** | **250G** | **300G** | **350G** | **400G** | **450G** | **500G** | **550G** | **600G** }

commit

To configure the card in the muxponder slice mode, use the following commands.

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number* **client-rate { 100GE|OTU4}**

hw-module location *location* **mxponder-slice trunk-rate** { **100G** | **200G** | **300G** | **400G** | **500G** | **600G** }

commit

Examples

The following is a sample in which the card is configured in the muxponder mode with a 550G trunk payload.

```
RP/0/RP0/CPU0:ios#config
Tue Oct 15 01:24:56.355 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder trunk-rate 550G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder mode with a 500G trunk payload.

```
RP/0/RP0/CPU0:ios#config
Sun Feb 24 14:09:33.989 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder client-rate OTU4
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder trunk-rate 500G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 0 mode with a 500G trunk payload.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 trunk-rate 500G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 1 mode with a 400G trunk payload.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured with mixed client rates in the muxponder slice mode.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 23 06:10:22.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-rate OTU4 trunk-rate
 500G
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 client-rate 100GE trunk-rate
 500G
RP/0/RP0/CPU0:ios(config)#commit
```

Verifying the Card Configuration

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder
Fri Mar 15 11:48:48.344 IST

Location:             0/2
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port            Mapper/Trunk Port   CoherentDSP0/2/0/0  CoherentDSP0/2/0/1
                     Traffic Split Percentage

HundredGigECtrlr0/2/0/2  ODU40/2/0/0/1               100                   0
HundredGigECtrlr0/2/0/3  ODU40/2/0/0/2               100                   0
HundredGigECtrlr0/2/0/4  ODU40/2/0/0/3               100                   0
HundredGigECtrlr0/2/0/5  ODU40/2/0/0/4               100                   0
HundredGigECtrlr0/2/0/6  ODU40/2/0/0/5               100                   0
HundredGigECtrlr0/2/0/7  ODU40/2/0/1/1                 0                 100
HundredGigECtrlr0/2/0/8  ODU40/2/0/1/2                 0                 100
HundredGigECtrlr0/2/0/9  ODU40/2/0/1/3                 0                 100
HundredGigECtrlr0/2/0/10 ODU40/2/0/1/4                 0                 100
HundredGigECtrlr0/2/0/11 ODU40/2/0/1/5                 0                 100
```

The following is a sample ouput of the coupled mode configuration where the shared client port is highlighted.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Tue Oct 15 01:25:57.358 UTC

Location:             0/1
Client Bitrate:       100GE
Trunk  Bitrate:       550G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port           Mapper/Trunk Port    CoherentDSP0/1/0/0 CoherentDSP0/1/0/1
                   Traffic Split Percentage

HundredGigECtrlr0/1/0/2    ODU40/1/0/0/1             100                   0
HundredGigECtrlr0/1/0/3    ODU40/1/0/0/2             100                   0
HundredGigECtrlr0/1/0/4    ODU40/1/0/0/3             100                   0
HundredGigECtrlr0/1/0/5    ODU40/1/0/0/4             100                   0
HundredGigECtrlr0/1/0/6    ODU40/1/0/0/5             100                   0
HundredGigECtrlr0/1/0/7    ODU40/1/0/0/6              50                  50
HundredGigECtrlr0/1/0/8    ODU40/1/0/1/1               0                 100
HundredGigECtrlr0/1/0/9    ODU40/1/0/1/2               0                 100
HundredGigECtrlr0/1/0/10   ODU40/1/0/1/3               0                 100
HundredGigECtrlr0/1/0/11   ODU40/1/0/1/4               0                 100
HundredGigECtrlr0/1/0/12   ODU40/1/0/1/5               0                 100
```

The following is a sample ouput of all the muxponder slice 0 configurations.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice  0
Fri Mar 15 06:04:18.348 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/1/0/0
                               Traffic Split Percentage

HundredGigECtrlr0/1/0/2        ODU40/1/0/0/1                     100
HundredGigECtrlr0/1/0/3        ODU40/1/0/0/2                     100
```

```
HundredGigECtrlr0/1/0/4         ODU40/1/0/0/3                       100
HundredGigECtrlr0/1/0/5         ODU40/1/0/0/4                       100
HundredGigECtrlr0/1/0/6         ODU40/1/0/0/5                       100
```

The following is a sample ouput of all the muxponder slice 1 configurations.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice 1
Fri Mar 15 06:11:50.020 UTC

Location:             0/1
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    TRUE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/8         ODU40/1/0/1/1                       100
HundredGigECtrlr0/1/0/9         ODU40/1/0/1/2                       100
HundredGigECtrlr0/1/0/10        ODU40/1/0/1/3                       100
HundredGigECtrlr0/1/0/11        ODU40/1/0/1/4                       100
```

The following is a sample ouput of the muxponder slice 1 configuration with client configured as OTU4.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/0 mxponder-slice 1

Wed Mar 11 13:59:11.073 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       OTU4
Trunk  Bitrate:       200G
Status:               Provisioned
Client Port                     Peer/Trunk Port            CoherentDSP0/0/0/1
                              Traffic Split Percentage
OTU40/0/0/8                     ODU40/0/0/1/1                       100
OTU40/0/0/9                     ODU40/0/0/1/2                       100
```

The following is a sample to verify the mixed client rate configuration in the muxponder slice mode.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Mon Mar 23 06:20:22.227 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       OTU4
Trunk  Bitrate:       500G
Status:               Provisioned
Client Port                     Peer/Trunk Port            CoherentDSP0/1/0/0
                                Traffic Split Percentage

OTU40/1/0/2                     ODU40/1/0/0/1                       100
OTU40/1/0/3                     ODU40/1/0/0/2                       100
OTU40/1/0/4                     ODU40/1/0/0/3                       100
OTU40/1/0/5                     ODU40/1/0/0/4                       100
OTU40/1/0/6                     ODU40/1/0/0/5                       100


Location:             0/1
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
```

```
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/1/0/1
                               Traffic Split Percentage

HundredGigECtrlr0/1/0/8        ODU40/1/0/1/1                         100
HundredGigECtrlr0/1/0/9        ODU40/1/0/1/2                         100
HundredGigECtrlr0/1/0/10       ODU40/1/0/1/3                         100
HundredGigECtrlr0/1/0/11       ODU40/1/0/1/4                         100
HundredGigECtrlr0/1/0/12       ODU40/1/0/1/5                         100
```

Use the following command to clear alarm statistics on the optics or coherent DSP controller.

clear counters controller *controllertype R/S/I/P*

The following is a sample in which the alarm statistics are cleared on the coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controller coherentDSP 0/1/0/0
Tue Jun 11 05:15:12.540 UTC

Port                                        : CoherentDSP 0/1/0/0
Controller State                            : Up
Inherited Secondary State                   : Normal
Configured Secondary State                  : Normal
Derived State                               : In Service
Loopback mode                               : None
BER Thresholds                              : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                      : Enable

Alarm Information:
LOS = 1 LOF = 1 LOM = 0
OOF = 1 OOM = 1 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 2      BDI = 2 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                             : None

Bit Error Rate Information
PREFEC  BER                                 : 8.8E-03
POSTFEC BER                                 : 0.0E+00

TTI :
        Remote hostname                     : P2B8
        Remote interface                    : CoherentDSP 0/1/0/0
        Remote IP addr                      : 0.0.0.0

FEC mode                                    : Soft-Decision 15

AINS Soak                                   : None
AINS Timer                                  : 0h, 0m
AINS remaining time                         : 0 seconds
RP/0/RP0/CPU0:ios#clear counters controller coherentDSP 0/1/0/0
Tue Jun 11 05:17:07.271 UTC
All counters are cleared
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/1
Tue Jun 11 05:20:55.199 UTC

Port                                        : CoherentDSP 0/1/0/1
Controller State                            : Up
Inherited Secondary State                   : Normal
Configured Secondary State                  : Normal
Derived State                               : In Service
Loopback mode                               : None
BER Thresholds                              : SF = 1.0E-5  SD = 1.0E-7
```

```
Performance Monitoring                            : Enable

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                   : None

Bit Error Rate Information
PREFEC  BER                                       : 1.2E-02
POSTFEC BER                                       : 0.0E+00

TTI :
        Remote hostname                           : P2B8
        Remote interface                          : CoherentDSP 0/1/0/1
        Remote IP addr                            : 0.0.0.0

FEC mode                                          : Soft-Decision 15

AINS Soak                                         : None
AINS Timer                                        : 0h, 0m
AINS remaining time                               : 0 seconds
```

Regeneration Mode

In an optical transmission system, 3R regeneration helps extend the reach of the optical communication links by reamplifying, reshaping, and retiming the data pulses. Regeneration helps to correct any distortion of optical signals by converting it to an electrical signal, processing that electrical signal, and then retransmitting it again as an optical signal.

In Regeneration (Regen) mode, the OTN signal is received on a trunk port and the regenerated OTN signal is sent on the other trunk port of the line card and the other way round. In this mode, only the trunk optics controller and coherentDSP controllers are created.

Configuring the Card in Regen Mode

The supported trunk rates for the different cards are:

- 1.2T card—100G to 600G in multiples of 100G
- 1.2TL card—200G to 400G in multiples of 100G
- 2-QDD-C card—200G to 400G in multiples of 100G

To configure regen mode on 1.2T, 1.2TL, and 2-QDD-C cards, use the following commands:

configure

hw-module location *location*

regen

trunk-rate *trunk-rate*

commit

exit

Example

The following is a sample to configure the regen mode on 1.2T, 1.2TL, and 2-QDD-C line cards with the trunk-rate 300.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0
RP/0/RP0/CPU0:ios(config-hwmod)#regen
RP/0/RP0/CPU0:ios(config-regen)#trunk-rate 300
RP/0/RP0/CPU0:ios(config-regen)#commit
RP/0/RP0/CPU0:ios(config-regen)#exit
```

Verifying the Regen Mode

The following is a sample to verify the regen mode.

show hw-module location *location* **regen**

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 regen
Mon Mar 25 09:50:42.936 UTC

Location:             0/0
Trunk  Bitrate:       400G
Status:               Provisioned
East Port              West Port
CoherentDSP0/0/0/0      CoherentDSP0/0/0/1
```

The terms, East Port and West Port are used to represent OTN signal regeneration at the same layer.

Configuring the BPS

You can configure the Bits per Symbol (BPS) to 3.4375 to support 300G trunk configurations on 75 GHz networks using the following commands:

configure

controller optics *R/S/I/P* **bits-per-symbol 3.4375**

commit

The following is a sample in which the BPS is configured to 3.4375.

```
RP/0/RP0/CPU0:ios#configure
Wed Mar 27 14:12:49.932 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/3/0/0 bits-per-symbol 3.4375
RP/0/RP0/CPU0:ios(config)#commit
```

Viewing BPS and Baud Rate Ranges

To view the the BPS for a specific range use the following command:

show controller optics *R/S/I/P* **bps-range** *bps-range* **| include** *data-rate* **| include** *fec-type*

```
RP/0/RP0/CPU0:ios#show controllers optics 0/3/0/0 bps-range 3 3.05 | include 300G | include
 SD27
Thu Mar 28 03:01:39.751 UTC
300G            SD27            3.0000000       69.4350994
300G            SD27            3.0078125       69.2547485
300G            SD27            3.0156250       69.0753320
300G            SD27            3.0234375       68.8968428
300G            SD27            3.0312500       68.7192736
```

```
300G            SD27            3.0390625       68.5426174
300G            SD27            3.0468750       68.3668671
```

To view the baud for a specific range use the following command:

show controller optics *R/S/I/P* **baud-rate-range** *baud-range* **| include** *data-rate* **| include** *fec-type*

```
RP/0/RP0/CPU0:ios#show controllers optics 0/3/0/0 baud-rate-range 43 43.4 | include 300G |
 include SD27
Thu Mar 28 03:12:36.521 UTC
300G            SD27            4.8046875       43.3545986
300G            SD27            4.8125000       43.2842178
300G            SD27            4.8203125       43.2140651
300G            SD27            4.8281250       43.1441394
300G            SD27            4.8359375       43.0744397
300G            SD27            4.8437500       43.0049648
```

Configuring the Trunk Rate for BPSK

From R7.2.1 onwards, you can configure trunk rates of 50G, 100G, and 150G to support Binary Phase-Shift Keying (BPSK) modulation. The BPSK modulation enables information to be carried over radio signals more efficiently.

You can configure trunk rates for BPSK using CLI, NetConf YANG, and OC models.

The following table list the 50G, 100G, and 150G trunk rates with the supported BPSK modulation:

Trunk Rate	BPSK Modulation
50G	1 to 1.4453125
100G	1 to 2.890625
150G	1.453125 to 4.3359375

To configure the trunk rate for BPSK modulation, enter the following commands:

configure

hw-module location *location* **mxponder**

trunk-rate {50G | 100G | 150G}

commit

The following example shows how to configure trunk rate to 50G:

```
RP/0/RP0/CPU0:(config)#hw-module location 0/0 mxponder
RP/0/RP0/CPU0:(config-hwmod-mxp)#trunk-rate 50G
RP/0/RP0/CPU0:(config-hwmod-mxp)#commit
```

Viewing the BPSK Trunk Rate Ranges

To view the trunk rate configured for the BPSK modulation, use the following **show** commands:

```
RP/0/RP0/CPU0:ios(hwmod-mxp)#show hw-module location 0/0 mxponder

Tue Feb 25 11:13:41.934 UTC
```

```
Location:             0/0
Client Bitrate:       100GE
Trunk  Bitrate:       50G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/0
CoherentDSP0/0/0/1
                                Traffic Split Percentage


HundredGigECtrlr0/0/0/2         ODU40/0/0/0                              50
      50


RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/0
Thu Mar  5 07:12:55.681 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Green

 Optics Status

        Optics Type:  DWDM optics
        DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
        Wavelength=1552.524nm

        Alarm Status:
        -------------
        Detected Alarms: None


        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 2
        HIGH-TX-PWR = 0            LOW-TX-PWR = 0
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 1.97 dBm
        RX Power = 1.58 dBm
        RX Signal Power = 0.60 dBm
        Frequency Offset = 386 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          4.9      -12.0           0.0          0.0
        Tx Power Threshold(dBm)          3.5      -10.1           0.0          0.0
```

```
LBC Threshold(mA)                N/A        N/A          0.00         0.00

Configured Tx Power = 2.00 dBm
Configured CD High Threshold = 180000 ps/nm
Configured CD lower Threshold = -180000 ps/nm
Configured OSNR lower Threshold = 0.00 dB
Configured DGD Higher Threshold = 180.00 ps
Baud Rate =  34.7175521851 GBd
Bits per Symbol = 1.0000000000  bits/symbol
Modulation Type: BPSK
Chromatic Dispersion -9 ps/nm
Configured CD-MIN -180000 ps/nm  CD-MAX 180000 ps/nm
Polarization Mode Dispersion = 0.0 ps
Second Order Polarization Mode Dispersion = 125.00 ps^2
Optical Signal to Noise Ratio = 34.60 dB
SNR = 20.30 dB
Polarization Dependent Loss = 0.20 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 2.00 ps
Filter Roll Off Factor : 0.100
Rx VOA Fixed Ratio : 15.00 dB
Enhanced Colorless Mode : 0
Enhanced SOP Tolerance Mode : 0
NLEQ Compensation Mode : 0
Cross Polarization Gain Mode : 0
Cross Polarization Weight Mode : 0
Carrier Phase Recovery Window : 0
Carrier Phase Recovery Extended Window : 0


AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

OTN-XP Card

The following section describes the supported configurations and procedures to configure the card modes on the line card.

LC Mode on OTN-XP Card

When you install the OTN-XP card in the Cisco NCS 1004 chassis, it is in the POWERED_ON state. The **LCMODE is not configured for line card** alarm is present on the card and the LED status is AMBER.

```
sysadmin-vm:0_RP0# show platform
Thu Mar  26 21:38:07.305 UTC+00:00
Location  Card Type               HW State      SW State      Config State
--------------------------------------------------------------------------------
0/0       NCS1K4-LC-FILLER        PRESENT       N/A           NSHUT
0/1       NCS1K4-OTN-XP           POWERED_ON    N/A           NSHUT
0/RP0     NCS1K4-CNTLR-K9         OPERATIONAL   OPERATIONAL   NSHUT
0/FT0     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/FT1     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/FT2     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/PM0     NCS1K4-AC-PSU           OPERATIONAL   N/A           NSHUT
0/SC0     NCS1004                 OPERATIONAL   N/A           NSHUT

sysadmin-vm:0_RP0# show alarms brief system active
Thu Mar  26 21:38:34.394 UTC+00:00
```

```
-------------------------------------------------------------------------------
Active Alarms
-------------------------------------------------------------------------------
Location          Severity      Group           Set time            Description
-------------------------------------------------------------------------------
0                 major         environ         03/26/20 20:23:11   Power Module redundancy
 lost.
0                 critical      environ         03/26/20 20:23:29   Fan: One or more LCs
missing, running fans at max speed.
0/1               not_alarmed   shelf           03/26/20 21:38:26   LCMODE is not configured
 for line card
sysadmin-vm:0_RP0#
```

```
sysadmin-vm:0_RP0# show led location 0/1
Thu Mar  26 21:39:05.101 UTC+00:00
============================================================
Location  LED Name                       Mode          Color
============================================================
0/1
          0/1-Status LED                 WORKING       AMBER
sysadmin-vm:0_RP0#
```

You must select a datapath mode by configuring the LC mode. OTN-XP does not have a default LC mode. After the LC mode is configured using the CLI,the card transitions to the OPERATIONAL state, the alarm clears, and the LED status turns to GREEN.

The LC modes supported on the OTN-XP card are:

- 10G-GREY-MXP
- 4x100G-MXP-400G-TXP
- 40x10G-4x100G-MXP
- 4x100GE-MXP-DD
- 400GE-TXP-DD
- FC-MXP
- OTUCn-REGEN
- 2x100GE-TXP-MXP

Note 100G-TXP LC mode is not supported.

Only one LC mode can be configured on the OTN-XP card at a time. When the LC mode is changed using the CLI, the **LCMODE changed, delete the datapath config and reload line card** alarm is present on the card and the DP FPD is in disabled state. To clear the alarm and enable the DP FPD, delete the the existing datapath configuration and reload the line card to apply the new LC mode to make the card operational.

If a LC mode requires a different FPGA configuration, and the package is not available, the **OTN_XP_DP_FPD_PKG is missing, please install the package to proceed** alarm is present on the card. To clear the alarm, install the OTN_XP_DP_FPD_PKG file. After the package installation is complete, the required FPGA image is copied from the OTN_XP_DP_FPD_PKG file to the card, the card is automatically reloaded, and the card becomes operational.

Note The LC mode configuration is a shared plane configuration. The configuration does not enter the preconfigured state when the line card is not available.

Configuring the LC Mode

Note

- Ensure the OTN_XP_DP_FPD_PKG file is installed before configuring the LC mode.
- When you insert an OTN-XP line card having a lower FPD version, you must configure a LC mode which is supported on the software release that the line card is loaded with. You cannot upgrade the FPD of a line card if you configure a LC mode supported only in a higher software release.
- The LC_CPU_MOD_FW version on a new OTN-XP line card is 7.3.1. Support for new LC modes or features from version 7.5.1 or higher, such as OTUCn-REGEN mode in 7.5.2, is not available in this line card software. When you install an OTN-XP card for the first time in an NCS 1004 chassis with the controller card software version of 7.5.1 or higher, you must upgrade LC_CPU_MOD_FW, to ensure the availability and support for the LC modes or features that are supported supported in the XR software version. You must configure an LC mode which is supported in the 7.3.1 XR software, such as 4x100G-MXP-400G-TXP, and bring the card to OPERATIONAL state to upgrade the line card software.

To configure the LC mode on the OTN-XP card, use the following commands:

configure

lc-module location *location* **lcmode** *mode*

commit

Example

To view the LC modes available on the OTN-XP card, use the following command:

```
RP/0/RP0/CPU0:ios#sh lc-module location 0/0 lcmode all
Wed Sep 29 14:41:51.487 UTC
States: A-Available     R-Running       C-Configured

Node    Lcmode_Supported    Owner    Options(State)                HW_Ver
-------------------------------------------------------------------------------
0/0     Yes                 None     10G-GREY-MXP (A)               3.0
                                     4x100G-MXP-400G-TXP (A)        2.0
```

From Release 7.3.2 onwards, you can view the hybrid mode options that are supported on the OTN-XP card. To view the LC modes supported on the OTN-XP card, use the following command:

```
RP/0/RP0/CPU0:ios#show lc-module location 0/3 lcmode all
Wed Aug 11 17:06:29.538 UTC
States: A-Available     R-Running       C-Configured

Node    Lcmode_Supported    Owner    Options(State)                HW_Ver
-------------------------------------------------------------------------------
0/3     Yes                 CLI      10G-GREY-MXP (A)               3.0
                                     4x100G-MXP-400G-TXP (A)        2.0
```

```
                                    40x10G-4x100G-MXP (A)          3.0
                                    4x100GE-MXP-DD (R/C)           7.0
```

Note The 100G-TXP mode is listed when using the **show lc-module location lcmode all** command, but the configuration on 100G-TXP mode is not supported.

The following is a sample in which the OTN-XP card is configured in the 10G-GREY-MXP mode.

```
RP/0/RP0/CPU0:ios#configure
Thu Mar 26 21:40:51.495 UTC
RP/0/RP0/CPU0:ios(config)#lc-module location 0/1 lcmode 10G-GREY-MXP
RP/0/RP0/CPU0:ios(config)#commit
```

Verifying the LC Mode Configuration

The following is a sample output of a successful 10G-GREY-MXP LC mode configuration after which the card transitions to the OPERATIONAL state, the alarm clears, and the LED status turns to GREEN.

```
RP/0/RP0/CPU0:ios(config)#do show platform
Thu Mar 26 21:41:17.206 UTC
Node               Type                         State             Config state
--------------------------------------------------------------------------------
0/0                NCS1K4-LC-FILLER             PRESENT           NSHUT
0/1                NCS1K4-OTN-XP                OPERATIONAL       NSHUT
0/RP0/CPU0         NCS1K4-CNTLR-K9(Active)      IOS XR RUN        NSHUT
0/FT0              NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT1              NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT2              NCS1K4-FAN                   OPERATIONAL       NSHUT
0/PM0              NCS1K4-AC-PSU                OPERATIONAL       NSHUT
0/SC0              NCS1004                      OPERATIONAL       NSHUT
RP/0/RP0/CPU0:ios(config)#do show alarms brief system active
Thu Mar 26 21:41:29.641 UTC

------------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------------
Location        Severity     Group            Set Time                    Description



------------------------------------------------------------------------------------
0               Major        Environ          03/26/2020 20:23:11 UTC     Power Module
redundancy lost.


0               Critical     Environ          03/26/2020 20:23:29 UTC     Fan: One or more
LCs missing, running fans at max speed.

RP/0/RP0/CPU0:ios(config)#end
RP/0/RP0/CPU0:ios#show lc-module location 0/1 lcmode all
Thu Mar 26 21:41:58.780 UTC
States: A-Available     R-Running       C-Configured

Node    Lcmode_Supported    Owner    Options(State)                HW_Ver
--------------------------------------------------------------------------------
0/1     Yes                 CLI      10G-GREY-MXP (R/C)             3.0
                                     4x100G-MXP-400G-TXP (A)        2.0
RP/0/RP0/CPU0:ios#show lc-module location 0/1 lcmode
Thu Mar 26 21:42:18.997 UTC
```

```
Node    Lcmode_Supported    Owner    Running                Configured
------------------------------------------------------------------------------
0/1     Yes                 CLI      10G-GREY-MXP           10G-GREY-MXP
RP/0/RP0/CPU0:ios#admin
Thu Mar 26 21:42:38.525 UTC

root connected from 192.0.2.3 using ssh on sysadmin-vm:0_RP0
sysadmin-vm:0_RP0# show led location 0/1
Thu Mar  26 21:42:45.337 UTC+00:00
============================================================
Location  LED Name                      Mode         Color
============================================================
0/1
          0/1-Status LED                WORKING      GREEN
```

Example

The following is a sample in which the LC mode is changed from 10G-GREY-MXP to the 4x100G-MXP-400G-TXP mode. In this sample, the datapath configuration is deleted and the card is reloaded to apply the new LC mode.

```
RP/0/RP0/CPU0:ios#show lc-module location all lcmode
Thu Sep 30 10:19:29.853 UTC

Node    Lcmode_Supported    Owner    Running                Configured
------------------------------------------------------------------------------
0/0     Yes                 CLI      10G-GREY-MXP           10G-GREY-MXP
0/1     No                  N/A      N/A                    N/A
0/2     No                  N/A      N/A                    N/A
0/3     No                  N/A      N/A                    N/A


RP/0/RP0/CPU0:ios#configure
Thu Sep 30 10:19:32.818 UTC
Current Configuration Session  Line       User     Date                      Lock
00001000-000051f7-00000000     vty1       root     Wed Sep 29 15:26:00 2021
RP/0/RP0/CPU0:ios(config)#no lc-module location 0/0 lcmode 10g-GREY-MXP
RP/0/RP0/CPU0:ios(config)#commit
Thu Sep 30 10:20:34.086 UTC
RP/0/RP0/CPU0:ios(config)#do show alarms brief system active
Thu Sep 30 10:20:52.950 UTC

------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------
Location        Severity     Group            Set Time                    Description



------------------------------------------------------------------------------
0/PM0           Major        Environ          09/29/2021 14:41:59 UTC    Power Module Output
 Disabled


0               Major        Environ          09/29/2021 14:42:15 UTC    Power Module
redundancy lost.


0               Critical     Environ          09/29/2021 14:42:25 UTC    Fan: One or more
LCs missing, running fans at max speed.
```

```
0/0             NotAlarmed   Shelf            09/30/2021 10:20:34 UTC    LCMODE changed,
delete the datapath config and reload line card



RP/0/RP0/CPU0:ios#configure
Thu Sep 30 10:21:41.281 UTC
Current Configuration Session  Line       User     Date                       Lock
00001000-000051f7-00000000     vty1       root     Wed Sep 29 15:26:00 2021
RP/0/RP0/CPU0:ios(config)#no hw-module location 0/0
RP/0/RP0/CPU0:ios(config)#commit
Thu Sep 30 10:21:49.982 UTC
RP/0/RP0/CPU0:ios(config)#

RP/0/RP0/CPU0:ios#show platform
Thu Sep 30 10:22:08.482 UTC
Node              Type                         State             Config state
--------------------------------------------------------------------------------
0/0               NCS1K4-OTN-XP                OPERATIONAL       NSHUT
0/2               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/3               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/RP0/CPU0        NCS1K4-CNTLR-K9(Active)      IOS XR RUN        NSHUT
0/FT0             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT1             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT2             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/PM0             NCS1K4-AC-PSU                OPERATIONAL       NSHUT
0/SC0             NCS1004                      OPERATIONAL       NSHUT
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#admin
Thu Sep 30 10:23:55.937 UTC
Last login: Thu Sep 30 04:32:57 2021 from 192.0.2.3
root connected from 192.0.2.3 using ssh on sysadmin-vm:0_RP0
sysadmin-vm:0_RP0# hw-module location 0/0 reload
Thu Sep  30 10:24:17.938 UTC+00:00
Reloading the module will be traffic impacting if not properly drained. Continue to Reload
 hardware module ? [no,yes] yes
result Card graceful reload request on 0/0 succeeded.
sysadmin-vm:0_RP0#show platform
Thu Sep  30 10:25:16.876 UTC+00:00
Location  Card Type               HW State      SW State      Config State
--------------------------------------------------------------------------------
0/0       NCS1K4-OTN-XP           POWERED_ON    N/A           NSHUT
0/2       NCS1K4-LC-FILLER        PRESENT       N/A           NSHUT
0/3       NCS1K4-LC-FILLER        PRESENT       N/A           NSHUT
0/RP0     NCS1K4-CNTLR-K9         OPERATIONAL   OPERATIONAL   NSHUT
0/FT0     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/FT1     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/FT2     NCS1K4-FAN              OPERATIONAL   N/A           NSHUT
0/PM0     NCS1K4-2KW-AC           OPERATIONAL   N/A           NSHUT
0/SC0     NCS1004-K9              OPERATIONAL   N/A           NSHUT

sysadmin-vm:0_RP0#exit
RP/0/RP0/CPU0:ios#show lc-module location all lcmode
Thu Sep 30 10:29:08.183 UTC

Node    Lcmode_Supported    Owner    Running                Configured
--------------------------------------------------------------------------------
0/0     Yes                 None     Not running            Not configured
0/1     No                  N/A      N/A                    N/A
0/2     No                  N/A      N/A                    N/A
0/3     No                  N/A      N/A                    N/A
```

```
RP/0/RP0/CPU0:ios#show platform
Thu Sep 30 10:29:36.075 UTC
Node              Type                         State             Config state
--------------------------------------------------------------------------------
0/0               NCS1K4-OTN-XP                POWERED_ON        NSHUT
0/2               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/3               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/RP0/CPU0        NCS1K4-CNTLR-K9(Active)      IOS XR RUN        NSHUT
0/FT0             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT1             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT2             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/PM0             NCS1K4-AC-PSU                OPERATIONAL       NSHUT
0/SC0             NCS1004                      OPERATIONAL       NSHUT
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#configure
Thu Sep 30 10:29:57.997 UTC
Current Configuration Session  Line       User     Date                      Lock
00001000-000051f7-00000000     vty1       root     Wed Sep 29 15:26:00 2021
RP/0/RP0/CPU0:ios(config)#lc-module location 0/0 lcmode 4x100G-MXP-400G-TXP
RP/0/RP0/CPU0:ios(config)#commit
Thu Sep 30 10:30:11.312 UTC
RP/0/RP0/CPU0:ios(config)#end
RP/0/RP0/CPU0:ios#show lc-module location all lcmode
Thu Sep 30 10:40:56.480 UTC

Node    Lcmode_Supported    Owner    Running                Configured
--------------------------------------------------------------------------------
0/0     Yes                 CLI      4x100G-MXP-400G-TXP    4x100G-MXP-400G-TXP
0/1     No                  N/A      N/A                    N/A
0/2     No                  N/A      N/A                    N/A
0/3     No                  N/A      N/A                    N/A

RP/0/RP0/CPU0:ios# RP/0/RP0/CPU0:ios#show  platform
Thu Sep 30 10:41:25.093 UTC
Node              Type                         State             Config state
--------------------------------------------------------------------------------
0/0               NCS1K4-OTN-XP                OPERATIONAL       NSHUT
0/2               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/3               NCS1K4-LC-FILLER             PRESENT           NSHUT
0/RP0/CPU0        NCS1K4-CNTLR-K9(Active)      IOS XR RUN        NSHUT
0/FT0             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT1             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/FT2             NCS1K4-FAN                   OPERATIONAL       NSHUT
0/PM0             NCS1K4-AC-PSU                OPERATIONAL       NSHUT
0/SC0             NCS1004                      OPERATIONAL       NSHUT
RP/0/RP0/CPU0:ios#
```

Example: 4x100GE-MXP-DD LC Mode

To view the LC modes available on the OTN-XP card, use the following command: The following is a sample in which the OTN-XP card is configured in the 4x100GE-MXP-DD mode.

```
RP/0/RP0/CPU0:ios#show lc-module location all lcmode all
Thu Sep 30 10:43:47.536 UTC
States: A-Available     R-Running       C-Configured

Node    Lcmode_Supported    Owner    Options(State)                HW_Ver
--------------------------------------------------------------------------------
0/0     Yes                 CLI      100G-TXP (A)                   3.0
                                     10G-GREY-MXP (A)               3.0
                                     4x100G-MXP-400G-TXP (A)        2.0
```

```
                                        40x10G-4x100G-MXP (A)          3.0
                                        4x100GE-MXP-DD (R/C)           7.0
                                        400GE-TXP-DD (A)               1.0
                                        FC-MXP (A)                     4.0
                                        OTUCn-REGEN (A)                8.0
                                        2x100GE-TXP-MXP (R/C)          9.0
0/1     No                  N/A         N/A                            N/A
0/2     No                  N/A         N/A                            N/A
0/3     No                  N/A         N/A                            N/A
```

Example: OTUCn-REGEN Mode

The following is a sample to configure the OTUCn-REGEN mode:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  4 16:52:18.021 UTC
RP/0/RP0/CPU0:ios(config)#lc-module location 0/2 lcmode OTUCn-REGEN
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample to verify the OTUCn-REGEN LC mode.

```
RP/0/RP0/CPU0:ios#sh lc-module location 0/2 lcmode
Fri Feb  4 17:00:09.842 UTC

Node    Lcmode_Supported    Owner      Running          Configured
------------------------------------------------------------------------------------------------
0/2       Yes                CLI       OTUCn-REGEN       OTUCn-REGEN
```

Example: FC-MXP Mode

The following is a sample to configure the OTN-XP card in FC-MXP mode:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  5 15:53:17.023 UTC
RP/0/RP0/CPU0:ios(config)#lc-module location 0/2 lcmode FC-MXP
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample to verify the FC-MXP mode configured on the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show lc-module location 0/2 lcmode
Fri Feb  4 16:13:32.745 UTC

Node    Lcmode_Supported    Owner    Running     Configured
------------------------------------------------------------------------------------------------
0/2          Yes             CLI     FC-MXP       FC-MXP

RP/0/RP0/CPU0:ios#
```

Example: 2x100GE-TXP-MXP

The following is a sample to configure the OTN-XP card in 2x100GE-TXP-MXP mode:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  5 15:53:17.023 UTC
RP/0/RP0/CPU0:ios(config)#lc-module location 0/0 lcmode 2x100GE-TXP-MXP
RP/0/RP0/CPU0:ios(config)#commit
```

Following is a sample output that verifies the 2x100GE-TXP-MXP mode configured on the OTN-XP card:

```
RP/0/RP0/CPU0:ios#sh lc-module location 0/0 lcmode
Mon Nov  7 10:41:48.398 UTC

Node    Lcmode_Supported    Owner    Running                Configured
```

```
--------------------------------------------------------------------------------
0/0     Yes                 CLI      2x100GE-TXP-MXP    2x100GE-TXP-MXP
```

Regeneration Mode on OTN-XP Card

Table 4: Feature History Table

Feature	Release Information	Description
Regeneration Mode Support on the OTN-XP Card	Cisco IOS XR Release 7.5.2	The OTN-XP card now supports the OTUCn-REGEN LC mode for regeneration. This mode allows regeneration of the DWDM channels across trunk ports of the OTN-XP card and significantly extends the reach of the service. You can configure 200G and 400G trunk rates on the card. The OTN-XP card supports the OTUCn-REGEN mode as a part of the LC mode for regeneration.

The OTUCn-REGEN mode supports the following features:

- Trunk rate—400G and 200G
- Modulation type:
  - 16-QAM for 400G
  - 8-QAM, 16-QAM, and QPSK for 200G
- Trunk optics—ONS-CFP2D-400G-C
- Supported features—Loopbacks, TTI, AINS

Limitations

GCC0 is not supported on the REGEN trunks.

Configuring Regen Mode on OTN-XP Card

Note You must configure the OTUCn-REGEN LC mode on the OTN-XP card before performing this configuration. See Example: OTUCn-REGEN Mode, on page 30.

The following is a sample for configuring regen mode with 400G trunk rate:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  4 16:53:48.018 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 regen
RP/0/RP0/CPU0:ios(config-regen)#trunk-rate 400G
```

```
RP/0/RP0/CPU0:ios(config-regen)#commit
Fri Feb  4 16:54:05.920 UTC
```

The following is a sample to verify the 400G trunk rate configured in the regen mode.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 regen
Fri Feb  4 16:58:51.716 UTC

Location:             0/2
Trunk  Bitrate:       400G
Status:               Provisioned
East Port                   West Port
---------                   ---------
CoherentDSP0/2/0/12         CoherentDSP0/2/0/13
```

The following is a sample for configuring regen mode with 200G trunk rate:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  4 16:53:48.018 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 regen
RP/0/RP0/CPU0:ios(config-regen)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-regen)#commit
Fri Feb  4 16:54:05.920 UTC
```

If you want to change the modulation format for the 200G trunk rate, see Configuring 8QAM Modulation for 200G Muxponder Mode, on page 71.

The following is a sample to verify the 200G trunk rate configured in the regen mode.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 regen
Fri Feb  4 16:58:51.716 UTC

Location:             0/2
Trunk  Bitrate:       200G
Status:               Provisioned
East Port                   West Port
---------                   ---------
CoherentDSP0/2/0/12         CoherentDSP0/2/0/13
```

FC-MXP Mode on OTN-XP Card

Table 5: Feature History Table

Feature	Release Information	Description
FC-MXP Mode Support on the OTN-XP Card	Cisco IOS XR Release 7.5.2	The OTN-XP card now supports FC-MXP LC mode for Fiber Channel (FC) support. You can configure 16G FC with 400G trunk rate.

Table 6: Feature History Table

Feature	Release Information	Description
32G FC-MXP Mode Support on the OTN-XP Card	Cisco IOS XR Release 7.7.1	The OTN-XP card now supports 32G FC-MXP LC mode for Fiber Channel (FC) support, in addition to the 16G FC-MXP mode that was supported already. You can configure 32G FC with 400G trunk rate on slice 0.

From Release 7.5.2 onwards, the OTN-XP card supports FC-MXP LC mode for Fiber Channel (FC) support. You can configure 16G FC with 400G trunk rate on both the slices.

From Release 7.7.1, 32GFC can be configured on slice 0.

FC Mode	Supported Slices	Slice 0 Ports	Slice 1 Ports	Client Payloads	Trunk Rate	Client Optics	Trunk Optics	Modulation Type
16G FC-MXP	Slice 0 and Slice 1	Clients: 1, 6, 7, 9, 10, 11 and Trunk: 12	Clients: 0, 2, 3, 4, 5, 8 and Trunk: 13	16G FC	400G (OTUC4) on both the slices	ONS-QC-16GFC-SW	ONS-CFP2D-400G-C DP04CFP2-M25-K9	16-QAM
32G FC-MXP	Slice 0	Clients: 9, 10, 11 and Trunk: 12	NA	32G FC	400G (OTUC4) on slice 0	DS-SFP-4X32G-SW	ONS-CFP2D-400G-C DP04CFP2-M25-K9	16-QAM

The FC-MXP mode supports the following features:

- Loopback
- PRBS
- AINS
- Laser squelch

The FC-MXP mode supports the following alarms:

- SIGLOSS—Signal Loss
- SYNCLOSS—Loss of Synchronization
- NOS—Not-Operational Primitive Sequence

Limitations:

- The combination of 16G FC and 32G FC configurations is not supported on the same slice.
- GCC0 and GCC1 are not supported.

- Local Fault and Remote Fault Ethernet alarms are not supported.
- FC32 only supports SIGLOSS alarm.
- FC32 does not support statistics counters.

Configuring the OTN-XP Card in 16G FC Muxponder Mode

Note You must configure the FC-MXP LC mode on the OTN-XP card before performing this configuration. See Example: FC-MXP Mode, on page 30.

To configure the OTN-XP card in 16G FC-MXP mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **lane** *lane-number* **client-type fc16**

commit

Example:

The following is a sample to configure 16G FC muxponder mode on slice 0 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Fri Feb  4 16:06:59.967 UTC
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#
```

The following is a sample to verify the 16G FC muxponder mode configured on slice 0 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder-slice 0
Fri Feb  4 16:15:10.984 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       FC16G
Trunk  Bitrate:       400G
Status:               Provisioned
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/12
                                Traffic Split Percentage

SixteenGigFibreChanCtrlr0/2/0/1/1       ODU-FLEX0/2/0/12/1/1                        100
SixteenGigFibreChanCtrlr0/2/0/1/2       ODU-FLEX0/2/0/12/1/2                        100
SixteenGigFibreChanCtrlr0/2/0/1/3       ODU-FLEX0/2/0/12/1/3                        100
SixteenGigFibreChanCtrlr0/2/0/1/4       ODU-FLEX0/2/0/12/1/4                        100
SixteenGigFibreChanCtrlr0/2/0/6/1       ODU-FLEX0/2/0/12/6/1                        100
SixteenGigFibreChanCtrlr0/2/0/6/2       ODU-FLEX0/2/0/12/6/2                        100
SixteenGigFibreChanCtrlr0/2/0/6/3       ODU-FLEX0/2/0/12/6/3                        100
SixteenGigFibreChanCtrlr0/2/0/6/4       ODU-FLEX0/2/0/12/6/4                        100
SixteenGigFibreChanCtrlr0/2/0/7/1       ODU-FLEX0/2/0/12/7/1                        100
SixteenGigFibreChanCtrlr0/2/0/7/2       ODU-FLEX0/2/0/12/7/2                        100
SixteenGigFibreChanCtrlr0/2/0/7/3       ODU-FLEX0/2/0/12/7/3                        100
SixteenGigFibreChanCtrlr0/2/0/7/4       ODU-FLEX0/2/0/12/7/4                        100
SixteenGigFibreChanCtrlr0/2/0/9/1       ODU-FLEX0/2/0/12/9/1                        100
SixteenGigFibreChanCtrlr0/2/0/9/2       ODU-FLEX0/2/0/12/9/2                        100
SixteenGigFibreChanCtrlr0/2/0/9/3       ODU-FLEX0/2/0/12/9/3                        100
SixteenGigFibreChanCtrlr0/2/0/9/4       ODU-FLEX0/2/0/12/9/4                        100
SixteenGigFibreChanCtrlr0/2/0/10/1      ODU-FLEX0/2/0/12/10/1                       100
SixteenGigFibreChanCtrlr0/2/0/10/2      ODU-FLEX0/2/0/12/10/2                       100
SixteenGigFibreChanCtrlr0/2/0/10/3      ODU-FLEX0/2/0/12/10/3                       100
SixteenGigFibreChanCtrlr0/2/0/10/4      ODU-FLEX0/2/0/12/10/4                       100
SixteenGigFibreChanCtrlr0/2/0/11/1      ODU-FLEX0/2/0/12/11/1                       100
SixteenGigFibreChanCtrlr0/2/0/11/2      ODU-FLEX0/2/0/12/11/2                       100
SixteenGigFibreChanCtrlr0/2/0/11/3      ODU-FLEX0/2/0/12/11/3                       100
SixteenGigFibreChanCtrlr0/2/0/11/4      ODU-FLEX0/2/0/12/11/4                       100

RP/0/RP0/CPU0:ios#
```

The following is a sample to configure 16G FC muxponder mode on slice 1 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 lane 1 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 lane 2 client-type fc16
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 lane 3 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 lane 4 client-type fc16
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Fri Feb  4 16:06:59.967 UTC
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#
```

The following is a sample to verify the 16G FC muxponder mode configured on slice 1 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder-slice 1
Fri Feb  4 16:15:10.984 UTC

Location:             0/2
Slice ID:             1
Client Bitrate:       FC16G
Trunk  Bitrate:       400G
Status:               Provisioned
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/13
                                Traffic Split Percentage

SixteenGigFibreChanCtrlr0/2/0/0/1       ODU-FLEX0/2/0/13/0/1                        100
SixteenGigFibreChanCtrlr0/2/0/0/2       ODU-FLEX0/2/0/13/0/2                        100
SixteenGigFibreChanCtrlr0/2/0/0/3       ODU-FLEX0/2/0/13/0/3                        100
SixteenGigFibreChanCtrlr0/2/0/0/4       ODU-FLEX0/2/0/13/0/4                        100
SixteenGigFibreChanCtrlr0/2/0/2/1       ODU-FLEX0/2/0/13/2/1                        100
SixteenGigFibreChanCtrlr0/2/0/2/2       ODU-FLEX0/2/0/13/2/2                        100
SixteenGigFibreChanCtrlr0/2/0/2/3       ODU-FLEX0/2/0/13/2/3                        100
SixteenGigFibreChanCtrlr0/2/0/2/4       ODU-FLEX0/2/0/13/2/4                        100
SixteenGigFibreChanCtrlr0/2/0/3/1       ODU-FLEX0/2/0/13/3/1                        100
SixteenGigFibreChanCtrlr0/2/0/3/2       ODU-FLEX0/2/0/13/3/2                        100
SixteenGigFibreChanCtrlr0/2/0/3/3       ODU-FLEX0/2/0/13/3/3                        100
SixteenGigFibreChanCtrlr0/2/0/3/4       ODU-FLEX0/2/0/13/3/4                        100
SixteenGigFibreChanCtrlr0/2/0/4/1       ODU-FLEX0/2/0/13/4/1                        100
SixteenGigFibreChanCtrlr0/2/0/4/2       ODU-FLEX0/2/0/13/4/2                        100
SixteenGigFibreChanCtrlr0/2/0/4/3       ODU-FLEX0/2/0/13/4/3                        100
SixteenGigFibreChanCtrlr0/2/0/4/4       ODU-FLEX0/2/0/13/4/4                        100
SixteenGigFibreChanCtrlr0/2/0/5/1       ODU-FLEX0/2/0/13/5/1                       100
SixteenGigFibreChanCtrlr0/2/0/5/2       ODU-FLEX0/2/0/13/5/2                       100
SixteenGigFibreChanCtrlr0/2/0/5/3       ODU-FLEX0/2/0/13/5/3                       100
SixteenGigFibreChanCtrlr0/2/0/5/4       ODU-FLEX0/2/0/13/5/4                       100
SixteenGigFibreChanCtrlr0/2/0/8/1       ODU-FLEX0/2/0/13/8/1                       100
SixteenGigFibreChanCtrlr0/2/0/8/2       ODU-FLEX0/2/0/13/8/2                       100
SixteenGigFibreChanCtrlr0/2/0/8/3       ODU-FLEX0/2/0/13/8/3                       100
SixteenGigFibreChanCtrlr0/2/0/8/4       ODU-FLEX0/2/0/13/8/4                       100

RP/0/RP0/CPU0:ios#
```

Configuring the OTN-XP Card in 32G FC Muxponder Mode

Note You must configure the FC-MXP LC mode on the OTN-XP card before performing this configuration. See Example: FC-MXP Mode, on page 30.

Note The Production Software Maintenance Updates (SMU) for the Cisco IOS-XR Release 7.7.1 (ncs1004-sysadmin-7.7.1.CSCwb01852.tar) is mandatory to configure the latest port configurations for the 32G FC muxponder mode.

To configure the OTN-XP card in 32G FC-MXP mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **lane** *lane-number* **client-type fc32**

commit

Example:

The following is a sample to configure 32G FC muxponder mode on slice 0 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  4 16:24:53.964 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 1 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 2 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 3 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 lane 4 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 3 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 4 client-type fc32
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Fri Feb  4 16:26:46.550 UTC
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#
```

The following is a sample to verify the 32G FC muxponder mode configured on slice 0 of the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder-slice 0
Fri Feb  4 16:31:37.494 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       FC32G
Trunk  Bitrate:       400G
Status:               Provisioned
Client Port                    Mapper/Trunk Port          CoherentDSP0/2/0/12
                               Traffic Split Percentage

ThirtyTwoGigFibreChanCtrlr0/2/0/9/1     ODU-FLEX0/2/0/12/9/1                        100
ThirtyTwoGigFibreChanCtrlr0/2/0/9/2     ODU-FLEX0/2/0/12/9/2                        100
ThirtyTwoGigFibreChanCtrlr0/2/0/9/3     ODU-FLEX0/2/0/12/9/3                        100
ThirtyTwoGigFibreChanCtrlr0/2/0/9/4     ODU-FLEX0/2/0/12/9/4                        100
ThirtyTwoGigFibreChanCtrlr0/2/0/10/1     ODU-FLEX0/2/0/12/10/1                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/10/2     ODU-FLEX0/2/0/12/10/2                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/10/3     ODU-FLEX0/2/0/12/10/3                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/10/4     ODU-FLEX0/2/0/12/10/4                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/11/1     ODU-FLEX0/2/0/12/11/1                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/11/2     ODU-FLEX0/2/0/12/11/2                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/11/3     ODU-FLEX0/2/0/12/11/3                       100
ThirtyTwoGigFibreChanCtrlr0/2/0/11/4     ODU-FLEX0/2/0/12/11/4                       100

RP/0/RP0/CPU0:ios#
```

Supported Pluggables for OTN-XP Card

Table 7: Feature History

Feature Name	Release Information	Description
FC Mode Support on DP04CFP2-M25-K9 Pluggable	Cisco IOS XR Release 7.7.1	The 16G FC and 32G FC muxponder modes support is added to the trunk pluggable DP04CFP2-M25-K9 on the OTN-XP card. This is in addition to the 4x100 muxponder and 400G-TXP modes that were supported previously.

The OTN-XP card supports the following trunk and client pluggables:

Trunk Pluggables

- ONS-CFP2D-400G-C
- QDD-400G-ZRP-S
- CFP2-WDM-DETS-1HL
- DP04CFP2-M25-K9

Note

- Starting from the Release 7.5.2, DP04CFP2-M25-K9 supports 4x100 muxponder and 400G-TXP modes.
- Starting from the Release 7.7.1, DP04CFP2-M25-K9 supports 16G FC and 32G FC muxponder modes.

Client Pluggables

- QSFP-100G-LR4
- QSFP-100G-FR-S
- QSFP-100G-SR4-S
- QSFP-100G-CWDM4-S
- QSFP-100G-LR4-S
- QSFP-100G-AOC
- QSFP-100G-PSM4
- QSFP-100G-DR-S
- QSFP-4x10-MLR

- QSFP-40G-SR4=
- QDD-400G-FR4-S
- QDD-400G-DR4-S
- QDD-400G-LR8-S
- ONS-QC-16GFC-SW
- DS-SFP-4X32G-SW
- QSFP-100G-LR-S
- ONS-QSFP28-LR4

See Supported Pluggables, for the complete list of pluggables supported by NCS 1004.

Muxponder Configuration on OTN-XP Card

The OTN-XP card has two trunk ports and 12 client ports. The muxponder configuration supports two slices, 0 and 1. You can configure mxponder-slice 0, mxponder-slice 1, or both. Each mxponder-slice supports 10 client interfaces.

From Release 7.3.1 onwards, the OTN-XP card supports two trunk ports for CFP2 DCO on port 12 and port 13, and 8 client ports. For configuration, see Configuring the Muxponder Mode for 4x100G MXP, on page 50.

From Release 7.3.2 onwards, the OTN-XP card supports two trunk ports for QDD ZRP on port 9 and port 11, and 8 client ports. For configuration, see Configuring the Muxponder Mode for 4x100GE-MXP-DD, on page 78.

From Release 7.5.1 onwards, the OTN-XP card supports two trunk ports for QDD ZRP on port 9 and port 11, and the supported operating modes are 400G-TXP-DD, 3X100GE MXP, and 2X100GE MXP. For configuration, see Configuring the Transponder Mode for 400GE-TXP-DD, on page 86. The client rates 2x100GE and 3x100G are supported as part of the 4x100GE-MXP-DD mode. For configurations, see Configuring the Muxponder Mode for 2x100GE-MXP-DD, on page 80 and Configuring the Muxponder Mode for 3x100GE-MXP-DD, on page 83.

Table 8: Feature History

Feature Name	Release Information	Description
400 TXP or MXP modes with CFP2 DCO for OTN-XP Card	Cisco IOS XR Release 7.3.1	On the OTN-XP card, you can configure a single 400GE or 4x100G payload that is received over the client port as a 400G signal over DWDM on the line side. The card improves efficiency, performance, and flexibility for customer networks allowing 400GE or 4x100G client transport over 400G WDM wavelength. Commands modified: • **controller coherentDSP** • **show controller coherentDSP**
OC192 and STM64 clients on OTN-XP card	Cisco IOS XR Release 7.10.1	40x10G-4x100G-MXP card mode on the OTN-XP card now supports OC192/STM64 clients in the 40x10G mode. This allows you to use the OTN-XP card to handle OC192 SONET and STM64 SDH payloads.

Table 9: Hardware Module Configuration with Client to Trunk Mapping

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
10G Grey Muxponder	10G-GREY-MXP	OTU2, OTU2e, or 10 GE	Mxponder-slice 0—Client ports 4, 5, and 2 are mapped to the trunk port 0. Mxponder-slice 1—Client ports 7, 6, and 11 are mapped to the trunk port 1. Each client port consists of four lanes, 1, 2, 3, and 4. The lanes 3 and 4 can only be configured for ports 2 and 11. It is not mandatory to configure all 10 client lanes for a slice.	100G
400G-MXP	4x100G-MXP -400G-TXP	100GE, OTU4	Mxponder-slice 0—Client ports 1, 6, 7, and 10 are mapped to the trunk port 12. Mxponder-slice 1—Client ports 0, 4, 5, and 8 are mapped to the trunk port 13.	400G
400G-TXP	4x100G-MXP -400G-TXP	400GE	Mxponder-slice 0—Client port 10 is mapped to the trunk port 12. Mxponder-slice 1—Client port 8 is mapped to the trunk port 13.	400G
40x10G	40x10G-4x100G-MXP	STM64, OC192, OTU2, OTU2e, or 10 GE	Mxponder-slice 0—10G Client ports 0, 1, 2, 3, 4, 5, 6, 7, 9, and 11 mapped to the trunk port 12. Each client port consists of four lanes, 1, 2, 3, and 4.	400G CFP2

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
30x10G	40x10G-4x100G-MXP	OTU2, OTU2e, or 10 GE	Mxponder-slice 0—10G Client ports 0, 1, 2, 3, 4, 5, 9, and 11 are mapped to the trunk port 12. The client ports 0, 1, 2, 3, 4, 5, and 9 are configured for all four lanes, 1, 2, 3, and 4. The client port 11 is configured for lanes 1 and 2.	300G CFP2
20x10G + 2x100G	40x10G-4x100G-MXP	10 GE, 100 GE, OTU2, OTU2e, or OTU4	Mxponder-slice 0—The following 100G and 10G client ports are mapped to trunk port 12. • 100G client port—0 and 1 • 10G client port—4, 5, 6, and 7 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—11 and 2 are configured for lanes 3 and 4.	400G CFP2

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
10x10G + 3 x 100G	40x10G-4x100G-MXP	10GE, 100GE, OTU4, OTU2, or OTU2e	Mxponder-slice 0—The following 100G and 10G client ports are mapped to a trunk port 12. • 100G client port—0, 1, and 6 • 10G client port—4 and 5 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—2 are configured for lanes 3 and 4.	400G CFP2
20x10G + 1 x 100G	40x10G-4x100G-MXP	10 GE, 100 GE, OTU2, OTU2e, or OTU4	Mxponder-slice 0—The following 100G and 10G client ports are mapped to trunk port 12. • 100G client port—0 • 10G client port—1, 4,5 and 9 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—2 is configured for lanes 3 and 4 and 11 is configured for lanes 1 and 2.	300G CFP2

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
30x10G + 1 x 100G	40x10G-4x100G-MXP	10 GE, 100 GE, OTU2, OTU2e, or OTU4	Mxponder-slice 0—The following 100G and 10G client ports are mapped to trunk port 12. • 100G client port—6 • 10G client port—0, 1, 2, 3, 4, 5, and 9 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—11 is configured for lanes 1 and 2.	400G CFP2
10x10G + 2 x 100G	40x10G-4x100G-MXP	10 GE, 100 GE, OTU2, OTU2e, or OTU4	Mxponder-slice 0—The following 100G and 10G client ports are mapped to trunk port 12. • 100G client port—0 and 1 • 10G client port—4, and 5 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—2 is configured for lanes 3 and 4.	300G CFP2

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
10x10G + 1 x 100G	40x10G-4x100G-MXP	10 GE and 100 GE	Mxponder-slice 0—The following 100G and 10G client ports are mapped to trunk port 12. • 100G client port—0 • 10G client port—4, and 5 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—2 is configured for lanes 3 and 4. • Mxponder-slice 1—The following 100G and 10G client ports are mapped to trunk port 13. • 100G client port—1 • 10G client port—6, and 7 are configured for all four lanes, 1, 2, 3, and 4. • 10G client port—11 is configured for lanes 3 and 4.	200G CFP2
200G Muxponder	200G-FOIC2-oFEC-QPSK-1-S 200G-FOIC2-oFEC-8QAM-1-E	OTU4, 100GE	Mxponder-slice 0—Client ports 7 and 10 mapped to the trunk port 12. Mxponder-slice1—Client ports 5 and 8 mapped to the trunk port 13.	200G CFP2

Hardware Module Configuration	Line Card Mode	Client Port Rate	Client to Trunk Mapping	Trunk Rate
QDD ZRP	4x100GE-MXP-DD	100GE	Mxponder-slice 0—Client ports 1, 6, 7, and 10 are mapped to the trunk port 11. Mxponder-slice 1—Client ports 0, 4, 5, and 8 are mapped to the trunk port 9.	400G
QDD ZRP	400GE-TXP-DD	400GE	Mxponder-slice 0—Client port 10 is mapped to the trunk port 11. Mxponder-slice 1—Client port 8 is mapped to the trunk port 9.	400G
	4x100GE-MXP-DD	100GE	Mxponder-slice 0—Client ports 1, 7, and 10 are mapped to the trunk port 11. Mxponder-slice 1—Client ports 4, 5, and 8 are mapped to the trunk port 9.	300G
	4x100GE-MXP-DD	100GE	Mxponder-slice 0—Client ports 7, and 10 are mapped to the trunk port 11. Mxponder-slice 1—Client ports 4 and 5 are mapped to the trunk port 9.	200G

Note OC192 and STM64 are supported only on Cisco 4x10G QSFP+ MLR Pluggable Optics Module as the client pluggable.

QDD ZRP Limitations

- Hold of timer and Idle insertion are not supported on 400GE Client for 400G-TXP mode.
- Local Fault and Remote Fault ethernet alarms are not supported on 400GE Client for 400G-TXP mode.

- Far-end PM counters on Coherent DSP controllers are not supported for 400G-TXP and 4x100G MXP modes.
- QDD ZRP alarms appear with *Flexo* label due to absence of a separate ZRP layer.

Limitations for STM64 and OC192 in 40x10G-4x100G-MXP Mode

- OC192 and STM64 are available only in 400G trunk mode and only slice 0 is supported.
- Data communication channel for STM64 and OC192 is available only in path monitoring mode.
- PRBS, AINS, and SNMP are not supported for OC192 and STM64.
- The OC192 and STM64 controllers do not support the **controller description** command.
- Current and History PM counters do not support flex and 30 second bucket types.
- OOF alarm (out-of-frame) is not supported.

Configuring the Muxponder Mode for 10G Grey Muxponder

Note The LC mode must be configured to 10G-GREY-MXP on the OTN-XP card before you perform this configuration.

To configure the OTN-XP card in the muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 100G

client-port-rate *client-port-number* **lane** *lane-number* **client-type { 10GE | OTU2 | OTU2e}**

commit

Example

The following is a sample in which the OTN-XP card is configured with mixed client rates in the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 100G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Tue Apr 21 09:26:12.308 UTC
```

```
Location:             0/1
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       100G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port   Mapper/Trunk Port    Peer/Trunk Port    OTU40/0/0/0
              Traffic Split Percentage
OTU20/0/0/2/3            NONE          ODU20/0/0/0/2/3       100
OTU20/0/0/2/4            NONE          ODU20/0/0/0/2/4       100
TenGigECtrlr0/0/0/4/1 ODU2E0/0/0/0/4/1       NONE            100
```

Configuring DAC Rate for 400G Muxponder Modes

Table 10: Feature History

Feature Name	Release	Description
DAC Configuration Support for 400GE, 4x100G, or 400G Regen modes	Cisco IOS XR Release 7.5.2	On the OTN-XP card, you can configure the Digital-to-Analog (DAC) rate for the 400GE, 4x100G, or 400G Regen modes with CFP2 DCO pluggable. Based on the DAC rate configured, pulse shaping and modem setting is set on the CFP2 DCO trunk pluggable.

From Release 7.5.2 onwards, you can configure DAC rate to set the bookended mode for the 400GE, 4x100G, or 400G Regen modes on the OTN-XP card.

DAC Supported Modes

The following operating modes are supported on the CFP2 coherent pluggable module for the OTN-XP card:

Table 11: DAC Supported Modes

Network Configuration Mode	Trunk Rate	Data Path	Line Framing	FEC Type	Modulation Format	BPS	Baud Rate (GBd)	Pulse Shaping	Mode Type
400G	400G	400G TXP, 4x100G MXP, 400G Regen	FlexO-4	oFEC	16 QAM	4	63.1	1	Enhanced

The following table provides the pulse shaping and modem setting values for the respective DAC rates.

Note The default pulse shaping is *1.5* and mode type is *Standard* for the supported modes.

Table 12: DAC Rate

DAC Rate	Pulse Shape	Modem Setting
1	0	Standard
1.25	1	Enhanced
1.5	1	Standard
2	0	Enhanced

To configure the DAC rate for OTN-XP card in the 400G TXP, 4x100G MXP, and 400G Regen modes, use the following commands:

configure

controller optics *Rack/Slot/Instance/Port* **dac-rate** 1x1.25

commit

The following is a sample in which DAC rate is configured.

```
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12 dac-Rate 1x1.25
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

Verifying the DAC Rate Configuration

The following is a sample to verify the DAC rate configuration for the 400G TXP, 4x100G MXP, and 400G Regen modes in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show controllers optics 0/2/0/12
Wed Apr 13 15:00:10.044 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Red

DAC RATE: 1x1.25

Configured DAC RATE: 1x1.25
```

Configuring the Muxponder Mode for 4x100G MXP

Note The LC mode must be configured to 4x100G-MXP-400G-TXP on the OTN-XP card before you perform this configuration. See Configuring the LC Mode, on page 25.

To configure the OTN-XP card in the 4x100 muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type OTU4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Mon Nov 30 01:32:57.338 UTC

Location: 0/1
Slice ID: 0
Client Bitrate: 100GE
Trunk Bitrate: 400G
Status: Provisioned
LLDP Drop Enabled: FALSE
ARP Snoop Enabled: FALSE
Client Port Mapper/Trunk Port CoherentDSP0/1/0/12
Traffic Split Percentage

HundredGigECtrlr0/1/0/1 ODU40/1/0/12/1 100
```

```
Location: 0/1
Slice ID: 1
Client Bitrate: OTU4
Trunk Bitrate: 400G
Status: Provisioned
Client Port Peer/Trunk Port CoherentDSP0/1/0/13
Traffic Split Percentage

OTU40/1/0/8 ODUC40/1/0/13 100
```

Configuring the Muxponder Mode for 400G TXP

Note The LC mode must be configured to 4x100G-MXP-400G-TXP on the OTN-XP card before you perform this configuration.

To configure the OTN-XP card in the 400G TXP mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type 400GE**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Mon Nov 30 01:36:14.514 UTC

Location: 0/1
Slice ID: 0
```

```
Client Bitrate: 400GE
Trunk Bitrate: 400G
Status: Provisioned
LLDP Drop Enabled: FALSE
ARP Snoop Enabled: FALSE
Client Port Mapper/Trunk Port CoherentDSP0/1/0/12
Traffic Split Percentage

FourHundredGigECtrlr0/1/0/10 ODU-FLEX0/1/0/12/10 100


Location: 0/1
Slice ID: 1
Client Bitrate: 400GE
Trunk Bitrate: 400G
Status: Provisioned
LLDP Drop Enabled: FALSE
ARP Snoop Enabled: FALSE
Client Port Mapper/Trunk Port CoherentDSP0/1/0/13
Traffic Split Percentage

FourHundredGigECtrlr0/1/0/8 ODU-FLEX0/1/0/13/8 100
```

Static TPN and TS allocation for TXP-MXP-Grey muxponder modes

The OTN-XP card uses the following mapping of tributary port numbers, tributary slots, and clients for the various TXP and MXP configurations.

Table 13: TPN-TS Mapping in 400GE TXP Configuration

Slice	Client Port	Client Rate	Trunk Port	Trunk Rate	TPN	TS
0	10	400GE	12	400G	1	1.1 to 4.20
1	8	400GE	13	400G	1	1.1 to 4.20

Table 14: TPN-TS Mapping in 4 X 100G MXP Configuration

Slice	Client Port	Client Rate	Trunk Port	Trunk Rate	TPN	TS
0	10	100GE/OTU4	12	400G	1	1.1 to 1.20
	7	100GE/OTU4			2	2.1 to 2.20
	6	100GE/OTU4			3	3.1 to 3.20
	1	100GE/OTU4			4	4.1 to 4.20
1	8	100GE/OTU4	13	400G	1	1.1 to 1.20
	5	100GE/OTU4			2	2.1 to 2.20
	4	100GE/OTU4			3	3.1 to 3.20
	0	100GE/OTU4			4	4.1 to 4.20

Table 15: TPN-TS Mapping in 10 X 10G Grey Muxponder Configuration

Slice	Client Port	Client Lane	Client Rate	Trunk Port	Trunk Rate	TPN	TS
0	4	1	10GE/OTU2/OTU2e	0	100G	1	1–8
		2	10GE/OTU2/OTU2e			2	9–16
		3	10GE/OTU2/OTU2e			3	17–24
		4	10GE/OTU2/OTU2e			4	25–32
	5	1	10GE/OTU2/OTU2e			5	33–40
		2	10GE/OTU2/OTU2e			6	41–48
		3	10GE/OTU2/OTU2e			7	49–56
		4	10GE/OTU2/OTU2e			8	57–64
	2	3	10GE/OTU2/OTU2e			9	65–72
		4	10GE/OTU2/OTU2e			10	73–80
1	7	1	10GE/OTU2/OTU2e	1	100G	1	1–8
		2	10GE/OTU2/OTU2e			2	9–16
		3	10GE/OTU2/OTU2e			3	17–24
		4	10GE/OTU2/OTU2e			4	25–32
	6	1	10GE/OTU2/OTU2e			5	33–40
		2	10GE/OTU2/OTU2e			6	41–48
		3	10GE/OTU2/OTU2e			7	49–56
		4	10GE/OTU2/OTU2e			8	57–64
	11	3	10GE/OTU2/OTU2e			9	65–72
		4	10GE/OTU2/OTU2e			10	73–80

Configuring the Muxponder Mode for 40x10G Muxponder

To configure the OTN-XP card in the 40x10G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E | STM64 | OC192}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the port 0 lane 1 has OC192 payload and lane 2 has STM64 payload. Only slice 0 configuration is supported for OC192/STM64.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 1 client-type oc192
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 2 client-type stm64
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 40x10G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/3 mxponder
Wed Jun  2 17:57:36.647 UTC

Location:             0/3
Slice ID:             0
Client Bitrate:       10GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/3/0/12
                               Traffic Split Percentage

TenGigECtrlr0/3/0/0/1          ODU2E0/3/0/12/0/1                          100
TenGigECtrlr0/3/0/0/2          ODU2E0/3/0/12/0/2                          100
TenGigECtrlr0/3/0/0/3          ODU2E0/3/0/12/0/3                          100
TenGigECtrlr0/3/0/0/4          ODU2E0/3/0/12/0/4                          100
TenGigECtrlr0/3/0/1/1          ODU2E0/3/0/12/1/1                          100
TenGigECtrlr0/3/0/1/2          ODU2E0/3/0/12/1/2                          100
TenGigECtrlr0/3/0/1/3          ODU2E0/3/0/12/1/3                          100
TenGigECtrlr0/3/0/1/4          ODU2E0/3/0/12/1/4                          100
TenGigECtrlr0/3/0/2/1          ODU2E0/3/0/12/2/1                          100
TenGigECtrlr0/3/0/2/2          ODU2E0/3/0/12/2/2                          100
TenGigECtrlr0/3/0/2/3          ODU2E0/3/0/12/2/3                          100
TenGigECtrlr0/3/0/2/4          ODU2E0/3/0/12/2/4                          100
TenGigECtrlr0/3/0/3/1          ODU2E0/3/0/12/3/1                          100
TenGigECtrlr0/3/0/3/2          ODU2E0/3/0/12/3/2                          100
TenGigECtrlr0/3/0/3/3          ODU2E0/3/0/12/3/3                          100
TenGigECtrlr0/3/0/3/4          ODU2E0/3/0/12/3/4                          100
TenGigECtrlr0/3/0/4/1          ODU2E0/3/0/12/4/1                          100
TenGigECtrlr0/3/0/4/2          ODU2E0/3/0/12/4/2                          100
TenGigECtrlr0/3/0/4/3          ODU2E0/3/0/12/4/3                          100
TenGigECtrlr0/3/0/4/4          ODU2E0/3/0/12/4/4                          100
TenGigECtrlr0/3/0/5/1          ODU2E0/3/0/12/5/1                          100
TenGigECtrlr0/3/0/5/2          ODU2E0/3/0/12/5/2                          100
TenGigECtrlr0/3/0/5/3          ODU2E0/3/0/12/5/3                          100
TenGigECtrlr0/3/0/5/4          ODU2E0/3/0/12/5/4                          100
TenGigECtrlr0/3/0/6/1          ODU2E0/3/0/12/6/1                          100
TenGigECtrlr0/3/0/6/2          ODU2E0/3/0/12/6/2                          100
TenGigECtrlr0/3/0/6/3          ODU2E0/3/0/12/6/3                          100
TenGigECtrlr0/3/0/6/4          ODU2E0/3/0/12/6/4                          100
TenGigECtrlr0/3/0/7/1          ODU2E0/3/0/12/7/1                          100
TenGigECtrlr0/3/0/7/2          ODU2E0/3/0/12/7/2                          100
TenGigECtrlr0/3/0/7/3          ODU2E0/3/0/12/7/3                          100
TenGigECtrlr0/3/0/7/4          ODU2E0/3/0/12/7/4                          100
TenGigECtrlr0/3/0/9/1          ODU2E0/3/0/12/9/1                          100
TenGigECtrlr0/3/0/9/2          ODU2E0/3/0/12/9/2                          100
TenGigECtrlr0/3/0/9/3          ODU2E0/3/0/12/9/3                          100
TenGigECtrlr0/3/0/9/4          ODU2E0/3/0/12/9/4                          100
TenGigECtrlr0/3/0/11/1         ODU2E0/3/0/12/11/1                         100
TenGigECtrlr0/3/0/11/2         ODU2E0/3/0/12/11/2                         100
TenGigECtrlr0/3/0/11/3         ODU2E0/3/0/12/11/3                         100
TenGigECtrlr0/3/0/11/4         ODU2E0/3/0/12/11/4                         100
```

Configuring the Muxponder Mode for 30x10G

To configure the OTN-XP card in the 30x10G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 300G

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2E | OTU2}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 300G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 30x10G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Wed Jun  2 17:56:40.574 UTC
```

```
Location:             0/1
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                   Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/1/0/12

                                 Traffic Split Percentage

OTU20/1/0/0/3                        NONE      ODU20/1/0/12/0/3
    100
OTU20/1/0/1/3                        NONE      ODU20/1/0/12/1/3
    100
OTU20/1/0/2/3                        NONE      ODU20/1/0/12/2/3
    100
OTU20/1/0/3/3                        NONE      ODU20/1/0/12/3/3
    100
OTU20/1/0/4/3                        NONE      ODU20/1/0/12/4/3
    100
OTU20/1/0/5/3                        NONE      ODU20/1/0/12/5/3
    100
OTU20/1/0/9/3                        NONE      ODU20/1/0/12/9/3
    100
OTU2E0/1/0/0/2                       NONE      ODU2E0/1/0/12/0/2
    100
OTU2E0/1/0/1/2                       NONE      ODU2E0/1/0/12/1/2
    100
OTU2E0/1/0/2/2                       NONE      ODU2E0/1/0/12/2/2
    100
OTU2E0/1/0/3/2                       NONE      ODU2E0/1/0/12/3/2
    100
OTU2E0/1/0/4/2                       NONE      ODU2E0/1/0/12/4/2
    100
OTU2E0/1/0/5/2                       NONE      ODU2E0/1/0/12/5/2
    100
OTU2E0/1/0/9/2                       NONE      ODU2E0/1/0/12/9/2
    100
TenGigECtrlr0/1/0/0/1          ODU2E0/1/0/12/0/1             NONE
    100
TenGigECtrlr0/1/0/0/4          ODU2E0/1/0/12/0/4             NONE
    100
TenGigECtrlr0/1/0/1/1          ODU2E0/1/0/12/1/1             NONE
    100
TenGigECtrlr0/1/0/1/4          ODU2E0/1/0/12/1/4             NONE
    100
TenGigECtrlr0/1/0/2/1          ODU2E0/1/0/12/2/1             NONE
    100
TenGigECtrlr0/1/0/2/4          ODU2E0/1/0/12/2/4             NONE
    100
TenGigECtrlr0/1/0/3/1          ODU2E0/1/0/12/3/1             NONE
    100
TenGigECtrlr0/1/0/3/4          ODU2E0/1/0/12/3/4             NONE
    100
TenGigECtrlr0/1/0/4/1          ODU2E0/1/0/12/4/1             NONE
    100
TenGigECtrlr0/1/0/4/4          ODU2E0/1/0/12/4/4             NONE
    100
TenGigECtrlr0/1/0/5/1          ODU2E0/1/0/12/5/1             NONE
    100
TenGigECtrlr0/1/0/5/4          ODU2E0/1/0/12/5/4             NONE
    100
```

```
TenGigECtrlr0/1/0/9/1           ODU2E0/1/0/12/9/1            NONE
    100
TenGigECtrlr0/1/0/9/4           ODU2E0/1/0/12/9/4            NONE
    100
```

Configuring Hybrid Modes Using 40x10G-4x100G-MXP

Table 16: Feature History

Feature Name	Release Information	Description
Hybrid Modes Using 40x10G-4x100G-MXP	Cisco IOS XR Release 7.3.2	With the 40x10G-4x100G-MXP muxponder mode support, you can configure the following hybrid modes: • 20x10G + 2x100G • 10x10G + 3 x 100G With the 40x10G-4x100G-MXP muxponder mode support, you have flexibility to choose a combination of 10G and 100G client rates across different OTN and Ethernet client rates.

Table 17: Feature History

Feature Name	Release Information	Description
Support for 10x10G + 2 x 100G, 20x10G + 1 x 100G, and 30x10G + 1 x 100G Hybrid Modes	Cisco IOS XR Release 7.5.1	You can configure different client rates across the ports depending on the bandwidth requirement, using the following hybrid modes: • 30x10G + 1 x 100G • 10x10G + 2 x 100G • 20x10G + 1 x 100G

With the 40x10G-4x100G-MXP muxponder mode support, you can configure the following hybrid modes:

- 20x10G + 2x100G
- 10x10G + 3 x 100G
- 30x10G + 1 x 100G
- 10x10G + 2 x 100G
- 20x10G + 1 x 100G

For more information on the client to trunk mapping for each of the mode, see #unique_6 unique_6_Connect_42_table_yft_mn1_srb.

Configuring the Muxponder Mode for 20x10G-2x100G

To configure the OTN-XP card in the 20x10G-2x100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type OTU4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 20x10G-2x100G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Wed Jun  2 18:00:58.201 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       MIXED
```

```
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                 Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/1/0/12

                              Traffic Split Percentage

OTU40/1/0/1                       NONE      ODU40/1/0/12/1                      100
OTU2E0/1/0/2/3                    NONE      ODU2E0/1/0/12/2/3
    100
OTU2E0/1/0/2/4                    NONE      ODU2E0/1/0/12/2/4
    100
OTU2E0/1/0/4/1                    NONE      ODU2E0/1/0/12/4/1
    100
OTU2E0/1/0/4/2                    NONE      ODU2E0/1/0/12/4/2
    100
OTU2E0/1/0/4/3                    NONE      ODU2E0/1/0/12/4/3
    100
OTU2E0/1/0/4/4                    NONE      ODU2E0/1/0/12/4/4
    100
OTU2E0/1/0/5/1                    NONE      ODU2E0/1/0/12/5/1
    100
OTU2E0/1/0/5/2                    NONE      ODU2E0/1/0/12/5/2
    100
OTU2E0/1/0/5/3                    NONE      ODU2E0/1/0/12/5/3
    100
OTU2E0/1/0/5/4                    NONE      ODU2E0/1/0/12/5/4
    100
TenGigECtrlr0/1/0/6/1          ODU2E0/1/0/12/6/1                          100
TenGigECtrlr0/1/0/6/2          ODU2E0/1/0/12/6/2                          100
TenGigECtrlr0/1/0/6/3          ODU2E0/1/0/12/6/3                          100
TenGigECtrlr0/1/0/6/4          ODU2E0/1/0/12/6/4                          100
TenGigECtrlr0/1/0/7/1          ODU2E0/1/0/12/7/1                          100
TenGigECtrlr0/1/0/7/2          ODU2E0/1/0/12/7/2                          100
TenGigECtrlr0/1/0/7/3          ODU2E0/1/0/12/7/3                          100
TenGigECtrlr0/1/0/7/4          ODU2E0/1/0/12/7/4                          100
TenGigECtrlr0/1/0/11/3          ODU2E0/1/0/12/11/3                          100
TenGigECtrlr0/1/0/11/4          ODU2E0/1/0/12/11/4                          100
HundredGigECtrlr0/1/0/0        ODU40/1/0/12/0        NONE                        100
```

Configuring the Muxponder Mode for 10 x 10G-3 x 100G

To configure the OTN-XP card in the 10 x 10G and 3 x 100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 21 09:21:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type OTU4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 10 x 10G and 3 x 100G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Wed Jun  2 18:00:58.201 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                  Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/1/0/12

                                Traffic Split Percentage

OTU40/1/0/1                        NONE      ODU40/1/0/12/1                        100
OTU2E0/1/0/2/3                     NONE      ODU2E0/1/0/12/2/3
    100
OTU2E0/1/0/2/4                     NONE      ODU2E0/1/0/12/2/4
    100
OTU2E0/1/0/4/1                     NONE      ODU2E0/1/0/12/4/1
    100
OTU2E0/1/0/4/2                     NONE      ODU2E0/1/0/12/4/2
    100
OTU2E0/1/0/4/3                     NONE      ODU2E0/1/0/12/4/3
    100
OTU2E0/1/0/4/4                     NONE      ODU2E0/1/0/12/4/4
    100
OTU2E0/1/0/5/1                     NONE      ODU2E0/1/0/12/5/1
    100
OTU2E0/1/0/5/2                     NONE      ODU2E0/1/0/12/5/2
    100
OTU2E0/1/0/5/3                     NONE      ODU2E0/1/0/12/5/3
    100
OTU2E0/1/0/5/4                     NONE      ODU2E0/1/0/12/5/4
    100
HundredGigECtrlr0/1/0/0         ODU40/1/0/12/0        NONE                          100
HundredGigECtrlr0/1/0/6         ODU40/1/0/12/6        NONE                          100
```

Configuring Hybrid Modes for 20x10G + 1 x 100G Over 300G

To configure the OTN-XP card in the 20x10G + 1x100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 300G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E}**

commit

The following is a sample in which the OTN-XP card is configured with the 20x10G + 1x100G mode on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#configure
Sun Jul 25 12:43:00.399 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios#configure
Sun Jul 25 12:43:00.399 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type otu2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type otu2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type otu2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample to verify the 20x10G + 1x100G muxponder configuration in the OTN-XP card:

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 0
Sun Jul 25 13:11:01.829 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                  Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/2/0/12
```

```
                                Traffic Split Percentage

OTU20/2/0/1/3                          NONE      ODU20/2/0/12/1/3
    100
OTU20/2/0/2/3                          NONE      ODU20/2/0/12/2/3
    100
OTU20/2/0/4/3                          NONE      ODU20/2/0/12/4/3
    100
OTU20/2/0/4/4                          NONE      ODU20/2/0/12/4/4
    100
OTU2E0/2/0/1/4                         NONE      ODU2E0/2/0/12/1/4
    100
OTU2E0/2/0/2/1                         NONE      ODU2E0/2/0/12/2/1
    100
OTU2E0/2/0/2/2                         NONE      ODU2E0/2/0/12/2/2
    100
TenGigECtrlr0/2/0/1/1          ODU2E0/2/0/12/1/1             NONE
    100
TenGigECtrlr0/2/0/1/2          ODU2E0/2/0/12/1/2             NONE
    100
TenGigECtrlr0/2/0/2/4          ODU2E0/2/0/12/2/4             NONE
    100
TenGigECtrlr0/2/0/4/1          ODU2E0/2/0/12/4/1             NONE
    100
TenGigECtrlr0/2/0/4/2          ODU2E0/2/0/12/4/2             NONE
    100
TenGigECtrlr0/2/0/5/1          ODU2E0/2/0/12/5/1             NONE
    100
TenGigECtrlr0/2/0/5/2          ODU2E0/2/0/12/5/2             NONE
    100
TenGigECtrlr0/2/0/5/3          ODU2E0/2/0/12/5/3             NONE
    100
TenGigECtrlr0/2/0/5/4          ODU2E0/2/0/12/5/4             NONE
    100
TenGigECtrlr0/2/0/9/1          ODU2E0/2/0/12/9/1             NONE
    100
TenGigECtrlr0/2/0/9/2          ODU2E0/2/0/12/9/2             NONE
    100
TenGigECtrlr0/2/0/9/3          ODU2E0/2/0/12/9/3             NONE
    100
TenGigECtrlr0/2/0/9/4          ODU2E0/2/0/12/9/4             NONE
    100
TenGigECtrlr0/2/0/11/1         ODU2E0/2/0/12/11/1            NONE
    100
TenGigECtrlr0/2/0/11/2         ODU2E0/2/0/12/11/2            NONE
    100
HundredGigECtrlr0/2/0/0        ODU40/2/0/12/0        NONE                            100
                        100

RP/0/RP0/CPU0:ios#show lc-module location 0/2 lcmode
Sun Jul 25 15:28:16.324 UTC

Node    Lcmode_Supported    Owner    Running                        Configured
------------------------------------------------------------------------------
0/2                    Yes               CLI      40x10G-4x100G-MXP      40x10G-4x100G-MXP
```

Configuring Hybrid Modes for 30x10G + 1 x 100G Over 400G

To configure the OTN-XP card in the 30x10G + 1x100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E}**

commit

The following is a sample in which the OTN-XP card is configured with 300G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#configure
Sun Jul 25 12:43:00.399 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios#configure
Sun Jul 25 12:43:00.399 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 2 client-type otu2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 3 client-type otu2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type otu2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type otu2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 1 client-type otu2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 2 client-type otu2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type otu2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 3 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 9 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample to verify the 30x10G + 1x100G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 0
Sun Jul 25 13:11:01.829 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
```

```
ARP Snoop Enabled:    FALSE
Client Port                  Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/2/0/12

                                Traffic Split Percentage

OTU20/2/0/0/2                       NONE      ODU20/2/0/12/0/2
    100
OTU20/2/0/1/3                       NONE      ODU20/2/0/12/1/3
    100
OTU20/2/0/2/3                       NONE      ODU20/2/0/12/2/3
    100
OTU20/2/0/4/3                       NONE      ODU20/2/0/12/4/3
    100
OTU20/2/0/4/4                       NONE      ODU20/2/0/12/4/4
    100
OTU2E0/2/0/0/3                      NONE      ODU2E0/2/0/12/0/3
    100
OTU2E0/2/0/1/4                      NONE      ODU2E0/2/0/12/1/4
    100
OTU2E0/2/0/2/1                      NONE      ODU2E0/2/0/12/2/1
    100
OTU2E0/2/0/2/2                      NONE      ODU2E0/2/0/12/2/2
    100
TenGigECtrlr0/2/0/0/1          ODU2E0/2/0/12/0/1             NONE
    100
TenGigECtrlr0/2/0/0/4          ODU2E0/2/0/12/0/4             NONE
    100
TenGigECtrlr0/2/0/1/1          ODU2E0/2/0/12/1/1             NONE
    100
TenGigECtrlr0/2/0/1/2          ODU2E0/2/0/12/1/2             NONE
    100
TenGigECtrlr0/2/0/2/4          ODU2E0/2/0/12/2/4             NONE
    100
TenGigECtrlr0/2/0/3/1          ODU2E0/2/0/12/3/1             NONE
    100
TenGigECtrlr0/2/0/3/2          ODU2E0/2/0/12/3/2             NONE
    100
TenGigECtrlr0/2/0/3/3          ODU2E0/2/0/12/3/3             NONE
    100
TenGigECtrlr0/2/0/3/4          ODU2E0/2/0/12/3/4             NONE
    100
TenGigECtrlr0/2/0/4/1          ODU2E0/2/0/12/4/1             NONE
    100
TenGigECtrlr0/2/0/4/2          ODU2E0/2/0/12/4/2             NONE
    100
TenGigECtrlr0/2/0/5/1          ODU2E0/2/0/12/5/1             NONE
    100
TenGigECtrlr0/2/0/5/2          ODU2E0/2/0/12/5/2             NONE
    100
TenGigECtrlr0/2/0/5/3          ODU2E0/2/0/12/5/3             NONE
    100
TenGigECtrlr0/2/0/5/4          ODU2E0/2/0/12/5/4             NONE
    100
TenGigECtrlr0/2/0/9/1          ODU2E0/2/0/12/9/1             NONE
    100
TenGigECtrlr0/2/0/9/2          ODU2E0/2/0/12/9/2             NONE
    100
TenGigECtrlr0/2/0/9/3          ODU2E0/2/0/12/9/3             NONE
    100
TenGigECtrlr0/2/0/9/4          ODU2E0/2/0/12/9/4             NONE
    100
TenGigECtrlr0/2/0/11/1         ODU2E0/2/0/12/11/1            NONE
    100
TenGigECtrlr0/2/0/11/2         ODU2E0/2/0/12/11/2            NONE
```

```
    100
HundredGigECtrlr0/2/0/6         ODU40/2/0/12/0        NONE                            100
                       100

RP/0/RP0/CPU0:ios#show lc-module location 0/2 lcmode
Sun Jul 25 15:28:16.324 UTC

Node    Lcmode_Supported    Owner    Running                          Configured
------------------------------------------------------------------------------------
0/2                    Yes               CLI      40x10G-4x100G-MXP      40x10G-4x100G-MXP
```

Configuring Hybrid Modes for 10x10G + 2 x 100G Over 300G

To configure the OTN-XP card in the 10x10G + 2x100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 300G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | OTU2 | OTU2E}**

commit

The following is a sample in which the OTN-XP card is configured with 300G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#configure
Sun Jul 25 12:43:00.399 UTC
RP/0/RP0/CPU0:ios(config)#
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type otu4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type OTU2e
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample to verify the 10x10G + 2x100G muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/2 mxponder-slice 0
Sun Jul 25 14:57:40.806 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                  Mapper/Trunk Port   Peer/Trunk Port       CoherentDSP0/2/0/12

                                  Traffic Split Percentage
```

```
OTU40/2/0/1                             NONE      ODU40/2/0/12/1                          100
OTU2e0/2/0/2/4                           NONE      ODU2e0/2/0/12/2/4
      100
OTU20/2/0/4/3                           NONE      ODU20/2/0/12/4/3
    100
OTU20/2/0/4/4                           NONE      ODU20/2/0/12/4/4
    100
TenGigECtrlr0/2/0/2/3           ODU2E0/2/0/12/2/3             NONE
    100
TenGigECtrlr0/2/0/4/1           ODU2E0/2/0/12/4/1             NONE
    100
TenGigECtrlr0/2/0/4/2           ODU2E0/2/0/12/4/2             NONE
    100
TenGigECtrlr0/2/0/5/1           ODU2E0/2/0/12/5/1             NONE
    100
TenGigECtrlr0/2/0/5/2           ODU2E0/2/0/12/5/2             NONE
    100
TenGigECtrlr0/2/0/5/3           ODU2E0/2/0/12/5/3             NONE
    100
TenGigECtrlr0/2/0/5/4           ODU2E0/2/0/12/5/4             NONE
    100
HundredGigECtrlr0/2/0/0         ODU40/2/0/12/0        NONE                              100

RP/0/RP0/CPU0:ios#show lc-module location 0/2 lcmode
Sun Jul 25 15:28:16.324 UTC

Node    Lcmode_Supported    Owner    Running                         Configured
------------------------------------------------------------------------------
0/2                    Yes               CLI      40x10G-4x100G-MXP      40x10G-4x100G-MXP
```

Configuring Hybrid Mode for 10x10G + 1x100G Over 200G

Table 18: Feature History

Feature Name	Release Information	Description
10x10G + 1x100G Hybrid Mode for OTN-XP Card	Cisco IOS XR Release 7.7.1	A new hybrid mode 10x10G + 1x100G over 200G trunk rate is introduced for OTN-XP card. This mode is configurable on both slice 1 and slice 0. This feature provides you the flexibility to choose a combination of 10G and 100G client rates simultaneously on both slices of the OTN-XP card.

To configure the OTN-XP card in the 10 x 10G and 1 x 100G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 200G

client-port-rate *client-port-number* **client-type {100GE | OTU4}**

client-port-rate *client-port-number* **lane** *lane-number* **client-type {10GE | 100GE}**

commit

Example

The following is a sample in which the OTN-XP card is configured with 200G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 22 10:51:44.460 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Example

The following is a sample in which the OTN-XP card is configured with 200G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#config
Tue Apr 22 11:01:44:55.250 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 11 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 10 x 10G and 1 x 100G muxponder configuration in the OTN-XP card.

```
RRP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 0
Fri Jun 17 15:55:43.520 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/12
```

```
                                Traffic Split Percentage

TenGigECtrlr0/2/0/2/3           ODU2E0/2/0/12/2/3                           100
TenGigECtrlr0/2/0/2/4           ODU2E0/2/0/12/2/4                           100
TenGigECtrlr0/2/0/4/1           ODU2E0/2/0/12/4/1                           100
TenGigECtrlr0/2/0/4/2           ODU2E0/2/0/12/4/2                           100
TenGigECtrlr0/2/0/4/3           ODU2E0/2/0/12/4/3                           100
TenGigECtrlr0/2/0/4/4           ODU2E0/2/0/12/4/4                           100
TenGigECtrlr0/2/0/5/1           ODU2E0/2/0/12/5/1                           100
TenGigECtrlr0/2/0/5/2           ODU2E0/2/0/12/5/2                           100
TenGigECtrlr0/2/0/5/3           ODU2E0/2/0/12/5/3                           100
TenGigECtrlr0/2/0/5/4           ODU2E0/2/0/12/5/4                           100
HundredGigECtrlr0/2/0/0         ODU40/2/0/12/0                       100

RP/0/RP0/CPU0:ios#
```

Configuring the Muxponder Mode for 200G on OTN-XP Card

Table 19: Feature History

Feature Name	Release Information	Description
Muxponder Configuration for 200G Trunk with QPSK and 8QAM Modulation	Cisco IOS XR Release 7.3.2	The OTN-XP card supports up to 200G trunk rate with QPSK and 8QAM modulation using CFP2. This feature enhances the signal reachability with reduced noise and can support the 50GHz network. Commands modified: • hw-module (OTN-XP Card)

To configure the OTN-XP card in the 200G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 200G

commit

The following is a sample configuration of 200G trunk rate on the mxponder-slice 0 mode for OTN-XP card:

```
RP/0/RP0/CPU0:ios#config
Wed Jun  2 17:17:59.409 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample configuration of 200G trunk rate on the mxponder-slice 1 mode for OTN-XP card:

```
RP/0/RP0/CPU0:ios#config
Wed Jun  2 17:17:59.409 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 xponder-capabilities mxponder-slice 0
Wed Jun  2 17:02:33.088 UTC

Location: 0/1

Trunk-Port(s): 12

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          1, 6, 7, 10

Trunk-bandwidth: 400G
Client-port                      Supported client rates
     1                            OTU4, 100GE
     6                            OTU4, 100GE
     7                            OTU4, 100GE
     10                           OTU4, 100GE, 400GE

Trunk-bandwidth: 300G
Client-port                      Supported client rates
     6                            OTU4, 100GE
     7                            OTU4, 100GE
     10                           OTU4, 100GE

Trunk-bandwidth: 200G
Client-port                      Supported client rates
     7                            OTU4, 100GE
     10                           OTU4, 100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 xponder-capabilities mxponder-slice 1
Wed Jun  2 17:02:50.133 UTC

Location: 0/1

Trunk-Port(s): 13

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          0, 4, 5, 8

Trunk-bandwidth: 400G
Client-port                      Supported client rates
     0                            OTU4, 100GE
     4                            OTU4, 100GE
     5                            OTU4, 100GE
     8                            OTU4, 100GE, 400GE

Trunk-bandwidth: 300G
Client-port                      Supported client rates
     4                            OTU4, 100GE
     5                            OTU4, 100GE
     8                            OTU4, 100GE

Trunk-bandwidth: 200G
Client-port                      Supported client rates
```

```
5                                 OTU4, 100GE
8                                 OTU4, 100GE
```

Configuring 8QAM Modulation for 200G Muxponder Mode

By default, QPSK is the modulation format, when you configure 200G trunk rate.

The following operating modes are supported on the DP04CFP2 coherent pluggable module:

Table 20: DP04CFP2 Supported Modes

Network Configuration Mode	Client Type	Trunk Rate	Data Path	Line Framing	FEC Type	Modulation Format	BPS	Baud Rate (GBd)	Pulse Shaping	Mode Type
200G-FOIC2-oFEC-QPSK-1-S (Default mode)	2xFOIC1.2	200G	FlexO Str	FlexO-2	oFEC	QPSK	2	63.1	1	Standard
200G-FOIC2-oFEC-8QAM-1-E	2xFOIC1.2	200G	FlexO Str	FlexO-2	oFEC	8QAM	3	42.1	1	Enhanced

Use the following commands to change the modulation format to 8QAM:

configure

controller optics *Rack/Slot/Instance/Port* **bits-per-symbol** 3

commit

The following is a sample in which 8QAM modulation is configured.

```
RP/0/RP0/CPU0:ios#config
Wed Jun  2 17:21:59.409 UTC
RP/0/RP0/CPU0:ios(config)#controller optics0/1/0/12 bits-per-symbol 3
RP/0/RP0/CPU0:ios(config-optics)#commit
```

Verifying the 8QAM Modulation Configuration

```
RP/0/RP0/CPU0:ios#show controllers optics 0/1/0/12
Wed Jun  2 17:17:29.652 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  <Unknown> DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
         Wavelength=1552.524nm

         Alarm Status:
         -------------
         Detected Alarms: None
```

```
        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 1
        HIGH-TX-PWR = 0            LOW-TX-PWR = 1
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 1
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = 1.47 dBm
        RX Signal Power = 17.67 dBm
        Frequency Offset = 82 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -31.5           0.0          0.0
        Tx Power Threshold(dBm)          3.0      -12.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 96000 ps/nm
        Configured CD lower Threshold = -96000 ps/nm
        Configured OSNR lower Threshold = 13.70 dB
        Configured DGD Higher Threshold = 67.00 ps
        Baud Rate =  42.2082633972 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 2 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 72.00 ps^2
        Optical Signal to Noise Ratio = 34.10 dB
        SNR = 18.40 dB
        Polarization Dependent Loss = 1.20 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 2.00 ps

Transceiver Vendor Details

        Form Factor            : Not set
        Fiber Connector Type: Not Set
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius


AINS Soak               : None
AINS Timer              : 0h, 0m
```

```
 AINS remaining time      : 0 seconds
```

Configuring the Muxponder Mode for 300G on OTN-XP Card

Table 21: Feature History

Feature Name	Release Information	Description
Muxponder Configuration for 300G Trunk with 8QAM Modulation	Cisco IOS XR Release 7.3.2	The OTN-XP card supports up to 300G trunk rate with 8QAM modulation using CFP2. This feature improves the signal reachability with decreased noise. Commands modified: • hw-module (OTN-XP Card)

To configure the OTN-XP card in the 300G muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 300G

commit

The following is a sample configuration of 300G trunk rate on the mxponder-slice 0 mode for OTN-XP card:

```
RP/0/RP0/CPU0:ios#config
Wed Jun  2 17:17:59.409 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample configuration of 300G trunk rate on the mxponder-slice 1 mode for OTN-XP card:

```
RP/0/RP0/CPU0:ios#config
Wed Jun  2 17:17:59.409 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 xponder-capabilities mxponder-slice 0
Wed Jun  2 17:02:33.088 UTC

Location: 0/1

Trunk-Port(s): 12
```

```
Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          1, 6, 7, 10

Trunk-bandwidth: 400G
Client-port                       Supported client rates
     1                             OTU4, 100GE
     6                             OTU4, 100GE
     7                             OTU4, 100GE
     10                            OTU4, 100GE, 400GE

Trunk-bandwidth: 300G
Client-port                       Supported client rates
     6                             OTU4, 100GE
     7                             OTU4, 100GE
     10                            OTU4, 100GE

Trunk-bandwidth: 200G
Client-port                       Supported client rates
     7                             OTU4, 100GE
     10                            OTU4, 100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 xponder-capabilities mxponder-slice 1
Wed Jun  2 17:02:50.133 UTC

Location: 0/1

Trunk-Port(s): 13

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          0, 4, 5, 8

Trunk-bandwidth: 400G
Client-port                       Supported client rates
     0                             OTU4, 100GE
     4                             OTU4, 100GE
     5                             OTU4, 100GE
     8                             OTU4, 100GE, 400GE

Trunk-bandwidth: 300G
Client-port                       Supported client rates
     4                             OTU4, 100GE
     5                             OTU4, 100GE
     8                             OTU4, 100GE

Trunk-bandwidth: 200G
Client-port                       Supported client rates
     5                             OTU4, 100GE
     8                             OTU4, 100GE
```

By default, 8QAM is the modulation format, when you configure 300G trunk rate.

The following operating mode is supported on the DP04CFP2 coherent pluggable module:

Table 22: DP04CFP2 Supported Modes

Network Configuration Mode	Client Type	Trunk Rate	Data Path	Line Framing	FEC Type	Modulation Format	BPS	Baud Rate (GBd)	Pulse Shaping	Mode Type
300G-FOIC3-oFEC-8QAM-1-S (Default mode)	3xFOIC1.2	300G	FlexO Str	FlexO-3	oFEC	8QAM	3	63.1	1	Standard

The following sample shows the supported client rates for 300G trunk rate and the provisioning status of slice 1:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice 1
Fri Jul 23 16:04:42.279 UTC

Location:             0/1
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/13
                                Traffic Split Percentage
HundredGigECtrlr0/1/0/4         ODU40/1/0/13/8                 100
HundredGigECtrlr0/1/0/5         ODU40/1/0/13/5                 100
HundredGigECtrlr0/1/0/8         ODU40/1/0/13/8                 100
```

The following sample shows the default 8QAM modulation format for the 300G trunk rate:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/1/0/12
Wed Jun  2 17:17:29.652 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  <Unknown> DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
         Wavelength=1552.524nm

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 1
         HIGH-TX-PWR = 0            LOW-TX-PWR = 1
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 1
         WVL-OOL = 0                MEA  = 0
```

```
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = 1.47 dBm
        RX Signal Power = 17.67 dBm
        Frequency Offset = 82 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -31.5           0.0          0.0
        Tx Power Threshold(dBm)          3.0      -12.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 96000 ps/nm
        Configured CD lower Threshold = -96000 ps/nm
        Configured OSNR lower Threshold = 13.70 dB
        Configured DGD Higher Threshold = 67.00 ps
        Baud Rate =  42.2082633972 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 2 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 72.00 ps^2
        Optical Signal to Noise Ratio = 34.10 dB
        SNR = 18.40 dB
        Polarization Dependent Loss = 1.20 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 2.00 ps

Transceiver Vendor Details

        Form Factor            : Not set
        Fiber Connector Type: Not Set
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius



AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Configuring the Muxponder Mode for 4x100GE-MXP-DD

Table 23: Feature History

Feature Name	Release Information	Description
4X100GE MXP modes with QDD ZRP for OTN-XP Card	Cisco IOS XR Release 7.3.2	On the OTN-XP card, you can configure a single 4x100GE payload that isreceived over the client port as a 400GE signal over DWDM on the line side. The card improves efficiency, performance, and flexibility for customer networksallowing 4x100GE client transport over 400GE WDM wavelength.

From Release 7.3.2 onwards, you can configure the 4x100GE-MXP-DD muxponder mode on the OTN-XP card.

Restrictions for Port Group Mapping

The following table explains about the port mapping when the mxponder-slice 0 is at the near end and is connected to the mxponder-slice 1 at the far end:

Table 24: Port Group Mapping for Shared-Client-Group-Bandwidth

Slice Configuration - Client Port	Shared-Client-Group-Bandwidth	Shared-Group-Client-Ports
Slice 0	400G	1, 6, 7, 10
Slice 1	400G	8, 0, 4, 5

Table 25: Port Group Mapping for Trunk-Bandwidth

Trunk-Bandwidth	Slice Configuration - Client Port	Supported Client Rates	Client-Ports
400G	Slice 0	100G	1, 6, 7, 10
	Slice 1	100G	8, 0, 4, 5
300G	Slice 0	100G	6, 7, 10
	Slice 1	100G	8, 4, 5
200G	Slice 0	100G	7, 10
	Slice 1	100G	4, 5

The traffic flows from the near-end slice-0 to the far-end slice-1 client ports:

- The port 1 traffic reaches port 8

- The port 6 traffic reaches port 0
- The port 7 traffic reaches port 4
- The port 10 traffic reaches port 5

The following table describes the QSFP DD trunk port to the slice-0 client port and slice-1 client port mapping:

Table 26: QSFP DD Trunk Port to the Slice-0 and Slice-1 Client Port Mapping

QSFP-DD Trunk Port	Slice 0 - Client Port	Slice 1 - Client Port
0	Port 10	Port 5
1	Port 7	Port 4
2	Port 6	Port 0
3	Port 1	Port 8

Configuring the Muxponder Mode for 4x100GE-MXP-DD

To configure the OTN-XP card in the 4x100GE-MXP-DD muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type 100GE**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 0 mode.

```
RP/0/RP0/CPU0:ios#configure
Tue Jun 15 20:20:17.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 6 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 7 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Tue Jun 15 20:20:57.532 UTC
```

Verifying the Muxponder Configuration

The following is a sample to verify the 4x100GE-MXP-DD muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder
Tue Jun 15 20:21:46.587 UTC

Location:             0/0
```

```
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/0/0/11
                               Traffic Split Percentage

HundredGigECtrlr0/0/0/1                 -                           100
HundredGigECtrlr0/0/0/6                 -                           100
HundredGigECtrlr0/0/0/7                 -                           100
HundredGigECtrlr0/0/0/10                -                           100
```

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#configure
Tue Jun 15 20:22:13.981 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Verifying the Muxponder Configuration

The following is a sample to verify the 4x100GE-MXP-DD muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 1
Tue Jun 15 20:23:06.217 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/0/0/9
                               Traffic Split Percentage

HundredGigECtrlr0/0/0/0                 -                           100
HundredGigECtrlr0/0/0/4                 -                           100
HundredGigECtrlr0/0/0/5                 -                           100
HundredGigECtrlr0/0/0/8                 -                           100


RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder
Tue Jun 15 20:23:46.650 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/0/0/11
```

```
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/1                 -                           100
HundredGigECtrlr0/0/0/6                 -                           100
HundredGigECtrlr0/0/0/7                 -                           100
HundredGigECtrlr0/0/0/10                -                           100


Location:             0/0
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/9
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/0                 -                           100
HundredGigECtrlr0/0/0/4                 -                           100
HundredGigECtrlr0/0/0/5                 -                           100
HundredGigECtrlr0/0/0/8                 -                           100
```

Configuring the Muxponder Mode for 2x100GE-MXP-DD

Table 27: Feature History

Feature Name	Release Information	Description
2X100GE MXP modes with QDD ZRP for OTN-XP Card	Cisco IOS XR Release 7.5.1	On the OTN-XP card, you can configure two 2x100GE payloads that are received over the client port as a 200GE signal over DWDM on the line side. The 2x100GE-MXP-DD muxponder mode improves efficiency, performance, and flexibility for customer networks allowing 2x100GE client transport over 200GE WDM wavelength.

From Release 7.5.1 onwards, you can configure the 2x100GE-MXP-DD muxponder mode on the OTN-XP card.

Note The LC mode must be configured to 4x100GE-MXP-DD on the OTN-XP card before you perform this configuration.

Two slices of 2x100GE-MXP-DD can be configured with the same LC mode on the OTN-XP card.

Restrictions on the port group mapping exist when the mxponder-slice 0 is at the near end and is connected to the mxponder-slice 1 at the far end. For more details, see Restrictions for Port Group Mapping , on page 77.

To configure the OTN-XP card in the 2x100GE-MXP-DD muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 200G

client-port-rate *client-port-number* **client-type 100GE**

commit

Example

The following is a sample in which the OTN-XP card is configured with 200G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#configure
Tue Jun 15 20:20:17.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Tue Jun 15 20:20:57.532 UTC
```

Verifying the Supported Client Rates for each Trunk Rate

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 0
Fri Jul 23 15:35:43.059 UTC

Location: 0/0

Trunk-Port(s): 11

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          1, 6, 7, 10

Trunk-bandwidth: 400G
Client-port                      Supported client rates
     1                            100GE
     6                            100GE
     7                            100GE
     10                           100GE

Trunk-bandwidth: 300G
Client-port                      Supported client rates
     6                            100GE
     7                            100GE
     10                           100GE

Trunk-bandwidth: 200G
Client-port                      Supported client rates
     7                            100GE
     10                           100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 1
Wed Sep 15 00:30:47.433 UTC

Location: 0/0

Trunk-Port(s): 9
```

```
Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          8, 0, 4, 5

Trunk-bandwidth: 400G
Client-port                      Supported client rates
     8                           100GE
     0                           100GE
     4                           100GE
     5                           100GE

Trunk-bandwidth: 300G
Client-port                      Supported client rates
     8                           100GE
     4                           100GE
     5                           100GE

Trunk-bandwidth: 200G
Client-port                      Supported client rates
     4                           100GE
     5                           100GE
```

Verifying the Running Configuration

The following is a sample to verify the provisioned slice and client port information for 2x100GE-MXP-DD muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show running config
hw-module location 0/2
 mxponder-slice 0
  trunk-rate 200G
  client-port-rate 1 client-type 100GE
  client-port-rate 7 client-type 100GE
 !
hw-module location  0/1
 mxponder-slice 1
  trunk-rate 200G
  client-port-rate 4 client-type 100GE
  client-port-rate 5 client-type 100GE
 !
!
```

Verifying the Muxponder Configuration

The following is a sample to verify the 2x100GE-MXP-DD muxponder configuration in the OTN-XP card for mxponder-slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 0
Fri Jul 23 16:04:42.279 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/11
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/7                 -                           100
HundredGigECtrlr0/0/0/10                -                           100
```

The following is a sample to verify the 2x100GE-MXP-DD muxponder configuration in the OTN-XP card for mxponder-slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 1
Tue Jun 15 20:21:46.587 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/11
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/4                 -                           100
HundredGigECtrlr0/0/0/5                 -                           100
```

Configuring the Muxponder Mode for 3x100GE-MXP-DD

Table 28: Feature History

Feature Name	Release Information	Description
3x100GE MXP modes with QDD ZRP for OTN-XP Card	Cisco IOS XR Release 7.5.1	On the OTN-XP card, you can configure two 3x100GE payloads that are received over the client port as a 300GE signal over DWDM on the line side. The 3x100GE-MXP-DD muxponder mode improves efficiency, performance, and flexibility for customer networks allowing 3x100GE client transport over 300GE WDM wavelength.

From Release 7.5.1 onwards, you can configure the 3x100GE-MXP-DD muxponder mode on the OTN-XP card.

Note The LC mode must be configured to 4x100GE-MXP-DD on the OTN-XP card before you perform this configuration.

Two slices of 3x100GE-MXP-DD can be configured with the same LC mode on the OTN-XP card.

Restrictions on the port group mapping exist when the mxponder-slice 0 is at the near end and is connected to the mxponder-slice 1 at the far end. For more details, see Restrictions for Port Group Mapping , on page 77.

To configure the OTN-XP card in the 3x100GE-MXP-DD muxponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 300G

client-port-rate *client-port-number* **client-type 100GE**

commit

Example

The following is a sample in which the OTN-XP card is configured with 300G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#configure
Tue Jun 15 20:20:17.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Tue Jun 15 20:20:57.532 UTC
```

Verifying the Supported Client Rates for each Trunk Rate

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 0
Fri Jul 23 15:35:43.059 UTC

Location: 0/0

Trunk-Port(s): 11

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          1, 6, 7, 10

Trunk-bandwidth: 400G
Client-port                      Supported client rates
     1                           100GE
     6                           100GE
     7                           100GE
     10                          100GE

Trunk-bandwidth: 300G
Client-port                      Supported client rates
     6                           100GE
     7                           100GE
     10                          100GE

Trunk-bandwidth: 200G
Client-port                      Supported client rates
     7                           100GE
     10                          100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 1
Wed Sep 15 00:30:47.433 UTC

Location: 0/0

Trunk-Port(s): 9

Port Group Restrictions:
```

```
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          8, 0, 4, 5

Trunk-bandwidth: 400G
Client-port                       Supported client rates
     8                             100GE
     0                             100GE
     4                             100GE
     5                             100GE

Trunk-bandwidth: 300G
Client-port                       Supported client rates
     8                             100GE
     4                             100GE
     5                             100GE

Trunk-bandwidth: 200G
Client-port                       Supported client rates
     4                             100GE
     5                             100GE
```

Verifying the Running Configuration

The following is a sample to verify the provisioned slice and client port information for 3x100GE-MXP-DD muxponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show running config
hw-module location 0/2
 mxponder-slice 0
  trunk-rate 300G
  client-port-rate 1 client-type 100GE
  client-port-rate 7 client-type 100GE
  client-port-rate 10 client-type 100GE
 !
hw-module location  0/1
 mxponder-slice 1
  trunk-rate 300G
  client-port-rate 4 client-type 100GE
  client-port-rate 5 client-type 100GE
  client-port-rate 8 client-type 100GE
 !
!
```

Verifying the Muxponder Configuration

The following is a sample to verify the 3x100GE-MXP-DD muxponder configuration in the OTN-XP card for mxponder-slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 0
Tue Jun 15 20:21:46.587 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/0/0/11
                               Traffic Split Percentage

HundredGigECtrlr0/0/0/1                 -                           100
HundredGigECtrlr0/0/0/7                 -                           100
```

```
HundredGigECtrlr0/0/0/10                -                              100
```

The following is a sample to verify the 3x100GE-MXP-DD muxponder configuration in the OTN-XP card for mxponder-slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 1
Tue Jun 15 20:21:46.587 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/11
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/8                 -                              100
HundredGigECtrlr0/0/0/4                 -                              100
HundredGigECtrlr0/0/0/5                 -                              100
```

Configuring the Transponder Mode for 400GE-TXP-DD

Table 29: Feature History

Feature Name	Release Information	Description
400GE TXP mode with QDD ZRP for OTN-XP Card	Cisco IOS XR Release 7.5.1	On the OTN-XP card, you can configure two 400GE payloads that are received over the client port as a 400GE signal over DWDM on the line side. The 400GE-TXP-DD muxponder mode improves efficiency, performance, and flexibility for customer networks allowing 400GE client transport over 400GE WDM wavelength.

From Release 7.5.1 onwards, you can configure the 400GE-TXP-DD transponder mode on the OTN-XP card.

Note The LC mode must be configured to 400GE-TXP-DD on the OTN-XP card before you perform this configuration.

Two slices of 400GE-TXP-DD can be configured with the same LC mode on the OTN-XP card.

Restrictions on the port group mapping exist when the mxponder-slice 0 is at the near end and is connected to the mxponder-slice 1 at the far end. For more details, see Restrictions for Port Group Mapping , on page 77.

To configure the OTN-XP card in the 400GE-TXP-DD transponder mode, use the following commands:

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number*

trunk-rate 400G

client-port-rate *client-port-number* **client-type 100GE**

commit

Example

The following is a sample in which the OTN-XP card is configured with 400G trunk rate on the mxponder-slice 1 mode.

```
RP/0/RP0/CPU0:ios#configure
Tue Jun 15 20:20:17.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Tue Jun 15 20:20:57.532 UTC
```

Verifying the Supported Client Rates for each Trunk Rate

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 0
Wed Oct 27 16:14:35.693 UTC

Location: 0/0

Trunk-Port(s): 11

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          10

Trunk-bandwidth: 400G
Client-port                       Supported client rates
     10                           100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 xponder-capabilities mxponder-slice 1
Wed Oct 27 16:16:37.524 UTC

Location: 0/0

Trunk-Port(s): 9

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     400G                          8

Trunk-bandwidth: 400G
Client-port                       Supported client rates
     8                            400GE
```

Verifying the Running Configuration

The following is a sample to verify the provisioned slice and client port information for 400GE-TXP-DD transponder configuration in the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show running config
hw-module location 0/0
```

```
 mxponder-slice 0
  trunk-rate 400G
  client-port-rate 10 client-type 100GE
 !
hw-module location  0/0
 mxponder-slice 1
  trunk-rate 400G
  client-port-rate 8 client-type 100GE
 !
!
```

Verifying the Muxponder Configuration

The following is a sample to verify the 400GE-TXP-DD transponder configuration in the OTN-XP card for mxponder-slice 0.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 0
Fri Jul 23 16:04:42.279 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       400GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/11
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/1                 -                           100
HundredGigECtrlr0/0/0/6                 -                           100
HundredGigECtrlr0/0/0/7                 -                           100
HundredGigECtrlr0/0/0/10                -                           100
```

The following is a sample to verify the 400GE-TXP-DD transponder configuration in the OTN-XP card for mxponder-slice 1.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder-slice 1
Tue Jun 15 20:21:46.587 UTC

Location:             0/0
Slice ID:             1
Client Bitrate:       400GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/9
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/8                 -                           100
HundredGigECtrlr0/0/0/0                 -                           100
HundredGigECtrlr0/0/0/4                 -                           100
HundredGigECtrlr0/0/0/5                 -                           100
```

Verifying the Client Ethernet Controller Status

The following is a sample to verify the Client Ethernet Controller Status of the 400GE-TXP-DD transponder configuration in the OTN-XP card.

```
P/0/RP0/CPU0:ios#show controller hundredGigECtrlr 0/0/0/1
Fri Jul 23 16:07:11.541 UTC
Operational data for interface HundredGigECtrlr0/0/0/1:


State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled


Phy:
    Media type: Not known


Autonegotiation disabled.


Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

Configuring Inverse Muxponder on OTN-XP Card for 400GE Client

Table 30: Feature History

Feature Name	Release Information	Feature Description
Inverse Muxponder Configuration on OTN-XP Card	Cisco IOS XR Release 7.3.2	The OTN-XP card supports inverse multiplexing for 400GE client over 2x200G CFP2 trunk ports. This feature allows you to split the 400GE client signal and carry it over 2x200G trunks thereby increasing the ease of signal reachability. Commands modified: • hw-module (OTN-XP Card) • controller coherentDSP

You can configure the OTN-XP card to support inverse multiplexing for 400GE client over 2x200G CFP2 trunk ports. To configure the inverse muxponder datapath, use the following commands:

configure

hw-module location *location*

mxponder

trunk-rate 200G

client-port-rate *client-port-number* **client-type 400GE**

commit

end

The following sample configures inverse muxponder for 400G:

```
RP/0/RP0/CPU0:ios #Configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 10 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following sample verifies the inverse muxponder configuration:

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder
Wed Jun  9 23:16:59.478 UTC

Location:             0/0
Client Bitrate:       400GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port         CoherentDSP0/0/0/12
CoherentDSP0/0/0/13
                               Traffic Split Percentage

FourHundredGigECtrlr0/0/0/10  ODU-FLEX0/0/0/12/10          50                  50
```

Alarm Correlation in Inverse Muxponder

When any service-affecting alarm is raised on the trunk port 12 or 13, the alarms are reported on the ports as follows:

- Port 12—Flexo alarms (FLEXO_LOS, FLEXO_LOL, FLEXO_GIDM, FLEXO_FMM, FLEXO_LOF, and FLEXO_LOM) and OTU alarms (LOD, AIS, LOS, LOM, LOD, and TIM)
- Port 13—Flexo alarms except Flexo MM and GIDM.

Both ports 12 and 13 go down when any service-affecting alarm is raised.

Example:

Shut down the trunk port 12:

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/3/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#shutdown
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Thu Sep 30 14:12:48.416 UTC
```

The following sample verifies that when trunk port 12 is shut down, LOS alarm is raised and the trunk port 13 also goes down.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/12
Thu Sep 30 14:12:54.604 UTC

Port                                          : CoherentDSP 0/2/0/12
```

```
Controller State                                : Down
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 200.0Gb/s

Alarm Information:
LOS = 2 LOF = 0 LOM = 0
OOF = 1 OOM = 0 AIS = 1
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 1
FLEXO-LOF = 0
Detected Alarms                                 : LOS

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 0.00 dB

Q-Margin                                        : 0.00dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 1, 2,
        Rx GID                                  : 0
        RX IID                                  : 0, 0,

Flexo Peers Information:
        Controller                              : CoherentDSP0_2_0_13
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds


RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/13
Thu Sep 30 14:12:59.330 UTC

Port                                            : CoherentDSP 0/2/0/13
Controller State                                : Down
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 200.0Gb/s

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
```

```
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 1
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 15.80 dB

Q-Margin                                        : 9.50dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 3, 4,
        Rx GID                                  : 1
        RX IID                                  : 3, 4,

Flexo Peers Information:
        Controller                              : CoherentDSP0_2_0_12
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

You can perform the following configurations on the DSPcontroller ports:

- Flexo Parameter Update on Inverse Muxponder Configuration on the OTN-XP Card, on page 232
- Configure TTI on Inverse Muxponder Configuration on the OTN-XP Card
- Configure Loopback on Inverse Muxponder Configured on the OTN-XP Card

2x100GE-TXP-MXP on OTN-XP Card

Table 31: Feature History Table

Feature Name	Release Information	Description
NCS1K4-OTN-XP Line Card Interoperability	Cisco IOS XR Release 7.8.1	This feature allows the NCS1K4-OTN-XP card with CFP2-DCO 200G pluggable to interoperate with the NCS2K-400G-XP and NCS4K-4H-OPW-QC2 cards. Using the new 2x100GE-TXP-MXP mode for the OTN-XP card, you can configure 1x100GE or 2x100GE payloads over 100G or 200G DWDM on the line side, respectively. The interoperation improves customer networks' efficiency, performance, and flexibility, allowing 100GE-TXP traffic over 100G DWDM or 2x100GE-MXP traffic over 200G DWDM wavelengths on each slice.

2x100GE-TXP-MXP mode supports the following features:

- PRBS
- GCCO
- Loopbacks
- Laser Squelch
- TTI
- LLDP
- AINS
- Modulation type:
  - QPSK for 100G
  - 16-QAM for 200G

Table 32: 100GE-TXP Mapping in 2XTXP-MXP Configuration

Slice	Client	Trunk Port
0	0	12

Slice	Client	Trunk Port
1	4	13

Table 33: 2x100GE-MXP Mapping in 2XTXP-MXP Configuration

Slice	Client	Trunk Port
0	0 & 1	12
1	4 & 5	13

- Trunk rate—100G and 200G
- Trunk optics—CFP2-WDM-DETS-1HL

The following is a sample to configure OTN-XP card in the LC mode 2x100GE-TXP-MXP transponder mode:

```
RP/0/RP0/CPU0:ios#configure
Fri Feb  4 16:52:18.021 UTC
RP/0/RP0/CPU0:ios(config)#lc-module location 0/2 lcmode 2x100GE-TXP-MXP
RP/0/RP0/CPU0:ios(config)#commit
```

To verify the OTN-XP card in the 2x100GE-TXP-MXP transponder mode, use the following commands:

```
RP/0/RP0/CPU0:ios#sh lc-module location 0/0 lcmode
Sat Nov 12 22:58:47.917 UTC

Node    Lcmode_Supported    Owner    Running                Configured
------------------------------------------------------------------------------
0/0     Yes                 CLI      2x100GE-TXP-MXP           2x100GE-TXP-MXP
```

The following is a sample in which the OTN-XP card is configured with 100G trunk rate on the mxponder-slice 0.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 100G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the OTN-XP card can be verified with 100G trunk rate on the mxponder-slice 0.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 0
Mon Oct 10 14:06:41.807 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       100G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/2/0/12
                               Traffic Split Percentage

HundredGigECtrlr0/2/0/0        ODU40/2/0/12/0                        100
```

The following is a sample in which the OTN-XP card is configured with 200G trunk rate on the mxponder-slice 0.

```
LC slot# 2, client port 0 & 1, and client type 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the OTN-XP card can be verified with 200G trunk rate on the mxponder-slice 0.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 0
Mon Oct 10 14:06:41.807 UTC

Location:             0/2
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/12
                                Traffic Split Percentage

HundredGigECtrlr0/2/0/0         ODU40/2/0/12/0                        100
HundredGigECtrlr0/2/0/1         ODU40/2/0/12/1                        100
```

The following is a sample in which the OTN-XP card is configured with 200G trunk rate on the mxponder-slice 1.

```
LC slot# 2, client port 0 & 1, and client type 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2 mxponder-slice 1
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

The following is a sample in which the OTN-XP card can be verified with 200G trunk rate on the mxponder-slice 1.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 mxponder-slice 1
Mon Oct 10 14:06:41.807 UTC

Location:             0/2
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/13
                                Traffic Split Percentage

HundredGigECtrlr0/2/0/4         ODU40/2/0/13/4                        100
HundredGigECtrlr0/2/0/5         ODU40/2/0/13/5                        100
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 0

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 xponder-capabilities mxponder-slice 0
Mon Oct 10 14:32:15.881 UTC

Location: 0/2

Trunk-Port(s): 12
```

```
Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     200G                          0
     200G                          1

Trunk-bandwidth: 100G
Client-port                       Supported client rates
     0                            100GE

Trunk-bandwidth: 200G
Client-port                       Supported client rates
     0                            100GE
     1                            100GE
```

The following is the sample output for verifying the supported client rates for each trunk rate configured in muxponder slice 1

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/2 xponder-capabilities mxponder-slice 1
Mon Oct 10 14:32:15.881 UTC

Location: 0/2

Trunk-Port(s): 12

Port Group Restrictions:
Shared-Client-Group-Bandwidth    Shared-Group-Client-Ports
     200G                          4
     200G                          5

Trunk-bandwidth: 100G
Client-port                       Supported client rates
     4                            100GE

Trunk-bandwidth: 200G
Client-port                       Supported client rates
     4                            100GE
     5                            100GE
```

Interoperation between OTN-XP and NCS2K and NCS4K:

Interoperation between GCC0 on the line side and PRBS supports the following features:

- PRBS (between OTN-XP Card & NCS4K-4H-OPW-QC2 mapper ODU4s)
- GCC0 between trunk interfaces (200G/100G)
- TTI
- Laser Squelch
- LLDP
- PRBS
- Loopbacks
- AINS
- GCC0

Note GCC0 interoperability between the OTN-XP card and 400G-XP-LC/NCS4K-4H-OPW-QC2 line card is not supported on 100G trunk rate in Slice 0.

Table 34: Hardware Module Configuration with Client to Trunk Mapping

OTN-XP Card-NCS4K-4H-OPW-QC2 Card						
	OTN-XP Card Side			NCS4K-4H-OPW-QC2 Card Side		
Data path	Slice	Trunk Port	Client Port	Slice	Trunk Port	Client Port
100G	0	12	0	N/A	11	0
200G	0	12	0	N/A	11	0
	0		1	N/A		6

OTN-XP Card - 400G-XP-LC Card						
	OTN-XP Card Side			400G-XP-LC Side		
Data path	Slice	Trunk Port	Client Port	Slice	Trunk Port	Client Port
100G	0	12	0	4	12	10
200G	0	12	0	3	12	9
	0		1	4		10

Note Interoperation between OTN-XP and NCS2K works only with 11.30 version.

Note Flex-coherent is not supported, so no GID/IID configuration and flex-o alarms are supported.

Note High switching time is observed with 100G trunks with CD minimum(-10000 ps/nm) or maximum(10000 ps/nm) configured as default.

2-QDD-C Line Card

Table 35: Feature History

Feature Name	Release Information	Description
NCS1K4-2-QDD-C-K9 C-Band Line Card	Cisco IOS XR Release 7.3.1	NCS 1004 supports the NCS1K4-2-QDD-C-K9 C-Band line card. The card has eight client ports (QSFP28 and QSFP-DD) and two DWDM dual sub-channel module trunk ports. Each trunk port is capable of 200, 300, and 400 Gbps line rate with fine control of modulation format, baud-rate, and forward error correction. The trunk ports are software configurable. The line card supports module and slice configurations. Command added: • **controller fourHundredGigECtrlr**

The following section describes the supported configurations and procedures to configure the card modes on the 2-QDD-C line card.

Limitations for 2-QDD-C

- Flex Ethernet is not supported.
- A single 400GE cannot be split and use as 4x 100GE due to hardware limitations.

Unsupported Features for 2-QDD-C in R731

The following features are not supported in R7.3.1 for 2-QDD-C card:

- OTU4 client
- Layer 1 encryption
- GCC remote node management
- Line rates of 50G, 100G, 150G, 250G, and 350G

2-QDD-C Card Modes

The 2-QDD-C line cards support module and slice configurations.

The line cards have two trunk ports (0 and 1) and 8 client ports (2 through 9) each. You can configure the line card in two modes:

- Muxponder—In this mode, both trunk ports are configured with the same trunk rate. The client-to-trunk mapping is in a sequence in vertical order.
- Muxponder slice—In this mode, each trunk port is configured independent of the other with different trunk rates. The client-to-trunk mapping is fixed in vertical order. For Trunk 0, the client ports are 2 through 5. For Trunk 1, the client ports are 6 through 9.

Sub 50G Configuration

Table 36: Feature History

Feature Name	Release Information	Description
Support for n x 50G Rate	Cisco IOS XR Release 7.5.1	You can now configure sub 50G muxponder mode in a combination of trunk and client rates for 2-QDD-C cards.

You can configure sub 50G muxponder mode in the following combination of trunk and client rates:

- 100GE Muxponder mode:
  - 1x100GE and 2x50G
  - 3x100GE and 2x150G
  - 5x100GE and 2x250G
  - 7x100GE and 2x350G
- OTU4 Muxponder mode:
  - 1xOTU4 and 2x50G
  - 3xOTU4 and 2x150G
  - 5xOTU4 and 2x250G
  - 7xOTU4 and 2x350G

The following table displays the port configuration for the supported data rates.

Trunk Data Rate (per trunk)	Total Configured Data rate	Trunk Ports	Client Ports for Trunk 0 (100G)	Shared Client Port (50G per trunk)	Client Ports for Trunk 1 (100G)
50G	100G	0, 1	-	2	-
150G	300G	0, 1	2	3	4
250G	500G	0, 1	2, 3	4	5, 6

Trunk Data Rate (per trunk)	Total Configured Data rate	Trunk Ports	Client Ports for Trunk 0 (100G)	Shared Client Port (50G per trunk)	Client Ports for Trunk 1 (100G)
350G	700G	0, 1	2, 3, 4	5	6, 7, 8

From Release 7.5.2, 2-QDD-C cards support an alternate port configuration for Sub 50G (split client port mapping) that you configure using CLI. The following table displays the port configuration for the supported data rates.

Trunk Data Rate (per trunk)	Total Configured Data rate	Trunk Ports	Client Ports for Trunk 0 (100G)	Shared Client Port (50G per trunk)	Client Ports for Trunk 1 (100G)
50G	100G	0, 1	-	5	-
150G	300G	0, 1	2	5	6
250G	500G	0, 1	2, 3	5	6, 7
350G	700G	0, 1	2, 3, 4	5	6, 7, 8

For information on how to configure split client port mapping, see Configure Split Client Port Mapping, on page 10

Coupled Mode Restrictions

The following restrictions apply to the coupled mode configuration:

- Both trunk ports must be configured with the same bits-per-symbol or baud rate and must be sent over same fiber and direction.
- The chromatic dispersion must be configured to the same value for both trunk ports.
- When trunk internal loopback is configured, it must be done for both trunk ports. Configuring internal loopback on only one trunk results in traffic loss.
- Fault on a trunk port of a coupled pair may cause errors on all clients including those running only on the unaffected trunk port.

Supported Data Rates for 2-QDD-C Card

The following table displays the client and trunk ports that are enabled for the muxponder configuration.

Trunk Data Rate	Card Support	Client Data Rate	Client Optics	Trunk Ports	Client Ports
200	2-QDD-C	100GE, OTU4	QSFP-28	0, 1	2, 3, 4, 5
300	2-QDD-C	100GE, OTU4	QSFP-28	0, 1	2, 3, 4, 5, 6, 7
400	2-QDD-C	100GE, OTU4	QSFP-28	0, 1	2, 3, 4, 5, 6, 7, 8, 9
400	2-QDD-C	400GE	QSFP-DD	0, 1	4,8

The following table displays the client and trunk ports that are enabled for the muxponder slice 0 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	2-QDD-C	100GE, OTU4	0	2
200	2-QDD-C	100GE, OTU4	0	2, 3
300	2-QDD-C	100GE, OTU4	0	2, 3, 4
400	2-QDD-C	100GE, OTU4	0	2, 3, 4, 5
400	2-QDD-C	400GE	0	4

The following table displays the client and trunk ports that are enabled for the muxponder slice 1 configuration.

Trunk Data Rate	Card Support	Client Data Rate	Trunk Ports	Client Ports
100	2-QDD-C	100GE, OTU4	1	6
200	2-QDD-C	100GE, OTU4	1	6, 7
300	2-QDD-C	100GE, OTU4	1	6, 7, 8
400	2-QDD-C	100GE, OTU4	1	6, 7, 8, 9
400	2-QDD-C	400GE	1	8

The following table displays the trunk parameter ranges for the 2-QDD-C card.

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
150G	27%	1.453125	4.335938	24.02079	71.67494
200G	27%	2	4.40625	31.51	69.43
250G	27%	2.414063	6	28.93129	71.9069
300G	27%	2.8984375	6	34.7175497	71.8681352
350G	27%	3.382813	6	40.5038	71.84047
400G	27%	3.8671875	6	46.2900663	71.8197392
150G	15%	1.320313	3.9375	24.02079	71.67494
200G	15%	1.7578125	5.25	24.02079115	71.74209625
250G	15%	2.195313	6	26.27274	71.80592
300G	15%	3.8203125	6	31.52728839	49.51525048
350G	15%	3.070313	6	36.78184	71.87901
400G	15%	3.8671875	6	42.03638452	71.9018782

Note The recommended value for 6 BPS for corresponding line rates are listed below:

Trunk Payload	FEC	BPS	GBd
300G	27%	6	34.7175
350G	27%	6	40.5038
400G	15%	6	42.0364

For more information on the QDD-C card, see the data sheet.

Configuring the Card Mode for 2-QDD-C Card

From R7.3.1, you can configure the 2-QDD-C line card in the module (muxponder) or slice configuration (muxponder slice).

To configure the card in the muxponder mode, use the following commands:

- **configure**

  hw-module location *location* **mxponder client-rate {100GE | OTU4 }**

  hw-module location *location* **mxponder trunk-rate** {**100G** | **150G** | **200G** | **250G** | **300G** | **350G** | **400G** }

  commit

- **configure**

  hw-module location *location* **mxponder client-rate { 400GE}**

  hw-module location *location* **mxponder trunk-rate { 200G** | **400G** }

  commit

To configure the card in the muxponder slice mode, use the following commands.

configure

hw-module location *location* **mxponder-slice** *mxponder-slice-number* **client-rate { 100GE | 400GE}**

hw-module location *location* **mxponder-slice** *mxponder-slice-number* **trunk-rate** { **100G** | **200G** | **300G** | **400G**}

commit

Examples

The following is a sample in which the card is configured in the muxponder mode with a 400G trunk rate.

```
RP/0/RP0/CPU0:ios#config
Tue Oct 15 01:24:56.355 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder client-rate 100GE
```

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 0 mode with a 400G trunk rate.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder slice 1 mode with a 400G trunk rate.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 client-rate 100GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample in which the card is configured in the muxponder mode with a 400GE trunk rate.

```
RP/0/RP0/CPU0:west#configure
Thu Oct  7 11:43:01.914 IST
RP/0/RP0/CPU0:west(config)#hw-module location 0/2 mxponder trunk-rate 4
400G  450G
RP/0/RP0/CPU0:west(config)#hw-module location 0/2 mxponder trunk-rate 400G
RP/0/RP0/CPU0:west(config)#hw-module location 0/2 mxponder client-rate 400GE
RP/0/RP0/CPU0:west(config)#commit
```

Configuring Mixed Client Traffic Mode

Table 37: Feature History

Feature Name	Release Information	Description
Mixed Client Traffic Mode Configuration	Cisco IOS XR Release 7.5.1	You can now configure the client traffic mode on each trunk port of the 2-QDD-C card independently. This feature provides flexibility to carry both OTN and Ethernet client traffic on the 2-QDD-C card at the same time across two slices.

Feature Name	Release Information	Description
Enhanced Mixed Mode Client Traffic Configuration on 2-QDD-C Card	Cisco IOS XR Release 7.10.1	This feature is an upgrade of earlier mixed-mode configuration on 2-QDD-C card that required reprovision of all client ports to switch between provisioning Ethernet or OTU interfaces. This enhancement makes the 2-QDD-C card smarter to delegate OTU and Ethernet traffic in the same slice simultaneously, avoiding the need to reprovision the client ports. This enhancement provides you with greater flexibility to configure both Ethernet and OTU interfaces for different client ports on the same slice in the 2-QDD-C card without disrupting the client traffic. Enable this enhancement with the following keywords on the **hw-module** command: • **client-port-rate <2-5> | <6-9>** • **client-type <100GE | OTU4>**

You can configure the client traffic mode on each trunk in a line card independently. This provides flexibility for the same card to carry both OTN and Ethernet client traffic at the same time across 2 slices.

100G, 200G, and 300G trunk rates are supported on both the slices (slice 0 and slice 1) with different client modes (100GE/OTU4).

From R7.10.1, you can configure both Ethernet and OTU interfaces on different client ports on each trunk in the 2-QDD-C line card independently. This enhancement gives you flexibility on the same 2-QDD-C line card to carry both OTN and Ethernet client traffic at the same time in the same slice for each trunk rates.

An additional 400G trunk rate is supported on both the slices (slice 0 and slice 1) with different client modes (100GE/OTU4).

Different-Slice Mixed Client Traffic Mode

To configure the card in mixed client traffic mode, use the following commands:

```
hw-module location R/S
mxponder-slice 0
  trunk-rate [100G|200G|300G|400G]
  client-rate [100GE|OTU4]
!
 mxponder-slice 1
  trunk-rate [100G|200G|300G|400G]
  client-rate [OTU4|100GE]
!
!
```

The following is a sample in which the card is configured with mixed client rates in the muxponder slice 0 and 1 mode.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 23 06:10:22.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-rate OTU4 trunk-rate
 400G
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 1 client-rate 100GE trunk-rate
 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following configuration is a sample of the mixed client traffic mode in different slices.

Example 1:

```
hw-module location 0/0
 mxponder-slice 0
  trunk-rate 400G
  client-rate OTU4
 !
 mxponder-slice 1
  trunk-rate 400G
  client-rate 100GE
 !
!
```

Verifying Card Configuration

```
RP/0/RP0/CPU0:ios#show hw-module location 0/0 mxponder

Location:             0/0
Slice ID:             0
Client Bitrate:       OTU4
Trunk  Bitrate:       400G
Status:               Provisioned
Client Port                     Peer/Trunk Port            CoherentDSP0/0/0/0
                                Traffic Split Percentage

OTU40/0/0/2                     ODU40/0/0/0/1                     100
OTU40/0/0/3                     ODU40/0/0/0/2                     100
OTU40/0/0/4                     ODU40/0/0/0/3                     100
OTU40/0/0/5                     ODU40/0/0/0/4                     100


Location:             0/0
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
Client Port                     Peer/Trunk Port            CoherentDSP0/0/0/1
                                Traffic Split Percentage
HundredGigECtrlr0/0/0/6         ODU40/0/0/1/1                       100
HundredGigECtrlr0/0/0/7         ODU40/0/0/1/2                       100
HundredGigECtrlr0/0/0/8         ODU40/0/0/1/3                       100
HundredGigECtrlr0/0/0/9         ODU40/0/0/1/4                       100
```

The following configuration is a sample in which both the slices use the same client mode.

Example 2:

```
hw-module location 0/3
 mxponder
  trunk-rate 350G
  client-rate 100GE
 !
!
```

Verifying Card Configuration

```
RP/0/RP0/CPU0:ios#show hw-module location 0/3 mxponder
Fri Nov 26 12:21:16.174 UTC

Location:             0/3
Client Bitrate:       100GE
Trunk  Bitrate:       350G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/3/0/0
CoherentDSP0/3/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/3/0/2         ODU40/3/0/0/1                          100
       0
HundredGigECtrlr0/3/0/3         ODU40/3/0/0/2                          100
       0
HundredGigECtrlr0/3/0/4         ODU40/3/0/0/3                          100
       0
HundredGigECtrlr0/3/0/5         ODU40/3/0/0/4                           50
      50
HundredGigECtrlr0/3/0/6         ODU40/3/0/1/1                            0
     100
HundredGigECtrlr0/3/0/7         ODU40/3/0/1/2                            0
     100
HundredGigECtrlr0/3/0/8         ODU40/3/0/1/3                            0
     100
```

Same-Slice Mixed Client Traffic Mode

To configure the card in mixed client traffic mode in same slice, use the following commands:

```
hw-module location R/S
mxponder-slice 0
  trunk-rate [100G|200G|300G|400G]
  client-port-rate 2 client-type <100GE|OTU4>
!
!
 mxponder-slice 1
  trunk-rate [100G|200G|300G|400G]
client-port-rate 2 client-type <100GE|OTU4>
!
!
```

The following is a sample in which the card is configured with mixed client rates in the muxponder slice 0 mode.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 23 06:10:22.227 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-port-rate 2
client-type OTU4 trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 client-port-rate 3
client-type 100GE trunk-rate 400G
RP/0/RP0/CPU0:ios(config)#commit
```

The following configuration is a sample of the mixed client port rate in same slice.

```
hw-module location 0/0
 mxponder-slice 0
  trunk-rate 200G
  client-port-rate 2 client-type 100G
  client-port-rate 3 client-type otu4
 !
 mxponder-slice 1
```

```
  trunk-rate 400G
  client-port-rate 4 client-type 100G
  client-port-rate 8 client-type otu4
 !
!
```

QXP Card

Table 38: Feature History

Feature Name	Release Information	Description
Digital-to-Analog (DAC) support for NCS1K4-QXP-K9 card	Cisco IOS XR Release 7.8.1	DAC support is now enabled on NCS1K4-QXP-K9 card for 2x100G, 3x100G, 4x100G, and 400G operating modes. DAC helps in the optimization of digital-to-analog signal conversion.
Forward Error Correction (FEC) support on QXP card for Ethernet controllers	Cisco IOS XR Release 7.8.1	FEC is now supported by the 100GE Ethernet controller on the NCS1K4-QXP-K9 card. FEC is supported for all pluggables except QSFP-100G-LR4-S and ONS-QSFP28-LR4.
Idle insertion on NCS1K4-QXP-K9 card	Cisco IOS XR Release 7.8.1	Idle insertion refers to the idles that are inserted in the traffic stream from the trunk port to the client port for the duration of the configured holdoff-time. Whenever a fault occurs on the trunk port, you can hold the propagation of local faults using idle insertion. Idle insertion is now enabled on 100GE or 400GE controllers for the NCS1K4-QXP-K9 card.
NCS1K4-QXP-K9 card support for 2x100GE and 3x100GE operating mode configurations	Cisco IOS XR Release 7.8.1	Support is enabled for 2x100GE and 3x100GE operating mode configurations through Open Config and CLI on NCS1K4-QXP-K9 card.

Feature Name	Release Information	Description
Cisco 400G QSFP-DD High-Power (Bright) Optical module support on QXP card	Cisco IOS XR Release 7.10.1	The QXP card now supports Cisco 400G QSFP-DD High-Power (Bright) Optical module. This pluggable provides higher output power compared to other ZR Pluggables (QDD-400G-ZR-S and QDD-400G-ZRP-S). This allows users to interconnect directly to add/drop ports without additional amplifiers which improves performance and saves cost for end to end services.
OTN Datapath on QXP card trunk ports	Cisco IOS XR Release 7.10.1	The QXP line card now supports OTN standard based trunk transmission with Cisco 400G QSFP-DD High-Power (Bright) Optical Module. This allows trunk connections from the QXP card to be connected to other OpenROADMcompliant trunk devices.

The NCS1K4-QXP-K9 3.2T QSFP-DD DCO Transponder Line Card has eight client ports (QSFP-DD) and eight trunk ports (QSFP-DD ZR+). Each line card supports up to 3.2 Tbps traffic. The client rates that are supported are 400GE, 4x100GE, and 100GE Ethernet only. The modulation formats supported are 16 QAM for 400GE Txp/4x100GE Mxp and QPSK for 100GE Txp.

The QXP line card provides up to 16 QSFP-DD ports (eight QSFP-DD client ports and eight QSFP-DD trunk ports). The supported operating modes are:

- 400GE-TXP
- 4X100GE MXP
- 100GE TXP

From R7.8.1 onwards the operating modes listed below are supported.

- 3x100GE MXP
- 2x100GE MXP

The QXP card has 8 slices. Each slice consists of one client and one trunk port with a slice capacity of 400G. The total capacity is 3.2T.

Table 39: Slice and Port Mapping on the QXP Card

Slice	Trunk Port	Client Port
0	0	1
1	2	3

Slice	Trunk Port	Client Port
2	4	5
3	6	7
4	8	9
5	10	11
6	12	13
7	14	15

Restriction The QXP card uses only the first 6 client and trunk ports (first 6 slices) when installed in an NCS1004 chassis.

Note

- When you use OPENROADM trunk mode by configuring the **trunk-mode OR** command, use only alternate slices on the QXP card. Either use slices 0, 2, 4, 6 or 1, 3, 5, 7.

Supported Data Rates for QXP Card

The following table displays the client and trunk ports that are enabled for transponder and muxponder modes.

Operating mode	Card Support	Client Data Rate	Client Optics	Trunk Ports	Client Ports
400GE-TXP	QXP Card	400G	QDD-400G-DR4-S, QDD-400G-FR4-S, QDD-AOCxM, QDD-4X100G-FR-S	0,2,4,6,8,10,12,14	1,3,5,7,9,11,13,15
4X100GE MXP	QXP Card	4X100G Break out	QDD-400G-DR4-S, QDD-4X100G-LR-S, QDD-4X100G-FR-S	0,2,4,6,8,10,12,14	1,3,5,7,9,11,13,15
3X100GE MXP	QXP Card	3X100G Break out	QDD-400G-DR4-S, QDD-4X100G-LR-S, QDD-4X100G-FR-S	0,2,4,6,8,10,12,14	1,3,5,7,9,11,13,15
2X100GE MXP	QXP Card	2X100G Break out	QDD-400G-DR4-S, QDD-4X100G-LR-S, QDD-4X100G-FR-S	0,2,4,6,8,10,12,14	1,3,5,7,9,11,13,15
100GE TXP	QXP Card	100G	QSFP28-100G-LR4, QSFP28-100G-LR-S, QSFP28-100G-DR-S, QSFP28-100G-FR-S	0,2,4,6,8,10,12,14	1,3,5,7,9,11,13,15

Configure 400G Transponder Mode

Use the following commands to configure and provision 400G TXP.

hw-module location *location*

mxponder-slice *slice-number*

trunk-rate 400G

trunk-mode [ZR | OR]

client-port-rate *port-number***client-type 400GE**

The following is a sample configuration of configuring a 400G TXP.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 400GE
```

The following is a sample output of **show hw-module location** *location* **mxponder-slice** *slice-number* when configured in 400G Transponder Mode.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/0 mxponder-slice 0
Sat Jun 25 21:32:58.799 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       400GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                    Mapper/Trunk Port          CoherentDSP0/0/0/0
                               Traffic Split Percentage

FourHundredGigECtrlr0/0/0/1                 -                           100
```

Note The **trunk-mode** command allows you to choose between OTN and ethernet traffic on the trunk port.

Configure 100G Transponder Mode

Use the following commands to configure and provision 100G TXP.

hw-module location *location*

mxponder-slice *slice-number*

trunk-rate 100G

client-port-rate *port-number***client-type 100GE**

The following is a sample configuration of configuring a 100G TXP.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 100G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 100GE
```

The following is a sample output of **show hw-module location** *location* **mxponder-slice** *slice-number* when configured in 100G Transponder Mode.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/0 mxponder-slice 0
Sat Jun 25 21:58:15.417 UTC

Location:             0/0
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       100G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/0/0/0
                                Traffic Split Percentage

HundredGigECtrlr0/0/0/1                 -                            100
```

Configure 400G Muxponder Mode

Use the following commands to configure and provision 400G MXP.

hw-module location *location*

mxponder-slice *slice-number*

trunk-rate 400G

client-port-rate *port-number* **lane** *lane-number***client-type 100GE**

The following is a sample configuration of configuring a 400G MXP.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 4 client-type 100GE
```

The following is a sample output of **show hw-module location** *location* **mxponder-slice** *slice-number* when configured in 400G MXP Mode.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/3 mxponder-slice 1
Sat Jun 25 23:03:20.823 UTC

Location:             0/3
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/3/0/2
                                Traffic Split Percentage

HundredGigECtrlr0/3/0/3/1               -                            100
HundredGigECtrlr0/3/0/3/2               -                            100
HundredGigECtrlr0/3/0/3/3               -                            100
HundredGigECtrlr0/3/0/3/4               -                            100
```

Configure 2x100G Muxponder Mode

Use the following commands to configure and provision 2x100G MXP.

hw-module location *location*

mxponder-slice *slice-number*

trunk-rate 200G

client-port-rate *port-number* **lane** *lane-number***client-type 100GE**

The following is a sample configuration of configuring a 2x100G MXP.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 100GE
```

The following is a sample output of **show hw-module location** *location* **mxponder-slice** *slice-number* when configured in 2x100G MXP Mode.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/3 mxponder-slice 1
Sat Jun 25 23:03:20.823 UTC

Location:             0/3
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       200G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/3/0/2
                                Traffic Split Percentage

HundredGigECtrlr0/3/0/3/1               -                           100
HundredGigECtrlr0/3/0/3/2               -                           100
```

Configure 3x100G Muxponder Mode

Use the following commands to configure and provision 3x100G MXP.

hw-module location *location*

mxponder-slice *slice-number*

trunk-rate 300G

client-port-rate *port-number* **lane** *lane-number***client-type 100GE**

The following is a sample configuration of configuring a 3x100G MXP.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 300G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 2 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 lane 3 client-type 100GE
```

The following is a sample output of **show hw-module location** *location* **mxponder-slice** *slice-number* when configured in 3x100G MXP Mode.

```
RP/0/RP0/CPU0:ios#sh hw-module location 0/3 mxponder-slice 1
Sat Jun 25 23:03:20.823 UTC

Location:             0/3
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/3/0/2
                                Traffic Split Percentage

HundredGigECtrlr0/3/0/3/1               -                          100
HundredGigECtrlr0/3/0/3/2               -                          100
HundredGigECtrlr0/3/0/3/3               -                          100
```

DAC Supported Modes for NCS1K4-QXP-K9 Card

From R7.8.1 DAC support is enabled on the NCS1K4-QXP-K9 card for 2x100G, 3x100G, 4x100G, and 400G operating modes. The following table provides the details of the respective DAC rates for the different trunk rates for NCS1K4-QXP-K9 card.

Table 40: DAC Supported Data Rates for NCS1K4-QXP-K9 Card

Trunk Rate	Modulation Format	Default Value	Modified DAC Supported
100G	QPSK	1x1.50	N/A
200G	QPSK	1x1	1x1.50
200G	8QAM	1x1.25	N/A
200G	16-QAM	1x1.25	N/A
300G	8-QAM	1x1	1x1.25,1x1.50,1x2
400G	16-QAM	1x1	1x1.50

The following example changes the DAC rate to 1x1.5 on an optics controller.

```
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/0
RP/0/RP0/CPU0:ios(config-Optics)#dac-Rate 1x1.50
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

Note

- Changing the DAC turns the laser Off and then back on for the optics. This is a traffic impacting operation.
- The DAC rate configuration must match on both ends of a connection.

FEC Support on QXP Card

From R7.8.1 onwards, FEC support is enabled on 100GE Ethernet controller for all pluggables except the LR4 pluggables such as QSFP-100G-LR4-S and ONS-QSFP28-LR4 for the NCS1K4-QXP-K9 card. For more information on FEC refer to the FEC, on page 130 section.

Cisco 400G QSFP-DD High-Power (Bright ZR+) Optical Module Support on QXP Card

From R7.10.1, QXP card supports Cisco 400G QSFP-DD High-Power (Bright) Optical Modules. The Bright QSFP-DD operates as Ethernet or OTN transponder.

Use the following commands to configure OTN data path on the Bright ZR+ pluggable optical modules. The **trunk-mode OR** refers to OpenROADM.

hw-module location *location*

mxponder-slice 1 *slice-number*

trunk-mode OR

trunk-rate *rate*

Use the following commands to configure Ethernet data path on the Bright ZR+ pluggable optical modules.

hw-module location *location*

mxponder-slice 1 *slice-number*

trunk-mode ZR

trunk-rate *rate*

The following is a sample configuration of configuring a 4x100G OTN trunk on a Bright ZR+ pluggable.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder-slice 4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-mode OR
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  client-port-rate 9 lane 1 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  client-port-rate 9 lane 2 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  client-port-rate 9 lane 3 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  client-port-rate 9 lane 4 client-type 100GE
```

The following is a sample configuration of configuring Ethernet trunk on a Bright ZR+ pluggable.

```
RP/0/RP0/CPU0:ios#configure
Tue Apr 11 19:29:20.132 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder-slice 4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-mode ZR
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
```

The following is a sample configuration of setting 0dBm transmit power on a Bright ZR+ pluggable.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/2
RP/0/RP0/CPU0:ios(config-Optics)#transmit-power 0
Thu Mar  9 13:02:30.662 UTC
WARNING! Changing TX power can impact traffic
RP/0/RP0/CPU0:ios(config-Optics)#commit
Thu Mar  9 13:02:31.566 UTC
```

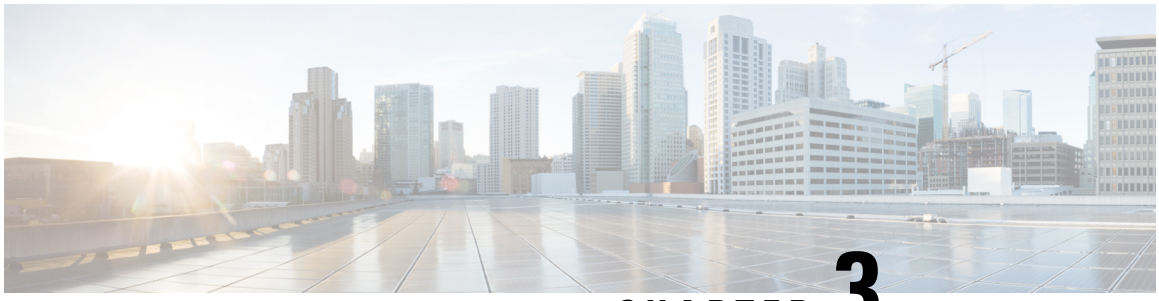
The following is a sample output of the **show controllers optics** command, with the transmit power set to 0 dBm.

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/8
Thu Apr 13 13:54:33.163 UTC
 Controller State: Up
 Transport Admin State: In Service
 Laser State: On
 LED State: Green
```

```
Optics Status
        Optics Type:  QSFP-DD DWDM
        DWDM carrier Info: C BAND, MSA ITU Channel=49, Frequency=193.70THz,
        Wavelength=1547.715nm
        Alarm Status:
        -------------
        Detected Alarms: None
        LOS/LOL/Fault Status:
        Alarm Statistics:
        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 4
        HIGH-TX-PWR = 0            LOW-TX-PWR = 1
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 4
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.00 dBm
        RX Power = -10.50 dBm
        RX Signal Power = -10.35 dBm
        Frequency Offset = 199 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -24.5           0.0          0.0
        Tx Power Threshold(dBm)          0.0      -16.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = 0.00 dBm
        Configured CD High Threshold = 52000 ps/nm
        Configured CD lower Threshold = -52000 ps/nm
        Configured OSNR lower Threshold = 21.10 dB
        Configured DGD Higher Threshold = 67.00 ps
```

Table 41: Operating Modes Supported for Bright ZR+ Pluggable Modules on QXP Card

Operating mode	Modulation	FEC
4x100GE MXP	16-QAM	CFEC
4x100GE MXP	16-QAM	OFEC
3x100GE MXP	8QAM	OFEC
2x100GE MXP	QPSK	OFEC
400GE TXP	16-QAM	CFEC
400GE TXP	16-QAM	OFEC
100GE TXP	QPSK	OFEC



CHAPTER 3

Configuring Controllers

There are three types of controllers for the line card. The controllers are the optics controller, the ethernet controller, and the coherent DSP controller. This chapter describes the procedures used to configure these controllers.

Note Unless otherwise specified, “line cards” refers to 1.2T and 1.2TL line cards.

- AINS, on page 118
- FEC, on page 130
- Laser Squelching, on page 143
- Idle Insertion, on page 150
- Idle Insertion for Ethernet Controllers, on page 157
- LLDP Drop, on page 159
- Link Layer Discovery Protocol (LLDP) Support on Management Interface, on page 163
- Daisy Chain Support on Management Ports, on page 167
- DHCP Client, on page 169
- MAC Address Snooping on Client Ports, on page 172
- Transmit Shutdown, on page 174
- Loopback, on page 178
- Restore Factory Settings, on page 195
- Headless Mode, on page 197
- Trail Trace Identifier, on page 197
- Chromatic dispersion, on page 206
- Transmit Power, on page 208
- Laser Bias Current High Threshold, on page 211
- Differential Group Delay Threshold, on page 213
- Optical Signal to Noise Ratio, on page 215
- Chromatic Dispersion Threshold, on page 217
- Receive Power Threshold, on page 219
- Transmit Power Threshold, on page 221
- Frequency, on page 223
- Pseudo Random Binary Sequence, on page 223
- FlexO GID and IID, on page 231

• FPD, on page 236
• Automatic Protection Switching (APS) on OTN XP Card, on page 237

AINS

The Automatic-In-Service (AINS) feature allows the controller to automatically move to the automatic-in-service state after the maintenance window is completed. A soak time period is associated with the AINS state. The controller automatically moves to the In-Service state after the soak time period is completed. During the AINS maintenance window, alarms are not propagated to the EMS/NMS monitoring system.

You can configure AINS on the client ports of the card.

AINS States

The following table lists the AINS states.

State	Description
None	AINS is not enabled on the controller or the soak time period is complete.
Pending	AINS is configured on the controller. However, the soak time period has not started because either the primary state of controller is in Shutdown, Admin down, or Not ready state or the secondary state is in Maintenance state. AINS can also move to Pending state if alarms are raised during the soak time period.
Running	AINS is enabled on the controller. The primary state of the controller is Up and the secondary state is AINS.

If there are any service-affecting alarms when AINS is runring on ethernet or optics controllers, the AINS state moves to Pending state. When the alarms are cleared, the AINS state moves to Running state.

The AINS soak time period restarts when there are line card reloads, XR reloads, line card warm reloads, power cycles, or alarm conditioning.

Soak Time Period

You can configure the soak time period to be between 1 minute to 48 hours.

All alarms are suppressed during the AINS state. When the optical and ethernet alarms are raised on the port during the soak time period, the AINS state moves to Pending. These alarms are not displayed in the output of the **show alarms brief card location 0/RP0/CPU0 active** command but in the output of the **show alarms brief card location 0/RP0/CPU0 conditions** command. When all the alarms clear, the soak time period starts, and the AINS state moves to Running. When the soak time period expires, the port moves to IS state.

Configuring AINS

To configure AINS on a muxponder, use the following command:

configure

hw-module location *location* **mxponder client-port-ains-soak hours** *hours* **minutes** *minutes*

commit

The following is a sample in which all client ports are configured with AINS with soak time period specified to be 15 minutes.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder client-port-ains-soak hours 0
minutes 15
RP/0/RP0/CPU0:ios(config)#commit
```

To configure AINS on a muxponder slice, use the following command:

configure

hw-module location *location* **mxponder-slice** *slice-number* **client-port-ains-soak hours** *hours* **minutes** *minutes*

commit

The following is a sample in which slice 0 client ports are configured with AINS with soak time period specified to be 40 minutes.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder-slice 0 client-port-ains-soak
hours 0 minutes 40
RP/0/RP0/CPU0:ios(config)#commit
```

Disabling AINS

To disable AINS on all muxponder client ports, set the hours and minutes to 0. Use the following commands:

configure

hw-module location *location* **mxponder client-port-ains-soak hours** *hours* **minutes** *minutes*

commit

The following is a sample in which AINS is disabled on all client ports.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder client-port-ains-soak hours 0
minutes 0
RP/0/RP0/CPU0:ios(config)#commit
```

To disable AINS on a muxponder slice, set the hours and minutes to 0. Use the following command:

configure

hw-module location *location* **mxponder-slice** *slice-number* **client-port-ains-soak hours** *hours* **minutes** *minutes*

commit

The following is a sample in which AINS is disabled on all client ports of slice 0.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder-slice 0 client-port-ains-soak
hours 0 minutes 0
RP/0/RP0/CPU0:ios(config)#commit
```

Displaying the AINS Configuration

The AINS Soak field in the output indicates the current state of AINS. The current state can be None, Pending, or Running. The Total Duration field indicates the total soak time period that is configured. The Remaining Duration field indicates the soak time that remains, after which, the AINS state moves to None.

This example displays the ethernet controller statistics with AINS Soak in running state.

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/1/0/2

Thu Feb 21 19:52:55.001 UTC
Operational data for interface HundredGigECtrlr0/1/0/2:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: Running
      Total Duration: 0 hour(s) 15 minute(s)
      Remaining Duration: 0 hour(s) 5 minute(s) 37 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

This example displays the ethernet controller statistics with AINS Soak in pending state.

```
RP/0/RP0/CPU0:ios#show controllers HuC 0/0/0/2
Thu Mar 12 13:52:12.129 UTC
Operational data for interface HundredGigECtrlr0/0/0/2:

State:
    Administrative state: enabled
    Operational state: Down (Reason: State undefined)
    LED state: Red On
    Maintenance: Disabled
    AINS Soak: Pending
      Total Duration: 0 hour(s) 30 minute(s)
      Remaining Duration: 0 hour(s) 30 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known
    Alarms:
        Current:
            Local Fault
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 9

Autonegotiation disabled.
```

```
Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

This example displays the optics controller statistics with AINS Soak in running state.

```
RP/0/RP0/CPU0:ios#show controller optics 0/1/0/3

Thu Feb 21 19:45:41.088 UTC

 Controller State: Up

 Transport Admin State: Automatic In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  Grey optics

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 0
         HIGH-TX-PWR = 0            LOW-TX-PWR = 0
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 0
         WVL-OOL = 0                MEA  = 0
         IMPROPER-REM = 0
         TX-POWER-PROV-MISMATCH = 0

         Performance Monitoring: Enable

         THRESHOLD VALUES
         ----------------

         Parameter                High Alarm  Low Alarm  High Warning  Low Warning
         -----------------------  ----------  ---------  ------------  -----------
         Rx Power Threshold(dBm)          4.9      -12.0           0.0          0.0
         Tx Power Threshold(dBm)          3.5      -10.1           0.0          0.0
         LBC Threshold(mA)                N/A        N/A          0.00         0.00

         LBC High Threshold = 98 %
         Polarization parameters not supported by optics

        Total TX Power = 6.39 dBm

        Total RX Power = 5.85 dBm
```

```
        Lane  Laser Bias    TX Power    RX Power  Output Frequency
        ----  ----------  ----------  ----------  ----------------
          1      75.0 %    0.59 dBm    0.63 dBm  230.43 THz
          2      68.6 %    0.06 dBm   -0.68 dBm  230.43 THz
          3      69.0 %    0.26 dBm   -0.63 dBm  230.43 THz
          4      69.1 %    0.56 dBm   -0.10 dBm  230.43 THz

Transceiver Vendor Details

        Form Factor            : QSFP28
        Name                   : CISCO-FINISAR
        Part Number            : FTLC1152RGPL-C2
        Rev Number             : CISCO-FINISAR
        Serial Number          : FNS22150LEC
        PID                    : QSFP-100G-CWDM4-S
        VID                    : V02
        CISCO-FINISAR
        Date Code(yy/mm/dd)    : 18/04/11
        Fiber Connector Type: LC
        Sonet Application Code: Not Set
        Ethernet Compliance Code: 100GBASE-CWDM4

Transceiver Temperature : 32 Celsius


AINS Soak               : Running
AINS Timer              : 0h, 15m
AINS remaining time     : 771 seconds
```

When the soak time expires, AINS state changes from Running to None. The Transport Admin State of optics controller changes from Automatic In Service to In Service.

```
RP/0/RP0/CPU0:ios# show controllers optics 0/1/0/3


Thu Feb 21 20:02:34.126 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

        Optics Type:  Grey optics

        Alarm Status:
        -------------
        Detected Alarms: None


        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 0
        HIGH-TX-PWR = 0            LOW-TX-PWR = 0
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
```

```
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                 High Alarm  Low Alarm  High Warning  Low Warning
        ------------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          4.9      -12.0           0.0          0.0
        Tx Power Threshold(dBm)          3.5      -10.1           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 98 %
        Polarization parameters not supported by optics

       Total TX Power = 6.41 dBm

       Total RX Power = 5.85 dBm

        Lane  Laser Bias    TX Power    RX Power  Output Frequency
        ----  ----------  ----------  ----------  ----------------
          1      74.9 %    0.60 dBm    0.63 dBm  230.43 THz
          2      68.6 %    0.06 dBm   -0.70 dBm  230.43 THz
          3      69.0 %    0.30 dBm   -0.63 dBm  230.43 THz
          4      69.1 %    0.57 dBm   -0.11 dBm  230.43 THz

Transceiver Vendor Details

        Form Factor            : QSFP28
        Name                   : CISCO-FINISAR
        Part Number            : FTLC1152RGPL-C2
        Rev Number             : CISCO-FINISAR
        Serial Number          : FNS22150LEC
        PID                    : QSFP-100G-CWDM4-S
        VID                    : V02
        CISCO-FINISAR
        Date Code(yy/mm/dd)    : 18/04/11
        Fiber Connector Type: LC
        Sonet Application Code: Not Set
        Ethernet Compliance Code: 100GBASE-CWDM4

Transceiver Temperature : 32 Celsius


AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Configuring AINS on OTN-XP Card

You can configure the default AINS settings for all controllers on the OTN-XP card using the shared plane configuration. The configuration is applied to any line card that is installed in the NCS 1004. Use the following commands:

configure

ains-soak hours *hours* **minutes** *minutes*

commit

The following is a sample in which all the controllers on the OTN-XP card are configured with AINS with soak time period specified to be two minutes.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#ains-soak hours 0 minutes 2
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#do show controllers optics 0/1/0/0
Tue Apr 28 11:50:15.431 UTC

Controller State: Down

Transport Admin State: Automatic In Service

Laser State: On

LED State: Red

 Optics Status

         Optics Type:  100G QSFP28 LR4

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 0
         HIGH-TX-PWR = 0            LOW-TX-PWR = 0
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 0
         WVL-OOL = 0                MEA  = 0
         IMPROPER-REM = 0
         TX-POWER-PROV-MISMATCH = 0

         Performance Monitoring: Enable

         THRESHOLD VALUES
         ----------------

         Parameter                High Alarm  Low Alarm  High Warning  Low Warning
         -----------------------  ----------  ---------  ------------  -----------
         Rx Power Threshold(dBm)          4.9      -12.0           0.0          0.0
         Tx Power Threshold(dBm)          3.5      -10.1           0.0          0.0
         LBC Threshold(mA)                N/A        N/A          0.00         0.00

         LBC High Threshold = 98 %
         Polarization parameters not supported by optics

        Total TX Power = 7.74 dBm

        Total RX Power = -40.00 dBm

         Lane  Laser Bias    TX Power    RX Power  Output Frequency
         ----  ----------  ----------  ----------  ----------------
           1      67.2 %    1.85 dBm  -40.00 dBm  231.39 THz
           2      67.9 %    1.55 dBm  -40.00 dBm  230.59 THz
           3      67.5 %    1.58 dBm  -40.00 dBm  229.79 THz
           4      66.8 %    1.89 dBm  -40.00 dBm  230.25 THz
```

```
Transceiver Vendor Details

        Form Factor             : QSFP28
        Name                    : CISCO-FINISAR
        Part Number             : 10-3204-01
        Rev Number              : B
        Serial Number           : FNS20510YUB
        PID                     : ONS-QSFP28-LR4
        VID                     : V01
        Date Code(yy/mm/dd)     : 16/12/15
        Fiber Connector Type: LC
        Otn Application Code: 4I1-9D1F
        Sonet Application Code: Not Set
        Ethernet Compliance Code: 100GBASE-LR4

Transceiver Temperature : 27 Celsius


AINS Soak               : Pending
AINS Timer              : 0h, 2m
AINS remaining time     : 120 seconds
```

To override the default AINS settings on a specific controller, use the following commands:

automatic-in-service controller optics *R/S/I/P* **hours** *hours* **minutes** *minutes*

Note This configuration does not persist after an RP reload operation.

The following is a sample in which the optics controller on the OTN-XP card is configured with a soak time period of 45 minutes.

```
RP/0/RP0/CPU0:ios#automatic-in-service controller optics 0/1/0/0 hours 0 minutes 45
Tue Apr 28 11:55:15.666 UTC
RP/0/RP0/CPU0:ios#show controllers optics 0/1/0/0
Tue Apr 28 11:55:30.323 UTC

Controller State: Down

Transport Admin State: Automatic In Service

Laser State: On

LED State: Red

 Optics Status

        Optics Type:  100G QSFP28 LR4

        Alarm Status:
        -------------
        Detected Alarms: None


        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
```

```
        HIGH-RX-PWR = 0            LOW-RX-PWR = 0
        HIGH-TX-PWR = 0            LOW-TX-PWR = 0
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          4.9      -12.0           0.0          0.0
        Tx Power Threshold(dBm)          3.5      -10.1           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 98 %
        Polarization parameters not supported by optics

       Total TX Power = 7.74 dBm

       Total RX Power = -40.00 dBm

        Lane  Laser Bias    TX Power    RX Power  Output Frequency
        ----  ----------  ----------  ----------  ----------------
          1      67.2 %    1.85 dBm  -40.00 dBm  231.39 THz
          2      67.9 %    1.55 dBm  -40.00 dBm  230.59 THz
          3      67.5 %    1.58 dBm  -40.00 dBm  229.79 THz
          4      66.8 %    1.89 dBm  -40.00 dBm  230.25 THz

Transceiver Vendor Details

        Form Factor            : QSFP28
        Name                   : CISCO-FINISAR
        Part Number            : 10-3204-01
        Rev Number             : B
        Serial Number          : FNS20510YUB
        PID                    : ONS-QSFP28-LR4
        VID                    : V01
        Date Code(yy/mm/dd)    : 16/12/15
        Fiber Connector Type: LC
        Otn Application Code: 4I1-9D1F
        Sonet Application Code: Not Set
        Ethernet Compliance Code: 100GBASE-LR4

Transceiver Temperature : 27 Celsius


AINS Soak               : Pending
AINS Timer              : 0h, 45m
AINS remaining time     : 2700 seconds
```

From Release 7.5.2 onwards, AINS is supported on the 16G FC and 32FC controllers.

The following is a sample to configure AINS in 16G FC controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#automatic-in-service controller SixteenGigFibreChanCtrlr  0/0/0/6/2
```

```
 hours 0 minutes 15
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample to verify AINS in 16G FC controller.

```
 show controllers SixteenGigFibreChanCtrlr 0/0/0/6/2 on XR

+++ 15:04:25 ne(default) exec +++
show controllers SixteenGigFibreChanCtrlr 0/0/0/6/2

Wed Apr 13 15:04:25.206 UTC

Operational data for Fibre Channel controller SixteenGigFibreChanCtrlr0/0/0/6/2

State:
        Admin State           : Up
        Operational state     : Up
        LED state             : Green On
        Secondary admin state : Automatic In Service
        AINS Soak             : Running
            Total Duration    : 0 hour(s) 15 minute(s)
            Remaining Duration: 0 hour(s) 14 minute(s) 29 second(s)
        Laser Squelch         : Disabled

Performance Monitoring is enabled

Operational values:
        Speed                    : 16 Gbps
        Loopback                 : None
        BER monitoring:
            Not supported
        Hold-off Time            : 0 ms
        Forward Error Correction : Not Configured
```

The following is a sample to configure AINS in 32G FC controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#automatic-in-service controller ThirtyTwoGigFibreChanCtrlr
0/0/0/0/4 hours 0 minutes 15
RP/0/RP0/CPU0:ios(config)#commit
```

The following is a sample to verify AINS in 32G FC controller.

```
 show controllers ThirtyTwoGigFibreChanCtrlr 0/0/0/0/4 on XR

+++ 15:04:25 ne(default) exec +++
show controllers ThirtyTwoGigFibreChanCtrlr 0/0/0/0/4

Wed Apr 13 15:04:25.393 UTC

Operational data for Fibre Channel controller ThirtyTwoGigFibreChanCtrlr0/0/0/0/4

State:
        Admin State           : Up
        Operational state     : Up
        LED state             : Green On
        Secondary admin state : Automatic In Service
        AINS Soak             : Running
            Total Duration    : 0 hour(s) 15 minute(s)
            Remaining Duration: 0 hour(s) 14 minute(s) 29 second(s)
        Laser Squelch         : Disabled

Performance Monitoring is enabled

Operational values:
```

```
        Speed                    : 32 Gbps
        Loopback                 : None
        BER monitoring:
            Not supported
        Hold-off Time            : 0 ms
        Forward Error Correction : Standard(Reed Solomon)

RP/0/RP0/CPU0:ne#
```

From Release 7.5.2 onwards, AINS is supported on the controllers for OTUCn-REGEN mode.

The following is a sample to configure AINS on the coherentDSP controllers for OTUCn-REGEN mode.

```
RP/0/RP0/CPU0:ios#automatic-in-service controller coherentDSP 0/0/0/12 hours 0 minutes 15
Tue May 24 17:51:06.979 UTC
```

The following sample verifies the AINS configured on the coherentDSP controllers for OTUCn-REGEN mode.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/12
Tue May 24 17:52:29.558 UTC

Port                                          : CoherentDSP 0/0/0/12
Controller State                              : Up
Inherited Secondary State                     : Automatic-In-Service
Configured Secondary State                    : Automatic-In-Service
Derived State                                 : Automatic-In-Service
Loopback mode                                 : None
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 400.0Gb/s

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 4 OOM = 0 AIS = 3
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 3 TIM = 1
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 2
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 2
FLEXO-LOF = 0
Detected Alarms                               : None

Bit Error Rate Information
PREFEC  BER                                   : 1.42E-04
POSTFEC BER                                   : 0.00E+00
Q-Factor                                      : 11.10 dB

Q-Margin                                      : 4.60dB

TTI :
        Remote hostname                       : ios
        Remote interface                      : CoherentDSP 0/0/0/12
        Remote IP addr                        : 0.0.0.0

FEC mode                                      : O_FEC

Flexo-Mode                                    : Enable
Flexo Details:
        Tx GID                                : 10
        TX IID                                : 1, 2, 3, 4,
        Rx GID                                : 10
        RX IID                                : 1, 2, 3, 4,

AINS Soak                                     : Running
```

```
AINS Timer                                          : 0h, 15m
AINS remaining time                                 : 855 seconds
```

The following is a sample to configure AINS globally on the OTUCn-REGEN mode.

```
RP/0/RP0/CPU0:ios#configure terminal
Tue May 24 17:51:34.545 UTC
RP/0/RP0/CPU0:ios(config)#ains-soak hours 0 minutes 15
RP/0/RP0/CPU0:ios(config)#commit
Tue May 24 17:51:44.144 UTC
```

The following sample verifies the AINS configured on the coherentDSP controllers:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Tue May 24 17:52:34.445 UTC

Controller State: Up

Transport Admin State: Automatic In Service

Laser State: On

LED State: Green

Optics Status

        Optics Type:  CFP2 DWDM
        DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
        Wavelength=1552.524nm

        Alarm Status:
        -------------
        Detected Alarms: None


        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 0
        HIGH-TX-PWR = 0            LOW-TX-PWR = 0
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 1
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 0
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.47 dBm
        RX Power = 0.40 dBm
        RX Signal Power = 0.30 dBm
        Frequency Offset = -1358 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         3.0      -25.5           0.0          0.0
        Tx Power Threshold(dBm)         3.0      -12.0           0.0          0.0
        LBC Threshold(mA)               N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
```

```
        Configured Tx Power = 0.50 dBm
        Configured CD High Threshold = 48000 ps/nm
        Configured CD lower Threshold = -48000 ps/nm
        Configured OSNR lower Threshold = 22.00 dB
        Configured DGD Higher Threshold = 67.00 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 4.0000000000  bits/symbol
        Modulation Type: 16QAM
        Chromatic Dispersion -1 ps/nm
        Configured CD-MIN -24000 ps/nm  CD-MAX 24000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 52.00 ps^2
        Optical Signal to Noise Ratio = 35.90 dB
        SNR = 19.40 dB
        Polarization Dependent Loss = 1.70 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 2.00 ps
        DAC RATE:  1x1.50

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3555-01
        Rev Number             : A0
        Serial Number          : ACA25420007
        PID                    : DP04CFP2-M25-K9
        VID                    : VES1
        Date Code(yy/mm/dd)    : 21/09/28
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 41 Celsius


AINS Soak               : Running
AINS Timer              : 0h, 15m
AINS remaining time     : 850 seconds
```

FEC

Table 42: Feature History

Feature Name	Release Information	Description
Forward Error Correction (FEC) support on QXP card for Ethernet controllers	Cisco IOS XR Release 7.8.1	FEC is now supported by the 100GE Ethernet controller on the NCS1K4-QXP-K9 card. FEC is supported for all pluggables except QSFP-100G-LR4-S and ONS-QSFP28-LR4.

Forward error correction (FEC) is a feature that is used for controlling errors during data transmission. This feature works by adding data redundancy to the transmitted message using an algorithm. This redundancy

allows the receiver to detect and correct a limited number of errors occurring anywhere in the message, instead of having to ask the transmitter to resend the message.

FEC States for Ethernet Controller

The following table lists the FEC states for the Ethernet controller.

State	Description
None	FEC is not enabled on the Ethernet controller.
Standard	Standard (Reed-Solomon) FEC is enabled on the Ethernet controller.

FEC configuration is automatically enabled for only the pluggables that support Auto-FEC. If you manually configure FEC, the manual configuration overrides the Auto-FEC.

The supported pluggables for Auto-FEC are:

- QSFP-100G-SR4-S
- QSFP-100G-CWDM4-S
- QSFP-100G-SM-SR
- QSFP-100G-AOC-1M
- QSFP-100G-AOC-3M
- QSFP-100G-AOC-10M
- QDD-400-AOC15M
- QDD-400G-FR4-S
- QSFP-100G-ER4L
- QDD-400G-DR4-S
- QDD-400G-LR8-S
- QDD-4X100G-LR-S

The LR4 pluggable is a 1310nm long range band pluggable that does not require you to enable FEC.

The software automatically enables FEC mode on the pluggables installed in the Cisco NCS 1004. When you upgrade the software of an NCS 1004 with pluggables in the FEC disabled mode, traffic is affected.

The following sample shows the running FEC configuration on the LR4 pluggable:

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/0/0/4
Thu Aug  8 15:41:20.857 IST
Operational data for interface HundredGigECtrlr0/0/0/4:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
```

```
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

The following sample shows the running FEC configuration on the non LR4 pluggable:

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/0/0/2
Thu Aug  8 15:41:56.457 IST
Operational data for interface HundredGigECtrlr0/0/0/2:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 66

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Configuring FEC on the Ethernet Controller

Note The FEC configuration is not required for the supported pluggables. The configuration is required only in the case of non-Cisco qualified non-LR4 pluggables.

To configure FEC on the Ethernet controller, use the following command:

configure

controller HundredGigECtrlr *R/S/I/P* **fec { none | standard }**

commit

The following sample shows how to configure FEC on the Ethernet controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/1/0/10 fec standard
RP/0/RP0/CPU0:ios(config)#commit
```

The following sample shows the running FEC configuration on the Ethernet controller:

```
RP/0/RP0/CPU0:BH-SIT2#show controller HundredGigECtrlr 0/1/0/10
Tue Jul 16 15:30:30.165 IST
Operational data for interface HundredGigECtrlr0/1/0/10:

State:
    Administrative state: enabled
    Operational state: Down (Reason: State undefined)
    LED state: Red On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known
    Alarms:
        Current:
            Loss of Frequency Sync Data
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

FEC States for CoherentDSP Controller

The following table lists the FEC states for the coherentDSP controllers.

Table 43: FEC State for CoherentDSP Controllers

State	Description
EnhancedSD15	FEC Soft-Decision 15.
EnhancedSD27	FEC Soft-Decision 27. Default.

Q-Margin Support

Q-margin is an important optical parameter that characterizes the health of an optical link. The Q-margin value is calculated based on the average bit error rate (BER) in the optical link.

Table 44: Feature History

Feature Name	Release Information	Description
Enhanced Q-Margin Support	Cisco IOS XR Release 7.3.1	Enhanced Q-Margin is supported for Forward error correction (FEC) and performance monitoring on CoherentDSP Controllers for 1.2T and 1.2TL cards. This enhanced Q-margin value is calculated based on the maximum number of errors for each frame. An attribute that is called **instantaneous Q-margin** is displayed in the output of the **show controllers coherentDSP** command. The lower the delta value between the instantaneous Q-margin value with the Q-margin value, the better the FEC performance of the NCS 1004 system. The instantaneous Q-margin values thus help you to optimize the system with continuous error correction in subsea transport networks. Command modified: • **show controllers coherentDSP**

Enhanced Q-Margin Support

From Release 7.3.1 onwards, enhanced Q-Margin is supported for Forward error correction (FEC) and performance monitoring on CoherentDSP controllers for 1.2T and 1.2TL cards. Enhanced Q-margin provides

a better error free signal in the optical link. The enhanced Q-margin value is calculated based on the maximum number of errors per frame. An attribute that is called instantaneous Q-margin is displayed in the output of the **show controllers coherentDSP** command. The lower the delta value between the instantaneous Q-margin value with the Q-margin value, the better the FEC performance of the NCS 1004 system.

To view Q-margin and enhanced Q-margin values for FEC on CoherentDSP controllers, see Verifying FEC on CoherentDSP Controllers, on page 135.

To view Q-margin and enhanced Q-margin values for performance monitoring on CoherentDSP controllers, see Configure PM parameters, on page 249.

Configuring FEC on CoherentDSP Controllers

To configure FEC on the CoherentDSP controller, use the following command:

configure

controller coherentDSP *R/S/I/P*

fec {**EnhancedSD15** | **EnhancedSD27**}

commit

The following sample shows how to configure FEC on the CoherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/0
RP/0/RP0/CPU0:ios(config-CoDSP)#fec EnhancedSD15
Tue Feb 25 11:25:52.670 UTC
WARNING! Changing FEC mode can impact traffic
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

Verifying FEC on CoherentDSP Controllers

The following sample shows the FEC configuration on the CoherentDSP controller:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/0

Tue Feb 25 11:26:08.235 UTC


Port                                          : CoherentDSP 0/0/0/0
Controller State                              : Up
Inherited Secondary State                     : Normal
Configured Secondary State                    : Normal
Derived State                                 : In Service
Loopback mode                                 : None
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 50.0Gb/s
Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                               : None
Bit Error Rate Information
PREFEC  BER                                   : 0.00E+00
POSTFEC BER                                   : 0.00E+00
```

```
Q-Factor                                        : 0.00 dB
Q-Margin                                        : -5.00dB
Instantaneous Q_margin                        :  0 dB


TTI :
       Remote IP addr                           : 0.0.0.0
FEC mode                                        : Soft-Decision 15

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Configuring FEC on OTN-XP Card

FEC is supported on the CoherentDSP controllers for the OTN-XP card and O-FEC is the default FEC option configured on the card.

From Release 7.3.1 onwards, CFP2-DCO trunk is configured with 0-FEC.

Note The options enhanced SD15 and SD17 are not supported on the OTN-XP card.

Note CFEC is not supported on CFP2-DCO.

To configure FEC on the CoherentDSP controller for the OTN-XP card, use the following command:

configure

controller coherentDSP *R/S/I/P*

fec OFEC

commit

The following sample shows how to configure O-FEC option on the CoherentDSP controller for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/0
RP/0/RP0/CPU0:ios(config-CoDSP)#fec OFEC
Tue Feb 25 11:25:52.670 UTC
WARNING! Changing FEC mode can impact traffic
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

Verifying FEC on OTN-XP Card

The following sample shows the FEC configuration on the CoherentDSP controller for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/0

Tue Feb 25 11:26:08.235 UTC
```

```
Port                                            : CoherentDSP 0/0/0/0
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 50.0Gb/s
Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                 : None
Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 0.00 dB
Q-Margin                                        : -5.00dB
Instantaneous Q_margin                        :  0 dB


TTI :
        Remote IP addr                          : 0.0.0.0
FEC mode                                        : O-FEC

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Configuring FEC on OTN-XP Card – QDD-400G-ZRP

The QDD-400G-ZRP coherent trunk pluggable supports two types of FEC modes on the OTN-XP card:

- C-FEC
- O-FEC

You can configure the required FEC mode on the OTN-XP card.

This table lists the datapath trunk rates supported for the QDD-400G-ZRP pluggable module on the OTN-XP card.

Table 45: Datapath trunk rate supported for QDD-400G-ZRP pluggable modules on OTN-XP Card

Datapath trunk rate	Modulation	FEC
4x100GE MXP	16-QAM	CFEC
4x100GE MXP	16-QAM	OFEC
3x100GE MXP	8QAM	OFEC
2x100GE MXP	QPSK	OFEC
400GE TXP	16-QAM	CFEC

Datapath trunk rate	Modulation	FEC
400GE TXP	16-QAM	OFEC

Note On configuring datapath for 400G trunk rate, the default FEC mode enabled is C-FEC mode. For 200G and 300G trunk rates, the only FEC mode enabled is O-FEC mode.

Note You can migrate from CFEC to OFEC or OFEC to CFEC mode for the 400G trunk rate. The change in FEC mode affects traffic.

To configure FEC on the CoherentDSP controller for the OTN-XP card, use the following command:

configure

controller coherentDSP *R/S/I/P*

fec *fec-type*

commit

The following sample shows how to configure O-FEC option on the CoherentDSP controller for the OTN-XP card:

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/11
RP/0/RP0/CPU0:ios(config-CoDSP)#fec OFEC
Fri Jul 23 18:19:31.204 UTC
WARNING! Changing FEC mode can impact traffic
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Fri Jul 23 18:19:32.835 UTC
```

The following sample shows how to configure C-FEC option on the CoherentDSP controller for the OTN-XP card:

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/11
RP/0/RP0/CPU0:ios(config-CoDSP)#fec CFEC
Fri Jul 23 18:19:31.204 UTC
WARNING! Changing FEC mode can impact traffic
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Fri Jul 23 18:19:32.835 UTC
```

Verifying FEC on OTN-XP Card – QDD-400G-ZRP

The following sample shows the FEC configuration on the CoherentDSP controller with the trunk controller status as C-FEC for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controller coherentDSP 0/0/0/11
Fri Jul 23 17:36:45.342 UTC

Port                                        : CoherentDSP 0/0/0/11
Controller State                            : Up
Inherited Secondary State                   : Normal
Configured Secondary State                  : Normal
Derived State                               : In Service
Loopback mode                               : None
BER Thresholds                              : SF = 1.0E-5  SD = 1.0E-7
```

```
Performance Monitoring                            : Enable
Bandwidth                                         : 400.0Gb/s

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms                                   : None

Bit Error Rate Information
PREFEC  BER                                       : 9.02E-04
POSTFEC BER                                       : 0.00E+00
Q-Factor                                          : 9.90 dB

Q-Margin                                          : 2.70dB

TTI :
        Remote IP addr                            : 0.0.0.0

FEC mode                                          : C_FEC

Flexo-Mode                                        : Enable
Flexo Details:
        Tx GID                                    : 0
        Rx GID                                    : 0

AINS Soak                                         : None
AINS Timer                                        : 0h, 0m
AINS remaining time                               : 0 seconds
```

The following sample shows the FEC configuration on the CoherentDSP controller with the trunk controller status as O-FEC for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controller coherentDSP 0/0/0/11
Fri Jul 23 17:46:51.775 UTC

Port                                              : CoherentDSP 0/0/0/11
Controller State                                  : Up
Inherited Secondary State                         : Normal
Configured Secondary State                        : Normal
Derived State                                     : In Service
Loopback mode                                     : None
BER Thresholds                                    : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                            : Enable
Bandwidth                                         : 400.0Gb/s

Alarm Information:
LOS = 3 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 1
Detected Alarms                                   : None

Bit Error Rate Information
PREFEC  BER                                       : 3.36E-04
POSTFEC BER                                       : 0.00E+00
```

```
Q-Factor                                        : 10.60 dB

Q-Margin                                        : 4.10dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 0
        Rx GID                                  : 0

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

The following sample shows the FEC configuration on the optics controller with the trunk controller optics as C-FEC for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controller optics 0/0/0/11
Wed Sep 15 00:36:24.383 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  QSFP-DD DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=49, Frequency=193.70THz,
         Wavelength=1547.715nm

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 1
         HIGH-TX-PWR = 0            LOW-TX-PWR = 1
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 1
         WVL-OOL = 0                MEA  = 0
         IMPROPER-REM = 0
         TX-POWER-PROV-MISMATCH = 0
         Laser Bias Current = 0.0 %
         Actual TX Power = -8.09 dBm
         RX Power = -7.31 dBm
         RX Signal Power = -7.67 dBm
         Frequency Offset = 81 MHz
 Performance Monitoring: Enable

         THRESHOLD VALUES
```

```
        ----------------

        Parameter                 High Alarm  Low Alarm  High Warning  Low Warning
        ------------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -23.5           0.0          0.0
        Tx Power Threshold(dBm)          0.0      -16.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = -7.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 24.00 dB
        Configured DGD Higher Threshold = 40.00 ps
        Baud Rate =  59.8437500000 GBd
        Bits per Symbol = 4.0000000000  bits/symbol
        Modulation Type: 16QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -2400 ps/nm  CD-MAX 2400 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.40 dB
        SNR = 17.30 dB
        Polarization Dependent Loss = 0.40 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 3.00 ps

 Transceiver Vendor Details

        Form Factor            : QSFP-DD
        Name                   : CISCO
        Part Number            : 10-3496-01
        Rev Number             : 11
        Serial Number          : 210153241
        PID                    : QDD-400G-ZRP-S
        VID                    : ES04
        Date Code(yy/mm/dd)    : 20/21/01
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set
 Transceiver Temperature : 57 Celsius
 AINS Soak               : None
 AINS Timer              : 0h, 0m
 AINS remaining time     : 0 seconds
```

The following sample shows the FEC configuration on the optics controller with the trunk controller optics as O-FEC for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controller optics 0/3/0/9
Wed Sep 15 00:41:22.027 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

        Optics Type:  QSFP-DD DWDM
```

```
        DWDM carrier Info: C BAND, MSA ITU Channel=49, Frequency=193.70THz,
        Wavelength=1547.715nm

        Alarm Status:
        -------------
        Detected Alarms: None


        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 0            LOW-RX-PWR = 3
        HIGH-TX-PWR = 0            LOW-TX-PWR = 5
        HIGH-LBC = 0               HIGH-DGD = 0
        OOR-CD = 0                 OSNR = 4
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 6
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = -11.10 dBm
        RX Power = -11.56 dBm
        RX Signal Power = -11.62 dBm
        Frequency Offset = -66 MHz
Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -24.5           0.0          0.0
        Tx Power Threshold(dBm)          0.0      -16.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = -10.00 dBm
        Configured CD High Threshold = 52000 ps/nm
        Configured CD lower Threshold = -52000 ps/nm
        Configured OSNR lower Threshold = 21.10 dB
        Configured DGD Higher Threshold = 67.00 ps
        Baud Rate =  60.1385467980 GBd
        Bits per Symbol = 4.0000000000  bits/symbol
        Modulation Type: 16QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -13000 ps/nm  CD-MAX 13000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 24.00 ps^2
        Optical Signal to Noise Ratio = 35.70 dB
        SNR = 19.40 dB
        Polarization Dependent Loss = 0.20 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

 Transceiver Vendor Details

        Form Factor            : QSFP-DD
        Name                   : CISCO-ACACIA
        Part Number            : DP04QSDD-E
        Rev Number             : A
        Serial Number          : ACA2524006W
        PID                    : QDD-400G-ZRP-S
        VID                    : V01
```

```
        Date Code(yy/mm/dd)     : 21/06/18
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 62 Celsius
AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Laser Squelching

Table 46: Feature History

Feature Name	Release Information	Feature Description
Laser Squelching	Cisco IOS XR Release 7.8.1	Laser Squelching is now triggered using a new interrupt mechanism to detect faults in the client or trunk connections. Compared to the earlier poll-based triggers, the new interrupt-based mechanism makes the protection switching considerably faster. This feature is supported on the following line cards with 100GE client rate with the ONS-QSFP28-LR4 pluggable: • NCS1K4-1.2T-K9 • NCS1K4-1.2TL-K9 • NCS1K4-OTN-XP • NCS1K4-2-QDD-C-K9

You can enable laser squelching on Ethernet controllers. When laser squelching is enabled, the laser is shut down in the event of trunk faults (LOS, LOF), and a SQUELCHED alarm is raised on the mapped client port.

In previous releases, implementation was based on a poll mechanism and client squelch was supported only in case of trunk fault scenarios. From 7.8.1 release squelching uses an interrupt based method. Hence squelching happens faster when compared to previous releases. Squelch happens for client alarms also like Ingress LF, LOA, and CSF (not for egress client alarms) in addition to trunk fault cases. Fast squelching helps to achieve faster protection switching. See Protection Switching Use Cases, on page 144. This feature is supported on the following line cards with 100GE client rate with the ONS-QSFP28-LR4 pluggable:

- NCS1K4-1.2T-K9
- NCS1K4-1.2TL-K9
- NCS1K4-OTN-XP
- NCS1K4-2-QDD-C-K9

To configure laser squelching on the Ethernet controllers, use the following commands:

configure

controller HundredGigECtrlr *Rack/Slot/Instance/Port*

laser-squelch

commit

The following is a sample where laser squelching is enabled on the Ethernet controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/1/0/10
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#laser-squelch
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the laser squelch status on the controller.

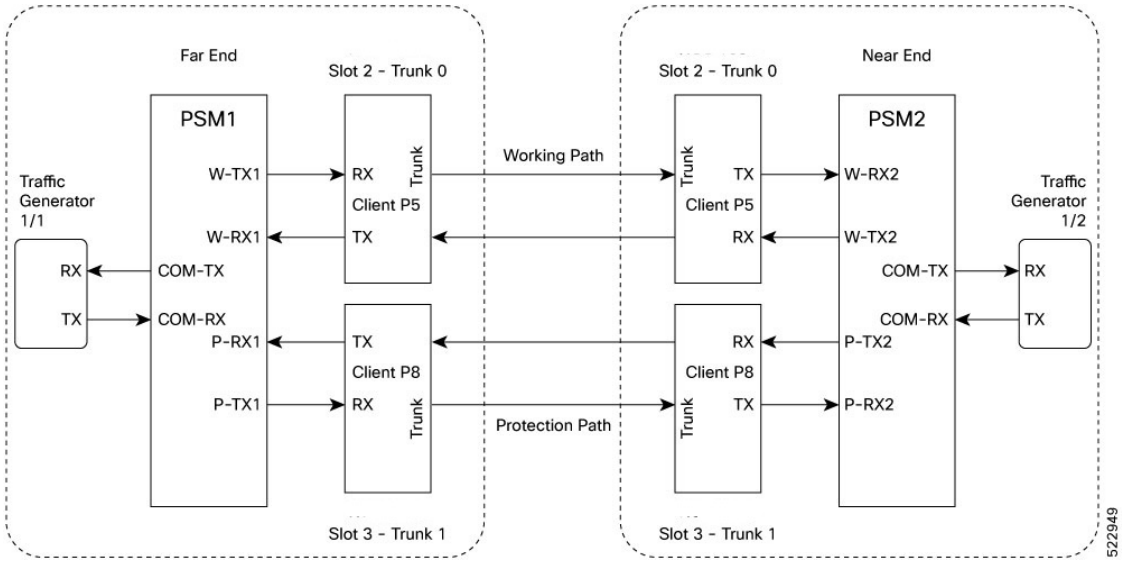
```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/1/0/10
Fri Feb 22 15:18:47.011 UTC
Operational data for interface HundredGigECtrlr0/1/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Protection Switching Use Cases

Fast-Squelching provides increased protection switching speed when there's a trunk fault or a client fault.

The following sample topology includes a Far End (FE) station and a Near End (NE) station. Each station includes an NCS 1004 node having two line cards. The nodes are connected to the respective Traffic generators through a Protection Switching Module (PSM).

Figure 1: Reference Topology for Protection Switching



Protection Switching Principle (Trunk fault)

If there is a fiber cut in the trunk working path from the FE station to the NE station, an LOS alarm is raised on the NE working trunk. This results in the squelching of all client ports mapped to the working NE trunk port. As the laser of the client port is squelched, LOS is reported on the W-RX2 port of the PSM2. As the received optical power on the W-RX2 port of PSM is below the threshold, PSM2 switches to receive the optical signal in the P-RX2 port instead of the W-RX2 port. Hence switching happens for traffic from work to protect in FE station to NE station direction. In this way, bidirectional switching is implemented.

In the case of a unidirectional trunk fault, switching happens in one direction as explained above. In the other direction, when LOS is received at the W-RX2 port of PSM2, W-TX2 sends LOS for 25 milliseconds. When LOS is reported on the NE client port, fault gets propagated over the trunk, resulting in the squelching of FE station client ports. Finally, the LOS on the PSM port results in switching in this direction as well.

Protection Switching Principle (Client fault)

When a client failure happens on the FE station, a Client Signal Failure (CSF) alarm is raised on the NE station trunk. The CSF on the trunk results in the squelching of the corresponding client port, and the PSM switching happens. In summary, a fault on the NE station client RX port results in CSF on the FE station trunk, and the switching happens. And, a fault on the NE station client TX port results in LOS on the PSM ports, and the switching happens.

Note

- PSM must be in the standalone mode.
- PSM alarm threshold must be set to +/ –3 dBm from the actual power received in the PSM RX port.
- If line card protection is required, the working and protect path must be configured in two different line cards.
- If only client protection is required, the working and protection path can be configured in the same line card.
- If the LC trunk configuration is x50 rate, then we can't use a single-line card for work and protection due to x50 coupled mode limitations (coupled trunk).
- Manual switch, Force switch, and lock-out protection on PSM, result in bidirectional switching.

The PSM supports both C2B (4x100G-MXP-400G-TXP) and C3B (40x10G-4x100G-MXP) cards. The C2B card allows client data rates of 100G and 400G, while the C3B card supports client data rates of 10G and 100G. The PSM supports upto 400G with the GL-2 pluggable module.

The PAM4 and QDD pluggable modules take more than 50ms to recover traffic when switching between working and protection modes.

The GL-2 DP04CFP2-M25-K9 pluggable module supports both Flexcoh streaming and ODUCn termination modes. Compared to Flexcoh streaming mode, the ODUCn termination mode with chromatic dispersion configuration of +/- 10000 has a shorter switching time.

The following table lists 400G client pluggables that do not support switching from the active path to the protection path within 50 milliseconds:

Operating Mode	Pluggables not Supporting Switching within 50 ms
400GE to 2×400G	QDD-400G-DR4-S
	QDD-400G-FR4-S

Note According to the QDD pluggable standard, if an electrical (TP1) or optical (TP3) signal is interrupted or disturbed at a client pluggable, and the Clock and Data Recovery (CDR) inside the QDD loses the lock, the QDD pluggable requires some time to recover the signal. Hence, when a fault is identified on the trunk port, the client port Rx loses the lock. The recovery takes between 1.5 to 2 seconds depending on the pluggable standard.

Configuring Laser Squelching on OTN-XP Card

From R7.2.1 onwards, laser squelching is supported on 10GE controllers for the OTN-XP card.

From R7.3.1 onwards, laser squelching is supported on 100GE or 400GE controllers for the OTN-XP card.

From Release 7.5.2 onwards, laser squelching is supported on the 16G FC and 32G FC controllers for the OTN-XP card.

Configuring Laser Squelching on 10GE Controllers

To configure laser squelching on the 10GE controllers for the OTN-XP card, use the following commands:

configure

controller tenGigECtrlr *Rack/Slot/Instance/Port/Lanenumber*

laser-squelch

commit

The range of *Lanenumber* is from 1 to 4.

The following is a sample where laser squelching is enabled on the 10GE controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller tenGigECtrlr 0/0/0/4/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#laser-squelch
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the laser squelch status on the 10GE controller.

```
P/0/RP0/CPU0:ios#show controllers tenGigECtrlr 0/0/0/4/1
Wed May 6 06:28:29.603 UTC
Operational data for interface TenGigECtrlr0/0/0/4/1:

State:
Administrative state: enabled
Operational state: Up
LED state: Green On
Maintenance: Disabled
AINS Soak: None
Total Duration: 0 hour(s) 0 minute(s)
Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
Laser Squelch: Enabled

Phy:
Media type: Not known

Autonegotiation disabled.

Operational values:
Speed: 10Gbps
Duplex: Full Duplex
Flowcontrol: None
Loopback: None (or external)
Inter-packet gap: standard (12)
BER monitoring:
Not supported
Holdoff Time: 0ms
```

Configuring Laser Squelching on 100GE Controllers

To configure laser squelching on the 100GE controllers for the OTN-XP card, use the following commands:

configure

controller HundredGigECtrlr *Rack/Slot/Instance/Port*

laser-squelch

commit

The following is a sample where laser squelching is enabled on the 100GE controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/0/0/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#laser-squelch
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the laser squelch status on the 100GE controller.

```
RP/0/RP0/CPU0:ios#show controller hundredGigECtrlr 0/0/0/1
Fri Jul 23 16:07:11.541 UTC
Operational data for interface HundredGigECtrlr0/0/0/1:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0
Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Configuring Laser Squelching on 400GE Controllers

To configure laser squelching on the 400GE controllers for the OTN-XP card, use the following commands:

configure

controller fourHundredGigECtrlr *Rack/Slot/Instance/Port*

laser-squelch

commit

The following is a sample where laser squelching is enabled on the 400GE controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller fourHundredGigECtrlr 0/0/0/8
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#laser-squelch
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the laser squelch status on the 400GE controller.

```
RP/0/RP0/CPU0:ios#show controller fourhundredGigECtrlr 0/0/0/8
Fri Jul 23 16:07:11.541 UTC
Operational data for interface fourHundredGigECtrlr0/0/0/8:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known

Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0
Autonegotiation disabled.

Operational values:
    Speed: 400Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Configuring Laser Squelching on 16G FC and 32G FC Controllers

The following is a sample where laser squelching is enabled on the 16G FC controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller SixteenGigFibreChanCtrlr 0/1/0/0/2 laser-squelch
RP/0/RP0/CPU0:ios(config)#commit
Sat Apr  9 13:03:26.746 UTC
RP/0/RP0/CPU0:ios(config)#end
```

The following is a sample where laser squelching is enabled on the 32G FC controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller ThirtyTwoGigFibreChanCtrlr 0/1/0/6/4 laser-squelch
RP/0/RP0/CPU0:ios(config)#commit
Sat Apr  9 13:05:26.746 UTC
RP/0/RP0/CPU0:ios(config)#end
```

The following sample verifies the laser squelching enabled on the 16G FC controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show controllers SixteenGigFibreChanCtrlr 0/1/0/0/2

+++ 13:03:44 fe(default) exec +++
show controllers SixteenGigFibreChanCtrlr 0/1/0/0/2

Sat Apr  9 13:03:43.743 UTC
```

```
Operational data for Fibre Channel controller SixteenGigFibreChanCtrlr0/1/0/0/2

State:
 Admin State           : Up
 Operational state     : Up
 LED state             : Green On
 Secondary admin state : Normal
 AINS Soak             : None
     Total Duration    : 0 hour(s) 0 minute(s)
     Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
 Laser Squelch         : Enabled

Performance Monitoring is enabled

Operational values:
 Speed                    : 16 Gbps
 Loopback                 : None
 BER monitoring:
     Not supported
 Hold-off Time            : 0 ms
 Forward Error Correction : Not Configured
RP/0/RP0/CPU0:ios#
```

The following sample verifies the laser squelching enabled on the 32G FC controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#show controllers ThirtyTwoGigFibreChanCtrlr 0/1/0/6/4

+++ 13:03:44 fe(default) exec +++
show controllers ThirtyTwoGigFibreChanCtrlr 0/1/0/6/4

Sat Apr  9 13:03:43.923 UTC

Operational data for Fibre Channel controller ThirtyTwoGigFibreChanCtrlr0/1/0/6/4

State:
 Admin State           : Up
 Operational state     : Up
 LED state             : Green On
 Secondary admin state : Normal
 AINS Soak             : None
     Total Duration    : 0 hour(s) 0 minute(s)
     Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
 Laser Squelch         : Enabled

Performance Monitoring is enabled

Operational values:
 Speed                    : 32 Gbps
 Loopback                 : None
 BER monitoring:
     Not supported
 Hold-off Time            : 0 ms
 Forward Error Correction : Standard(Reed Solomon)
```

Idle Insertion

When a fault occurs on the trunk port, you can hold the propagation of local faults using the idle insertion feature. This feature is enabled on the ethernet controller by configuring the hold-off timer.

When the fault occurs on the trunk, idles are inserted in the traffic stream from the trunk port to the client port for the duration of the configured holdoff-time. If the trunk port remains faulty beyond the configured holdoff-time, a local fault is transmitted towards the client device. If the trunk recovers from the fault before the holdoff-time expires, traffic resumes.

This feature can be used on customer deployments to prevent reset of client ports during a PSM switchover.

You can enable the idle insertion feature by using the following commands:

configure

controller HundredGigECtrlr *Rack/Slot/Instance/Port*

holdoff-time trunk-fault *time-value*

The range of *timevalue* is from 0 ms to 3000 ms.

The following is a sample for enabling the hold off -timer in 100GE controllers:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/1/0/10
RP/0/RP0/CPU0:ios (config-eth-ctrlr)#holdoff-time trunk-fault 3000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

To view the hold-off time that is configured on 100GE controller, use the following command:

show controllers hundredGigECtrlr *Rack/Slot/Instance/Port*

Example

```
RP/0/RP0/CPU0:ios#show controllers HundredGigECtrlr 0/1/0/10
Fri Feb 22 18:58:06.888 UTC
Operational data for interface HundredGigECtrlr0/1/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 3000ms
```

Enabling Idle Insertion on OTN-XP Card

From R7.2.1 onwards, you can enable the idle insertion feature on the 10GE controller for OTN-XP card.

From R7.3.1 onwards, you can enable the idle insertion feature on 100GE or 400GE controllers for the OTN-XP card.

To enable idle insertion on the 10GE controller, enter the following commands:

configure

controller tenGigECtrlr *Rack/Slot/Instance/Port/Lanenumber*

holdoff-time trunk-fault *time-value*

commit

The range of *Lanenumber* is from 1 to 4 and the range of holdoff-time trunk-fault *time-value* is from 0 to 3000 ms.

The following is a sample for enabling the idle insertion feature in 10GE controllers:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller tenGigECtrlr 0/0/0/4/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#holdoff-time trunk-fault 2000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

To view the hold-off time that is configured on 10GE controllers, use the following command:

show controllers tenGigECtrlr *Rack/Slot/Instance/Port/Lanenumber*

Example

```
RP/0/RP0/CPU0:ios#show controllers TenGigECtrlr 0/0/0/4/1
Thu Mar 26 12:46:16.543 UTC
Operational data for interface TenGigECtrlr0/0/0/4/1:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 10Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    Inter-packet gap: standard (12)
    BER monitoring:
        Not supported
    Holdoff Time: 2000ms
```

Configuring Idle Insertion on 100GE Controllers

To configure idle insertion on the 100GE controllers for the OTN-XP card, use the following commands:

configure

controller HundredGigECtrlr *Rack/Slot/Instance/Port*

holdoff-time trunk-fault *time-value*

commit

The range of *timevalue* is from 0 ms to 3000 ms.

The following is a sample where idle insertion is enabled on the 100GE controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/0/0/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#holdoff-time trunk-fault 3000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the idle insertion status on the 100GE controller.

```
RP/0/RP0/CPU0:ios#show controller hundredGigECtrlr 0/0/0/1
Fri Jul 23 16:07:11.541 UTC
Operational data for interface HundredGigECtrlr0/0/0/1:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
        Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 3000ms
```

Configuring Idle Insertion on 400GE Controllers

To configure idle insertion on the 400GE controllers for the OTN-XP card, use the following commands:

configure

controller fourHundredGigECtrlr *Rack/Slot/Instance/Port*

holdoff-time trunk-fault *time-value*

commit

The following is a sample where idle insertion is enabled on the 400GE controller for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller fourHundredGigECtrlr 0/0/0/10
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#holdoff-time trunk-fault 2000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the idle insertion status on the 400GE controller.

```
RP/0/RP0/CPU0:ios#show controller fourhundredGigECtrlr 0/0/0/10
Fri Jul 23 16:07:11.541 UTC
Operational data for interface fourHundredGigECtrlr0/0/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 400Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
        Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 2000ms
```

Enable Idle Insertion on QXP Card

From R7.8.1 onwards, you can enable idle insertion on 100GE or 400GE controllers for the QXP card such as the NCS1K4-QXP-K9 card.

Configure Idle Insertion on 100GE Controllers

To configure idle insertion on the 100GE controllers for the NCS1K4-QXP-K9 card, use the following commands:

configure

controller HundredGigECtrlr *Rack/Slot/Instance/Port*

holdoff-time trunk-fault *time-value*

commit

The range of *timevalue* is from 0 ms to 3000 ms.

The following is a sample where idle insertion is enabled on the 100GE controller for the NCS1K4-QXP-K9 card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/1/0/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#holdoff-time trunk-fault 3000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the idle insertion status on the 100GE controller.

```
RP/0/RP0/CPU0:ios#show controller hundredGigECtrlr 0/1/0/1
Fri Jul 23 16:07:11.541 UTC
Operational data for interface HundredGigECtrlr0/1/0/1:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
        Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 3000ms
```

Configure Idle Insertion on 400GE Controllers

To configure idle insertion on the 400GE controllers for the NCS1K4-QXP-K9 card, use the following commands:

configure

controller fourHundredGigECtrlr *Rack/Slot/Instance/Port*

holdoff-time trunk-fault *time-value*

commit

The following is a sample where idle insertion is enabled on the 400GE controller for the NCS1K4-QXP-K9 card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller fourHundredGigECtrlr 0/0/0/10
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#holdoff-time trunk-fault 2000
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following is a sample to view the idle insertion status on the 400GE controller.

```
RP/0/RP0/CPU0:ios#show controller fourhundredGigECtrlr 0/0/0/10
Fri Jul 23 16:07:11.541 UTC
Operational data for interface fourHundredGigECtrlr0/0/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
Statistics:
FEC:
Corrected Codeword Count: 134967789
Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 400Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
        Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 2000ms
```

Idle Insertion for Ethernet Controllers

Table 47: Feature History

Feature Name	Release Information	Feature Description
Idle Insertion for Ethernet Controllers	Cisco IOS XR Release 7.5.2	Idle insertion for Ethernet controllers feature performs end-to-end link verification between 100GE or 400GE Ethernet controllers before bringing up the actual traffic. This feature enables you to perform pre-provisioning checks to isolate link errors in advance without any Ethernet testers. This feature is supported on the 1.2T C band, 1.2T L band, and 800G QSFP-DD Transponder line cards.

Idle insertion for Ethernet controllers feature allows you to perform end-to-end link verification between 100GE or 400GE Ethernet controllers before bringing up the actual traffic. This feature is supported on the 1.2T, 1.2TL, and 2-QDD-C cards.

Note OTU4 client rate is not supported.

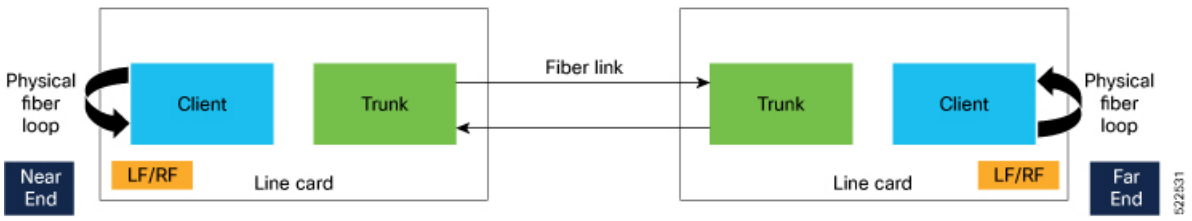
Idle frames can be inserted in both the ingress and egress directions on Ethernet controllers and the LOCAL-FAULT and REMOTE-FAULT alarms are cleared. The performance monitoring counters on the pcs layer are monitored to check for any errors on the link.

Warning Do not configure the Idle insertion for Ethernet controllers feature on the link that carries live traffic.

Recommended Topology for Link Verification

The following diagram describes the recommended topology for link verification:

Figure 2: Topology for Link Verification



The following steps describe the sequence for link verification using this topology:

1. Both the near-end and far-end clients have the LOCAL-FAULT alarm if the trunk is up on both the ends.
2. Enable idle ingress on the near-end client. The idle frame transmits toward the trunk link and reaches the far-end client. The LOCAL-FAULT alarm is then cleared on the far-end client.
3. As the far-end client has fiber loop, the idle frame is inserted again into the same client RX toward the trunk link and reaches the near-end client. The LOCAL-FAULT alarm is then cleared on the near-end client as well.
4. When you enable idle insertion on any client and in any direction, the idle frame transmits in loop similar to this topology and all the LOCAL-FAULT and the REMOTE-FAULT alarms are cleared.
5. The link can be monitored after all the alarms are cleared. The link has a problem if any alarm is reported during the link test.

Configuring Idle Insertion for Ethernet Controllers

Before You Begin:

- Do not configure idle frame insertion with hold-off timer.
- Do not configure PRBS on the trunk.

You can configure this feature by using the following commands:

configure

controller hundredGigECtrlr *Rack/Slot/Instance/Port*

insert-idle ingress

insert-idle egress

commit

end

The following is a sample for enabling the idle ingress and idle egress in 100GE controllers:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller hundredGigECtrlr 0/2/0/2
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#insert-idle ingress
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#insert-idle egress
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#end
```

You can disable this feature by using the following commands:

configure

controller hundredGigECtrlr *Rack/Slot/Instance/Port*

no insert-idle ingress

no insert-idle egress

commit

end

Limitation

After disabling the idle frame insertion feature, the LOCAL-FAULT or REMOTE-FAULT alarm may not appear again because the idle frames are in loop. Hence, you must break the idle frame loop in the link by performing either one of the following:

- Perform fiber OIR on either the near-end or far-end client port.
- Perform shut and unshut operation on any client port.

Verifying Idle Insertion Configuration for Ethernet Controllers

To verify the idle ingress and idle egress that is configured on the Ethernet controllers, use the following command:

RP/0/RP0/CPU0:ios# **show controllers hundredGigECtrlr** *Rack/Slot/Instance/Port*

Example

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/2/0/2
Wed Mar 30 06:56:58.878 UTC
Operational data for interface HundredGigECtrlr0/2/0/2:
State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

    Insert Idle Ingress: Enabled
    Insert Idle Egress: Enabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 0
Autonegotiation disabled.
Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

LLDP Drop

Link Layer Discovery Protocol (LLDP) Snooping is enabled by default on all ethernet controllers.

To verify the LLDP neighbors, use the following commands:

```
RP/0/RP0/CPU0:ios#show lldp neighbors detail
Tue Mar 12 11:49:20.819 IST
Capability codes:
        (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
        (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

------------------------------------------------
Local Interface: HundredGigECtrlr0/1/0/7
Chassis id: 008a.96cd.34e1
Port id: Hu0/0/0/4
Port Description - not advertised
System Name: ncs5500_node

System Description:
 6.1.4, NCS-5500

Time remaining: 116 seconds
Hold Time: 120 seconds
System Capabilities: R
Enabled Capabilities: R
Management Addresses - not advertised
Peer MAC Address: 00:8a:96:cd:34:10


------------------------------------------------
Local Interface: HundredGigECtrlr0/1/0/13
Chassis id: 008a.96cd.34e1
Port id: Hu0/0/0/5
Port Description - not advertised
System Name: ncs5500_node

System Description:
 6.1.4, NCS-5500

Time remaining: 90 seconds
Hold Time: 120 seconds
System Capabilities: R
Enabled Capabilities: R
Management Addresses - not advertised
Peer MAC Address: 00:8a:96:cd:34:14


Total entries displayed: 2


RP/0/RP0/CPU0:ios#show lldp neighbors
Tue Mar 12 16:17:56.713 IST
Capability codes:
        (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
        (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

Device ID       Local Intf              Hold-time  Capability     Port ID
ncs5500_node    HundredGigECtrlr0/1/0/7  120        R               Hu0/0/0/4
ncs5500_node    HundredGigECtrlr0/1/0/13 120        R               Hu0/0/0/5

Total entries displayed: 2
```

When you enable LLDP drop on the client controller ports of the muxponder or muxponder slice, the LLDP frames drop on the ports without forwarding.

Note LLDP on 400GE is not supported on the OTN-XP card.

Configuring LLDP Drop

You can configure the LLDP drop for a muxponder or muxponder slice. By default, the LLDP drop status is set to False. On enabling the LLDP Drop, the status is set to True.

To configure LLDP drop on a muxponder use the following command:

configure

hw-module location *location* **mxponder drop-lldp**

Note Use the **no** form of the command to disable LLDP drop.

commit

Limitation

- When you disable LLDP globally, the LLDP gets disabled on all the interfaces.

Note By default, LLDP is enabled for NCS 1004. But when you enable and disable LLDP in the global configuration mode, LLDP gets disabled on all the interfaces.

Workaround: You must enable LLDP globally or reload the Router.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios#hw-module location 0/1 mxponder drop-lldp
RP/0/RP0/CPU0:ios#commit
```

configure

hw-module location *location* **mxponder-slice** *slice-number* **drop-lldp**

Note Use the **no** form of the command to disable LLDP drop.

To configure LLDP drop on a muxponder slice, use the following command:

commit

The following is a sample in which slice 0 client ports are enabled with LLDP drop.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0 drop-lldp
RP/0/RP0/CPU0:ios(config)#commit
```

Verifying the Status of LLDP Drop

To verify the LLDP drop enabled status, use the following command.

```
RP/0/RP0/CPU0:ios#show hw-module location all mxponder
Fri Feb 22 13:22:19.281 UTC
```

```
Location:             0/0
Client Bitrate:       NONE
Trunk  Bitrate:       NONE
Status:               Not Provisioned


Location:             0/1
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/0
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/2         ODU40/1/0/0/0                     100
HundredGigECtrlr0/1/0/3         ODU40/1/0/0/1                     100
HundredGigECtrlr0/1/0/4         ODU40/1/0/0/2                     100
HundredGigECtrlr0/1/0/5         ODU40/1/0/0/3                     100
HundredGigECtrlr0/1/0/6         ODU40/1/0/0/4                     100


Location:             0/1
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/1/0/8         ODU40/1/0/1/0                     100
HundredGigECtrlr0/1/0/9         ODU40/1/0/1/1                     100
HundredGigECtrlr0/1/0/10        ODU40/1/0/1/2                     100
HundredGigECtrlr0/1/0/11        ODU40/1/0/1/3                     100
HundredGigECtrlr0/1/0/12        ODU40/1/0/1/4                     100


Location:             0/2
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/0
                                Traffic Split Percentage

HundredGigECtrlr0/2/0/2         ODU40/2/0/0/0                     100
HundredGigECtrlr0/2/0/3         ODU40/2/0/0/1                     100
HundredGigECtrlr0/2/0/4         ODU40/2/0/0/2                     100
HundredGigECtrlr0/2/0/5         ODU40/2/0/0/3                     100
HundredGigECtrlr0/2/0/6         ODU40/2/0/0/4                     100


Location:             0/2
Slice ID:             1
Client Bitrate:       100GE
Trunk  Bitrate:       500G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/2/0/1
                                Traffic Split Percentage

HundredGigECtrlr0/2/0/8         ODU40/2/0/1/0                     100
```

```
HundredGigECtrlr0/2/0/9         ODU40/2/0/1/1                      100
HundredGigECtrlr0/2/0/10        ODU40/2/0/1/2                      100
HundredGigECtrlr0/2/0/11        ODU40/2/0/1/3                      100
HundredGigECtrlr0/2/0/12        ODU40/2/0/1/4                      100


Location:             0/3
Slice ID:             0
Client Bitrate:       100GE
Trunk  Bitrate:       300G
Status:               Provisioned
LLDP Drop Enabled:    TRUE
Client Port                     Mapper/Trunk Port          CoherentDSP0/3/0/0
                                Traffic Split Percentage

HundredGigECtrlr0/3/0/2         ODU40/3/0/0/0                      100
HundredGigECtrlr0/3/0/3         ODU40/3/0/0/1                      100
HundredGigECtrlr0/3/0/4         ODU40/3/0/0/2                      100
```

Link Layer Discovery Protocol (LLDP) Support on Management Interface

The LLDP support on management interface feature requires a system to form LLDP neighborship over the system management interface, through which it advertises and learns LLDP neighbor information. This information about neighbors used to learn about the neighbors and in turn the topology of the devices for Operations, Administration, and Maintenance (OAM) purposes.

Advantages of LLDP

- Provides support on non-Cisco devices.
- Enables neighbor discovery between non-Cisco devices.

Limitation

- When you disable LLDP globally, the LLDP gets disabled on all the interfaces.

Note By default, LLDP is enabled for NCS 1004. But when you enable and disable LLDP in the global configuration mode, LLDP gets disabled on all the interfaces.

Workaround: You must enable LLDP globally or reload the Router.

Cisco Discovery Protocol (CDP) vs LLDP

The CDP is a device discovery protocol that runs over Layer 2. Layer 2 is also known as the data link layer that runs on all Cisco devices, such as routers, bridges, access servers, and switches. This protocol allows the network management applications to automatically discover and learn about other Cisco devices that connect to the network.

The LLDP is also a device discovery protocol that runs over Layer 2. This protocol allows the network management applications to automatically discover and learn about other non-Cisco devices that connect to the network.

Interoperability between non-Cisco devices using LLDP

LLDP is also a neighbor discovery protocol that is used by network devices to advertise information about themselves to other devices on the network. This protocol runs over the data link layer, which allows two systems running different network layer protocols to learn about each other.

With LLDP, the user can also access the information about a particular physical network connection. If the user uses a non-Cisco monitoring tool (through SNMP), LLDP helps you identify the Object Identifiers (OIDs) that the system supports. The following OIDs are supported:

- 1.0.8802.1.1.2.1.4.1.1.4
- 1.0.8802.1.1.2.1.4.1.1.5
- 1.0.8802.1.1.2.1.4.1.1.6
- 1.0.8802.1.1.2.1.4.1.1.7
- 1.0.8802.1.1.2.1.4.1.1.8
- 1.0.8802.1.1.2.1.4.1.1.9
- 1.0.8802.1.1.2.1.4.1.1.10
- 1.0.8802.1.1.2.1.4.1.1.11
- 1.0.8802.1.1.2.1.4.1.1.12

Neighbor Discovery

System advertises the LLDP TLV (Type Length Value) details over the management network using which other devices in the management network can learn about this device.

Configuring LLDP

- LLDP full stack functionality is supported on all three management interfaces supported in NCS 1004.
- You can selectively enable or disable LLDP on any of the management interfaces on demand.
- You can selectively enable or disable LLDP transmit or receive functionality at the management interface level.
- Information gathered using LLDP can be stored in the device Management Information Database (MIB) and queried with the Simple Network Management protocol (SNMP).
- LLDP operational data are available in both Command Line Interface and netconf-yang interface.

Enabling LLDP Globally

When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations.

Note You can override this default operation at the interface to disable receive or transmit operations.

The following table describes the global LLDP attributes that the user can configure:

Table 48:

Attribute	Default	Range	Description
Holdtime	120	0–65535	Specifies the hold time (in sec). Hold time refers to the time or duration that an LLDP device maintains the neighbor information before discarding.
Reinit	2	2–5	Delay (in sec) for LLDP initialization on any interface
Timer	30	5-65534	Specifies the rate at which LLDP packets are sent (in sec)

The following example shows the commands to configure LLDP globally. The global LLDP configuration enables LLDP on all the three management interfaces.

```
RP/0/RP0/CPU0:regen#configure terminal
RP/0/RP0/CPU0:regen(config)#lldp management enable
RP/0/RP0/CPU0:regen(config)#lldp holdtime 30
RP/0/RP0/CPU0:regen(config)#lldp reinit 2
RP/0/RP0/CPU0:regen(config)#commit
```

Verification

You can verify the LLDP configuration using the **show running-config lldp** command.

The output of **show running-config lldp** command is as follows:

```
RP/0/RP0/CPU0:regen#show running-config lldp
Tue Dec 10 10:36:11.567 UTC
lldp
timer 30
reinit 2
holdtime 120
management enable
!
```

You can verify the LLDP data using the **show lldp interface** and **show lldp neighbors** commands.

The output of **show lldp interface** command is as follows:

```
RP/0/RP0/CPU0:regen#show lldp interface
Thu Nov  7 08:45:22.934 UTC


MgmtEth0/RP0/CPU0/0:
        Tx: enabled
        Rx: enabled
        Tx state: IDLE
        Rx state: WAIT FOR FRAME


MgmtEth0/RP0/CPU0/1:
        Tx: enabled
        Rx: enabled
```

```
        Tx state: IDLE
        Rx state: WAIT FOR FRAME
```

The output of **show lldp neighbors** command is as follows:

```
RP/0/RP0/CPU0:M-131#show lldp neighbors
Mon Dec  2 11:01:20.143 CET
Capability codes:
        (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
        (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

Device ID       Local Intf              Hold-time  Capability     Port ID
[DISABLED]      MgmtEth0/RP0/CPU0/0     120        B               gi19
MYS-130         MgmtEth0/RP0/CPU0/1     120        R               MgmtEth0/RP0/CPU0/1
```

where [DISABLED] shows that the LLDP is disabled on the interface MgmtEth0/RP0/CPU0/0.

Enabling LLDP per Management Interface

The following example shows the commands to configure LLDP at the management interface level.

```
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/X
RP/0/RP0/CPU0:ios(config-if)#lldp enable
RP/0/RP0/CPU0:ios(config-if)#commit
```

Disabling LLDP Transmit and Receive Operations

The following example shows the commands to disable the LLDP transmit operations at the specified management interface.

```
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/X
RP/0/RP0/CPU0:ios(config-if)#lldp transmit disable
RP/0/RP0/CPU0:ios(config-if)#commit
```

The following example shows the commands to disable the LLDP receive operations at the specified management interface.

```
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/X
RP/0/RP0/CPU0:ios(config-if)#lldp receive disable
RP/0/RP0/CPU0:ios(config-if)#commit
```

Debugging LLDP Issues

The following commands are used for debugging issues in the LLDP functionality.

- **show lldp traffic**
- **debug lldp all**
- **debug lldp errors**
- **debug lldp events**
- **debug lldp packets**
- **debug lldp tlvs**
- **debug lldp trace**
- **debug lldp verbose**

Daisy Chain Support on Management Ports

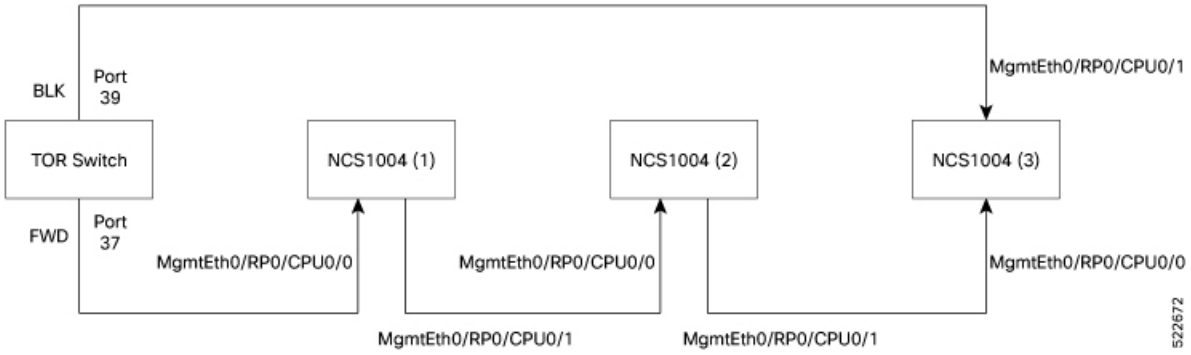
Table 49: Feature History

Feature Name	Release Information	Description
Daisy Chain Support on Management Ports	Cisco IOS XR Release 7.7.1	In a daisy chain arrangement, multiple NCS 1004 devices are connected to form a ring-like topology, and only the first and last nodes are connected to a Top-of-Rack (TOR) switch. The first connection serves as the main path for data transmission and carries the traffic, while the last connection acts as a backup or secondary path. If the primary path fails, the secondary path takes over and allows traffic to continue transmitting in the network. You can daisy chain up to five NCS 1004 nodes in the network.

Daisy Chain feature is supported only on the management ports 0 and 1 on the NCS 1004 chassis.

The following diagram shows the Daisy Chain topology. In this topology, three NCS 1004 nodes are connected to each other over the management ports.

Figure 3: Daisy Chain Topology



Configure Daisy Chain Support on Management Ports

Daisy Chain must be configured only on the management port 1.

Before You Begin:

- Enable Storm Control on TOR Switch, on page 168
- Disable DAD on Management Port, on page 168

- STP must be running on the TOR switch.
- Management port 0 must not be in shut down state and must be configured with either IPv4 or IPv6 address.
- Management port 1 must not be configured with either IPv4 or IPv6 address.
- LLDP is not supported on the management port 1 if Daisy Chain is configured.

To configure Daisy Chain on the management port 1, enter the following commands:

configure

interface mgmtEth 0/RP0/CPU0/1

no ipv4 address

no ipv6 address

bridge-port routed-interface mgmtEth 0/RP0/CPU0/0

Verify Daisy Chain

To verify the daisy chain that is configured on the management port, use the following commands.

```
!
interface MgmtEth0/RP0/CPU0/0
 ipv4 address 10.127.60.220 255.255.255.0
!
interface MgmtEth0/RP0/CPU0/1
 bridge-port routed-interface MgmtEth0/RP0/CPU0/0
!
interface MgmtEth0/RP0/CPU0/2
 shutdown
!
```

Enable Storm Control on TOR Switch

In Daisy Chain configuration, one of the Top of the Rack (TOR) switch ports is in blocked state, provided NCS 1004 forwards STP BPDU in periodic intervals. Three consecutive hello misses move the port state from blocked to forwarding state.

When the NCS 1004 node reboots, the other port state of the TOR switch changes from blocking to forwarding state. Hence, a loop is created momentarily when both the TOR switch ports are in forwarding state. This loop results in duplication of packets on the network. Hence, storm control must be enabled on the TOR switch.

Enter the following commands from the TOR switch to enable storm control.

errdisable recovery interval 60

errdisable recovery cause storm-control

Disable DAD on Management Port

By default, IPv6 Duplicate Address Detection (DAD) is enabled on the management ports. Similar to storm control scenario, when IPv6 is configured for a management port, DAD happens due to looping in the network.

Since DAD was enabled, management port will be down. In order to avoid management port being down due to momentary looping, DAD must be disabled on the management port 0.

Enter the following commands to disable DAD on the management port 0.

configure

interface mgmtEth0/RP0/CPU0/0

ipv6 nd dad attempts 0

DHCP Client

Table 50: Feature History

Feature Name	Release Information	Feature Description
Dynamic Host Configuration Protocol (DHCP) Client	Cisco IOS XR Release 7.3.2	The DHCP client enables the router interfaces to dynamically acquire the IPv4 or DHCPv4 or DHCPv6 server which is used to forward the response to the correct layer 2 address. The DHCP client ensures that configuration information reaches the correct device.

The Dynamic Host Configuration Protocol (DHCP) client functionality enables the router interfaces to dynamically acquire the IPv4 or DHCPv4 or DHCPv6 server, and forwards the responses back to the correct Layer 2 address so that the correct device gets the correct configuration information.

DHCP has the ability to allocate IP addresses only for a configurable period of time, called the lease period. If the client is required to retain this IP address for a longer period beyond the lease period, the lease period must be renewed before the IP address expires. The client renews the lease based on configuration that was sent from the server. The client unicasts a REQUEST message using the IP address of the server. When a server receives the REQUEST message and responds with an ACK message. The lease period of the client is extended by the lease time configured in the ACK message. DHCP client supports DHCPv6.

DHCP Client Options

You can configure DHCPv6 client on management Ethernet interfaces. You can configure different DHCPv6 client options to differentiate between clients as required. The different DHCPv6 client options are also configured to differentiate how a DHCPv6 client communicates with a DHCPv6 server. The different DHCPv6 client options that can be configured are:

- **DUID:** If the DUID DHCPv6 client option is configured on an interface, DHCPv6 client communicates with the DHCPv6 server through the link layer address.
- **Rapid Commit:** If the Rapid Commit DHCPv6 client option is configured on an interface, DHCPv6 client can obtain configuration parameters from the DHCPv6 server through a rapid two-step exchange (solicit and reply) instead of the default four-step exchange (solicit, advertise, request, and reply).
- **DHCP Options:** The various other DHCPv6 options that can be configured on a DHCPv6 client are:

- **Option 15:** Option 15 is also known as the User Class option and it is used by a DHCPv6 client to identify the type or category of users or applications it represents.
- **Option 16:** Option 16 is also known as the Vendor ID option and it is used by a DHCPv6 a client to identify the vendor that manufactured the hardware on which the client is running.
- **Option 23:** Option 23 is also known as the Domain name Server (DNS) option provides a list of one or more IPv6 addresses of DNS recursive name servers to which a client's DNS resolver can send DNS queries.
- **Option 24:** Option 24 is also known as the Domain List option and it specifies the domain search list that the client uses to resolve hostnames with the DNS.

- **DHCP Timers:** This option is used to set different timer value for DHCP client configurations. The various DHCP timer options are:
  - **Release-timeout:** It is used to set retransmission timeout value for the initial release message.
  - **Req-max-rt:** It is used to set the maximum retransmission timeout value for the request message.
  - **Req-max-rt:** It is used to set the maximum retransmission timeout value for the request message.
  - **Sol-max-delay:** It is used to set the maximum delay time of the first solicit message.
  - **Sol-max-rt:** It is used to set the maximum solicit retransmission time.
  - **Sol-time-out:** It is used to set the intial timeout value of the solicit message.

Enabling DHCP Client on Management Ethernet Interface

To enable DHCP client on the management Ethernet interface, use the following command:

configure

interface MgmtEth *R/S/I/P*

ipv6 address dhcp-client-options

duid linked-layer-address

rapid-commit

{**option 16 vendor** | **option 15 ciscoupnnp** | **option 23** | **option 24**}

timers sol-max-delay *value* | **timers sol-time-out** *value* | **timers req-timeout** *value* | **timers req-max-rt** *value* | **timers release-timeout** *value*}

ipv6 address dhcp

commit

The following sample shows to enable DHCP client on the management Ethernet interface:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#interface MgmtEth0/0/CPU0/0
RP/0/RP0/CPU0:ios(config)#ipv6 address dhcp-client-options
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#duid linked-layer-address
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#rapid-commit
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#option 16 vendor
```

```
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#option 15 ciscoupnnp
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#option 23
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#option 24
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers sol-max-delay 1
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers sol-time-out 1
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers sol-max-rt 120
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers req-timeout 1
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers req-max-rt 30
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#timers release-timeout 1
RP/0/RP0/CPU0:ios(config-dhcpv6-client)#commit
```

Verifying DHCP Client on Management Ethernet Interface

To verify DHCP client options on the management Ethernet interface, use the **show dhcp ipv6 client** and **show dhcp ipv6 client detail** commands:

```
RP/0/0/CPU0:ios#show dhcp ipv6 client
Thu Sep 17 10:45:44.493 IST

Interface name IPv6 Address State Lease Time Rem
---------------------- -------------- ---------------- --------------------
MgmtEth0/0/1/0 500:1::1a/128 BOUND 7116


RP/0/RP0/CPU0:ios#show dhcp ipv6 client detail
Thu Sep 17 10:45:48.880 IST

-----------------------------------------------------
Client Interface name : MgmtEth0/0/`/0
Client Interface handle : 0x4040
Client MACAddr : 0219.bc81.e750
Client State : BOUND
Client Link Local Address : fe80::19:bcff:fe81:e750
Client IPv6 Address (Dhcp) : 500:1::1a/128
Lease Remaining (in secs) : 7112
DUID : 000300010219bc81e750

Client Configuration
Timers
SOL_MAX_DELAY : 1 secs (00:00:01)
SOL_TIMEOUT : 1 secs (00:00:01)
SOL_MAX_RT : 120 secs (00:02:00)
REQ_TIMEOUT : 1 secs (00:00:01)
REQ_MAX_RT : 30 secs (00:00:30)
REL_TIMEOUT : 1 secs (00:00:01)

Options
RAPID-COMMIT : True
USER-CLASS : ciscoupnnp
VENDOR-CLASS : vendor
DNS-SERVERS : True
DOMAIN-LIST : True

DUID Type : DUID_LL

Server Information
Server Address : fe80::1a:19ff:fe03:99ca
Preference : 255
DUID : 000300010206826e2e00
Status : SUCCESS
IA-NA
Status : SUCCESS
```

```
IAID : 0x40400001
T1 : 3600 secs (01:00:00)
T2 : 5760 secs (01:36:00)
IA-ADDR
IA NA Address : 500:1::1a
Preferred Time : 7200 secs (02:00:00)
Valid Time : 7200 secs (02:00:00)
Flags : 0x0
-----------------------------------------------------
```

MAC Address Snooping on Client Ports

MAC address snooping allows you to learn the MAC address of the neighbor, that is connected to the client ports. You can enable ARP snooping on all client ports and learn the MAC address of neighbors through CLI.

This feature overcomes the limitation, where LLDP (Link Layer Discovery protocol) cannot be enabled in some networks.

Limitations

- When you enable or disable MAC address snooping on any slice, few packets are dropped during configuration.
- Open config interface for enabling or disabling MAC address snooping is not supported.
- SNMP MIB is not supported for the MAC address attribute.

Note When you enable MAC address snooping on client ports, it overrides LLDP.

Configuring MAC Address Snooping on Client Ports

You can configure MAC address or ARP snoop on slice in Muxponder slice mode using the following commands.

configure

hw-module location *location* **mxponder-slice** *slice-number*

client-rate 100GE

trunk-rate 600G { 100G | 150G | 200G | 250G | 300G | 350G | 400G | 450G | 500G | 550G | 600G }

arp-snoop

commit

Example

The following is a sample in which, MAC address or ARP snoop is configured on the client ports of slice 0 in Muxponder slice mode.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 16 19:30:33.933 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/3 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-rate 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 600G
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#arp-snoop
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
Mon Mar 16 19:30:52.636 UTC
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#end
```

The following is a sample in which, MAC address or ARP snoop is configured in Muxponder mode.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 16 19:08:17.154 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder arp-snoop
RP/0/RP0/CPU0:ios(config)#commit
```

The following sample shows the output of **show controllers hundredGigEctrlr** command, before configuring MAC address or ARP snoop on client ports.

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/1/0/2
Mon Mar 16 19:40:37.434 UTC
Operational data for interface HundredGigECtrlr0/1/0/2:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

Viewing Neighbor MAC Address

You can view the neighbor's physical address after enabling MAC address or ARP snoop using the following command. MAC address snoop output is enabled after ARP packets are received on the respective 100G client.

show controllers hundredGigEctrlr *R/S/I/P*

The following sample shows the neighbor's MAC address after configuring MAC address or ARP snoop on client ports.

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/1/0/2
Mon Mar 16 19:41:08.047 UTC
Operational data for interface HundredGigECtrlr0/1/0/2:

State:
```

```
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Disabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled
    Neighbor Address:
    0010.9400.5502

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
```

Transmit Shutdown

Transmit shut on trunk optics controller brings down the CFP2 transmit power.

From R7.3.2 onwards, you can configure transmit shut on optics controller in the 4x100GE-MXP-DD muxponder mode.

Configuring Transmit Shutdown on Trunk Optics Controller

To perform transmit shutdown, enter the following commands:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller Optics R/S/I/P
RP/0/RP0/CPU0:ios(config-Optics)#transmit-shutdown
RP/0/RP0/CPU0:ios(config-Optics)#commit
RP/0/RP0/CPU0:ios(config-Optics)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following is a sample to perform transmit shutdown on the trunk port:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller Optics 0/1/0/12
RP/0/RP0/CPU0:ios(config-Optics)#transmit-shutdown
RP/0/RP0/CPU0:ios(config-Optics)#commit
RP/0/RP0/CPU0:ios(config-Optics)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following is a sample to perform transmit shutdown on the trunk port in the 4x100GE-MXP-DD muxponder mode:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller Optics 0/1/0/9
RP/0/RP0/CPU0:ios(config-Optics)#transmit-shutdown
RP/0/RP0/CPU0:ios(config-Optics)#commit
RP/0/RP0/CPU0:ios(config-Optics)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Verifying Transmit Shutdown on Trunk Optics Controller

```
RP/0/RP0/CPU0:ios#sh controllers optics 0/1/0/12
Fri Feb 26 21:36:16.009 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: Off

LED State: Yellow

Optics Status

Optics Type: CFP2 DWDM
DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
Wavelength=1552.524nm

Alarm Status:
-------------
Detected Alarms:
LOW-TX-PWR

LOS/LOL/Fault Status:

Alarm Statistics:

-------------
HIGH-RX-PWR = 0 LOW-RX-PWR = 1
HIGH-TX-PWR = 0 LOW-TX-PWR = 1
HIGH-LBC = 0 HIGH-DGD = 0
OOR-CD = 0 OSNR = 0
WVL-OOL = 0 MEA = 0
IMPROPER-REM = 0
TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 %
Actual TX Power = -40.00 dBm
RX Power = -1.02 dBm
RX Signal Power = -12.20 dBm
Frequency Offset = -8 MHz

Performance Monitoring: Enable

THRESHOLD VALUES
----------------

Parameter High Alarm Low Alarm High Warning Low Warning
----------------------- ---------- --------- ------------ -----------
Rx Power Threshold(dBm) 4.9 -12.0 0.0 0.0
Tx Power Threshold(dBm) 3.5 -10.1 0.0 0.0
LBC Threshold(mA) N/A N/A 0.00 0.00

LBC High Threshold = 98 %
Configured Tx Power = -1.50 dBm
Configured CD High Threshold = 180000 ps/nm
Configured CD lower Threshold = -180000 ps/nm
Configured OSNR lower Threshold = 0.00 dB
Configured DGD Higher Threshold = 180.00 ps
Baud Rate = 63.1394691467 GBd
Bits per Symbol = 4.0000000000 bits/symbol
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
```

```
Configured CD-MIN -26000 ps/nm CD-MAX 26000 ps/nm
Polarization Mode Dispersion = 0.0 ps
Second Order Polarization Mode Dispersion = 16.00 ps^2
Optical Signal to Noise Ratio = 35.90 dB
SNR = 17.00 dB
Polarization Dependent Loss = 0.70 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 1.00 ps

Transceiver Vendor Details

Form Factor : CFP2
Name : CISCO-ACACIA
Part Number : 10-3500-01
Rev Number : 01
Serial Number : ACA24230026
PID : ONS-CFP2D-400G-C
VID : VES1
Date Code(yy/mm/dd) : 10/09/Í
Fiber Connector Type: LC
Otn Application Code: Not Set
Sonet Application Code: Not Set
Ethernet Compliance Code: Not set

Transceiver Temperature : 55 Celsius



AINS Soak : None
AINS Timer : 0h, 0m
AINS remaining time : 0 seconds


RP/0/RP0/CPU0:ios#show controller optics 0/1/0/9
Wed Sep 15 00:41:22.027 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  QSFP-DD DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=49, Frequency=193.70THz,
         Wavelength=1547.715nm

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 3
         HIGH-TX-PWR = 0            LOW-TX-PWR = 5
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 4
         WVL-OOL = 0                MEA  = 0
```

```
        IMPROPER-REM = 6
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = -11.10 dBm
        RX Power = -11.56 dBm
        RX Signal Power = -11.62 dBm
        Frequency Offset = -66 MHz

  Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                 High Alarm  Low Alarm  High Warning  Low Warning
        ------------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)          3.0      -24.5           0.0          0.0
        Tx Power Threshold(dBm)          0.0      -16.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 90 %
        Configured Tx Power = -10.00 dBm
        Configured CD High Threshold = 52000 ps/nm
        Configured CD lower Threshold = -52000 ps/nm
        Configured OSNR lower Threshold = 21.10 dB
        Configured DGD Higher Threshold = 67.00 ps
        Baud Rate =  60.1385467980 GBd
        Bits per Symbol = 4.0000000000  bits/symbol
        Modulation Type: 16QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -13000 ps/nm  CD-MAX 13000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 24.00 ps^2
        Optical Signal to Noise Ratio = 35.70 dB
        SNR = 19.40 dB
        Polarization Dependent Loss = 0.20 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : QSFP-DD
        Name                   : CISCO-ACACIA
        Part Number            : DP04QSDD-E
        Rev Number             : A
        Serial Number          : ACA2524006W
        PID                    : QDD-400G-ZRP-S
        VID                    : V01
        Date Code(yy/mm/dd)    : 21/06/18
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 62 Celsius
AINS Soak               : None
AINS Timer              : 0h, 0m
```

Loopback

Table 51: Feature History

Feature Name	Release Information	Description
Configuration Alarms for Loopback	Cisco IOS XR Release 7.8.1	A configuration alarm is now triggered whenever there is a change in the loopback configuration. This alarm helps in improving loopback status monitoring. You can now view the alarm details such as, the configuration time and date, description, severity, and location using the **show alarms brief system active** command.

You can configure the loopback on the CoherentDSP, FC, OTU, and Ethernet controllers to identify connection problems. The loopback can be configured only in the maintenance mode. Use the **controller** ***controller-type*** and the **secondary-admin-state maintenance** commands to place the controllers in the maintenance mode.

From R7.8.1, loopback configuration alarm details for each controller are triggered whenever there is a change in the loopback configuration. Details such as, location of the controller, severity, configuration date and time, and description are available in the output of the **show alarms brief system active** and **show alarms brief history** commands.

Note Internal and line loopbacks are supported on the FC, OTU, and Ethernet controllers whereas only internal loopbacks are supported on the CoherentDSP controllers.

Configuring Loopback on the 1.2T Card

To configure the loopback, use the following commands:

configure

controller *controllertype Rack/Slot/Instance/Port*

sec-admin-state maintenance

loopback [**line** | **internal**]

commit

Example 1

The following example shows how a line loopback is configured on the Ethernet controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 1/0/1/10 secondary-admin-state
maintenance
```

```
RP/0/RP0/CPU0:ios(config)#commit
Fri Feb 22 19:49:46.504 UTC
RP/0/RP0/CPU0:ios(config)#exit
```

The following example shows how to verify a line loopback configured on the Ethernet controller.

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/1/0/10
Fri Feb 22 19:50:08.328 UTC
Operational data for interface HundredGigECtrlr0/1/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Enabled
    AINS Soak: Pending
      Total Duration: 0 hour(s) 30 minute(s)
      Remaining Duration: 0 hour(s) 30 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 0

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/1/0/10 loopback line
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/1/0/10
Fri Feb 22 20:01:00.521 UTC
Operational data for interface HundredGigECtrlr0/1/0/10:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Enabled
    AINS Soak: Pending
      Total Duration: 0 hour(s) 30 minute(s)
      Remaining Duration: 0 hour(s) 30 minute(s) 0 second(s)
    Laser Squelch: Enabled

Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 0
            Uncorrected Codeword Count: 6

Autonegotiation disabled.
```

```
Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: Line
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Example 2

The following example shows how to verify an internal loopback configured on the coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP  0/0/0/0
Fri Mar 13 22:00:20.951 UTC

Port                                          : CoherentDSP 0/0/0/0
Controller State                              : Up
Inherited Secondary State                     : Normal
Configured Secondary State                    : Maintenance
Derived State                                 : Maintenance
Loopback mode                                 : Internal
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 200.0Gb/s

Alarm Information:
LOS = 0 LOF = 1 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 3 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                               : None

Bit Error Rate Information
PREFEC  BER                                   : 0.00E+00
POSTFEC BER                                   : 0.00E+00
Q-Factor                                      : 16.70 dB

Q-Margin                                      : 0.99dB

TTI :
        Remote hostname                       : ios
        Remote interface                      : CoherentDSP 0/0/0/0
        Remote IP addr                        : 0.0.0.0

FEC mode                                      : Soft-Decision 27

AINS Soak                                     : None
AINS Timer                                    : 0h, 0m
AINS remaining time                           : 0 seconds
```

Configuring Loopback on OTN-XP Card

From R7.2.1 onwards, OTN-XP card supports loopback on the OTU2, OTU2e, OTU4, 10GE, and CoherentDSP controllers.

From R7.3.2 onwards, OTN-XP card supports loopback on the 100GE and 400GE controllers.

From R7.5.2 onwards, OTN-XP card supports loopback on the 16G FC and 32G FC controllers.

The CoherentDSP controller supports both line and internal.

To configure the loopback on the controllers, use the following commands:

configure

controller *controller type Rack/Slot/Instance/Port/Lane number*

sec-admin-state maintenance

loopback [**line** | **internal**]

commit

The range of *Lane number* is 1–4.

Restriction From R7.10.1, OTN-XP card supports loopback on STM64 and OC192 controllers. You must use **no sec-admin-state** command instead of **sec-admin-state normal**.

Example 1

The following example shows how an internal loopback is configured on the 10GE controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller tenGigECtrlr 0/0/0/5/2
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following example shows how to verify an internal loopback configured on the 10GE controller.

```
RP/0/RP0/CPU0:ios#show controllers tenGigECtrlr 0/0/0/5/2
Thu Apr 23 10:47:48.020 UTC
Operational data for interface TenGigECtrlr0/0/0/5/2:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Enabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled

Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 10Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: Internal
    Inter-packet gap: standard (12)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

Example 2

The following example shows how a line loopback is configured on the OTU2e controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller otu2e 0/0/0/11/3
RP/0/RP0/CPU0:ios(config-otu2e)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-otu2e)#loopback line
RP/0/RP0/CPU0:ios(config-otu2e)#commit
Thu Apr 23 10:55:19.319 UTC
RP/0/RP0/CPU0:ios(config-otu2e)#end
```

The following example shows how to verify a line loopback configured on the OTU2e controller.

```
RP/0/RP0/CPU0:ios#show controllers otu2e 0/0/0/11/3
Thu Apr 23 10:55:28.014 UTC

Port                                        : OTU2E 0/0/0/11/3
Controller State                            : Up
Inherited Secondary State                   : Normal
Configured Secondary State                  : Maintenance
Derived State                               : Maintenance
Loopback mode                               : Line
BER Thresholds                              : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                      : Enable
Bandwidth                                   : 10.0Gb/s

Alarm Information:
LOS = 0 LOF = 1 LOM = 0
OOF = 1 OOM = 1 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                             : None

Bit Error Rate Information
PREFEC  BER                                 : 0.00E+00
POSTFEC BER                                 : 0.00E+00

TTI :
        Remote hostname                     : ios
        Remote interface                    : OTU2E 0/0/0/11/3
        Remote IP addr                      : 0.0.0.0

FEC mode                                    : STANDARD

AINS Soak                                   : None
AINS Timer                                  : 0h, 0m
AINS remaining time                         : 0 seconds
```

Example 3

The following example shows how an internal loopback is configured on the OTU2 controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller otu2 0/0/0/5/1
RP/0/RP0/CPU0:ios(config-otu2)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-otu2)#loopback internal
RP/0/RP0/CPU0:ios(config-otu2)#commit
Thu Apr 23 11:01:00.562 UTC
RP/0/RP0/CPU0:ios(config-otu2)#end
```

The following example shows how to verify an internal loopback configured on the OTU2 controller.

```
RP/0/RP0/CPU0:ios#show controllers otU2 0/0/0/5/1
Thu Apr 23 11:01:04.126 UTC

Port                                            : OTU2 0/0/0/5/1
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Maintenance
Derived State                                   : Maintenance
Loopback mode                                   : Internal
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 10.0Gb/s

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00

TTI :
        Remote hostname                         : SM-TRC SAPI-SECSM-TRC DA
        Remote IP addr                          : 209.165.200.229

FEC mode                                        : STANDARD

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Example 4

The following example shows how an internal loopback is configured on the OTU4 controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller otu4 0/0/0/0
RP/0/RP0/CPU0:ios(config-otu4)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-otu4)#loopback internal
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Apr 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how to verify an internal loopback configured on the OTU4 controller.

```
RP/0/RP0/CPU0:ios#show controllers otu4 0/0/0/0
Thu Apr 23 11:05:30.281 UTC

Port                                            : OTU4 0/0/0/0
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Maintenance
Derived State                                   : Maintenance
Loopback mode                                   : Internal
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
```

```
Bandwidth                                          : 100.0Gb/s

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                    : None

Bit Error Rate Information
PREFEC  BER                                        : 0.00E+00
POSTFEC BER                                        : 0.00E+00

TTI :
        Remote hostname                            : ios
        Remote interface                           : OTU4 0/0/0/0
        Remote IP addr                             : 0.0.0.0

FEC mode                                           : STANDARD

AINS Soak                                          : None
AINS Timer                                         : 0h, 0m
AINS remaining time                                : 0 seconds
```

Example 5

The following example shows how an internal loopback is configured on the 16G FC controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller SixteenGigFibreChanCtrlr 0/2/0/1/1
RP/0/RP0/CPU0:ios(config-SixteenGigFibreChanCtrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-SixteenGigFibreChanCtrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-SixteenGigFibreChanCtrlr)#commit
Thu Apr 11 10:05:21.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how to verify the internal loopback configured on the 16G FC controller:

```
RP/0/RP0/CPU0:ios#show controller SixteenGigFibreChanCtrlr 0/1/0/0/2

Sat Apr  9 22:50:38.930 UTC

Operational data for Fibre Channel controller SixteenGigFibreChanCtrlr0/1/0/0/2

State:
 Admin State           : Up
 Operational state     : Up
 LED state             : Green On
 Secondary admin state : Maintenance
 AINS Soak             : None
     Total Duration    : 0 hour(s) 0 minute(s)
     Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
 Laser Squelch         : Disabled

Performance Monitoring is enabled

Operational values:
 Speed                   : 16 Gbps
 Loopback                : Internal
 BER monitoring:
     Not supported
```

```
 Hold-off Time            : 0 ms
 Forward Error Correction : Not Configured
RP/0/RP0/CPU0:ios#
```

Example 6

The following example shows how an internal loopback is configured on the 32G FC controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller ThirtyTwoGigFibreChanCtrlr 0/1/0/6/4
RP/0/RP0/CPU0:ios(config-ThirtyTwoGigFibreChanCtrlr)#loopback internal

RP/0/RP0/CPU0:ios(config-ThirtyTwoGigFibreChanCtrlr)#commit

Sat Apr  9 22:50:11.666 UTC
RP/0/RP0/CPU0:ios(config-ThirtyTwoGigFibreChanCtrlr)#end
```

The following example shows how to verify the internal loopback configured on the 32G FC controller:

```
RP/0/RP0/CPU0:ios#show controller ThirtyTwoGigFibreChanCtrlr 0/1/0/6/4

Sat Apr  9 22:50:39.082 UTC

Operational data for Fibre Channel controller ThirtyTwoGigFibreChanCtrlr0/1/0/6/4

State:
 Admin State           : Up
 Operational state     : Up
 LED state             : Green On
 Secondary admin state : Maintenance
 AINS Soak             : None
     Total Duration    : 0 hour(s) 0 minute(s)
     Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
 Laser Squelch         : Disabled

Performance Monitoring is enabled

Operational values:
 Speed                 : 32 Gbps
 Loopback              : Internal
 BER monitoring:
     Not supported
 Hold-off Time         : 0 ms
 Forward Error Correction : Standard(Reed Solomon)
RP/0/RP0/CPU0:ios#
```

Example: Loopback Configuration in 4X100G MXP on 100GE Controller

The following example shows how the client internal loopback is configured on the 100GE controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/2/0/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the client line loopback is configured on the 100GE controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/2/0/1
```

```
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback line
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the trunk internal is configured on the coherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/11
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback internal
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the trunk line is configured on the coherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/11
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback line
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

Example: Loopback Configuration in 400G-TXP on 400GE Controller

The following example shows how the client internal loopback is configured on the 400GE controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller FourHundredGigECtrlr 0/2/0/10
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the client line loopback is configured on the 100GE controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller FourHundredGigECtrlr 0/2/0/10
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback line
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the trunk internal is configured on the coherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/10
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback internal
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

The following example shows how the trunk line is configured on the coherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/10
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback line
RP/0/RP0/CPU0:ios(config-otu4)#commit
Thu Sep 23 11:05:22.429 UTC
RP/0/RP0/CPU0:ios(config-otu4)#end
```

Configure Loopback in Inverse Muxponder Configured on the OTN-XP Card

You can configure loopback on the coherentDSP controllers in the inverse muxponder configuration.

Note You must configure loopback on both trunk ports 12 and 13, otherwise traffic goes down.

The following example shows how loopback is configured on both the trunk ports:

```
RP/0/RP0/CPU0:ios#configure
Thu Sep 30 14:16:04.678 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback internal
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Thu Sep 30 14:16:19.594 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#controller coherentDSP 0/2/0/13
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback internal
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Thu Sep 30 14:16:32.390 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#
```

The following examples shows how to verify loopback configured on the OTN-XP card in the inverse muxponder configuration:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/12
Thu Sep 30 14:17:04.411 UTC

Port                                          : CoherentDSP 0/2/0/12
Controller State                              : Up
Inherited Secondary State                     : Normal
Configured Secondary State                    : Maintenance
Derived State                                 : Maintenance
Loopback mode                                 : Internal
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 200.0Gb/s

Alarm Information:
LOS = 2 LOF = 0 LOM = 0
OOF = 1 OOM = 0 AIS = 1
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 1
FLEXO-LOF = 0
Detected Alarms                               : None

Bit Error Rate Information
PREFEC  BER                                   : 2.46E-08
POSTFEC BER                                   : 0.00E+00
Q-Factor                                      : 14.60 dB
```

```
Q-Margin                                        : 8.30dB

TTI :
        Remote hostname                         : ios
        Remote interface                        : CoherentDSP 0/2/0/12
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 1, 2,
        Rx GID                                  : 1
        RX IID                                  : 1, 2,

Flexo Peers Information:
        Controller                              : CoherentDSP0_2_0_13
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds


RP/0/RP0/CPU0:ios#sh controllers coherentDSP 0/2/0/13
Thu Sep 30 14:17:08.140 UTC

Port                                            : CoherentDSP 0/2/0/13
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Maintenance
Derived State                                   : Maintenance
Loopback mode                                   : Internal
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 200.0Gb/s

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 1
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 15.70 dB

Q-Margin                                        : 9.50dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
```

```
        Tx GID                                  : 1
        TX IID                                  : 3, 4,
        Rx GID                                  : 1
        RX IID                                  : 3, 4,

Flexo Peers Information:
        Controller                              : CoherentDSP0_2_0_12
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Configuring Loopback on 2-QDD-C Card

From R7.3.1 onwards, 2-QDD-C card supports loopback on the 100 and 400GE controllers.

Note On applying client-side loopbacks, traffic is looped and does not continue in the 2-QDD-C card. QSFP squelching happens on applying internal loopback.

To configure the loopback on the controllers, use the following commands.

configure

controller *controllertype Rack/Slot/Instance/Port/Lanenumber*

sec-admin-state maintenance

loopback [**line** | **internal**]

commit

Example

The following example shows how an internal loopback is configured on a 100GE controller.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller hundredGigECtrlr 0/0/0/5
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following example shows how to verify the internal loopback configured on a 100GE controller.

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/0/0/5
Thu Apr 23 10:47:48.020 UTC
Operational data for interface hundredGigECtrlr0/0/0/5:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Enabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled
```

```
Phy:
    Media type: Not known

Autonegotiation disabled.

Operational values:
    Speed: 10Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: Internal
    Inter-packet gap: standard (12)
    BER monitoring:
        Not supported
    Holdoff Time: 0ms
```

From R7.5.2 onwards, Loopback is supported for the OTUCn-REGEN mode on the coherent DSP controller.

Example

The following example shows how to configure an internal loopback on a coherent DSP controller.

```
Sun Dec 26 14:34:02.733 UTC
RP/0/RP0/CPU0:ios(config)#controller CoherentDSP 0/3/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Sun Dec 26 14:34:03.437 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#end
```

The following example shows how to verify internal loopback configured on a coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controller CoherentDSP 0/3/0/12

Sun Dec 26 14:34:28.391 UTC

Port : CoherentDSP 0/3/0/12
Controller State : Up
Inherited Secondary State : Normal
Configured Secondary State : Maintenance
Derived State : Maintenance
Loopback mode : Line
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 200.0Gb/s

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 1 OOM = 0 AIS = 1
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 5 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms : None

Bit Error Rate Information
PREFEC BER : 3.12E-07
POSTFEC BER : 0.00E+00
Q-Factor : 14.00 dB

Q-Margin : 6.40dB

TTI :
Remote hostname : ios
Remote interface : CoherentDSP 0/2/0/13
```

```
Remote IP addr : 0.0.0.0

FEC mode : O_FEC

Flexo-Mode : Enable
Flexo Details:
Tx GID : 1
TX IID : 1, 2,
Rx GID : 1
RX IID : 1, 2,

AINS Soak : None
AINS Timer : 0h, 0m
AINS remaining time : 0 seconds
```

Configuring Loopback on the QXP Card

Example 1

The following example shows how to configure internal loopback on a coherent DSP controller.

```
RP/0/RP0/CPU0:ios#configure
Fri Jul 8 10:42:51.329 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/0
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback internal
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Fri Jul 8 10:43:48.644 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#end
```

The following example shows how to verify the internal loopback configured on a coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/0
Fri Jul 8 10:45:53.820 UTC
Port : CoherentDSP 0/0/0/0
Controller State : Down
Inherited Secondary State : Normal
Configured Secondary State : Maintenance
Derived State : Maintenance
Loopback mode : Internal
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s
Alarm Information:
LOS = 2 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms : LOS
Bit Error Rate Information
PREFEC BER : 5.00E-01
POSTFEC BER : 0.00E+00
Q-Factor : 0.00 dB
Q-Margin : 0.00dB
OTU TTI Received
FEC mode : C_FEC
Flexo-Mode : Enable
Flexo Details:
Tx GID : 0
Rx GID : 0
AINS Soak : None
```

```
AINS Timer : 0h, 0m
AINS remaining time : 0 seconds
```

Example 2

The following example shows how to configure line loopback on a coherent DSP controller.

```
RP/0/RP0/CPU0:ios#configure
Fri Jul 8 10:48:48.577 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/0
RP/0/RP0/CPU0:ios(config-CoDSP)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-CoDSP)#loopback line
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Fri Jul 8 10:49:26.809 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#end
```

The following example shows how to verify the line loopback configured on a coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/0
Fri Jul 8 10:49:44.073 UTC
Port : CoherentDSP 0/0/0/0
Controller State : Down
Inherited Secondary State : Normal
Configured Secondary State : Maintenance
Derived State : Maintenance
Loopback mode : Line
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s
Alarm Information:
LOS = 2 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms : LOS
Bit Error Rate Information
PREFEC BER : 5.00E-01
POSTFEC BER : 0.00E+00
Q-Factor : 0.00 dB
Q-Margin : 0.00dB
OTU TTI Received
FEC mode : C_FEC
Flexo-Mode : Enable
Flexo Details:
Tx GID : 0
Rx GID : 0
AINS Soak : None
AINS Timer : 0h, 0m
AINS remaining time : 0 seconds
```

Example 3

The following example shows how to configure internal loopback on the 400GE controller.

```
RP/0/RP0/CPU0:ios#configure
Fri Jul 8 11:19:26.286 UTC
RP/0/RP0/CPU0:ios(config)#controller FourHundredGigECtrlr 0/0/0/3
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

```
Fri Jul 8 11:19:47.496 UTC
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#end
```

The following example shows how to verify the internal loopback configured on the 400GE controller.

```
RP/0/RP0/CPU0:ios#show controllers FourHundredGigECtrlr 0/0/0/3
Fri Jul 8 11:19:59.597 UTC
Operational data for interface FourHundredGigECtrlr0/0/0/3:
State:
Administrative state: enabled
Operational state: Down (Reason: State undefined)
LED state: Red On
Maintenance: Enabled
AINS Soak: None
Total Duration: 0 hour(s) 0 minute(s)
Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
Laser Squelch: Disabled
Insert Idle Ingress: Disabled
Insert Idle Egress: Disabled
Phy:
Media type: Not known
Alarms:
Current:
Loss of Signal
Statistics:
FEC:
Corrected Codeword Count: 702710
Uncorrected Codeword Count: 1147
Autonegotiation disabled.
Operational values:
Speed: 400Gbps
Duplex: Full Duplex
Flowcontrol: None
Loopback: Internal
BER monitoring:
Not supported
Forward error correction: Standard (Reed-Solomon)
Holdoff Time: 0ms
```

Example 4

The following example shows how to configure line loopback on the 4X100GE MXP.

```
RP/0/RP0/CPU0:ios(config)#controller hundredGigECtrlr 0/3/0/1/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback line
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following example shows how to verify the line loopback configured on the 4X100GE MXP.

```
RP/0/RP0/CPU0:ios#sh controllers hundredGigECtrlr 0/3/0/1/1
Fri Jul 22 10:34:39.730 UTC
Operational data for interface HundredGigECtrlr0/3/0/1/1:

State:
    Administrative state: enabled
    Operational state: Up
    LED state: Green On
    Maintenance: Enabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled
    Insert Idle Ingress: Disabled
    Insert Idle Egress: Disabled
```

```
Phy:
    Media type: Not known
    Statistics:
        FEC:
            Corrected Codeword Count: 6110368                 Valid: True       Start time:
 13:10:41 Thu Jul 21 2022
            Uncorrected Codeword Count: 2771                  Valid: True       Start time:
 13:10:41 Thu Jul 21 2022
        PCS:
            Total BIP errors: 63700992                        Valid: True       Start time:
 13:10:41 Thu Jul 21 2022
            Total frame errors: 0                             Valid: False      Start time:
 13:10:41 Thu Jul 21 2022
            Total Bad SH: 0                                   Valid: False      Start time:
 13:10:41 Thu Jul 21 2022

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: Line
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Example 5

The following example shows how to configure internal loopback on the 4X100GE MXP.

```
RP/0/RP0/CPU0:ios#conf
RP/0/RP0/CPU0:ios(config)#controller hundredGigECtrlr 0/3/0/7/1
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#sec-admin-state maintenance
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#loopback internal
RP/0/RP0/CPU0:ios(config-eth-ctrlr)#commit
```

The following example shows how to verify the internal loopback configured on the 4X100GE MXP.

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/3/0/7/1
Fri Jul 22 10:40:34.928 UTC

Operational data for interface HundredGigECtrlr0/3/0/7/1:

State:
    Administrative state: enabled
    Operational state: Down (Reason: State undefined)
    LED state: Red On
    Maintenance: Enabled
    AINS Soak: None
      Total Duration: 0 hour(s) 0 minute(s)
      Remaining Duration: 0 hour(s) 0 minute(s) 0 second(s)
    Laser Squelch: Disabled
    Insert Idle Ingress: Disabled
    Insert Idle Egress: Disabled

Phy:
    Media type: Not known
    Alarms:
        Current:
            Loss of Signal
    Statistics:
```

```
        FEC:
            Corrected Codeword Count: 31426046
            Uncorrected Codeword Count: 2187

Autonegotiation disabled.

Operational values:
    Speed: 100Gbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: Internal
    BER monitoring:
        Not supported
    Forward error correction: Standard (Reed-Solomon)
    Holdoff Time: 0ms
```

Viewing Loopback Configuration Alarm

The following example shows how to view the loopback configuration alarms on the 2-QDD-C, 1.2TC, 1.2TL, OTN-XP, and QXP cards.

```
RP/0/RP0/CPU0:ios#show alarms brief system active
Tue Sep 13 17:43:35.212 UTC

------------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------------
Location        Severity     Group            Set Time                    Description


------------------------------------------------------------------------------------



         0/2             Minor        Controller       09/13/2022 17:34:32 UTC
HundredGigECtrlr0/2/0/2 - Internal Loopback Configured
0/2             Minor        Controller       09/13/2022 17:34:32 UTC
HundredGigECtrlr0/2/0/2 - Internal Loopback Configured
0/2             Minor        Controller       09/13/2022 17:34:32 UTC
HundredGigECtrlr0/2/0/8 - Line Loopback Configured


0/2             Major        Ethernet         09/13/2022 17:34:31 UTC
HundredGigECtrlr0/2/0/4 - Loss of Synchronization The Data Interface


0/2             Minor        Controller       09/13/2022 17:37:42 UTC    OTU40/2/0/8 -
Internal Loopback Configured

0/2             Minor        Controller       09/13/2022 17:39:19 UTC    CoherentDSP0/2/0/0
 - Internal Loopback Configured
```

Restore Factory Settings

Note Perform this operation only on the console port.

You can restore the factory settings on the NCS 1004. The entire system configuration, including usernames, passwords, and IP addresses, is removed. You can perform this operation only through the console port and not on the management interface. To restore NCS 1004 to factory settings, use the **commit replace** command. After the **commit replace** operation completes, you must perform the IOS XR reload operation.

The **commit best-effort** command merges the target configuration with the running configuration and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.

Example

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#commit replace
Tue Sep 24 09:36:59.430 UTC

This commit will replace or remove the entire running configuration. This
operation can be service affecting.
Do you wish to proceed? [no]: yes
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#reload
Tue Sep 24 09:38:12.881 UTC

Standby card not present or not Ready for failover. Proceed? [confirm]

Preparing system for backup. This may take a few minutes especially for large configurations.

        Status report: node0_RP0_CPU0: BACKUP INPROGRESS

        Status report: node0_RP0_CPU0: BACKUP HAS COMPLETED SUCCESSFULLY
[Done]

Proceed with reload?  [confirm]
Reloading node  0/RP0/CPU0

RL: Reboot initiated with code 1, cause User initiated graceful reload reboot_timeout 30
shutdown delay 0
RL: Shutdown initiated
Query the node to be reloaded
 NODE_IP of noded to be reloaded 192.0.2.4
sending stop hb
Cause: User initiated graceful reload
VM IP addr sent for reload 192.0.2.4

Received ack from sdrmgr for reload request.Returncode:0
successful disconnection from service
wd_disconnect_cb 548 CMP-WD disconnected successfully
Invmgr successful disconnection from service

RP/0/RP0/CPU0:ios#
Disconnecting from 'default-sdr--1' console. Continue(Y/N)?


Connecting to 'default-sdr--1' console
ÿûÿûÿûÿýbootlogd: ioctl(/dev/pts/2, TIOCCONS): Device or resource busy
/sbin/restorecon:  lstat(/etc/adjtime) failed:  No such file or directory
Configuring network interfaces... done.
Starting system message bus: dbus.
Starting OpenBSD Secure Shell server: sshd
sshd start/running, process 1739
Starting rpcbind daemon...done.
Starting random number generator daemonUnable to open file: /dev/tpm0
.
Starting system log daemon...0
```

```
Starting kernel log daemon...0
tftpd-hpa disabled in /etc/default/tftpd-hpa
Starting internet superserver: xinetd.
net.ipv4.ip_forward = 1
Libvirt not initialized for container instance
Starting crond: OK
SIOCADDRT: File exists

DBG_MSG: platform type is 0
[*] ima_policy have loaded, or IMA policy file does not exist
Start serial incoming on , Clearing ..
RP/0/RP0/CPU0:Sep 24 09:38:44.284 UTC: fpd-serv[256]: %PKT_INFRA-FM-3-FAULT_MAJOR :
ALARM_MAJOR :FPD-NEED-UPGRADE :DECLARE :0/PM0:

This (D)RP Node is not ready or active for login /configuration
.......
........
........

ios con0/RP0/CPU0 is now available

Press RETURN to get started.

!!!!!!!!!!!!!!!!!!! NO root-system username is configured. Need to configure root-system
username. !!!!!!!!!!!!!!!!!!!
```

Headless Mode

During process restarts, CPU reload, or removal of CPU, the NCS 1004 operates in headless mode for up to 72 hours. During this time, traffic is not impacted, although the control plane is not up and running. Fault propagation continues to operate for failures on client and trunk ports. However, you cannot provision anything nor view operational data with a non-functional CPU. Performance monitoring data based on 15 minutes and 24 hour intervals is not supported with a non-functional CPU.

Trail Trace Identifier

The Trail trace identifier (TTI) feature helps you to identify the signal from the source to the destination within the network. You can configure the TTI sent or expected string only in ASCII string format. When the expected TTI string does not match the received TTI trace string, the controller goes down and the OTUK-TIM alarm is raised. To configure TTI on the coherent DSP controllers, use the following commands:

configure

controller coherentDSP *R/S/I/P* **tti** {**sent | expected**} **ascii** *tti-string*

commit

Note The *tti-string* can have a maximum of 64 characters.

The following sample displays how to configure TTI on a coherent DSP controller with the sent and expected strings set to the same ASCII string. The state of the controller is up.

```
RP/0/RP0/CPU0:ios#config
Fri Mar 15 08:03:02.094 UTC
```

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/1/0/1 tti sent ascii 1234
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/1/0/1 tti expected ascii 1234
RP/0/RP0/CPU0:ios(config)#commit
Fri Mar 15 08:03:49.725 UTC
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/1
Fri Mar 15 08:04:06.290 UTC

Port                                          : CoherentDSP 0/1/0/1
Controller State                              : Up
Inherited Secondary State                     : Normal
Configured Secondary State                    : Normal
Derived State                                 : In Service
Loopback mode                                 : None
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 1 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                               : None

Bit Error Rate Information
PREFEC  BER                                   : 7.7E-03
POSTFEC BER                                   : 0.0E+00

OTU TTI Sent
     OPERATOR SPECIFIC  ASCII                 : 1234
                                              :
     OPERATOR SPECIFIC HEX                    : 31323334000000000000000000000000
                                              : 00000000000000000000000000000000
OTU TTI Received
     OPERATOR SPECIFIC  ASCII                 : 1234
                                              :
     OPERATOR SPECIFIC HEX                    : 31323334000000000000000000000000
                                              : 00000000000000000000000000000000
OTU TTI Expected
     OPERATOR SPECIFIC  ASCII                 : 1234
                                              :
     OPERATOR SPECIFIC HEX                    : 31323334000000000000000000000000
                                              : 00000000000000000000000000000000
FEC mode                                      : Soft-Decision 27

AINS Soak                                     : None
AINS Timer                                    : 0h, 0m
AINS remaining time                           : 0 seconds
```

The following example shows how to configure TTI on a coherent DSP controller with the sent and expected strings set to different ASCII strings. The state of the controller goes down and the TIM alarm is raised.

```
RP/0/RP0/CPU0:ios#config
Fri Mar 15 08:54:29.780 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/1/0/1 tti sent ascii 1234
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/1/0/1 tti expected ascii 5678
RP/0/RP0/CPU0:ios(config)#commit
Fri Mar 15 08:56:12.293 UTC
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/1
Fri Mar 15 08:56:33.910 UTC

Port                                          : CoherentDSP 0/1/0/1
```

```
Controller State                                : Down
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable

Alarm Information:
LOS = 1 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 3 TIM = 1
FECMISMATCH = 0 FEC-UNC = 0
Detected Alarms                                 : BDI TIM

Bit Error Rate Information
PREFEC  BER                                     : 8.2E-03
POSTFEC BER                                     : 0.0E+00

OTU TTI Sent
     OPERATOR SPECIFIC  ASCII                   : 1234
                                                :
     OPERATOR SPECIFIC HEX                      : 31323334000000000000000000000000
                                                : 00000000000000000000000000000000
OTU TTI Received
     OPERATOR SPECIFIC  ASCII                   : 1234
                                                :
     OPERATOR SPECIFIC HEX                      : 31323334000000000000000000000000
                                                : 00000000000000000000000000000000
OTU TTI Expected
     OPERATOR SPECIFIC  ASCII                   : 5678
                                                :
     OPERATOR SPECIFIC HEX                      : 35363738000000000000000000000000
                                                : 00000000000000000000000000000000
FEC mode                                        : Soft-Decision 27

AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Configure TTI on OTN-XP Card

You can configure the TTI sent or expected string in the full ASCII format, or Source Access Point Identifier (SAPI)/Destination Access Point Identifier (DAPI) format on OTU, ODU, ODU-flex, ODUCn, and coherentDSP controllers for the OTN-XP card.

From R7.3.1 onwards, coherentDSP controller supports only the full ASCII string format.

From R7.3.2 onwards, coherentDSP controller supports SAPI/DAPI string format in addition to the full ASCII string format.

You can configure TTI for the following muxponder modes:

- 10G-Grey-MXP
- 4x100G-MXP-400G-TXP

The following table lists the ASCII format that is supported on each muxponder mode for TTI:

Table 52: ASCII Format Supported on Each Muxponder Mode

Muxponder Mode	ASCII with Character String	Controller
10G Grey	Full ASCII 64-character	OTU2, OTU2E, OTU4, ODU4, ODU2E (10G mapper)
	SAPI ASCII 15-character	OTU2, OTU2E, OTU4, ODU4, ODU2E (10G mapper)
	DAPI ASCII 15-character	OTU2, OTU2E, OTU4, ODU4, ODU2E (10G mapper)
	Operator-specific ASCII 32-character	OTU2, OTU2E, OTU4, ODU4, ODU2E (10G mapper)
4x100G-MXP-400G-TXP	Full ASCII 64-character	OTU4, coherentDSP, ODUC4, ODU4 (100G mapper), and ODU-FLEX (400G mapper)
	SAPI ASCII 15-character	OTU4, coherentDSP, ODUC4, ODU4 (100G mapper), and ODU-FLEX (400G mapper)
	DAPI ASCII 15-character	OTU4, coherentDSP, ODUC4, ODU4 (100G mapper), and ODU-FLEX (400G mapper)
	Operator-specific ASCII 32-character	OTU4, coherentDSP, ODUC4, ODU4 (100G mapper), and ODU-FLEX (400G mapper)

To configure TTI, use the following commands:

configure

controller *controller-type R/S/I/P* **tti** {**sent | expected**} {**ascii | sapi ascii | dapi ascii | operator-specific ascii** } *tti-string*

commit

Note We recommend that you configure TTI in the SAPI/DAPI ASCII format.

Restriction

- For OC192 and STM 64 payloads, configure both sides for ASCII and hex on mapper ODU2.
- For OC192 and STM 64 payloads, do not edit operator specific hex on mapper ODU2. Instead, delete and create the operator specific hex.

The following is a sample configuration on ODU4 controller:

```
RP/0/RP0/CPU0:ios#configure
Thu Sep 30 17:19:11.804 UTC
Current Configuration Session  Line       User     Date                      Lock
00001000-00005c15-00000000     vty0       root     Thu Sep 30 11:50:12 2021
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti sent sapi ascii cisco123
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti expect sapi ascii 123cisco
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti sent dapi ascii dapistring123
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti expected dapi ascii 123stringdapi
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti sent operator-specific ascii
operation123
RP/0/RP0/CPU0:ios(config)#controller odU4 0/1/0/12/7 tti expected operator-specific ascii
123operator
RP/0/RP0/CPU0:ios(config)#commit
Thu Sep 30 17:21:49.521 UTC
RP/0/RP0/CPU0:ios(config)#end
```

The following is a sample output of the TTI configuration on the ODU4 controller:

```
RP/0/RP0/CPU0:ios#show controllers odU4 0/1/0/12/7
Thu Sep 30 17:22:30.658 UTC

Port                                        : ODU4 0/1/0/12/7
Controller State                            : Down
Inherited Secondary state                   : Normal
Configured Secondary state                  : Normal
Derived State                               : In Service
Loopback mode                               : None
BER Thresholds                              : SF = 1.0E-6  SD = 1.0E-7

Performance Monitoring                      : Enable

Path Monitoring Mode                        : Operational
PM TIM-CA state                             : Disable

Alarm Information:
AIS = 3 IAE = 0 BIAE = 0
SF_BER = 0      SD_BER = 0      BDI = 1
OCI = 0 LCK = 2 PTIM = 0
TIM = 1 CSF = 0 GFP LFD = 0
GFP LOCS = 0    GFP LOCCS = 0   GFP UPM = 0

Detected Alarms                             : TIM

ODU TTI Sent
     SAPI  ASCII                            : c i s c o 1 2 3
     SAPI  HEX                              : 00636973636F31323300000000000000000
     DAPI  ASCII                            : d a p i s t r i n g 1 2 3
     DAPI  HEX                              : 006461706973747269 6E673132330000
     OPERATOR SPECIFIC ASCII                : operation123
     OPERATOR SPECIFIC HEX                  :
6F7065726174696F6E3132330000000000000000000000000000000000000000
ODU TTI Received

ODU TTI Expected
     SAPI  ASCII                            : 1 2 3 c i s c o
     SAPI  HEX                              : 00313233636973636F00000000000000
     DAPI  ASCII                            : 1 2 3 s t r i n g d a p i
     DAPI  HEX                              : 00313233737472696E67646170690000
     OPERATOR SPECIFIC ASCII                : 123operator
     OPERATOR SPECIFIC HEX                  :
3132336F70657261746F72000000000000000000000000000000000000000000
```

```
AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds

Private Line Emulation(PLE) supported           : No
```

You can configure TTI on OTUCn-REGEN mode on the OTN-XP Card.

The following sample displays how to configure TTI on a coherent DSP controller port 12 on the OTUCn-REGEN mode.

```
Mon Dec 27 12:03:53.642 UTC
RP/0/RP0/CPU0:ios(config)#controller CoherentDSP 0/3/0/12 tti sent ascii 1234cisco
RP/0/RP0/CPU0:ios(config)#commit
Mon Dec 27 12:03:54.333 UTC
RP/0/RP0/CPU0:ios(config)#end
Mon Dec 27 12:03:55.434 UTC
RP/0/RP0/CPU0:ios(config)#controller CoherentDSP 0/3/0/12 tti expected ascii cisco1234
RP/0/RP0/CPU0:ios(config)#commit
Mon Dec 27 12:03:56.137 UTC
RP/0/RP0/CPU0:ios(config)#end
```

The following sample verifies the TTI configuration on the inverse muxponder configured on the OTUCn-REGEN mode.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/12
Tue May 24 17:49:14.301 UTC

Port                                            : CoherentDSP 0/0/0/12
Controller State                                : Up
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : In Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 400.0Gb/s

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 1 TIM = 1
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 1.55E-04
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 11.10 dB

Q-Margin                                        : 4.70dB

OTU TTI Sent
     FULL TTI ASCII                             : cisco123
                                                :
     FULL TTI HEX                               : 636973636F3132330000000000000000
                                                : 00000000000000000000000000000000
OTU TTI Received
     FULL TTI ASCII                             : 123cisco
                                                :
     FULL TTI HEX                               : 313233636973636F0000000000000000
```

```
                                                  : 00000000000000000000000000000000
OTU TTI Expected
     FULL TTI ASCII                               : 123cisco
                                                  :
     FULL TTI HEX                                 : 313233636973636F0000000000000000
                                                  : 00000000000000000000000000000000
FEC mode                                          : O_FEC

Flexo-Mode                                        : Enable
Flexo Details:
        Tx GID                                    : 1
        TX IID                                    : 1, 2, 3, 4,
        Rx GID                                    : 1
        RX IID                                    : 1, 2, 3, 4,

AINS Soak                                         : None
AINS Timer                                        : 0h, 0m
AINS remaining time                               : 0 seconds


RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#show running-config controller coherentDSP 0/0/0/12
Tue May 24 17:49:21.749 UTC
controller CoherentDSP0/0/0/12
tti
  expected ascii 123cisco
  sent ascii cisco123
!
!
```

Configure TTI on Inverse Muxponder Configuration on the OTN-XP Card

The following sample displays how to configure TTI on a coherent DSP controller port 12 on the OTN-XP in inverse muxponder configuration mode.

Note TTI configuration is not supported on the DSP controller port 13.

```
RP/0/RP0/CPU0:ios#configure
Thu Sep 30 14:18:13.288 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/2/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent sapi ascii cisco
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following sample verifies the TTI configuration on the inverse muxponder configured on the OTN-XP Card.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/12
Thu Sep 30 14:19:05.367 UTC

Port                                              : CoherentDSP 0/2/0/12
Controller State                                  : Up
Inherited Secondary State                         : Normal
Configured Secondary State                        : Maintenance
Derived State                                     : Maintenance
Loopback mode                                     : Internal
BER Thresholds                                    : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                            : Enable
Bandwidth                                         : 200.0Gb/s

Alarm Information:
```

```
LOS = 2 LOF = 0 LOM = 0
OOF = 1 OOM = 0 AIS = 1
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 1
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 4.11E-09
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 14.90 dB

Q-Margin                                        : 8.60dB

OTU TTI Sent
     SAPI  ASCII                                : c i s c o
     SAPI  HEX                                  : 00636973636F00000000000000000000
     DAPI  ASCII                                :
     DAPI  HEX                                  :
     OPERATOR SPECIFIC ASCII                    :
     OPERATOR SPECIFIC HEX                      :
CDCDCDCDED00DBBE2100000000000000050D9D29AD7F00007603BADC7698BADC
OTU TTI Received
     SAPI  ASCII                                : c i s c o
     SAPI  HEX                                  : 00636973636F00000000000000000000
FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 1, 2,
        Rx GID                                  : 1
        RX IID                                  : 1, 2,

Flexo Peers Information:
        Controller                              : CoherentDSP0_2_0_13
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

Enable TIM CA on Path Monitoring Layer

You can enable Trace Identifier Mismatch (TIM) consequent action (CA) on the Path Monitoring (PM) layer using the **pm-tim-ca** command on mapper ODUs for Ethernet controller. The TTI transmit string in the SAPI/DAPI format is not configurable on ODUs that are transparent.

For example, the clients that are supported are ODU4, ODU2, and ODU2E, and lower-order ODUs such as ODU2 or ODU2E.

You can configure **pm-tim-ca** only on mapper ODUs such as ODU2E (10G mapper), ODU4 (100G mapper), and ODU-FLEX (400G mapper).

To configure **pm-tim-ca** on mapper ODU in the 4x100G-MXP-400G-TXP muxponder mode, use the following commands

configure

controller *controller-type R/S/I/P*

pm-tim-ca

commit

Configure TTI on QXP Card

From R7.10.1, you can configure the TTI sent or expected string in the full ASCII format, or Source Access Point Identifier (SAPI)/Destination Access Point Identifier (DAPI) format on ODU-flex, ODU4, and coherentDSP controllers for the QXP card.

Note TTI operates only in **trunk mode OR**.

The following table lists the ASCII format that is supported for TTI:

ASCII with Character String	Controller
Full ASCII 64-character	CoherentDSP,odu4,odu-flex
SAPI ASCII 15-character	CoherentDSP,odu4,odu-flex
DAPI ASCII 15-character	CoherentDSP,odu4,odu-flex
Operator-specific ASCII 32-character	CoherentDSP,odu4,odu-flex

To configure TTI, use the following commands:

configure

controller *controller-type R/S/I/P* **tti** {**sent | expected**} {**ascii | sapi ascii | dapi ascii | operator-specific ascii** } *tti-string*

commit

The following is a sample configuration for FULL TTI for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent ascii cisco
RP/0/RP0/CPU0:ios(config-CoDSP)#tti  expected ascii cisco123
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following is a sample configuration for TTI HEX for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent hex 6E6E6E2A2A2A
RP/0/RP0/CPU0:ios(config-CoDSP)#tti  expected hex 3F4B4B4B3D3E3A
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following is a sample configuration for Operator specific TTI for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent operator-specific ascii hellooo
```

```
RP/0/RP0/CPU0:ios(config-CoDSP)#tti expected operator-specific ascii hellooo
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following is a sample configuration for Operator specific TTI HEX for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti  sent operator-specific hex
6E6E6E2A2A2A3D3E3A3A6E6E6E2A2A2A3D
RP/0/RP0/CPU0:ios(config-CoDSP)#tti  expected operator-specific hex 5A5A6D3A3B3C3F4B4B4B3D3E3A
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following is a sample configuration for SAPI for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent operator-specific ascii hellooo
RP/0/RP0/CPU0:ios(config-CoDSP)#tti expected operator-specific ascii hellooo
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

The following is a sample configuration for DAPI for coherentDSP controller

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/8
RP/0/RP0/CPU0:ios(config-CoDSP)#tti sent dapi ascii cisco123
RP/0/RP0/CPU0:ios(config-CoDSP)#tti  expected dapi ascii hello
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

Chromatic dispersion

You can configure chromatic dispersion on optics controllers. When you configure the maximum and minimum values for chromatic dispersion for any data rate, ensure the minimum difference between the configured values is equal to or greater than 1500 ps/nm.

The following table lists the default CD search range.

Data Rate	BPS	Card Support	Default CD Search Range
200G to 500G	BPS < = 3	1.2T, 1.2TL	-10,000 to 100,000 ps/nm
	3 < BPS <= 4	1.2T, 1.2TL	-10,000 to 80,000 ps/nm
	4 < BPS <=5	1.2T	-5,000 to 20,000 ps/nm
600G	BPS=5.2578125	1.2T	-2000 to 2,000 ps/nm
400G for 400G CFP2 DCO	BPS=4	OTN-XP	-24,000 to 24,000 ps/nm
400GE for ZRP	BPS=4	OTN-XP	For CFEC, -2,400 to 2,400 ps/nm For OFEC, -13,000 to 13,000 ps/nm
100G	BPS <= 3	2-QDD-C	-10000 to +100000 ps/nm

Data Rate	BPS	Card Support	Default CD Search Range
200G to 300G	BPS <= 3	2-QDD-C	-10000 to +100000 ps/nm
	3> BPS =4		-10000 to +80000 ps/nm
	4> BPS =5		-5000 to +200000 ps/nm
	5> BPS =6		-5000 to +100000 ps/nm
400G	BPS = 4	2-QDD-C	-10000 to +80000 ps/nm
	4> BPS =5		-5000 to +20000 ps/nm
	5> BPS =6		-5000 to +10000 ps/nm
400GE for ZRP	BPS=4	QXP	For CFEC, -2,400 to 2,400 ps/nm For OFEC, -160000 to 160000 ps/nm

Note The cd-min and cd-max values must be set for BPS values that are greater than 4 in the 1.2T card.

Note When the user provisions the cd-min and cd-max values that are outside the range through CLI, the provisioned values are accepted; however, only the actual values supported by the hardware are applied.

The following is a sample where chromatic dispersion is configured on the optics controller.

```
RP/0/RP0/CPU0:ios#configure
Mon Aug 19 19:31:42.115 UTC
RP/0/RP0/CPU0:ios(config)#controller  optics 0/1/0/1
RP/0/RP0/CPU0:ios(config-Optics)#cd-max 4000
RP/0/RP0/CPU0:ios(config-Optics)#cd-min -1000
RP/0/RP0/CPU0:ios(config-Optics)#commit
Mon Aug 19 19:35:24.697 UTC
RP/0/RP0/CPU0:ios(config-Optics)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show run controller optics 0/1/0/*
Mon Aug 19 19:57:41.859 UTC
controller Optics0/1/0/0
 transmit-power -15
 dwdm-carrier 50GHz-grid itu-ch 55
 enh-sop-tol-mode 1
 cross-pol-gain-mode 10
 lbc-high-threshold 5
!
controller Optics0/1/0/1
 description trunk power UP
 cd-min -1000
 cd-max 4000
 enh-colorless-mode 2
 enh-sop-tol-mode 3
```

```
 nleq-comp-mode 4
 cross-pol-gain-mode 2
 cross-pol-weight-mode 3
 cpr-win-mode 3
 cpr-ext-win-mode 8
 rx-voa fixed-ratio 1200
 filter-roll-off-factor 0.035
!
controller Optics0/1/0/5
 soak-time 10
!
```

Transmit Power

From Release 7.3.1 onwards, you can configure transmit power on the CFP2 DCO optics for the OTN-XP card. The value ranges from -10 to +1 dBm.

From Release 7.3.2 onwards, you can configure transmit power on the QDD ZRP optics for the OTN-XP card. The following are the value ranges for OFEC and CFEC:

FEC Types	Transmit Power (dBm)
OFEC	-13 to -9
CFEC	-10 to -6

To configure transmit power on the CFP2 DCO optics for the OTN-XP card, use the following commands:

```
RP/0/RP0/CPU0:ios#configure
Mon Aug 19 19:31:42.115 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/1/0/12
RP/0/RP0/CPU0:ios(config-Optics)#transmit-power -1.50
RP/0/RP0/CPU0:ios(config-Optics)#commit
Mon Aug 19 19:35:24.697 UTC
RP/0/RP0/CPU0:ios(config-Optics)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following is a sample in which transmit power of -1.50 dBm is configured on the CFP2 DCO optics for the OTN-XP card.

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#do show controllers optics 0/1/0/12
Mon Jan 18 19:05:26.009 UTC

Controller State: Down

Transport Admin State: In Service

Laser State: On

LED State: Green

Optics Status

Optics Type: CFP2 DWDM
DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
Wavelength=1552.524nm

Alarm Status:
```

```
-------------
Detected Alarms: None


LOS/LOL/Fault Status:

Alarm Statistics:

-------------
HIGH-RX-PWR = 0 LOW-RX-PWR = 0
HIGH-TX-PWR = 0 LOW-TX-PWR = 0
HIGH-LBC = 0 HIGH-DGD = 0
OOR-CD = 0 OSNR = 0
WVL-OOL = 0 MEA = 0
IMPROPER-REM = 0
TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 %
Actual TX Power = -1.47 dBm
RX Power = -0.86 dBm
RX Signal Power = 0.86 dBm
Frequency Offset = 0 MHz

Performance Monitoring: Enable

THRESHOLD VALUES
----------------

Parameter High Alarm Low Alarm High Warning Low Warning
----------------------- ---------- --------- ------------ -----------
Rx Power Threshold(dBm) 4.9 -12.0 0.0 0.0
Tx Power Threshold(dBm) 3.5 -10.1 0.0 0.0
LBC Threshold(mA) N/A N/A 0.00 0.00

LBC High Threshold = 98 %
Configured Tx Power = -1.50 dBm
Configured CD High Threshold = 180000 ps/nm
Configured CD lower Threshold = -180000 ps/nm
Configured OSNR lower Threshold = 0.00 dB
Configured DGD Higher Threshold = 180.00 ps
Baud Rate = 63.0999984741 GBd
Bits per Symbol = 4.0000000000 bits/symbol
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -26000 ps/nm CD-MAX 26000 ps/nm
Polarization Mode Dispersion = 0.0 ps
Second Order Polarization Mode Dispersion = 156.00 ps^2
Optical Signal to Noise Ratio = 35.80 dB
SNR = 10.50 dB
Polarization Dependent Loss = 0.00 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 1.00 ps
Filter Roll Off Factor : 0.000
Rx VOA Fixed Ratio : 0.00 dB
Enhanced Colorless Mode : 0
Enhanced SOP Tolerance Mode : 0
```

Transmit Power on QDD ZRP Optics

To configure transmit power on the QDD ZRP optics for the OTN-XP card, use the following commands:

```
RP/0/RP0/CPU0:ios#configure
Mon Aug 19 19:31:42.115 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/1/0/9
```

```
RP/0/RP0/CPU0:ios(config-Optics)#transmit-power -8
RP/0/RP0/CPU0:ios(config-Optics)#commit
Mon Aug 19 19:35:24.697 UTC
RP/0/RP0/CPU0:ios(config-Optics)#
```

The following is a sample in which transmit power of -8 dBm is configured on the QDD ZRP optics for the OTN-XP card:

```
RP/0/RP0/CPU0:ios#show controller optics 0/1/0/9
Wed Sep 15 00:36:24.383 UTC

 Controller State: Up

 Transport Admin State: In Service

 Laser State: On

 LED State: Green

 Optics Status

         Optics Type:  QSFP-DD DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=49, Frequency=193.70THz,
         Wavelength=1547.715nm

         Alarm Status:
         -------------
         Detected Alarms: None


         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 0            LOW-RX-PWR = 1
         HIGH-TX-PWR = 0            LOW-TX-PWR = 1
         HIGH-LBC = 0               HIGH-DGD = 0
         OOR-CD = 0                 OSNR = 1
         WVL-OOL = 0                MEA  = 0
         IMPROPER-REM = 0
         TX-POWER-PROV-MISMATCH = 0
         Laser Bias Current = 0.0 %
         Actual TX Power = -8 dBm
         RX Power = -7.31 dBm
         RX Signal Power = -7.67 dBm
         Frequency Offset = 81 MHz
 Performance Monitoring: Enable

         THRESHOLD VALUES
         ----------------

         Parameter                High Alarm  Low Alarm  High Warning  Low Warning
         -----------------------  ----------  ---------  ------------  -----------
         Rx Power Threshold(dBm)          3.0      -23.5           0.0          0.0
         Tx Power Threshold(dBm)          0.0      -16.0           0.0          0.0
         LBC Threshold(mA)                N/A        N/A          0.00         0.00

         LBC High Threshold = 90 %
         Configured Tx Power = -7.00 dBm
         Configured CD High Threshold = 2400 ps/nm
         Configured CD lower Threshold = -2400 ps/nm
         Configured OSNR lower Threshold = 24.00 dB
         Configured DGD Higher Threshold = 40.00 ps
```

```
        Baud Rate =  59.8437500000 GBd
        Bits per Symbol = 4.0000000000  bits/symbol
        Modulation Type: 16QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -2400 ps/nm  CD-MAX 2400 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.40 dB
        SNR = 17.30 dB
        Polarization Dependent Loss = 0.40 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 3.00 ps

Transceiver Vendor Details

        Form Factor            : QSFP-DD
        Name                   : CISCO
        Part Number            : 10-3496-01
        Rev Number             : 11
        Serial Number          : 210153241
        PID                    : QDD-400G-ZRP-S
        VID                    : ES04
        Date Code(yy/mm/dd)    : 20/21/01
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set
Transceiver Temperature : 57 Celsius
AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Laser Bias Current High Threshold

You can configure the threshold of the laser bias current flowing on the phusical pluggable port on the trunk optics controller. The range is 0 to 100%

To configure the laser bias current threshold, use the following command:

configure

controller optics *R/S/I/P*

lbc-high-threshold *lbc-value*

commit

The following sample configures the high laser bias threshold on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#lbc-high-threshold 55
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the high rlaser bias threshold configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service
```

```
Laser State: On

LED State: Yellow

Optics Status

        Optics Type:  CFP2 DWDM
        DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
        Wavelength=1552.524nm

        Alarm Status:
        -------------
        Detected Alarms:
                HIGH-RX-PWR   LOW-TX-PWR
                HIGH-DGD

        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 1             LOW-RX-PWR = 0
        HIGH-TX-PWR = 0             LOW-TX-PWR = 1
        HIGH-LBC = 0                HIGH-DGD = 6
        OOR-CD = 0                  OSNR = 0
        WVL-OOL = 0                 MEA  = 0
        IMPROPER-REM = 1
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = -0.53 dBm
        RX Signal Power = -1.20 dBm
        Frequency Offset = 63 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                 High Alarm  Low Alarm  High Warning  Low Warning
        ------------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
        Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 55 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 0.40 dB
        Configured DGD Higher Threshold = 0.30 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.10 dB
        SNR = 17.50 dB
        Polarization Dependent Loss = 0.50 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps
```

```
Transceiver Vendor Details

         Form Factor            : CFP2
         Name                   : CISCO-ACACIA
         Part Number            : 10-3500-01
         Rev Number             : 01
         Serial Number          : ACA24480037
         PID                    : ONS-CFP2D-400G-C
         VID                    : VES1
         Date Code(yy/mm/dd)    : 20/11/10
         Fiber Connector Type: LC
         Otn Application Code: Not Set
         Sonet Application Code: Not Set
         Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius



AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Differential Group Delay Threshold

You can configure the threshold value for the maximum acceptable differential group delay (DGD) on the trunk optics controllers. The DGD alarm is raised if DGD exceeds this value.

The range is 0–18000 (in the units of 0.01 ps).

To configure the maximum acceptable DGD, use the following command:

configure

controller optics *R/S/I/P*

dgd-high-threshold *dgd-value*

commit

The following sample configures the minimum acceptable DGD on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#dgd-high-threshold 30
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the maximum acceptable DGD configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Yellow

Optics Status

         Optics Type:  CFP2 DWDM
```

```
        DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
        Wavelength=1552.524nm

        Alarm Status:
        -------------
        Detected Alarms:
                HIGH-RX-PWR   LOW-TX-PWR
                HIGH-DGD

        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 1            LOW-RX-PWR = 0
        HIGH-TX-PWR = 0            LOW-TX-PWR = 1
        HIGH-LBC = 0               HIGH-DGD = 6
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 1
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = -0.53 dBm
        RX Signal Power = -1.20 dBm
        Frequency Offset = 63 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
        Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 55 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 0.40 dB
        Configured DGD Higher Threshold = 0.30 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.10 dB
        SNR = 17.50 dB
        Polarization Dependent Loss = 0.50 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3500-01
        Rev Number             : 01
        Serial Number          : ACA24480037
```

```
          PID                   : ONS-CFP2D-400G-C
          VID                   : VES1
          Date Code(yy/mm/dd)   : 20/11/10
          Fiber Connector Type: LC
          Otn Application Code: Not Set
          Sonet Application Code: Not Set
          Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius



AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Optical Signal to Noise Ratio

You can configure the minimum acceptable Optical Signal to Noise ratio (OSNR) value. The OSNR alarm is raised if OSNR goes below this value.

The range is 0–4000 (in units of 0.01db).

To configure the minimum acceptable OSNR, use the following command:

configure

controller optics *R/S/I/P*

osnr-low-threshold *osnr-value*

commit

The following sample configures the minimum acceptable OSNR on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#osnr-low-threshold 40
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the minimum acceptable OSNR configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Yellow

Optics Status

          Optics Type:  CFP2 DWDM
          DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
          Wavelength=1552.524nm

          Alarm Status:
          -------------
          Detected Alarms:
                  HIGH-RX-PWR   LOW-TX-PWR
```

```
        HIGH-DGD

LOS/LOL/Fault Status:

Alarm Statistics:

-------------
HIGH-RX-PWR = 1            LOW-RX-PWR = 0
HIGH-TX-PWR = 0            LOW-TX-PWR = 1
HIGH-LBC = 0               HIGH-DGD = 6
OOR-CD = 0                 OSNR = 0
WVL-OOL = 0                MEA  = 0
IMPROPER-REM = 1
TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 %
Actual TX Power = 0.97 dBm
RX Power = -0.53 dBm
RX Signal Power = -1.20 dBm
Frequency Offset = 63 MHz

Performance Monitoring: Enable

THRESHOLD VALUES
----------------

Parameter                High High Alarm  Low Alarm  High Warning  Low Warning
-----------------------  ----------  ---------  ------------  -----------
Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
LBC Threshold(mA)                N/A        N/A          0.00         0.00

LBC High Threshold = 55 %
Configured Tx Power = 1.00 dBm
Configured CD High Threshold = 2400 ps/nm
Configured CD lower Threshold = -2400 ps/nm
Configured OSNR lower Threshold = 0.40 dB
Configured DGD Higher Threshold = 0.30 ps
Baud Rate =  63.1394679230 GBd
Bits per Symbol = 3.0000000000  bits/symbol
Modulation Type: 8QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
Polarization Mode Dispersion = 0.0 ps
Second Order Polarization Mode Dispersion = 29.00 ps^2
Optical Signal to Noise Ratio = 36.10 dB
SNR = 17.50 dB
Polarization Dependent Loss = 0.50 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3500-01
        Rev Number             : 01
        Serial Number          : ACA24480037
        PID                    : ONS-CFP2D-400G-C
        VID                    : VES1
        Date Code(yy/mm/dd)    : 20/11/10
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set
```

```
Transceiver Temperature : 46 Celsius


AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Chromatic Dispersion Threshold

You can configure the minimum and maximum acceptable chromatic dispersion for the trunk optics controllers. The CD alarm is raised if the chromatic dispersion goes below the minimum or exceeds the maximum value.

The following is a sample of configuring the minimum and maximum chromatic dispersion threshold:

To configure the maximum nd minimum acceptable CD, use the following command:

configure

controller optics *R/S/I/P*

cd-high-threshold *cd-high*

cd-low-threshold *cd-low*

commit

The following sample configures the maximum and minimum acceptable CD on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#cd-high-threshold 2400
RP/0/RP0/CPU0:ios(config-Optics)#cd-low-threshold -2400
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the maximum and minimum acceptable CD configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Yellow

Optics Status

         Optics Type:  CFP2 DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
         Wavelength=1552.524nm

         Alarm Status:
         -------------
         Detected Alarms:
                 HIGH-RX-PWR   LOW-TX-PWR
                 HIGH-DGD

         LOS/LOL/Fault Status:
```

```
        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 1            LOW-RX-PWR = 0
        HIGH-TX-PWR = 0            LOW-TX-PWR = 1
        HIGH-LBC = 0               HIGH-DGD = 6
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 1
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = -0.53 dBm
        RX Signal Power = -1.20 dBm
        Frequency Offset = 63 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
        Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 55 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 0.40 dB
        Configured DGD Higher Threshold = 0.30 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.10 dB
        SNR = 17.50 dB
        Polarization Dependent Loss = 0.50 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3500-01
        Rev Number             : 01
        Serial Number          : ACA24480037
        PID                    : ONS-CFP2D-400G-C
        VID                    : VES1
        Date Code(yy/mm/dd)    : 20/11/10
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius
```

```
AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Receive Power Threshold

You can configure the high and low threshold of the total optical signal power of the received signal on the trunk optics controller.

The range is –400 to 300 (in the units of 0.1 dBm).

To configure the high and low receive power threshold, use the following command:

configure

controller optics *R/S/I/P*

rx-high-threshold *rx-high*

rx-low-threshold *rx-low*

commit

The following sample configures the high receive power threshold on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#rx-high-threshold -20
RP/0/RP0/CPU0:ios(config-Optics)#rx-low-threshold -30
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the high receive power threshold configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Yellow

Optics Status

         Optics Type:  CFP2 DWDM
         DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
         Wavelength=1552.524nm

         Alarm Status:
         -------------
         Detected Alarms:
                 HIGH-RX-PWR   LOW-TX-PWR
                 HIGH-DGD

         LOS/LOL/Fault Status:

         Alarm Statistics:

         -------------
         HIGH-RX-PWR = 1            LOW-RX-PWR = 0
```

```
        HIGH-TX-PWR = 0            LOW-TX-PWR = 1
        HIGH-LBC = 0               HIGH-DGD = 6
        OOR-CD = 0                 OSNR = 0
        WVL-OOL = 0                MEA  = 0
        IMPROPER-REM = 1
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = -0.53 dBm
        RX Signal Power = -1.20 dBm
        Frequency Offset = 63 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                 High Alarm  Low Alarm  High Warning  Low Warning
        ------------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
        Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 55 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 0.40 dB
        Configured DGD Higher Threshold = 0.30 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.10 dB
        SNR = 17.50 dB
        Polarization Dependent Loss = 0.50 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3500-01
        Rev Number             : 01
        Serial Number          : ACA24480037
        PID                    : ONS-CFP2D-400G-C
        VID                    : VES1
        Date Code(yy/mm/dd)    : 20/11/10
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius


AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Transmit Power Threshold

You can configure the high and low threshold of the total optical signal power of the transmitted signal on the trunk optics controller.

The range is –400 to 300 (in the units of 0.1 dBm).

To configure the high and low transmit power threshold, use the following command:

configure

controller optics *R/S/I/P*

tx-high-threshold *tx-high*

tx-low-threshold *tx-low*

commit

The following sample configures the high transmit power threshold on the controller optics:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/12
RP/0/RP0/CPU0:ios(config-Optics)#tx-high-threshold 40
RP/0/RP0/CPU0:ios(config-Optics)#tx-low-threshold 20
RP/0/RP0/CPU0:ios(config-Optics)#commit
```

The following sample shows the high transmit power threshold configured on the controller optics:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/12
Fri Nov 12 10:58:50.595 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: On

LED State: Yellow

Optics Status

        Optics Type:  CFP2 DWDM
        DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
        Wavelength=1552.524nm

        Alarm Status:
        -------------
        Detected Alarms:
                HIGH-RX-PWR   LOW-TX-PWR
                HIGH-DGD

        LOS/LOL/Fault Status:

        Alarm Statistics:

        -------------
        HIGH-RX-PWR = 1             LOW-RX-PWR = 0
        HIGH-TX-PWR = 0             LOW-TX-PWR = 1
        HIGH-LBC = 0                HIGH-DGD = 6
        OOR-CD = 0                  OSNR = 0
        WVL-OOL = 0                 MEA  = 0
        IMPROPER-REM = 1
```

```
        TX-POWER-PROV-MISMATCH = 0
        Laser Bias Current = 0.0 %
        Actual TX Power = 0.97 dBm
        RX Power = -0.53 dBm
        RX Signal Power = -1.20 dBm
        Frequency Offset = 63 MHz

        Performance Monitoring: Enable

        THRESHOLD VALUES
        ----------------

        Parameter                High Alarm  Low Alarm  High Warning  Low Warning
        -----------------------  ----------  ---------  ------------  -----------
        Rx Power Threshold(dBm)         -2.0       -3.0           0.0          0.0
        Tx Power Threshold(dBm)          4.0        2.0           0.0          0.0
        LBC Threshold(mA)                N/A        N/A          0.00         0.00

        LBC High Threshold = 55 %
        Configured Tx Power = 1.00 dBm
        Configured CD High Threshold = 2400 ps/nm
        Configured CD lower Threshold = -2400 ps/nm
        Configured OSNR lower Threshold = 0.40 dB
        Configured DGD Higher Threshold = 0.30 ps
        Baud Rate =  63.1394679230 GBd
        Bits per Symbol = 3.0000000000  bits/symbol
        Modulation Type: 8QAM
        Chromatic Dispersion 0 ps/nm
        Configured CD-MIN -48000 ps/nm  CD-MAX 48000 ps/nm
        Polarization Mode Dispersion = 0.0 ps
        Second Order Polarization Mode Dispersion = 29.00 ps^2
        Optical Signal to Noise Ratio = 36.10 dB
        SNR = 17.50 dB
        Polarization Dependent Loss = 0.50 dB
        Polarization Change Rate = 0.00 rad/s
        Differential Group Delay = 1.00 ps

Transceiver Vendor Details

        Form Factor            : CFP2
        Name                   : CISCO-ACACIA
        Part Number            : 10-3500-01
        Rev Number             : 01
        Serial Number          : ACA24480037
        PID                    : ONS-CFP2D-400G-C
        VID                    : VES1
        Date Code(yy/mm/dd)    : 20/11/10
        Fiber Connector Type: LC
        Otn Application Code: Not Set
        Sonet Application Code: Not Set
        Ethernet Compliance Code: Not set

Transceiver Temperature : 46 Celsius


AINS Soak               : None
AINS Timer              : 0h, 0m
AINS remaining time     : 0 seconds
```

Frequency

You can configure the frequency on trunk ports of the line card.

The following table lists the frequency range with grid spacing supported on the line card:

Line Card	Frequency Range (THz)	Default Frequency (THz)	Grid Spacing
1.2T	191.25 to 196.1	193.1	50GHz and 100MHz
1.2TL [1]	186.1 to 190.85	188.5	100MHz
OTN-XP CFP2 trunk	191.275 to 196.125	193.1	50GHz and 100MHz
OTN-XP QDD ZRP	191.275 to 196.125	193.70	6.25GHz, 50GHz, and 100MHz
2-QDD-C	191.15 to 196.1	193.1	50GHz and 100MHz

[1] Only non-ITU channels are supported

To configure the wavelength, use the following commands:

configure

controller optics *Rack/Slot/Instance/Port*

dwdm-carrier {**100MHz-grid frequency** *frequency*} | {**50GHz-grid** [**frequency** *frequency*}

commit

Pseudo Random Binary Sequence

Table 53: Feature History

Feature Name	Release Information	Feature Description
PRBS (Pseudo Random Binary Sequence) on ODU4	Cisco IOS XR Release 7.3.1	1.2T card supports PRBS on the ODU4 controller. This feature allows you to test whether the traffic is error free during link bring up without depending on the peer port.

The Pseudo Random Binary Sequence (PRBS) feature allows you to perform data integrity checks between the NCS1004 trunk links without enabling the actual client traffic.

You need to enable PRBS feature on both the transmitting and receiving NCS 1004 trunk ports. The transmitting trunk port generates a bit pattern and sends it to the peer NCS 1004 device. The device detects if the sent bit pattern is received.

From R7.3.1 onwards, you can configure PRBS on the NCS 1004 trunk port for the 2-QDD-C card.

You can configure NCS 1004 trunk port in any one of the following modes for PRBS on the 1.2T card:

- **Source mode —** The NCS 1004 at trunk port generates PRBS signal on the line continuously as per the configured PRBS pattern.
- **Sink mode —** The NCS 1004 at trunk port gets locked to the ingress signal according to the configured pattern, analyzes and reports the errors.
- **Source-Sink mode —** The NCS 1004 at trunk port acts as both the PRBS transmitter and receiver, that is, it generates PRBS signal as per the configured pattern, and also gets locked to the ingress signal with the same pattern, and reports the errors.

Note From R7.3.1 onwards, the 1.2T card supports PRBS on ODU4.

NCS 1004 trunk port supports the following PRBS patterns:

- **PRBS31 —** Sequence length is from 2^31 -1 bits.
- **PRBS23 —** Sequence length is from 2^23 -1 bits.
- **PRBS15 —** Sequence length is from 2^15 -1 bits.
- **PRBS7 —**Sequence length is from 2^7 -1 bits.

Tip We recommend that for higher datarates like 100G and 400G:

- use high sequence length PRBS patterns and
- use PRBS inverted pattern.

Limitations of PRBS

There are following limitations with the PRBS feature:

- There is no SNMP support to fetch the PRBS status or Performance Monitoring (PM).
- TTI functionality is not supported with PRBS.
- Loopback and PRBS configurations cannot coexist on a coherentDSP controller. Loopback configuration will be rejected if PRBS is already configured.
- PRBS on ODU4 is supported only when the slice is provisioned in OTN client mode.

PRBS on OTN-XP Card

From R7.2.1 onwards, the OTN-XP card supports PRBS on the mapper optical data unit (ODU2e).

Note ODU2e PRBS is not supported for OTU2E client rates.

NCS 1004 with the OTN-XP card, supports the following PRBS mode:

- **Source mode —** The NCS 1004 at trunk port generates PRBS signal on the line continuously as per the configured PRBS pattern.
- **Sink mode —** The NCS 1004 at trunk port gets locked to the ingress signal according to the configured pattern, analyzes and reports the errors.
- **Source-Sink mode —** The NCS 1004 at trunk port acts as both the PRBS transmitter and receiver, that is, it generates PRBS signal as per the configured pattern, and also gets locked to the ingress signal with the same pattern, and reports the errors.
- **invertedpn31 —**Inverted pattern. Sequence length is from 2^31 -1 bits.

NCS 1004 trunk port supports the following PRBS patterns:

- **PRBS31 —** Sequence length is from 2^31 -1 bits.
- **PRBS23 —** Sequence length is from 2^23 -1 bits.
- **PRBS15 —** Sequence length is from 2^15 -1 bits.
- **PRBS7 —**Sequence length is from 2^7 -1 bits.

Configuring Pseudo Random Binary Sequence

The secondary admin state of the coherentDSP or ODU4 controller must be set to maintenance before enabling PRBS.

To enable the PRBS on the trunk port, use the following configuration command at the coherentDSP controller:

controller coherentDSP *R/S/I/P* **prbs mode** {**source | sink | source-sink**} **pattern** {**pn31 | pn23 | pn15 | pn7**}

To enable the PRBS on the trunk port, use the following configuration command at the ODU4 controller:

controller odu4 *R/S/I/P/L* **opu prbs mode** {**source | sink | source-sink**} **pattern** {**pn31 | pn23 | pn15 | pn7**}

When the PRBS is enabled on the trunk ports, you can view the following impacts in the corresponding client ports:

- Client traffic is dropped in the direction of source to sink as the frames are overwritten by the PRBS pattern.
- Remote fault is raised on the client ports nearer to the PRBS sink.

The following are the ODU4 PRBS combinations. The client ports must have physical loop back in all the combinations.

- Near End client and Near End trunk ODU4
- Near End client and Far End client ODU4
- Near End client and Far End trunk ODU4
- Near End trunk and Far End trunk ODU4

The following sample diagram describes the ODU4 PRBS combination for Near End client and Near End trunk.

Verifying PRBS

You can monitor the status of Pseudo Random Binary Sequence (PRBS) on the CoherentDSP or ODU4 controller using the following command:

show controllers coherentDSP | ODU4 *R/S/I/P* **prbs-details**

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/0 prbs-details
Wed Nov  6 23:12:22.464 UTC

----------------------PRBS details------------------
PRBS Test               : Enable
PRBS Mode               : Source
PRBS Pattern            : PN7
PRBS Status             : Not Applicable

-----------------------------------------------------

RP/0/RP0/CPU0:ios#show controllers ODU4 0/3/0/8 prbs-details
Mon Jan 11 05:29:12.436 UTC

 -------------------------PRBS details----------------------------------
PRBS Test                       : Enable
PRBS Mode                       : Source
PRBS Pattern                    : PN7
PRBS Status                     : Not Applicable

-----------------------------------------------------

RP/0/RP0/CPU0:ios#show controllers ODU4 0/3/0/1/1 prbs-details
Mon Jan 11 05:27:56.370 UTC

 -------------------------PRBS details----------------------------------
PRBS Test                       : Enable
PRBS Mode                       : Sink
PRBS Pattern                    : PN7
PRBS Status                     : Locked

-----------------------------------------------------
```

- You cannot view any details, if the PRBS is not enabled on the trunk.
- PRBS status is shown as **Not Applicable**, when the mode is **Source**.
- PRBS status is shown as **unlocked**, when the signal is not locked on the receiving side in the **Sink** or **Source-Sink** mode.

Viewing PRBS Performance Monitoring Parameters

PRBS PM parameters are not available for the controllers in Source mode. PRBS PM parameters are reset when PRBS configuration changes on the controller.

To view the PRBS PM parameters on the coherentDSP controller, use the following command:

show controllers coherentDSP | ODU4 *R/S/I/P* **pm** {**current | history** }{**15-min|24-hour**} **prbs**

The following tables describes the fields of PRBS PM parameters.

Table 54: PRBS PM Parameters

PM Parameter	Description
EBC	Cumulative count of PRBS bit errors in the sampling interval (15-minute or 24-hour). PRBS bit errors are accumulated only if PRBS signal is locked.
FOUND-COUNT	Number of state transitions from signal unlocked state to signal locked state in the sampling interval. If state change is not observed in the interval, the count is 0.
LOST-COUNT	Number of state transitions from signal locked state to signal unlocked state in the sampling interval. If state change is not observed in the interval, the count is 0.
FOUND-AT-TS	Latest timestamp when the PRBS state moves from unlocked state to locked state in the sampling interval. If state change is not observed in the interval, the value is null.
CONFIG-PTRN	Configured PRBS pattern on the port.

```
RP/0/RP0:ios#show controllers coherentDSP 0/0/0/1 pm current 15-min prbs
Mon Feb 13 00:58:48.327 UTC

PRBS in the current interval [00:45:00 - 00:58:48 Mon Feb 13 2019]
PRBS current bucket type : Valid
EBC                      : 40437528165
FOUND-COUNT              : 1 FOUND-AT-TS : 00:51:22 Mon Feb 13 2019
LOST-COUNT               : 1 LOST-AT-TS  : 00:52:52 Mon Feb 13 2019
CONFIG-PTRN              : PRBS_PATTERN_PN31
Last clearing of "show controllers OTU" counters never

RP/0/RP0:ios#show controllers ODU4 0/3/0/1/1 pm current 15-min prbs
Mon Jan 11 00:58:48.327 UTC

PRBS in the current interval [00:45:00 - 00:58:48 Mon Jan 11 2021]
PRBS current bucket type : Valid
EBC                      : 40437528165
FOUND-COUNT              : 1 FOUND-AT-TS : 00:51:22 Mon Jan 11 2021
LOST-COUNT               : 1 LOST-AT-TS  : 00:52:52 Mon Jan 11 2021
CONFIG-PTRN              : PRBS_PATTERN_PN7
Last clearing of "show controllers ODU" counters never
```

Configuring PRBS on OTN-XP Card

To configure PBRS mode on the ODU2e controller, you must configure Optical Channel Payload Unit (OPU) on the ODU2e controller followed by the PRBS mode and the pattern. The PRBS supported pattern on the OTN-XP card is invertedPN31.

From R7.3.1 onwards, you can configure PRBS on client or mapper ODU4 and ODU flex controllers.

For fiber channel controllers, PRBS is supported on mapper ODU flex controllers.

Note ODU2e PRBS is not supported for OTU2E client rates.

To configure PRBS mode on the ODU2e controller, enter the following commands:

configure

controller odu2e | oduflex *R/S/I/P/client-port/lane-number*

secondary-admin-state maintenance

opu

prbs mode {**source | sink | source-sink**} **pattern invertedpn31** {**direction {system | line}**}

end

commit

The following example shows how to configure PRBS mode as source-sink with pattern as invertedpn31:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller odu2e0/2/0/12/3/2
RP/0/RP0/CPU0:ios(config-odu2e)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-odu2e)#opu
RP/0/RP0/CPU0:ios(config-Opuk)#prbs mode source-sink pattern invertedpn31
RP/0/RP0/CPU0:ios(config-Opuk)#end
RP/0/RP0/CPU0:ios(config-odu2e)#commit
```

The following is a sample output of **show controller odu2e** command.

```
RP/0/RP0/CPU0(config-odu2e)#show controller odu2e 0/2/0/12/3/2 prbs-details
Mon Mar 14 21:33:02.293 UTC

 -------------------------PRBS details----------------------------------
PRBS Test                       : Enable
PRBS Mode                       : Source-Sink
PRBS Pattern                    : INVERTED PN31
PRBS Status                     : Locked
PRBS Lock Time(in seconds)      : 1190
PRBS Bit Errors                 : 0

-------------------------------------------------
```

The following example shows how to configure PRBS mode as source-sink with pattern as invertedpn31 with direction as system:

```
RP/0/RP0/CPU0:ios#configure
Wed Nov 11 00:38:11.789 UTC
RP/0/RP0/CPU0:ios(config)#controller odu4 0/2/0/5
RP/0/RP0/CPU0:ios(config-odu4)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-odu4)#opu prbs mode source-sink pattern invertedpn31 direction
system
RP/0/RP0/CPU0:ios(config-odu4)#commit
Wed Nov 11 00:38:26.391 UTC
```

The following example shows how to configure PRBS mode as source-sink with pattern as invertedpn31 with direction as line:

```
RP/0/RP0/CPU0:ios#configure
Wed Nov 11 00:38:11.789 UTC
RP/0/RP0/CPU0:ios(config)#controller odu4 0/2/0/5
RP/0/RP0/CPU0:ios(config-odu4)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-odu4)#opu prbs mode source-sink pattern invertedpn31 direction
line
RP/0/RP0/CPU0:ios(config-odu4)#commit
Wed Nov 11 00:38:26.391 UTC
```

The following example shows how to configure PRBS on the mapper controller:

```
RP/0/RP0/CPU0:ios#configure
Thu Oct 7 13:17:27.267 UTC
RP/0/RP0/CPU0:ios(config)#controller odu4 0/2/0/13/8
RP/0/RP0/CPU0:ios(config-odu4)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-odu4)#opu prbs mode source-sink pattern invertedpn31
RP/0/RP0/CPU0:ios(config-odu4)#commit
```

The following example shows how to configure PRBS on the mapper controller with PRBS mode as source-sink and pattern as invertedpn31 with direction as line:

```
RP/0/RP0/CPU0:ios(config)#controller odu4 0/3/0/7
RP/0/RP0/CPU0:ios(config-odu4)#secondary-admin-state maintenance
RP/0/RP0/CPU0:ios(config-odu4)#opu prbs mode source-sink pattern invertedpn31 direction
line
RP/0/RP0/CPU0:ios(config-odu4)#commit
Tue Oct 12 13:17:07.840 UTC
```

Verifying PRBS on OTN-XP Card

You can monitor the status of PRBS on the ODU2e controller using the following command:

show controllers odu2e *R/S/I/P/client-port/client-lane* **prbs-details**

The following example displays the output of the PRBS configuration with PRBS mode as sink:

```
RP/0/RP0/CPU0:ios#show controllers odu2e 0/2/0/12/3/2 prbs-details
-------------------------PRBS details----------------------------------
PRBS Test               : Enable
PRBS Mode               : Sink
PRBS Pattern            : INVERTED PN31
PRBS Status             : Locked
```

The following example displays the output of the PRBS configuration with PRBS mode as source-sink:

```
RP/0/RP0/CPU0:ios#show controllers odu2e 0/2/0/12/3/2 prbs-details
-------------------------PRBS details----------------------------------
PRBS Test               : Enable
PRBS Mode               : Source-Sink
PRBS Pattern            : INVERTED PN31
PRBS Status             : Locked
```

The following example displays the output of the PRBS configuration on the mapper controller:

```
RP/0/RP0/CPU0:ios#show controllers odu4 0/2/0/13/8 prbs-details
Thu Oct 7 13:21:19.444 UTC

-------------------------PRBS details----------------------------------
```

```
PRBS Test : Enable
PRBS Mode : Source-Sink
PRBS Pattern : INVERTED PN31
PRBS Status : Locked

-------------------------------------------------
```

The following example displays the output of the PRBS configuration on the mapper controller with PRBS mode as source-sink and pattern as invertedpn31 with direction as line:

```
RP/0/RP0/CPU0:ios#show controllers odu4 0/3/0/7 prbs-detailsTue Oct 12 13:17:22.748 UTC
Tue Oct 12 13:17:22.748 UTC
------------------------PRBS details----------------------------------
PRBS Test : Enable
PRBS Mode : Source-Sink
PRBS Pattern : INVERTED PN31
PRBS Status : Unlocked
PRBS Direction : Line
PRBS Bit Errors : 0

-------------------------------------------------
```

Clearing Bit Errors and Lock Time for PRBS

Lock time is the time that is elapsed since the last PRBS lock is detected.

The following sample shows that bit errors are observed during the PRBS test:

```
RP/0/RP0/CPU0:ios#show controllers odu4 0/2/0/5 prbs-details
Fri Nov 13 03:21:44.191 UTC

------------------------PRBS details----------------------------------
PRBS Test : Enable
PRBS Mode : Source-Sink
PRBS Pattern : INVERTED PN31
PRBS Status : Locked
PRBS Direction : Line
PRBS Lock Time(in seconds) : 28
PRBS Bit Errors : 23776

-------------------------------------------------
```

To clear the lock time and bit errors before the PRBS test, use the **clear** command:

```
RP/0/RP0/CPU0:ios#clear controller odu4 0/2/0/5 prbs-details
Fri Nov 13 03:21:50.726 UTC
PRBS bit errors cleared
```

The following sample displays the bit errors and lock time are removed.

```
RP/0/RP0/CPU0:ios#show controllers odu4 0/2/0/5 prbs-details
Fri Nov 14 03:21:44.191 UTC
------------------------PRBS details----------------------------------
PRBS Test : Enable
PRBS Mode : Source-Sink
PRBS Pattern : INVERTED PN31
PRBS Status : Locked
PRBS Direction : Line
PRBS Lock Time(in seconds) : 2
PRBS Bit Errors : 0
```

FlexO GID and IID

In the 4x100G-MXP-400G-TXP LC mode, the OTN-XP card uses flexible OTN (flexO) interfaces on trunk ports. These flexO interfaces provide a flexible and interoperable mechanism to transport OTUCn signals by grouping standard lower rate interfaces. Each flexO interface group is identified by a flexO group identification (GID) number, which ranges 1–1,048,576. Each member of a flexO group is identified by a flexO instance identification (IID) number, which ranges 1–254.

From Release 7.3.1 onwards, flexO GID and IID configurations are supported on CoherentDSP controller for the OTN-XP card.

From Release 7.10.1 onwards, flexO GID and IID configurations are supported on CoherentDSP controller for the QXP card.

Configuring FlexO GID and IID

To configure flexO GID and IID on the coherentDSP controller, enter the following commands:

configure

controller coherentDSP *R/S/I/P*

flexo

gid <*gid-no*> **iid** <*iid-no*>

commit

Note You must configure the iid number based on the trunk bandwidth. You must add 1, 2, 3, and 4 iid numbers for 100G, 200G, 300G, and 400G respectively.

The following sample shows how to configure flexO GID and IID on the CoherentDSP controller:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP0/2/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#flexo
RP/0/RP0/CPU0:ios(config-CoDSP)#gid 2 iid 5,6,7,8
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
```

Verifying FlexO GID and IID

The following sample shows the flexO GID and IID configuration on the CoherentDSP controller:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/12

Tue Jan 12 11:26:08.235 UTC
Port : CoherentDSP 0/2/0/12
Controller State : Up
Inherited Secondary State : Normal
Configured Secondary State : Normal
Derived State : In Service
Loopback mode : None
```

```
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s

Alarm Information:
LOS = 0 LOF = 3 LOM = 0
OOF = 3 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 1
FLEXO-MM = 1 FLEXO-LOM = 0 FLEXO-RDI = 3
FLEXO-LOF = 0
Detected Alarms : None

Bit Error Rate Information
PREFEC BER : 6.43E-04
POSTFEC BER : 0.00E+00

TTI :
Remote hostname : ios
Remote interface : CoherentDSP 0/0/0/13
Remote IP addr : 0.0.0.0

FEC mode : O_FEC

Flexo-Mode : Enable
Flexo Details:
Tx GID : 2
TX IID : 5,6,7,8,
Rx GID : 2
RX IID 5,6,7,8,

AINS Soak : None
AINS Timer : 0h, 0m
AINS remaining time : 0 seconds
```

Flexo Parameter Update on Inverse Muxponder Configuration on the OTN-XP Card

By default, the value of flexo parameters for the coherentDSP controllers in the 400G inverse muxponder, are as follows:

- CoherentDSP 0/0/0/12—GID is 1 and IID is 1, 2.
- CoherentDSP 0/0/0/13—GID is 1 IID is 3, 4.

The following example displays the default configurations on the transmission side:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/13
Wed Jun  9 23:17:30.794 UTC

Port                                          : CoherentDSP 0/0/0/13
Controller State                              : Admin Down
Inherited Secondary State                     : Normal
Configured Secondary State                    : Normal
Derived State                                 : Out Of Service
Loopback mode                                 : None
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 200.0Gb/s
```

```
Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 0.00 dB

Q-Margin                                        : 0.00dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 3, 4,
        Rx GID                                  : 0
        RX IID                                  : 0, 0,

Flexo Peers Information:
        Controller                              : CoherentDSP0_0_0_12
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

For 400G inverse muxponder, the flexo configuration on these coherent DSP controllers must be such that GID is the same on coherentDSP controllers on both port 12 and port 13, and IIDs are in the incremental order. In case if one of the node configurations is invalid, the Provisioning Failed alarm is raised on that particular controller. The Provisioning Failed alarm moves to the slice level in case you perform a line card reload.

The following sample configures the same IID on the coherentDSP 0/0/0/12 as that of coherentDSP 0/0/0/13 and shows the resulting Provisioning Failed alarm.

```
RP/0/RP0/CPU0:ios#configure
Wed Jun  9 23:19:28.101 UTC
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#flexo gid 1 iid 3,4
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Wed Jun  9 23:20:08.971 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#end
RP/0/RP0/CPU0:ios#show alarms brief system active
Wed Jun  9 23:20:11.940 UTC

------------------------------------------------------------------------------------
Active Alarms
------------------------------------------------------------------------------------
Location Severity Group     Set Time                Description
```

```
--------------------------------------------------------------------------------

0/0      Major  Controller 06/09/202123:20:10 UTC CoherentDSP0/0/0/12-Provisioning Failed
```

The following sample configures incremental IIDs in the coherentDSP 0/0/0/12 and coherent DSP 0/0/0/13 and verifies that the Provisioning Failed alarms cleared:

```
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/0/12
RP/0/RP0/CPU0:ios(config-CoDSP)#flexo gid 1 iid 5,6
RP/0/RP0/CPU0:ios(config-CoDSP)#controller coherentDSP 0/0/0/13
RP/0/RP0/CPU0:ios(config-CoDSP)#flexo gid 1 iid 7,8
RP/0/RP0/CPU0:ios(config-CoDSP)#commit
Wed Jun  9 23:21:06.335 UTC
RP/0/RP0/CPU0:ios(config-CoDSP)#end
```

The following sample verifies the IID configurations in the coherentDSP 0/0/0/12 and coherentDSP 0/0/0/13:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/12
Wed Jun  9 23:21:15.321 UTC

Port                                          : CoherentDSP 0/0/0/12
Controller State                              : Admin Down
Inherited Secondary State                     : Normal
Configured Secondary State                    : Normal
Derived State                                 : Out Of Service
Loopback mode                                 : None
BER Thresholds                                : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                        : Enable
Bandwidth                                     : 200.0Gb/s

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms                               : None

Bit Error Rate Information
PREFEC  BER                                   : 0.00E+00
POSTFEC BER                                   : 0.00E+00
Q-Factor                                      : 0.00 dB

Q-Margin                                      : 0.00dB

TTI :
        Remote IP addr                        : 0.0.0.0

FEC mode                                      : O_FEC

Flexo-Mode                                    : Enable
Flexo Details:
        Tx GID                                : 1
        TX IID                                : 5, 6,
        Rx GID                                : 0
        RX IID                                : 0, 0,
```

```
Flexo Peers Information:
        Controller                              : CoherentDSP0_0_0_13
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds


RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/13
Wed Jun  9 23:21:20.348 UTC

Port                                            : CoherentDSP 0/0/0/13
Controller State                                : Admin Down
Inherited Secondary State                       : Normal
Configured Secondary State                      : Normal
Derived State                                   : Out Of Service
Loopback mode                                   : None
BER Thresholds                                  : SF = 1.0E-5  SD = 1.0E-7
Performance Monitoring                          : Enable
Bandwidth                                       : 200.0Gb/s

Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0        SF_BER = 0
SD_BER = 0      BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0     FLEXO_GIDM = 0
FLEXO-MM = 0    FLEXO-LOM = 0   FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms                                 : None

Bit Error Rate Information
PREFEC  BER                                     : 0.00E+00
POSTFEC BER                                     : 0.00E+00
Q-Factor                                        : 0.00 dB

Q-Margin                                        : 0.00dB

TTI :
        Remote IP addr                          : 0.0.0.0

FEC mode                                        : O_FEC

Flexo-Mode                                      : Enable
Flexo Details:
        Tx GID                                  : 1
        TX IID                                  : 7, 8,
        Rx GID                                  : 0
        RX IID                                  : 0, 0,

Flexo Peers Information:
        Controller                              : CoherentDSP0_0_0_12
        OTUCn rate                              : OTUC2


AINS Soak                                       : None
AINS Timer                                      : 0h, 0m
AINS remaining time                             : 0 seconds
```

The following sample verifies that the Provisioning Failed alarm was cleared:

```
RP/0/RP0/CPU0:ios#show alarms brief system history
```

```
--------------------------------------------------------------------------------
History Alarms
--------------------------------------------------------------------------------
----------------------------------
Location Severity Group       Set Time                Description
       Clear Time




--------------------------------------------------------------------------------------------------------------
0/2      Major   Controller  09/30/2021 14:42:01 UTC  CoherentDSP0/2/0/12 - Provisioning
Failed 09/30/2021 14:42:25 UTC
```

FPD

FPD command enables you to verify the status of the installed QDD ZRP pluggables. The following is the sample output for verifying the FPD status of the installed QDD ZRP pluggables.

```
RP/0/RP0/CPU0:ios#sh hw-module fpd
Fri Jul 23 12:47:52.106 UTC

Auto-upgrade:Disabled
                                                                          FPD Versions
                                                                          ==============
Location   Card type              HWver FPD device       ATR Status     Running   Programd
--------------------------------------------------------------------------------------------
0/0        NCS1K4-OTN-XPL         3.0   LC_CPU_MOD_FW        CURRENT     21.27     21.27
0/0        NCS1K4-OTN-XPL         7.0   LC_DP_MOD_FW         CURRENT      3.10      3.10
0/0        NCS1K4-OTN-XPL         2.0   LC_QSFPDD_PORT_11    CURRENT    161.2009  161.2009
0/0        NCS1K4-OTN-XPL         2.0   LC_QSFPDD_PORT_9     CURRENT    161.2009  161.2009
```

In the above sample output, `LC_QSFPDD_PORT_11` and `LC_QSFPDD_PORT_9` indicate the provisioning of the QDD ZRP pluggables in the trunk ports 11 and 9. To resume traffic, the FPDs must be in `CURRENT` state. For more details on the FPD command, see Command Reference for Cisco NCS 1004.

Automatic Protection Switching (APS) on OTN XP Card

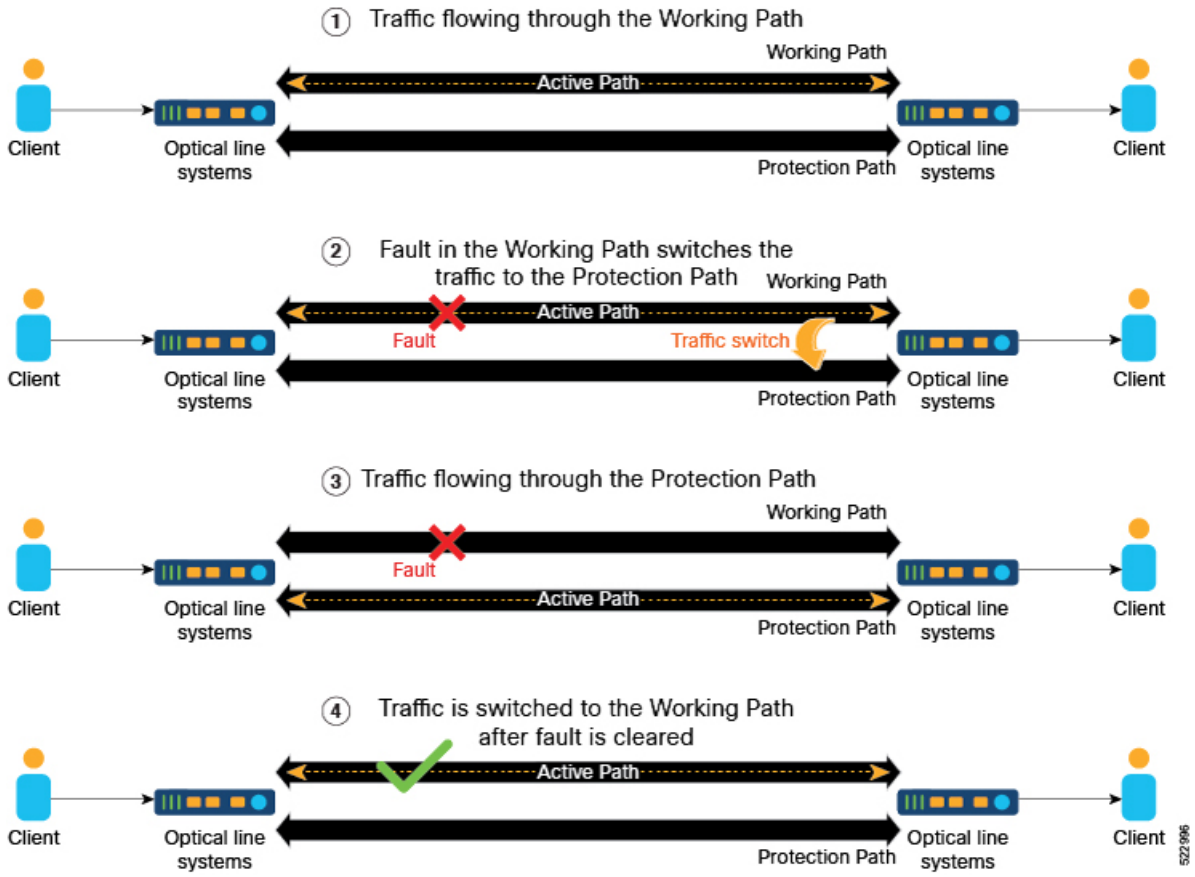
Table 55: Feature History

Feature Name	Release Information	Description
APS Support for 100G and 200G Trunk Bandwidth on OTN XP Cards	Cisco IOS XR Release 7.9.1	In addition to the 400G trunk bandwidth, you can now configure Automatic Protection Switching (APS) for the following trunk bandwidth combinations: • 200G DWDM with 2x100G clients • 100G QSFP28 Grey and 10x10G clients
Automatic Protection Switching (APS) on OTN XP Card	Cisco IOS XR Release 7.8.1	APS provides protection mechanism against optical fiber faults or signal failure. In case a failure is detected, live traffic is automatically moved from the working path to the protection path to prevent any data loss. Following trunk bandwidth combinations are supported: • 400G DWDM with 4x100G clients • 400G DWDM with 400G clients You can enable this feature using the **protected** keyword of the hw-module command.

APS allows you to configure protection switching on trunk ports of NCS1K4-OTN-XP Cards. Protection switching automatically switches traffic from one path to another working path if any signal failure occurs. This requires configuring working and protection paths on trunk ports where the protection path works as backup for the working path. Usually, the working path is the active path and carries traffic. The traffic immediately switches to the protection path if a signal failure occurs on the working path.

The following figure explains the working of APS where the traffic is immediately switched from working path to the protection path in case of a signal failure.

Figure 4: Automatic Protection Switching



If an electrical(TP1) or optical(TP3) signal is interrupted or disturbed at the client pluggable and the CDR inside the QDD loses the Rx lock, then the QDD pluggable requires 1.5 to 2 seconds to recover from the Rx loss.

You can employ both the GL-1 and GL-2 pluggable trunk protection features. The GL-2 pluggable supports both Flexcoh and Termination modes. The GL-2 Termination mode offers a shorter switching time compared to the Flexcoh mode, and supports both GL-2 and GL-1 pluggables.

The Pluggables Switching Time table lists the 100G and 400G client pluggables that either support or do not support switching from the active path to the protection path within 50 milliseconds.

Table 56: Pluggables Switching Time

Pluggables Supporting Switching within 50 ms		**Pluggables not Supporting Switching within 50 ms**	
100G	**400G**	**100GE**	**400GE**
ONS-QSFP28-LR4	-	QSFP-100G-FR-S	QDD-400G-DR4-S
QSFP-100G-LR-S	-	-	QDD-400G-FR4-S
QSFP-100G-SR4-S	-	-	-
QSFP-100G-DR-S	-	-	-

Pluggables Supporting Switching within 50 ms		Pluggables not Supporting Switching within 50 ms	
QSFP-100G-CWDM4-S	-	-	-

Note According to the QDD pluggable standard, if an electrical (TP1) or optical (TP3) signal is interrupted or disturbed at a client pluggable, and the Clock and Data Recovery (CDR) inside the QDD loses the lock, the QDD pluggable requires some time to recover the signal. Hence, when a fault is identified on the trunk port, the client port Rx loses the lock. The recovery takes between 1.5 to 2 s depending on the pluggable standard.

To configure APS on trunk ports, perform these tasks:

- Enable APS on Trunk
- Define the Working and Protecting Resources in an ODU Group Controller
- Configure Protection Attributes of an ODU Group Controller

Enable APS on Trunk

Before configuring APS, it must be enabled on the trunk ports of the NCS1K4-OTN-XP cards. Trunk protection on the NCS1K4-OTN-XP card supports 400G, 200G, and 100G bandwidth. For more details, see Client and Trunk Port Mapping on NCS1K4-OTN-XP Cards, on page 241.

To enable protection switching and configure the client and trunk bandwidth for DWDM, use the following commands:

hw-module location *location*

mxponder

protected

trunk-rate *trunk rate*

client-port-rate [*0* | *4* | *5* | *8*] **client-type** [*100GE* | *400GE*]

commit

To enable protection switching and configure the client and trunk bandwidth for 10G-Grey-MXP, use the following commands:

hw-module location *location*

mxponder

protected

trunk-rate *trunk rate*

client-port-rate [*4* | *5* | *2* | *8*] **lane** *lane number* **client-type** [*10GE* | *otu2* | *otu2e*]

commit

For more information about these commands, see Command Reference for Cisco NCS 1004.

Example 1

The following example shows how to configure protection datapath on slot 0 of the NCS1K4-OTN-XP card. This example also shows how you can configure client port at *400GE* to achieve a total bandwidth of 400G at the trunk port.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0
RP/0/RP0/CPU0:ios(config-hwmod)# mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  protected
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#  client-port-rate 8 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Example 2

The following example shows how to configure protection datapath on slot 2 of the NCS1K4-OTN-XP card. This example also shows how you can configure client ports *0*, *4*, *5* and *8* with a bandwidth of *100GE* each to achieve a total bandwidth of *400G* at the trunk port.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#protected
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 0 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Example 3

The following example shows how to configure protection datapath on slot 2 of the NCS1K4-OTN-XP card. This example also shows how you can configure client ports *5* and *8* with a bandwidth of *100GE* or *otu4* each to achieve a total bandwidth of *200G* at the trunk port.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#protected
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 200G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 client-type 100GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 8 client-type otu4
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Example 4

The following example shows how to configure protection datapath on slot 2 of the NCS1K4-OTN-XP card. This example also shows how you can configure client port *2* of *lane 3* and *lane 4*. This examples also shows how to configure client port *4* and *5* of *lane 1*, *lane 2*, *lane 3*, and *lane 4* with a bandwidth of *10GE* each to achieve a total bandwidth of *100G* at the trunk port.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#protected
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 100G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type 10GE
```

```
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Example 5

The following example shows how to configure protection datapath on slot 2 of the NCS1K4-OTN-XP card. This example shows how you can configure different client ports with mixed bandwidth on different lanes.

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/2
RP/0/RP0/CPU0:ios(config-hwmod)#mxponder
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#protected
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 100G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 1 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 2 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 4 lane 4 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 1 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 2 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 3 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 5 lane 4 client-type OTU2
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 3 client-type OTU2E
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 2 lane 4 client-type 10GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Client and Trunk Port Mapping on NCS1K4-OTN-XP Cards

The following table displays the port mapping that can be used to map client and trunk port bandwidth:

Table 57: Client/Trunk Port Mapping for NCS1K4-OTN-XP Cards

Trunk Bandwidth	Trunk Ports	Client Bandwidth	Client Ports (Slice 1)
400G	12, 13	400GE	8
400G	12, 13	4x100G	0, 4, 5 and 8
200G	12, 13	2x100G	5 and 8
100G	0, 1	10x10G	• Client Port 4 - Lanes 1, 2, 3, and 4 • Client Port 5 - Lanes 1, 2, 3, and 4 • Client Port 2 - Lanes 3 and 4

Define the Working and Protecting Resources in an ODU Group Controller

Before configuring the protection attributes, you must create an ODU group controller and configure the ports 12 or 13 as the working and/or protection paths.

Use the following commands to first create an ODU group controller and define the working and protection paths inside the group:

controller odu-group-mp *Group-ID* **signal** *client-signal-type* **odu-type** *type-of-the-odu*

protecting-controller [*ODUk Rack/Slot/Instance/Port*]

working-controller [*ODUk Rack/Slot/Instance/Port*]

Example

The following example shows how to create an ODU group *MP 2* with ODU type *ODUC4*. This example also shows how to configure the port numbers 12 and 13 as working and protection paths respectively.

```
RP/0/RP0/CPU0:ios(config)# controller Odu-Group-Mp 2 signal Otn odu-type ODUC4
RP/0/RP0/CPU0:ios(config)# protecting-controller ODUC4 0/2/0/13
RP/0/RP0/CPU0:ios(config)# working-controller ODUC4 0/2/0/12
```

Note If both APS and L1 Encryption are configured on the same ODU group controller, configure the GCC2 interface and the corresponding IP addresses for each trunk port separately. This ensures that any service impact on a trunk port does not affect the encryption functionality and also the independent working and protected paths for 1+1 trunk protection are maintained. For more details about configuring GCC interface, see Configuring the GCC Interface

Configure Protection Attributes of an ODU Group Controller

To configure the recommended protection attributes of an ODU group controller, use the following commands:

protection-attributes timers [**hold-off-time**] *timer*

protection-attributes protection-type [*APSbidi*]

protection-attributes protection-mode [**revertive wait-to-restore-time**] *timer*

protection-attributes connection mode [*snc-n*]

commit

Note

- *hold-off-time* is an optional parameter. If this parameter is not specified, the default parameter value is used.
- The recommended parameters for configuring protection attributes of an ODU Group Controller are *snc-n*, *APSbidi*, and *wait-to-restore-time*.

Example

The following example shows how to configure protection attributes **protection type**, **protection mode**, **connection mode**, and **timers** on the ODU group **MP 2**.

```
 RP/0/RP0/CPU0:ios(config)# protection-attributes timers hold-off-time 1000
 RP/0/RP0/CPU0:ios(config)# protection-attributes protection-type APSbidi
 RP/0/RP0/CPU0:ios(config)# protection-attributes protection-mode revertive
wait-to-restore-time 400
 RP/0/RP0/CPU0:ios(config)# protection-attributes connection-mode snc-n
```

For more details about these commands, see Command Reference for Cisco NCS 1004.

Manual Protection Switching

Protection switching is usually triggered automatically when a signal failure is detected. However, you can trigger a switch manually as well using the protection switching commands. The following table describes these switching commands and their priority levels that can be used to triggered a switch manually:

Table 58: Manual Protection Switching Commands Priority Levels

Priority	Priority Request	Description
1	lockout	Use this command to lockout the working path and prevent switching to the protection path. When lockout is configured, traffic is not switched to the protection path even if a failure is detected. This command has the highest priority level and overrides all the other protection switching commands. This means that if lockout is configured, you cannot trigger **force** and **manual** commands.
2	force	Use this command to trigger a force-switch from the working path to the protection path and vice-versa. This command cannot be triggered if **lockout** is configured.
3	manual	Use this to manually switch from the working path to the protection path or vice-versa in case a maintenance is scheduled on any of the paths. This command cannot be triggered if **lockout** and **force** is configured.

Perform a Lockout

If there is a signal failure on the protection path, you can prevent the switching of traffic from the working path to the protection path by locking out the working path. Lockout command has the highest priority and overrides all other switching commands.

To perform a lockout, use these commands:

odu-group-mp *Group-ID* **signal** *client-signal-type* **odu-type** *type-of-the-odu*

protection-switching operate lockout odu-dest [*ODUk Rack/Slot/Instance/Port*]

commit

Note The *ODUk Rack/Slot/Instance/Port* must be the working path.

Example

The following example shows how to configure lockout on an *ODUC4* on the working path *0/2/0/13*:

```
RP/0/RP0/CPU0:ios(config)#Odu-Group-Mp2 signal Otn odu-type ODUC4
RP/0/RP0/CPU0:ios(config-Odu-Group-Mp2)#protection-switching operate lockout odu-dest ODUC4
 0/2/0/13
RP/0/RP0/CPU0:ios(config-Odu-Group-Mp2)#commit
```

Perform a Manual Switch

If there are changes to be done during a scheduled maintenance window on the working or protection paths, you can perform a manual switch from the working path to the protection path or the opposite way. This command is overridden by the **force** and **lockout** commands.

To perform a manual switch, use the following commands:

odu-group-mp *Group-ID* **manual odu-dest** *ODUk Rack/Slot/Instance/Port*

Note The **manual** switch command works irrespective of the working or protection path *Rack/Slot/Instance/Port* and uses only the *Group-ID* to switch the traffic.

Example

The following example shows how to manually switch traffic:

```
RP/0/RP0/CPU0: odu-group mp 1 manual odu-dest ODUC4 0/2/0/13
```

Perform a Forced Switch

You can also perform a forced switch of traffic from the working path to the protection path and also, conversely, using the following commands:

odu-group-mp *Group-ID* **forced odu-dest** *ODUk Rack/Slot/Instance/Port*

Note The **forced** switch command works irrespective of the working or protection path *Rack/Slot/Instance/Port* and uses only the *Group-ID* to switch the traffic.

Example

The following example shows how to force switch the traffic:

```
RP/0/RP0/CPU0: odu-group mp 1 forced odu-dest ODUC4 0/2/0/13
```

Verify the APS Details on ODU Group Controller

You can verify the APS configuration details on ODU group controller such as, ODU group name, working path, protection path, and protection parameters, using the following command:

show controllers odu-group-mp *Group-ID*

Example

The following example shows how to verify the APS configuration details of the ODU group **MP 2**:

```
RP/0/RP0/CPU0:ios# show controllers odu-group-mp 2

ODU  Group Information
--------------------------------------------------------------
ODU GROUP ID                                 :  2
Controller State                             :  Up

 WORKING CONTROLLER

ODU NAME                                     :  ODUC4 0/0/0/12
ODU ROLE                                     :  WORKING
ODU STATE                                    :  Active_tx
Local Failure                                :  Yes
Remote Failure                               :  Yes

 PROTECTED CONTROLLER

ODU NAME                                     :  ODUC4 0/0/0/13
ODU ROLE                                     :  PROTECT
ODU STATE                                    :  Active
Local Failure                                :  No
Remote Failure                               :  No

PROTECTION PARAMETERS :
Connection Mode                              :  SNC_N
Protection Type                              :  1+1 Bidirectional Protection
Tcmid                                        :  0
Protection Mode                              :  Revertive
Hold off timer                               :  1000
Wait-to-restore timer                        :  400000 ms


Detected Alarms                              :  Switched To Protection
```

View the ODU Group Controller Hardware Details

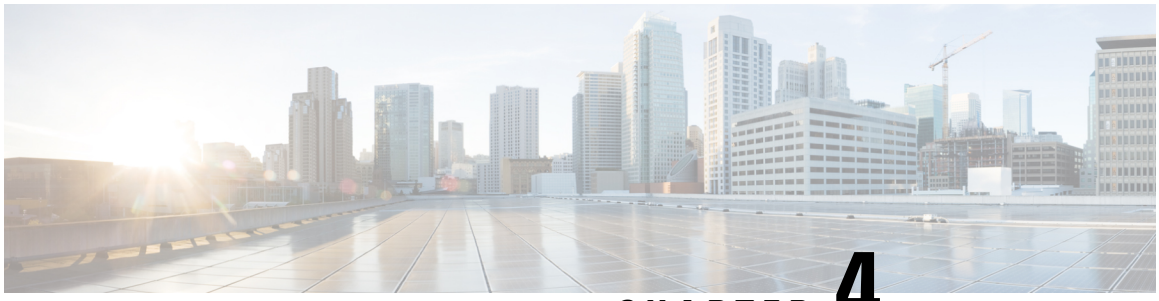
To verify the hardware details of the ODU group controller, use these commands:

show controllers odu-group-mp *Group-ID* **protection-detail**

Example

The following example shows how to view the ODU group controller *2* hardware details:

```
RP/0/RP0/CPU0:ios#show  controllers odu-group-mp 2 protection-detail

ODU  Group Information
--------------------------------------------------------------
LOCAL
                Request State                   : Signal Failed
                Request signal                  : 0
                Bridge signal                   : 1
                Bridge Status                   : 1+1
REMOTE
                Request State                   : Signal Failed
                Request signal                  : 0
                Bridge signal                   : 1
                Bridge Status                   : 1+1
WORKING
                Controller Name                 : ODUC40_0_0_12
                ODU STATE                       : Active_tx
                Local Failure                   : Signal Failure
                Remote Failure                  : Signal Failure
                WTR Left                        : 0 ms

PROTECT
                Controller Name                 : ODUC40_0_0_13
                ODU STATE                       : Active
                Local Failure                   : State Ok
                Remote Failure                  : State Ok
                WTR Left                        : 0 ms

Client
                Controller Name                 : ODUC40_0_0_0
                ODU STATE                       : Not Present


Wait to restore                                 : 400000 ms
Hold-off-timer                                  : 1000 ms
Current State                                   : Signal failed on Working
Previous State                                  : No Request State
```



CHAPTER 4

Performance Monitoring

- Performance monitoring parameters, on page 247
- Clear PM parameters, on page 262

Performance monitoring parameters

A performance monitoring parameter is a network data metric that

- allows service providers to gather, store, set thresholds for, and report network performance statistics,
- supports configurable data collection intervals, such as 30-second, 15-minute, or 24-hour periods, and
- simplifies troubleshooting by enhancing the quality of data collected directly from network equipment.

Service providers can configure and retrieve performance monitoring (PM) counters for various controllers, using intervals that suit operational needs. These parameters provide early detection of network issues, making it easier to maintain the network health and resolve problems quickly.

PM history persistence features

A PM history persistence feature is a performance monitoring capability that

- retains historical performance monitoring data for Optics, Ethernet, and coherent DSP controllers even after a line card reload, power cycle, or chassis upgrade,
- supports multiple time intervals, including 30-second, 15-minute, and 24-hour buckets, and
- provides access to historical PM data across software releases, except on unsupported

From Release 7.7.1, PM history parameters for Optics, Ethernet, and coherent DSP controllers are retained even after a line card cold reload, line card warm reload, XR reload, Calvados reload, RP reload, Hw-module all reload, power cycle, or upgrade of the NCS 1004 chassis.

After upgrading to a new software release, you can view historical PM data collected in the previous release. However, if new PM parameters are introduced in the upgrade, you may see an error when fetching the data, such as:

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/0/0/8 pm history 15-min ether 5
Tue Apr  5 22:05:56.750 UTC
pm_display_int_15min_ether_index: bag_decode failed ('bag' detected the 'fatal' condition
'An irresolvable version conflict prevented the specified bag from being decoded')
```

The time required to fill historical buckets is:

- 30-second bucket: 15 minutes (30 buckets)
- 15-minute bucket: 8 hours (32 buckets)
- 24-hour bucket: 24 hours (1 bucket)

PM counters are continuously updated in the current bucket for each interval. At the end of the timer, current data is rolled over into a historical bucket.

If a controller is deleted or removed, PM data is preserved for 3 hours. If the controller isn't restored in that time, the data is cleared.

Limitations

- PM history persistence is not supported on NCS1K4-QXP-K9 systems.
- If a reload occurs during the rollover process on NCS 1004, one of the following can happen:
  - The PM bucket may be missing and the next bucket marked as Invalid.
  - A bucket expiry message appears, for example:

    ```
    History data is empty. Verify at least one collection period is expired
    ```

  - A bucket interval may be marked Invalid, and counters could be zero or nonzero.

Instantaneous Q-margins

A Q-margin is a performance monitoring metric that

- measures signal quality in optical transmission systems,
- indicates the margin between observed signal quality and the threshold below which errors increase, and
- allows monitoring and troubleshooting of system stability for coherentDSP controllers on supported cards.

From Release 7.3.1 onwards, instantaneous Q-margin is supported for performance monitoring (PM) parameters on coherentDSP controllers for 1.2T and 1.2TL cards. For more information, see the supporting concept documentation.

In certain scenarios, some initial PM buckets can be displayed as valid, although the instantaneous Q-margin values in those buckets are shown as invalid. PM measurements are typically taken over 30 seconds, 15 minutes, and 24 hours intervals.

The following scenarios can cause initial PM buckets to show valid readings, despite instantaneous Q-margin values being invalid:

- Shutdown or no shutdown on optics
- Bit per second (BPS) change on optics
- Trunk rate change
- Fiber cut

To handle such cases, avoid relying on the initial PM buckets when monitoring instantaneous Q-margin values in these scenarios.

Sample output:

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/0 pm history flex-bin fec 1
Mon Sep 14 06:16:03.249 UTC

g709 FEC in interval 1 [06:15:50 - 06:16:00 Mon Sep 14 2020]

Flexible bin interval size: 10 seconds
FEC history bucket type : Invalid. ------- > Instantaneous Q_margin is invalid in this
bucket
 EC-BITS   : 38054                    UC-WORDS  : 0

                                  MIN                       AVG                       MAX
PreFEC BER                :    0E-15                     3.26E-08                  1.43E-07
PostFEC BER               :    0E-15                      0E-15                     0E-15
Q                         :    0.00                       5.73                      14.40
Q_margin                  :    -5.00                      -0.69                     9.40
Instantaneous Q_margin    :    -21474836.48               -8589934.59               0.00
```

At later intervals, PM buckets may be valid but instantaneous Q-margin values remain invalid, demonstrating why those initial readings should be disregarded.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/2/0/0 pm history 30-sec fec 1
Mon Sep 14 06:16:53.490 UTC

g709 FEC in interval 1 [06:16:00 - 06:16:30 Mon Sep 14 2020]

FEC history bucket type : Valid ------- > (Instantaneous Q_margin is invalid but the PM
bucket is valid. So these initial bins can be ignored)
 EC-BITS   : 431887                   UC-WORDS  : 0

                                  MIN                       AVG                       MAX
PreFEC BER                :    3.97E-09                  4.83E-08                  1.51E-07
PostFEC BER               :    0E-15                      0E-15                     0E-15
Q                         :    14.40                      14.48                     14.60
Q_margin                  :    9.30                       9.46                      9.60
Instantaneous Q_margin    :    -21474836.48               -5010784.19               14.42
```

Configure PM parameters

This task describes the steps to configure performance monitoring parameters, which are critical for monitoring device operation and analyzing the health of Optics, Ethernet, and coherent DSP controllers.

-

You can configure and view the performance monitoring parameters for the Optics, Ethernet, and coherent DSP controllers.

Table 59: Feature History

Feature Name	Release Information	Feature Description
PM History Persistence	Cisco IOS XR Release 7.7.1	PM history parameters for Optics, Ethernet, and coherent DSP controllers are now retained even after operation disruptive events like: • Various reload procedures • Power cycle • Operating system upgrade of the NCS 1004 chassis This functionality maintains prolonged access to performance history that is useful for device health monitoring.

Note Current and History PM counters do not support flex and 30 second bucket types for OC192 and STM64 controllers.

Follow these steps to configure PM parameters on Optics, Ethernet, or coherent DSP controllers using CLI commands.

Procedure

Step 1 Configure the controller using the command **configure controller** *controllertype R/S/I/P*

Step 2 Configure the pm parameters using the command **pm** { **15-min** | **30-sec** | **24-hour** } { **optics** | **ether** | **pcs** | **fec** | **otn** | **ocn** | **stm** } { **report** | **threshold** }

Example:

This is a sample in which the performance monitoring parameters of the Optics controller are configured at 24-hour intervals.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/1/5 pm 24-hour optics threshold osnr max 345
RP/0/RP0/CPU0:ios(config)#commit
```

Example:

This is a sample in which the performance monitoring parameters of the Ethernet controller are configured at 15-minute intervals.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller HundredGigECtrlr 0/3/0/0 pm 15-min pcs report bip enable
RP/0/RP0/CPU0:ios(config)#commit
```

Example:

This is a sample in which performance monitoring parameters of a Coherent DSP controller are configured 30-second intervals.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller coherentDSP 0/0/1/1 pm 30-sec fec threshold post-fec-ber max
OE-15
RP/0/RP0/CPU0:ios(config)#commit
```

Step 3 Use the command **performance-monitor-engine flex-bin interval** *duration*, to configure PM with flexible bin interval for a particular duraion.

Example:

This is a sample to configure PM with flexible bin interval as 1 second.

```
RP/0/RP0/CPU0:ios#configure
Thu Sep 30 10:38:39.729 UTC
RP/0/RP0/CPU0:ios(config)#performance-monitor-engine flex-bin interval 1
RP/0/RP0/CPU0:ios(config)#commit
Thu Sep 30 10:38:57.987 UTC
RP/0/RP0/CPU0:ios(config)#
```

Upon successful configuration, the specified controllers will begin collecting and retaining PM data according to the set parameters. You will be able to view historical and current performance metrics for troubleshooting and ongoing system health monitoring.

What to do next

Verify PM configuration and review collected PM data using show commands.

EC-BITS threshold values for NCS1K4-1.2T-K9 line card

Select an EC-BITS threshold based on the line rate, FEC type, performance monitoring interval, and the required Q-margin.

EC-BITS and thresholds

Bit errors corrected (BIEC) represent the number of bit errors that forward error correction (FEC) fixes during a performance monitoring interval. EC-BITS thresholds depend on the line rate and FEC type. The Q-margin determines the timing of threshold crossing alerts (TCAs).

- The 1 dBQ thresholds are conservative and raise alerts well before post-FEC errors occur.
- The 0.5 dBQ thresholds are more aggressive. These thresholds raise TCAs in the error-free region, but closer to the BER threshold that, when crossed, generates post-FEC errors.

This table provides the EC-BITS thresholds for the NCS1K4-1.2T-K9 line card.

Table 60: EC-BITS thresholds by traffic mode and Q-margin

Traffic mode	1 dBQ margin			0.5 dBQ margin		
	30-second threshold	15-minute threshold	24-hour threshold	30-second threshold	15-minute threshold	24-hour threshold
100G QPSK, 15% FEC	4.462727E+10	1.338818E+12	1.285265E+14	6.218553E+10	1.865566E+12	1.790943E+14
100G QPSK, 27% FEC	8.942437E+10	2.682731E+12	2.575422E+14	1.156877E+11	3.470632E+12	3.331807E+14
200G QPSK, 15% FEC	8.925453E+10	2.677636E+12	2.570530E+14	1.243711E+11	3.731132E+12	3.581887E+14
200G QPSK, 27% FEC	1.788487E+11	5.365462E+12	5.150843E+14	2.313755E+11	6.941264E+12	6.663614E+14
300G 8QAM, 15% FEC	1.338818E+11	4.016454E+12	3.855796E+14	1.865566E+11	5.596698E+12	5.372830E+14
300G 8QAM, 27% FEC	2.682731E+11	8.048193E+12	7.726265E+14	3.470632E+11	1.041190E+13	9.995421E+14
400G 16QAM, 15% FEC	1.785091E+11	5.355272E+12	5.141061E+14	2.487421E+11	7.462264E+12	7.163773E+14
400G 16QAM, 27% FEC	3.576975E+11	1.073092E+13	1.030169E+15	4.627510E+11	1.388253E+13	1.332723E+15
500G 32QAM, 15% FEC	2.231363E+11	6.694090E+12	6.426326E+14	3.109277E+11	9.327830E+12	8.954717E+14
500G 32QAM, 27% FEC	4.471218E+11	1.341365E+13	1.287711E+15	5.784387E+11	1.735316E+13	1.665903E+15
600G hybrid, 15% FEC	2.677636E+11	8.032908E+12	7.711591E+14	3.731132E+11	1.119340E+13	1.074566E+15

View PM parameters

Use this task to view performance monitoring (PM) parameters for Optics, Ethernet, and Coherent DSP controllers. This allows you to assess the operational status, diagnostics, and statistics of controller components for maintenance and troubleshooting.

Performance monitoring parameters provide insight into the health and performance of controller interfaces, including metrics such as optical power, frequency offset, and signal quality. Viewing these parameters helps network operators verify ongoing performance, detect anomalies, and ensure compliance with operational thresholds. Perform this task regularly to support proactive maintenance and rapid fault resolution.

Procedure

Use the command **show controllers** *controllertype R/S/I/P* **{ pm { current | history } { 30 sec | 15-min | 24-hour } { optics | ether | fec | otn | prbs}** *linenumber* }, to view the performance monitoring parameters for Optics, Ethernet, and Coherent DSP controllers.

Example:

This sample displays the current performance monitoring parameters of the Optics controller at 15-minute intervals. Client optics have four lanes.

```
RP/0/RP0/CPU0:ios#show controller optics 0/1/0/3 pm current 15-min optics 3

Sat Feb 9 19:33:42.480 UTC

Optics in the current interval [19:30:00 - 19:33:42 Sat Feb 9 2019]

Optics current bucket type : Valid
             MIN       AVG       MAX      Operational      Configured      TCA   Operational
Configured     TCA
                                          Threshold(min)   Threshold(min)  (min) Threshold(max)
Threshold(max) (max)
LBC[% ]      : 0.0       0.0       0.0      0.0              NA              NO   100.0
 NA              NO
OPT[dBm]     : -40.00    -40.00    -40.00   -30.00           NA              NO   63.32
 NA              NO
OPR[dBm]     : -40.00    -40.00    -40.00   -30.00           NA              NO   63.32
 NA              NO
FREQ_OFF[Mhz]: 0         0         0        0                NA              NO   0
 NA              NO
```

This sample displays the current performance monitoring parameters of the Optics controller 15-minute intervals. Trunk optics have one lane.

```
RP/0/RP0/CPU0:ios#show controller optics 0/2/0/1  pm  current 15-min optics 1


Sat Feb  9 11:19:15.234 UTC

Optics in the current interval [11:15:00 - 11:19:15 Sat Feb 9 2019]

Optics current bucket type : Valid
             MIN       AVG       MAX      Operational      Configured      TCA   Operational
Configured     TCA
                                          Threshold(min)   Threshold(min)  (min) Threshold(max)
Threshold(max) (max)
LBC[% ]      : 0.0       0.0       0.0      0.0              NA              NO   100.0
 NA              NO

OPT[dBm]     : -1.51     -1.49     -1.48    -30.00           NA              NO   63.32
 NA              NO

OPR[dBm]     : -9.11     -9.07     -9.03    -30.00           NA              NO   63.32
 NA              NO

CD[ps/nm]    : 13        15        18       -180000          NA              NO   180000
 NA              NO

DGD[ps ]     : 2.00      2.33      3.00     0.01             NA              NO   21474836.46
 NA              NO

SOPMD[ps^2]  : 5.00      33.02     79.00    0.01             NA              NO   21474836.46
```

```
 NA              NO

OSNR[dB]      : 31.50     31.97     32.50     0.01             NA              NO   21474836.46
 NA              NO

PDL[dB]       : 0.20      0.34      0.50      0.01             NA              NO   21474836.46
 NA              NO

PCR[rad/s]    : 0.00      19.92     93.00     0.01             NA              NO   21474836.46
 NA              NO

RX_SIG[dBm]   : -9.05     -9.02     -8.99     -30.00           NA              NO   63.32
 NA              NO

FREQ_OFF[Mhz]: -302       -178      -74       -1500            NA              NO   1500
 NA              NO
```

Performance monitoring parameter outputs for Ethernet, FEC, optics, and Fibre Channel controllers

This reference presents sample outputs from performance monitoring commands for multiple controller types and time intervals. It helps you understand what data is collected and how counters are displayed in operational environments.

This sample displays the current performance monitoring parameters of the Ethernet controller 15-minute intervals.

```
RP/0/RP0/CPU0:ios#show controller HundredGigECtrlr 0/1/0/2 pm current 15-min ether
Fri Aug 30 00:37:53.527 UTC

ETHER in the current interval [00:30:00 - 00:37:53 Fri Aug 30 2019]

ETHER current bucket type : Valid
RX-UTIL[%]                : 100.00              Threshold : 0.00         TCA(enable) : NO
TX-UTIL[%]                : 10.00               Threshold : 0.00         TCA(enable) : NO
RX-PKT                    : 3852414442          Threshold : 0            TCA(enable) : NO
STAT-PKT                  : 0                   Threshold : 0            TCA(enable) : NO
OCTET-STAT                : 5847965122956       Threshold : 0            TCA(enable) : NO
OVERSIZE-PKT              : 0                   Threshold : 0            TCA(enable) : NO
FCS-ERR                   : 0                   Threshold : 0            TCA(enable) : NO
LONG-FRAME                : 0                   Threshold : 0            TCA(enable) : NO
JABBER-STATS              : 0                   Threshold : 0            TCA(enable) : NO
64-OCTET                  : 0                   Threshold : 0            TCA(enable) : NO
65-127-OCTET              : 0                   Threshold : 0            TCA(enable) : NO
128-255-OCTET             : 0                   Threshold : 0            TCA(enable) : NO
256-511-OCTET             : 0                   Threshold : 0            TCA(enable) : NO
512-1023-OCTET            : 0                   Threshold : 0            TCA(enable) : NO
1024-1518-OCTET           : 0                   Threshold : 0            TCA(enable) : NO
IN-UCAST                  : 0                   Threshold : 0            TCA(enable) : NO
IN-MCAST                  : 0                   Threshold : 0            TCA(enable) : NO
IN-BCAST                  : 0                   Threshold : 0            TCA(enable) : NO
OUT-UCAST                 : 0                   Threshold : 0            TCA(enable) : NO
OUT-BCAST                 : 0                   Threshold : 0            TCA(enable) : NO
OUT-MCAST                 : 0                   Threshold : 0            TCA(enable) : NO
TX-PKT                    : 7053588067          Threshold : 0            TCA(enable) : NO
OUT-OCTET                 : 451429636288        Threshold : 0            TCA(enable) : NO
IFIN-ERRORS               : 0                   Threshold : 0            TCA(enable) : NO
IFIN-OCTETS               : 0                   Threshold : 0            TCA(enable) : NO
STAT-MULTICAST-PKT        : 0                   Threshold : 0            TCA(enable) : NO
```

```
STAT-BROADCAST-PKT        : 0                  Threshold : 0            TCA(enable) : NO
STAT-UNDERSIZED-PKT       : 0                  Threshold : 0            TCA(enable) : NO
IN_GOOD_BYTES             : 5847965122956      Threshold : 0            TCA(enable) : NO
IN_GOOD_PKTS              : 3852414442         Threshold : 0            TCA(enable) : NO
IN_DROP_OTHER             : 0                  Threshold : 0            TCA(enable) : NO
OUT_GOOD_BYTES            : 451429636288       Threshold : 0            TCA(enable) : NO
OUT_GOOD_PKTS             : 7053588067         Threshold : 0            TCA(enable) : NO
IN_PKT_64_OCTET           : 0                  Threshold : 0            TCA(enable) : NO
IN_PKTS_65_127_OCTETS     : 0                  Threshold : 0            TCA(enable) : NO
IN_PKTS_128_255_OCTETS    : 0                  Threshold : 0            TCA(enable) : NO
IN_PKTS_256_511_OCTETS    : 0                  Threshold : 0            TCA(enable) : NO
IN_PKTS_512_1023_OCTETS   : 0                  Threshold : 0            TCA(enable) : NO
IN_PKTS_1024_1518_OCTETS  : 3852414442         Threshold : 0            TCA(enable) : NO
OUT_PKT_64_OCTET          : 7053588067         Threshold : 0            TCA(enable) : NO
OUT_PKTS_65_127_OCTETS    : 0                  Threshold : 0            TCA(enable) : NO
OUT_PKTS_128_255_OCTETS   : 0                  Threshold : 0            TCA(enable) : NO
OUT_PKTS_256_511_OCTETS   : 0                  Threshold : 0            TCA(enable) : NO
OUT_PKTS_512_1023_OCTETS  : 0                  Threshold : 0            TCA(enable) : NO
OUT_PKTS_1024_1518_OCTETS : 0                  Threshold : 0            TCA(enable) : NO
TX_UNDERSIZED_PKT         : 0                  Threshold : 0            TCA(enable) : NO
TX_OVERSIZED_PKT          : 0                  Threshold : 0            TCA(enable) : NO
TX_JABBER                 : 0                  Threshold : 0            TCA(enable) : NO
TX_BAD_FCS                : 0                  Threshold : 0            TCA(enable) : NO
```

Note Performance monitoring statistics are not supported for IN-UCAST and OUT-UCAST counters for Ethernet clients.

Note If you set the LC mode on the OTN-XP card to 4x100G-MXP-400G-TXP-LC, the performance monitoring parameters for the 400GE controllers (fourHundredGigECtrlr) are unsupported in the card.

This sample displays the current FEC performance monitoring parameters of the Coherent DSP controller at 15-minute intervals.

```
RP/0/RP0/CPU0:ios#show controller coherentDSP 0/2/0/1 pm current 15-min  fec


Sat Feb  9 11:23:42.196 UTC

g709 FEC in the current interval [11:15:00 - 11:23:42 Sat Feb 9 2019]

FEC current bucket type : Valid
    EC-BITS   : 291612035786            Threshold : 903330                TCA(enable)  :
YES
    UC-WORDS  : 0                       Threshold : 5                     TCA(enable)  :
YES

                MIN       AVG        MAX    Threshold   TCA     Threshold   TCA
                                              (min)    (enable)   (max)    (enable)
PreFEC BER  : 7.1E-03   7.2E-03   8.1E-03     0E-15       NO     0E-15       NO
PostFEC BER :   0E-15     0E-15     0E-15     0E-15       NO     0E-15       NO
```

This sample displays the current PRBS performance monitoring parameters of the Coherent DSP controller 15-minute intervals.

```
RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/0/0/1 pm current 15-min prbs
```

```
Mon Feb 13 00:58:48.327 UTC
PRBS in the current interval [00:45:00 - 00:58:48 Mon Feb 13 2019]
PRBS current bucket type : Valid
EBC : 40437528165
FOUND-COUNT : 1 FOUND-AT-TS : 00:51:22 Mon Feb 13 2019
LOST-COUNT : 1 LOST-AT-TS : 00:52:52 Mon Feb 13 2019
CONFIG-PTRN : PRBS_PATTERN_PN31
Last clearing of "show controllers OTU" counters never
```

This sample displays the current PCS performance monitoring parameters of the Coherent DSP controller 30-second intervals.

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/0/0/2 pm current 30-sec pcs
Tue Nov 19 09:17:26.684 UTC

Ethernet PCS in the current interval [09:17:00 - 09:17:26 Tue Nov 19 2019]

Ethernet PCS current bucket type : Valid
BIP[00] : 0 Threshold : 0 TCA(enable) : NO
BIP[01] : 0 Threshold : 0 TCA(enable) : NO
BIP[02] : 0 Threshold : 0 TCA(enable) : NO
BIP[03] : 0 Threshold : 0 TCA(enable) : NO
BIP[04] : 0 Threshold : 0 TCA(enable) : NO
BIP[05] : 0 Threshold : 0 TCA(enable) : NO
BIP[06] : 0 Threshold : 0 TCA(enable) : NO
BIP[07] : 0 Threshold : 0 TCA(enable) : NO
BIP[08] : 0 Threshold : 0 TCA(enable) : NO
BIP[09] : 0 Threshold : 0 TCA(enable) : NO
BIP[10] : 0 Threshold : 0 TCA(enable) : NO
BIP[11] : 0 Threshold : 0 TCA(enable) : NO
BIP[12] : 0 Threshold : 0 TCA(enable) : NO
BIP[13] : 0 Threshold : 0 TCA(enable) : NO
BIP[14] : 0 Threshold : 0 TCA(enable) : NO
BIP[15] : 0 Threshold : 0 TCA(enable) : NO
BIP[16] : 0 Threshold : 0 TCA(enable) : NO
BIP[17] : 0 Threshold : 0 TCA(enable) : NO
BIP[18] : 0 Threshold : 0 TCA(enable) : NO
BIP[19] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[00] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[01] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[02] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[03] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[04] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[05] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[06] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[07] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[08] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[09] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[10] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[11] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[12] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[13] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[14] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[15] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[16] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[17] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[18] : 0 Threshold : 0 TCA(enable) : NO
FRM-ERR[19] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[00] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[01] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[02] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[03] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[04] : 0 Threshold : 0 TCA(enable) : NO
```

```
BAD-SH[05] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[06] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[07] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[08] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[09] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[10] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[11] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[12] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[13] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[14] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[15] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[16] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[17] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[18] : 0 Threshold : 0 TCA(enable) : NO
BAD-SH[19] : 0 Threshold : 0 TCA(enable) : NO
ES : 0 Threshold : 0 TCA(enable) : NO
SES : 0 Threshold : 0 TCA(enable) : NO
UAS : 0 Threshold : 0 TCA(enable) : NO
ES-FE : 0 Threshold : 0 TCA(enable) : NO
SES-FE : 0 Threshold : 0 TCA(enable) : NO
UAS-FE : 0 Threshold : 0 TCA(enable) : NO

Last clearing of "show controllers ETHERNET " counters never
RP/0/RP0/CPU0:BH1_P2A4#
```

This sample displays the history PCS performance monitoring parameters of the 100GE controller at 30-second intervals.

```
RP/0/RP0/CPU0:ios#show controllers hundredGigECtrlr 0/0/0/2 pm history 30-sec pcs 1
Tue Nov 19 09:27:49.169 UTC

Ethernet PCS in the current interval [09:27:00 - 09:27:30 Tue Nov 19 2019]

Ethernet PCS current bucket type : Valid
BIP[00] : 0
BIP[01] : 0
BIP[02] : 0
BIP[03] : 0
BIP[04] : 0
BIP[05] : 0
.........
BIP[16] : 0
BIP[17] : 0
BIP[18] : 0
BIP[19] : 0
FRM-ERR[00] : 0
FRM-ERR[01] : 0
FRM-ERR[02] : 0
......... 0
FRM-ERR[15] : 0
FRM-ERR[16] : 0
FRM-ERR[17] : 0
FRM-ERR[18] : 0
FRM-ERR[19] : 0
BAD-SH[00] : 0
BAD-SH[01] : 0
BAD-SH[02] : 0
.......
BAD-SH[18] : 0
BAD-SH[19] : 0
ES : 0
SES : 0
UAS : 0
ES-FE : 0
```

```
SES-FE : 0
UAS-FE : 0

Last clearing of "show controllers ETHERNET " counters never
RP/0/RP0/CPU0:BH1_P2A4#
```

This ssample displays the current performance monitoring parameters of the optics controller at 10-second intervals as flexi-bin.

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/0 pm current flex-bin optics 1
Thu May 21 07:43:38.964 UTC

Optics in the current interval [07:43:30 - 07:43:38 Thu May 21 2020]

Flexible bin interval size: 10 seconds

Optics current bucket type : Valid
            MIN       AVG       MAX      Operational      Configured      TCA   Operational
      Configured     TCA
                                          Threshold(min)   Threshold(min) (min) Threshold(max)
   Threshold(max) (max)
LBC[% ]      : 0.0       0.0       0.0      0.0              NA              NO   0.0
          NA               NO
OPT[dBm]     : -0.13     -0.10     -0.06    0.00             NA              NO   0.00
          NA               NO
OPR[dBm]     : -3.01     -2.96     -2.92    0.00             NA              NO   0.00
          NA               NO
CD[ps/nm]    : -3        -2        -1       0                NA              NO   0
          NA               NO
DGD[ps ]     : 1.00      1.67      2.00     0.00             NA              NO   0.00
          NA               NO
SOPMD[ps^2]  : 17.00     37.00     81.00    0.00             NA              NO   0.00
          NA               NO
OSNR[dB]     : 37.60     37.60     37.60    0.00             NA              NO   0.00
          NA               NO
PDL[dB]      : 0.60      0.66      0.70     0.00             NA              NO   0.00
          NA               NO
PCR[rad/s]   : 0.00      29.11     80.00    0.00             NA              NO   0.00
          NA               NO
RX_SIG[dBm]  : -3.49     -3.41     -3.36    0.00             NA              NO   0.00
          NA               NO
FREQ_OFF[Mhz]: 191       241       301      0                NA              NO   0
          NA               NO
SNR[dB]      : 14.50     14.62     14.70    0.00             NA              NO   0.00
          NA               NO
SNR-AX[dB]   : 17.10     17.19     17.30    0.00             NA              NO   0.00
          NA               NO
SNR-AY[dB]   : 11.90     12.06     12.10    0.00             NA              NO   0.00
          NA               NO
SNR-BX[dB]   : 0.00      0.00      0.00     0.00             NA              NO   0.00
          NA               NO
SNR-BY[dB]   : 0.00      0.00      0.00     0.00             NA              NO   0.00
          NA               NO
SOP-S1       : 0.50      0.55      0.59     0.00             NA              NO   0.00
          NA               NO
SOP-S2       : -0.59     -0.52     -0.48    0.00             NA              NO   0.00
          NA               NO
SOP-S3       : -0.67     -0.64     -0.60    0.00             NA              NO   0.00
          NA               NO

Last clearing of "show controllers OPTICS" counters never
```

This sample displays the history performance monitoring parameters of the optics controller at 10-second intervals as flexi-bin.

```
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/0 pm history flex-bin optics 1 bucket 1
Thu May 21 07:45:44.358 UTC

Optics in interval 1 [07:45:30 - 07:45:40 Thu May 21 2020]

Flexible bin interval size: 10 seconds

Optics history bucket type : Valid
              MIN       AVG       MAX
LBC[% ]     : 0.0       0.0       0.0
OPT[dBm]    : -0.12     -0.10     -0.04
OPR[dBm]    : -3.01     -2.97     -2.91
CD[ps/nm]   : -5        -4        -3
DGD[ps ]    : 1.00      1.50      2.00
SOPMD[ps^2]: 28.00      43.10     66.00
OSNR[dB]    : 37.60     37.60     37.60
PDL[dB]     : 0.60      0.65      0.70
PCR[rad/s] : 0.00       25.70     75.00
RX_SIG[dBm] : -3.49      -3.44     -3.37
FREQ_OFF[Mhz]: 235        272       330
SNR[dB]    : 14.60     14.64     14.80
SNR-AX[dB]: 17.20      17.25     17.30
SNR-AY[dB]: 11.90      12.02     12.20
SNR-BX[dB]: 0.00       0.00      0.00
SNR-BY[dB]: 0.00       0.00      0.00
SOP-S1     : 0.50      0.53      0.57
SOP-S2     : -0.58     -0.53     -0.49
SOP-S3     : -0.69     -0.65     -0.61
```

This sample displays the current FEC performance monitoring parameters of the coherentDSP controller as flexi-bin.

```
RP/0/0/CPU0:ios#show controllers coherentDSP 0/2/0/0 pm current flex-bin fec
Thu Apr  9 11:46:55.659 IST

g709 FEC in the current interval [11:46:50 - 11:46:55 Thu Apr 9 2020]

Flexible bin interval size: 10 seconds

FEC current bucket type : Valid
    EC-BITS   : 327                     Threshold : 0                       TCA(enable)  :
NO
    UC-WORDS  : 327                     Threshold : 0                       TCA(enable)  :
NO

                                  MIN       AVG        MAX    Threshold   TCA     Threshold
   TCA
                                                                 (min)    (enable)     (max)
 (enable)
PreFEC BER                    :   5.20E-14 5.40E-14   5.70E-14     0E-15        NO
0E-15       NO
PostFEC BER                   :   5.20E-14 5.40E-14   5.70E-14     0E-15        NO
0E-15       NO
Q[dB]                         :    0.52      0.54      0.57    0.00        NO      0.00
     NO
Q_Margin[dB]                  :    2.52      1.54      4.57    0.00        NO      0.00
     NO
Q_Margin Instantaneous [dB]    :    2.52      1.54      4.57    0.00        NO      0.00
     NO
```

```
Last clearing of "show controllers OTU" counters never
```

This sample displays the current OTN path monitor performance monitoring parameters of the ODU4 controller 1-second intervals.

```
P/0/RP0/CPU0:ios#show controllers odu4 0/2/0/13/8 pm current flex-bin otn pathmonitor
Wed Sep 22 12:47:09.497 UTC

g709 OTN in the current interval [12:47:08 - 12:47:08 Wed Sep 22 2021]

Flexible bin interval size: 1 seconds

OTN current bucket type : Valid
    ES-NE   : 0         Threshold : 0        TCA(enable)  : NO
    ESR-NE  : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    SES-NE  : 0         Threshold : 0        TCA(enable)  : NO
    SESR-NE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    UAS-NE  : 0         Threshold : 0        TCA(enable)  : NO
    BBE-NE  : 0         Threshold : 0        TCA(enable)  : NO
    BBER-NE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    FC-NE   : 0         Threshold : 0        TCA(enable)  : NO

    ES-FE   : 0         Threshold : 0        TCA(enable)  : NO
    ESR-FE  : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    SES-FE  : 0         Threshold : 0        TCA(enable)  : NO
    SESR-FE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    UAS-FE  : 0         Threshold : 0        TCA(enable)  : NO
    BBE-FE  : 0         Threshold : 0        TCA(enable)  : NO
    BBER-FE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    FC-FE   : 0         Threshold : 0        TCA(enable)  : NO


Last clearing of "show controllers ODU" counters never
RP/0/RP0/CPU0:ios#
```

This sample displays the current performance monitoring parameters of the 16G FC controller for 24 hours.

```
RP/0/RP0/CPU0:ios#show controllers sixteenGigFibreChanCtrlr 0/2/0/1/1 pm current 24-hour
fc
Thu Jan 6 19:13:07.222 UTC

FC in the current interval [00:00:00 - 19:13:07 Thu Jan 6 2022]

FC current bucket type : Valid
IFIN-OCTETS : 8691662359380 Threshold : 0 TCA(enable) : NO
RX-PKT : 4061524467 Threshold : 0 TCA(enable) : NO
IFIN-ERRORS : 0 Threshold : 0 TCA(enable) : NO
RX-BAD-FCS : 0 Threshold : 0 TCA(enable) : NO
IFOUT-OCTETS : 8691662359380 Threshold : 0 TCA(enable) : NO
TX-PKT : 4061524467 Threshold : 0 TCA(enable) : NO
TX-BAD-FCS : 0 Threshold : 0 TCA(enable) : NO
RX-FRAMES-TOO-LONG : 0 Threshold : 0 TCA(enable) : NO
RX-FRAMES-TRUNC : 0 Threshold : 0 TCA(enable) : NO
TX-FRAMES-TOO-LONG : 0 Threshold : 0 TCA(enable) : NO
TX-FRAMES-TRUNC : 0 Threshold : 0 TCA(enable) : NO

Last clearing of "show controllers FC" counters never
RP/0/RP0/CPU0:ios#
```

This sample displays the PM statistics for OTN path monitor with flexible bin interval as 1 second.

```
P/0/RP0/CPU0:ios#show controllers odu4 0/2/0/13/8 pm current flex-bin otn pathmonitor
Wed Sep 22 12:47:09.497 UTC
```

```
g709 OTN in the current interval [12:47:08 - 12:47:08 Wed Sep 22 2021]

Flexible bin interval size: 1 seconds

OTN current bucket type : Valid
    ES-NE   : 0         Threshold : 0        TCA(enable)  : NO
    ESR-NE  : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    SES-NE  : 0         Threshold : 0        TCA(enable)  : NO
    SESR-NE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    UAS-NE  : 0         Threshold : 0        TCA(enable)  : NO
    BBE-NE  : 0         Threshold : 0        TCA(enable)  : NO
    BBER-NE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    FC-NE   : 0         Threshold : 0        TCA(enable)  : NO

    ES-FE   : 0         Threshold : 0        TCA(enable)  : NO
    ESR-FE  : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    SES-FE  : 0         Threshold : 0        TCA(enable)  : NO
    SESR-FE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    UAS-FE  : 0         Threshold : 0        TCA(enable)  : NO
    BBE-FE  : 0         Threshold : 0        TCA(enable)  : NO
    BBER-FE : 0.00000   Threshold : 0.00000  TCA(enable)  : NO
    FC-FE   : 0         Threshold : 0        TCA(enable)  : NO


Last clearing of "show controllers ODU" counters never
```

Viewing PM Statistics

To view PM statistics for the Ethernet controllers, use this command:

```
RP/0/RP0/CPU0:ios#show controllers HundredGigECtrlr 0/0/0/2 stats
Fri Aug 30 13:10:33.123 IST
Statistics for interface HundredGigECtrlr0/0/0/2 (cached values):

Ingress:
    Input total bytes           = 1702197139760640
    Input good bytes            = 1702197139760640

    Input total packets         = 13298415154380
    Input 802.1Q frames         = 0
    Input pause frames          = 0
    Input pkts 64 bytes         = 0
    Input pkts 65-127 bytes     = 0
    Input pkts 128-255 bytes    = 13298415154380
    Input pkts 256-511 bytes    = 0
    Input pkts 512-1023 bytes   = 0
    Input pkts 1024-1518 bytes  = 0
    Input pkts 1519-Max bytes   = 0

    Input good pkts             = 13298415154380
    Input unicast pkts          = 0
    Input multicast pkts        = 0
    Input broadcast pkts        = 0

    Input drop overrun          = 0
    Input drop abort            = 0
    Input drop invalid VLAN     = 0
    Input drop invalid DMAC     = 0
    Input drop invalid encap    = 0
    Input drop other            = 0

    Input error giant           = 0
    Input error runt            = 0
```

```
    Input error jabbers         = 0
    Input error fragments       = 0
    Input error CRC             = 0
    Input error collisions      = 0
    Input error symbol          = 0
    Input error other           = 0

    Input MIB giant             = 0
    Input MIB jabber            = 0
    Input MIB CRC               = 0

Egress:
    Output total bytes          = 1702197139760640
    Output good bytes           = 1702197139760640

    Output total packets        = 13298415154380
    Output 802.1Q frames        = 0
    Output pause frames         = 0
    Output pkts 64 bytes        = 0
    Output pkts 65-127 bytes    = 0
    Output pkts 128-255 bytes   = 13298415154380
    Output pkts 256-511 bytes   = 0
    Output pkts 512-1023 bytes  = 0
    Output pkts 1024-1518 bytes = 0
    Output pkts 1519-Max bytes  = 0

    Output good pkts            = 13298415154380
    Output unicast pkts         = 0
    Output multicast pkts       = 0
    Output broadcast pkts       = 0

    Output drop underrun        = 0
    Output drop abort           = 0
    Output drop other           = 0

    Output error other          = 0

RP/0/RP0/CPU0:ios#
```

Note Performance monitoring statistics are not supported for the input unicast packets, output unicast packets, and input error fragments counters for Ethernet clients.

Clear PM parameters

Use this task to restore PM parameters on Ethernet or Coherent DSP controllers, ensuring that performance metrics are reset for accurate monitoring.

Clearing PM parameters is necessary when you want to reset the accumulated performance statistics for your controllers. This allows you to start fresh measurements and can be useful for troubleshooting, maintenance, or validating recent changes.

Procedure

Use the command **clear controller** *controllertype R/S/I/P* **pm**, to clear the performance monitoring parameters for Ethernet and Coherent DSP controllers.

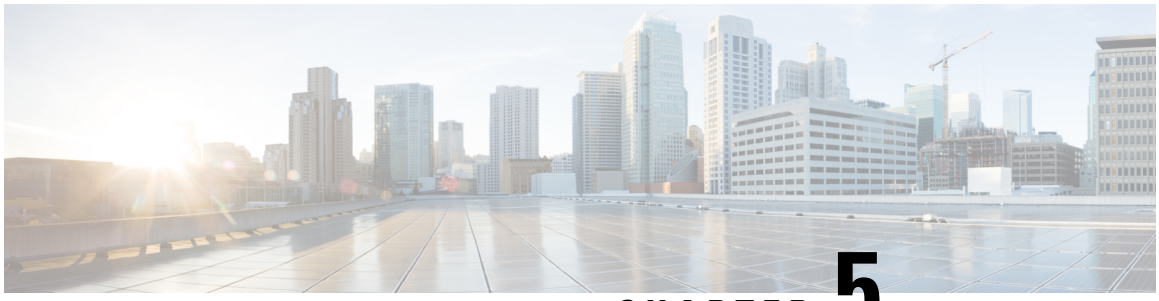
Example:

This sample clears the PM parameters on the Coherent DSP controller.

```
RP/0/RP0/CPU0:ios#show controller CD 0/0/0/0 pm current 15-min fec
Mon Jun 10 11:43:39.981 UTC

g709 FEC in the current interval [11:30:00 - 11:43:40 Mon Jun 10 2019]

FEC current bucket type : Invalid
    EC-BITS   : 308360273              Threshold : 903330                TCA(enable)  : YES
    UC-WORDS  : 131108352              Threshold : 5                     TCA(enable)  : YES

                MIN       AVG        MAX    Threshold   TCA     Threshold   TCA
                                              (min)    (enable)   (max)    (enable)
PreFEC BER  : 3.44E-02   3.45E-02   3.45E-02     0E-15        NO     0E-15       NO
PostFEC BER :   0E-15     0E-15     0E-15     0E-15        NO     0E-15       NO
Q           :    0.51      0.51      0.51    0.00        NO      0.00       NO
Q_Margin    :    0.00      0.00      0.00    0.00        NO      0.00       NO

Last clearing of "show controllers OTU" counters never
RP/0/RP0/CPU0:ios#clear controller coherentDSP 0/0/0/0 pm
Mon Jun 10 11:44:31.650 UTC
RP/0/RP0/CPU0:ios#show controller CD 0/0/0/0 pm current 15-min fec
Mon Jun 10 11:44:38.804 UTC

g709 FEC in the current interval [11:30:00 - 11:44:38 Mon Jun 10 2019]

FEC current bucket type : Invalid
    EC-BITS   : 0                      Threshold : 903330                TCA(enable)  : YES
    UC-WORDS  : 0                      Threshold : 5                     TCA(enable)  : YES

                MIN       AVG        MAX    Threshold   TCA     Threshold   TCA
                                              (min)    (enable)   (max)    (enable)
PreFEC BER  : 3.44E-02   3.44E-02   3.45E-02     0E-15        NO     0E-15       NO
PostFEC BER :   0E-15     0E-15     0E-15     0E-15        NO     0E-15       NO
Q           :    0.51      0.51      0.51    0.00        NO      0.00       NO
Q_Margin    :    0.00      0.00      0.00    0.00        NO      0.00       NO

Last clearing of "show controllers OTU" counters 00:00:07
```

This sample clears the PM parameters on the Ethernet controller.

```
RP/0/RP0/CPU0:ios#clear controller HundredGigECtrlr 0/0/0/2 pm
```



CHAPTER 5

IP Access Lists

- IP access control lists, on page 265

IP access control lists

An IP Access Control List (ACL) is a network traffic filtering mechanism that

- consists of ordered permit and deny statements evaluating IP addresses, protocols, and directions,
- controls packets by allowing or blocking their movement based on matching criteria,
- and is applied to interfaces, affecting traffic entering or leaving a system, but not traffic originating from the system itself.

ACLs are sequential lists that contain permit or deny statements for IP addresses and upper-layer protocols. These lists control packet flow by matching access list parameters to the information in the packet header. An access list becomes effective only when it is created and applied to an interface.

Packets can be filtered when they arrive at the device (ingress) or leave the device (egress). However, access lists cannot control traffic that originates from the system.

ACL processing paths:

There are two paths for interface packet filtering:

- Hardware programming path (fast path): Uses Ternary Content Addressable Memory (TCAM) via the packet filter Execution Agent for rapid ACL processing.
- Software programming path (slow path): Requires additional configuration through Interface Manager and NetIO, used for management interfaces.

Statistics for ACLs are collected separately for fast path and slow path packets. ACL information is stored globally on the route processor.

Examples of supported features in Cisco NCS 1004:

- Ingress ACL is supported for both IPv4 and IPv6.
- Management interface uses the slow packet path for ACL processing.
- Egress ACL: Self-originated packets are not controlled by ACLs, as these are already managed by the user. ACLs only filter forwarded packets/traffic, for both IPv4 and IPv6.

Best practices and requirements for configuring IP access control lists

These best practices, requirements, and guidelines should be followed when configuring IP ACLs.

- Requirement: Always include at least one permit statement in an access list; otherwise, all packets are denied.
- Best practice: Pay careful attention to the order of permit and deny statements. The order is critical because different orders may lead to different packet outcomes.
- Requirement: Only one access list is allowed for each interface, protocol, and direction.
- Best practice: Apply access lists only to management interfaces. ACLs do not filter traffic on other interfaces or controllers.
- Best practice: Always create the access list before attempting to apply it to an interface using the access-group command.
- Caution: To remove an access list, first remove its reference from the access group, and then remove the access list itself.
- Note: IP access lists cannot filter packets originating from the device itself; they filter only packets arriving at or leaving through an interface.

When you follow these best practices, IP access lists function as intended and protect network resources, preventing unintended packet loss and security breaches.

Configure an IP ACL

Use this procedure whenever you need to restrict or allow specific traffic types entering or leaving a network interface by specifying detailed access list rules.

Procedure

Step 1 Use the command **interface** *interface-type Rack/Slot/Instance/Port* to configure the interface in global configuration mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
Fri Oct 20 05:25:58.785 UTC
RP/0/RP0/CPU0:ios(config)#interface mgmtEth 0/RP0/CPU0/1t
```

Step 2 Use the command **ipv4 address** or **ipv6 address** in the interface configuration mode to configure IPv4 or IPv6 address for the interface.

Example:

```
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#ipv6 address 1000::1/64
```

Step 3 Use the **ipv4 | ipv6 access-group access-list-name {ingress | egress}** command, to configure the ACL at the IPv4 or IPv6 interface in the interface configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config-if)#ipv4 access-group IPV4_ICMP_DENY ingress
RP/0/RP0/CPU0:ios(config-if)#ipv4 access-group IPV4_ROUTER_FWD_TELNET_TRAFFIC_DENY egress
RP/0/RP0/CPU0:ios(config-if)#ipv6 access-group IPV6_SSH_DENY ingress
RP/0/RP0/CPU0:ios(config-if)#ipv6 access-group IPV6_ROUTER_FWD_TELNET_TRAFFIC_DENY egress
```

Step 4 Commit the configuration.

Example:

```
RP/0/RP0/CPU0:ios(config-ipv4-acl)# commit
RP/0/RP0/CPU0:ios(config-ipv6-acl)# commit
```

The IP access list is successfully applied to the selected interface. Traffic entering or leaving the interface is now filtered according to the rules you have defined for IPv4 and IPv6, helping enforce your network security policy.

Verify access control lists

Monitoring the number of packets processed by access control lists (ACLs) helps maintain network security and troubleshoot traffic flow. Use this task to review ACL statistics for both IPv4 and IPv6 to ensure filters are working as expected.

Procedure

Use the **show access-lists ipv4** or **show access-lists ipv6** command to verify the IPv4 or IPv6 ACLs.

Example:

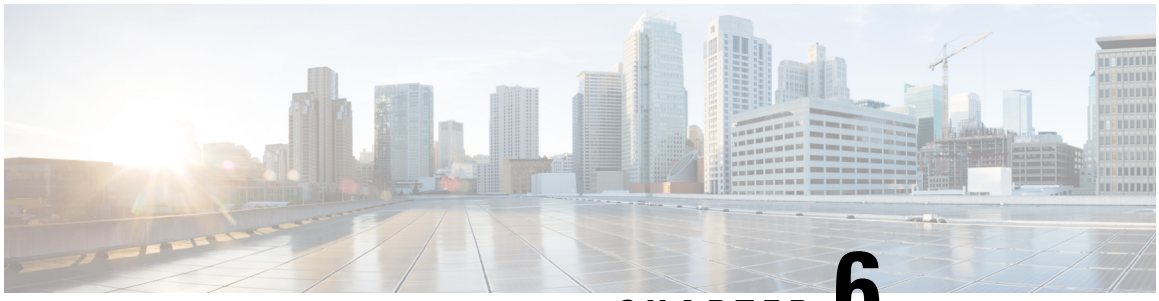
IPv4

```
RP/0/RP0/CPU0:ios#show access-lists ipv4
Wed Jan 17 09:52:12.448 IST
ipv4 access-list IPV4_ICMP_DENY
10 deny icmp any any (8 matches)
20 permit ipv4 any any (106 matches)
ipv4 access-list IPV4_ROUTER_FWD_TELNET_TRAFFIC_DENY
10 deny tcp any any eq telnet (3 matches)
20 permit ipv4 any any (6 matches)
```

Example:

IPv6

```
RP/0/RP0/CPU0:ios#show access-lists ipv6
Wed Jan 17 09:52:14.591 IST
ipv6 access-list IPV6_ROUTER_FWD_TELNET_TRAFFIC_DENY
10 deny tcp any any eq telnet (3 matches)
20 permit ipv6 any any (5 matches)
ipv6 access-list IPV6_SSH_DENY
10 deny tcp any any eq ssh (9 matches)
20 permit ipv6 any any (100 matches)
```



CHAPTER 6

Layer 1 Encryption

This chapter describes how to configure the IKEv2 protocol and layer 1 encryption for NCS 1004.

Note In this chapter, "layer 1 encryption" is referred to as "OTNSec".

Table 61: Feature History

Feature Name	Release Information	Feature Description
Encryption Support on 1.2TL Card	Cisco IOS XR Release 7.3.1	AES 256 GCM authenticated OTNSec encryption on 1.2TL line cards is supported. It uses only pre-shared keys for authentication. Optical encryption secures the communications link in and out of a facility, rendering all data undecipherable to hackers who tap into networks.
Encryption Support on OTN-XP Card	Cisco IOS XR Release 7.8.1	AES 256-GCM authenticated OTNSec encryption is supported on the OTN-XP card. The encryption is enabled on the ODUC4 controller. This encryption secures the data across different datapaths of the OTN-XP card.

Feature Name	Release Information	Feature Description
Encryption for 10G clients and 100GE clients on OTN-XP Card	Cisco IOS XR Release 7.9.1	OTN-XP card now supports AES 256-GCM authenticated OTNSec encryption for 10G and 100GE clients in the 40x10G-4x100G-MXP mode. As this authentication method uses a key size of 256 bits, it provides considerably strong cryptography acceptable by enterprise, and public sector organizations.

The Need for High Speed Encryption

Most of the emphasis on protecting networks today is focused on protecting data within data center. However, the infrastructure of networks that connect these data centers are as vulnerable to calculated attacks as the data centers themselves. As more sensitive information gets transmitted across fiber-optic networks, cyber criminals are increasingly turning their attention to intercepting the data when it travels across the network.

With the increase in network or fiber optic hacks, the need for data protection is paramount. Encryption of any data that leaves the data centers is becoming an important requirement for cloud operators. Optical encryption secures everything on the communications link in and out of a facility rendering all data undecipherable to any hacker that taps into the fiber strand. *Protecting data at high speeds or lines rates is a requirement for data centers today.*

The Cisco NCS 1004 brings to you AES256 based OTNSec encryption for 100GE and OTU4 clients. Encryption is supported on the 1.2T and 1.2TL cards.

From Release 7.8.1, layer 1 encryption is supported on the OTN-XP card. For OTN-XP card, different LC modes have different client-side controllers. Hence, encryption has to be enabled under the common ODUCn controller. In release 7.8.1, the OTNSec encryption is enabled under the ODUC4 controller for the following modes with a trunk rate of 400G and with CFP2 DCO Greylock 2 pluggable:

- 4x100G-MXP of 4x100G-MX-400G-TXP LC mode
- FC-MXP mode

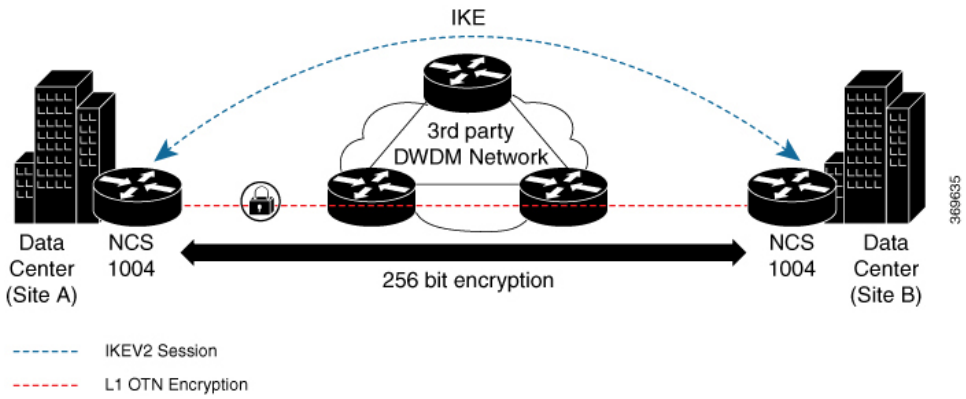
From Cisco IOS XR Release 7.9.1, in addition to the 10G client, with 40x10G-4x100G-MXP LCMode in OTN-XP card, you can enable encryption on the ODUC1, ODUC2, ODUC3, and ODUC4 controller for trunk rates 100-400G.

This encryption provides more flexibility across different LC modes and datapaths of the OTN-XP card.

OTNSec encryption uses the IKEv2 protocol to negotiate and establish the IKEv2 and OTNSec Security Associations (SA). IKEv2 is used for authentication of the devices in an encryption session, and the protocol provides pre-shared keys (PSK) or RSA certificate-based authentication. The IKEv2 datagrams are carried as payloads using the point-to-point protocol (PPP) over the GCC channel.

To implement this, an IKE session is established between the two endpoints, Site A and Site B, for overhead control plane communication between the two data centers. Data is then encrypted at Site A using OTNSec encryption and decrypted at Site B.

Figure 5: OTNSec Site-to-Site Example and Components



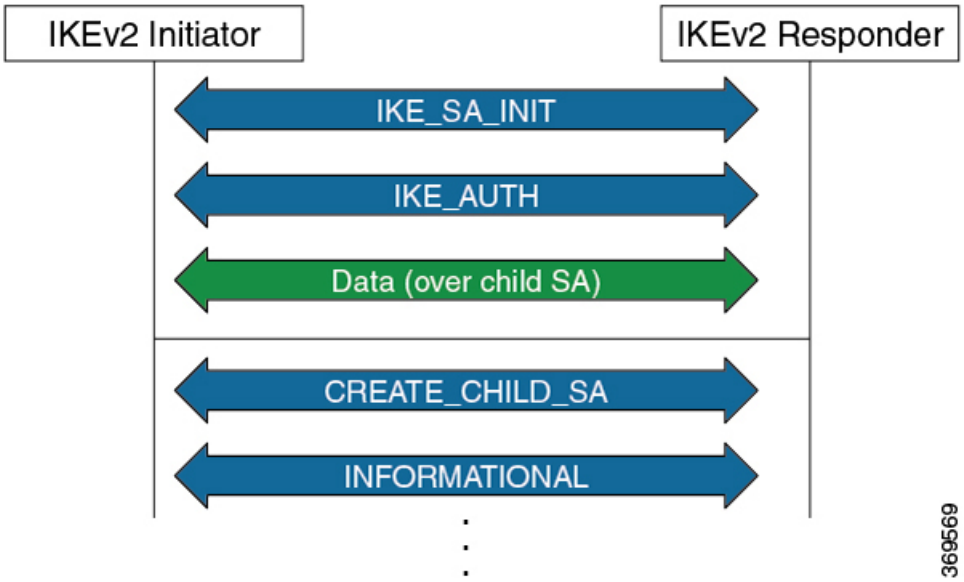
The recommended deployment is to have a single IKEv2 session running over a GCC2 channel per trunk port which creates the child SAs for each of the OTNSec controllers that are configured on the trunk port.

- IKEv2 Overview, on page 271
- OTNSec Encryption Overview, on page 273
- Prerequisites, on page 274
- Limitations, on page 275
- Configuration Workflow, on page 275
- Configuration Example, on page 284
- Verification, on page 287
- Troubleshooting, on page 288
- IKEv2 Certificate-Based Authentication, on page 288
- You May Be Interested In, on page 293

IKEv2 Overview

Internet Key Exchange Version 2 (IKEv2) is a request and response encryption that establishes and handles security associations (SA) in an authentication suite, such as OTNSec, to ensure secure traffic. IKE performs mutual authentication between two endpoints and establishes an IKE Security Association (SA). All IKE communications consist of pairs of messages that include a request and a response. The pair is called an exchange or a request-response pair. The first two exchanges of messages establishing an IKE SA are called the IKE_SA_INIT exchange and the IKE_AUTH exchange; subsequent IKE exchanges are called either CREATE_CHILD_SA exchanges or INFORMATIONAL exchanges. IKEv2 uses sequence numbers and acknowledgments to provide reliability, and mandates some error-processing logistics and shared state management (windowing). IKEv2 does not process a request until it determines the requester. This helps to mitigate DoS attacks. IKEv2 provides built-in support for Dead Peer Detection (DPD), which periodically confirms the availability of the peer node. When there is no response from the peer node, the system attempts to establish the session again.

Figure 6: IKEv2 Exchanges



IKEv2 is defined in RFC 7296 and consists of the following constructs:

- **Keyring**

  A keyring is a repository of symmetric and asymmetric pre-shared keys that is configured for a peer and identified using the IP address of the peer. The keyring is associated with an IKEv2 profile and therefore, caters to a set of peers that match the IKEv2 profile. This is a required configuration for the pre-shared keys authentication method that is used for NCS 1004.

Note The certificate-based authentication that uses RSA signatures can be used instead of the keyring. If both methods of authentication are configured, the certificate-based authentication takes precedence. See IKEv2 Certificate-Based Authentication, on page 288.

- **IKEv2 Profile**

  An IKEv2 profile is a repository of nonnegotiable parameters of the IKE SA, such as authentication method and services that are available to the authenticated peers that match the profile. The profile match lookup is done based on the IP address of the remote identity. For security purposes, the IKE SAs have a lifetime that is defined in the IKEv2 profile. The lifetime range, in seconds, is from 120 to 86400. The SAs are rekeyed proactively before the expiry of the lifetime. The default lifetime is 86400. An IKEv2 profile must be attached to an OTNSec configuration on the ODU4 controllers or ODUC4 controllers on both the IKEv2 initiator and responder. This is a required configuration.

Note Only one authentication method is supported for the local peer but multiple authentication methods can be configured for the remote peer.

If both methods of authentication are configured , keyring and certificate trustpoint (see, IKEv2 Certificate-Based Authentication, on page 288) in the profile, the remote peer can authenticate itself using either method. **authentication remote [pre-shared| rsa-signature]** can be used to exclusively control the remote authentication method. Similarly, **authentication local [pre-shared | rsa-signature]** can be used to exclusively configure local authentication method. If it is not configured, the certificate-based authentication takes precedence.

- **IKEv2 Proposal**

  An IKEv2 proposal is a collection of transforms that are used in the negotiation of IKE SAs as part of the IKE_SA_INIT exchange. The IKE2 proposal must be attached to an IKEv2 policy. This is an optional configuration. The transform types used in the negotiation are as follows:

  - Encryption algorithm
  - Integrity algorithm
  - Pseudo-Random Function (PRF) algorithm
  - Diffie-Hellman (DH) group

Note The IKEv2 proposal must have at least one algorithm of each type. It is possible to specify multiple algorithms for each type; the order in which the algorithms are specified determines the precedence.

- **IKEv2 Policy**

  IKEv2 employs policies that are configured on each peer to negotiate handshakes between the two peers. An IKEv2 policy contains proposals that are used to negotiate the encryption, integrity, PRF algorithms, and DH group in the SA_INIT exchange. An IKEv2 policy is selected based on the local IP address. This is an optional configuration.

Note The default IKEv2 proposal is used with default IKEv2 policy in the absence of any user-defined policy.

OTNSec Encryption Overview

OTNSec encryption in NCS 1004 has the following characteristics:

- The OTN layer 1 security is supported over the OPU client payload.

- The Galois-Counter-Mode (GCM) AES 256-bit security is the default cipher used for encryption and decryption of the OPU payloads.
- Each client offers an independent encrypted channel in each direction.
- There are two banks of 256-bit programmable key registers (current key and future key) that permit key updates through the software without interrupting traffic.

  Each key is associated with an Association Number [AN(1:0)] allowing up to four different numbers.
- Interhost key exchange is supported through communication over GCC.
- The encryption is supported in headless mode.

The OTNSec control plane generates two different keys, one for the transmit (Tx) side and the other for the receive (Rx) side. These keys are used by the line card to program the encryptor and decryptor blocks. These blocks encrypt and decrypt the data packets between the trunk ports of the two nodes. For security purposes, the keys have a lifetime. A key's lifetime specifies the time the key expires.

The key lifetime for the child SAs can be configured using the sak-rekey-interval which ranges from 30 seconds to 14 days. For example, if the sak-rekey-interval is configured for five minutes, a new key is generated by the OTNSec layer every five minutes. In the absence of a lifetime configuration, the default lifetime is 14.18 days. When the key reaches the maximum lifetime, it becomes invalid and the CRYPTO-KEY-EXPIRED alarm is raised. Volume-based rekeying is supported; it prevents the key from reaching the maximum lifetime. This allows the OTNSec layer to generate a new key when 70% of the lifetime (11 days) of the current key is over.

When the lifetime of the first key expires, it automatically rolls over to the next key. To achieve a hitless rollover, the lifetimes of the keys need to be overlapped so that for a certain period of time both keys are active. To maintain this seamless switchover, a key index table is maintained. Each key pair (Tx and Rx) is associated with an Association Number (AN). The index table allows up to four numbers (0,1, 2, and 3). When the keys are installed, the Rx AN number of node A must match the Tx AN number of node B. Also, the Tx AN number of node A must match the Rx AN number of node B. If there is a mismatch of the AN numbers between the peer nodes, the CRYPTO-INDEX-MISMATCH alarm is raised.

Prerequisites

- Ensure that the required k9sec.rpm package is installed.
- Configure the line card in the muxponder or muxponder slice mode using the following commands:
  - **1.2T Card:**
    - muxponder mode:

      hw-module location *location* **mxponder client-rate 100GE** | **OTU4**

      hw-module location *location* **mxponder trunk-rate** {**100G** | **200G** | **300G** | **400G** | **500G** | **600G**}
    - muxponder slice mode:

      hw-module location *location* **mxponder-slice** *mxponder-slice-number* **client-rate 100GE** | **OTU4**

      hw-module location *location* **mxponder-slice trunk-rate** { **100G** | **200G** | **300G** | **400G** | **500G** | **600G** }

- **1.2TL Card:**
  - muxponder mode:

    hw-module location *location* **mxponder client-rate 100GE** | **OTU4**

    hw-module location *location* **mxponder trunk-rate** {**200G** | **300G** | **400G** }

  - muxponder slice mode:

    hw-module location *location* **mxponder-slice** *mxponder-slice-number* **client-rate 100GE** | **OTU4**

    hw-module location *location* **mxponder-slice trunk-rate** { **200G** | **300G** | **400G** }

- **OTN-XP Card:**

  muxponder mode:

  hw-module location *location* **mxponder-slice** *mxponder-slice-number* trunk-rate **400G** **client-port-rate** *client-port-number* **lane** *lane-number* **client-type** { **10GE** | **100GE** |**OTU2** | **OTU2e** | **400GE** | **FC16** | **FC32**}

Limitations

- Traffic is impacted for a few seconds if the RP fails or GCC2 control plane goes down, during a key rollover.
- The sak-rekey-interval must be configured on the initiator and responder node.

Configuration Workflow

This section describes the workflow to configure IKEv2 and OTNSec encryption on NCS 1004. The authentication method used is pre-shared keys (PSKs).

Figure 7: L1 Encryption Workflow

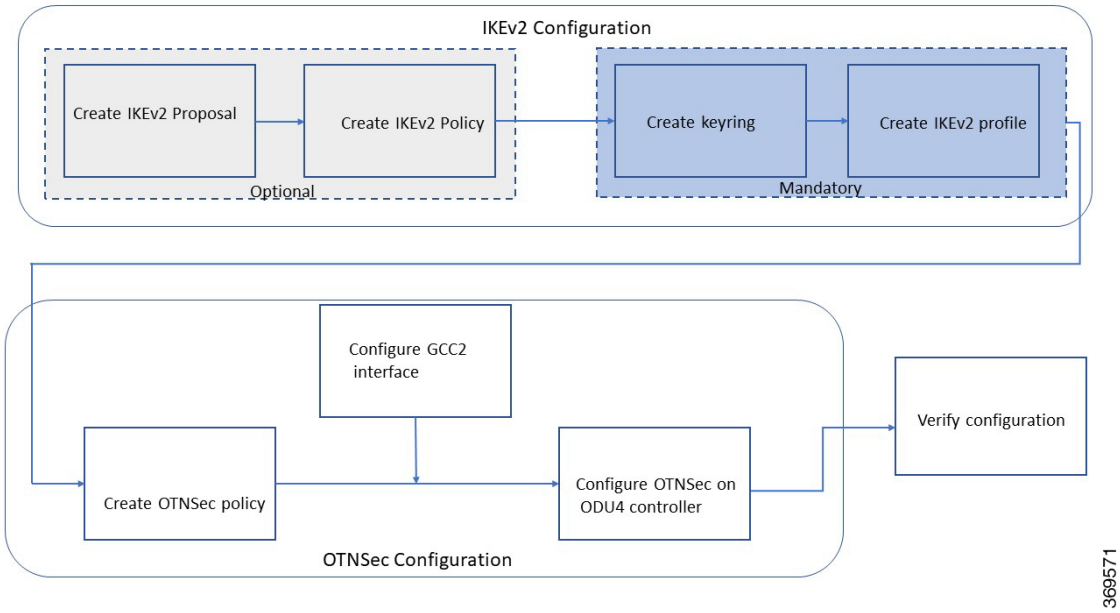


Table 62: Workflow for Configuring IKEv2 and OTNSec Encryption on NCS 1004

Workflow Sequence	Details
IKE Configuration	

Configuring an IKEv2 Proposal, on page 277	(Optional) Configure an IKEv2 proposal manually;otherwise, the default IKEv2 proposal is used in the default IKEv2 policy.
	The default IKEv2 proposal requires no configuration and is a collection of commonly used transforms types, which are as follows:
	`encryption cbc-aes-256`
	`integrity sha512, sha384`
	`prf sha512, sha384`
	`dh 19, 20, 21`

Configuring an IKEv2 Policy, on page 278	(Optional) Configure an IKEv2 policy manually;otherwise, the default proposal associated with the default policy is used for negotiation. **Note** An IKEv2 policy with no proposal is considered incomplete.
Configuring a Keyring, on page 279	Configure a keyring as the local or remote authentication method is a preshared key.

Workflow Sequence	Details
Configuring a IKEv2 Profile, on page 280	Configure an IKEv2 profile. **Note** • The IKEv2 profile must be attached to the OTNSec profile on both the IKEv2 initiator and the responder. • The DPD interval is 10 seconds. If there is no response from the peer node, it retries every two seconds with a maximum of five attempts. After five retries, the IKE session is brought down. NCS 1004 supports headless mode. Therefore, even though the control plane is down, traffic is not impacted because the encryption and decryption keys are still active on the line cards. The data path functions in a locally secure mode and the OTNSEC-LOCALLY-SECURED alarm is raised.
OTNSec Configuration	
Configuring an OTNSec Policy, on page 280	(Optional) Configure the OTNSec policy.
Configuring the GCC Interface, on page 281	Configure the GCC2 interface.
Configuring OTNSec on ODU4 Controllers, on page 282	Configure the ODU4 controller that is mapped to the HundredGigE controller for 1.2T and 1.2TL cards.
Configuring OTNSec on ODUC4 Controllers for OTN-XP Card, on page 283	Configure the ODUC4 controller in the OTN-XP card. The ODUC4 controller is mapped to the 400GE client for the TXP datapath, and 4x100GE clients for the MXP datapath.
Verification	
Verification, on page 287	Verify the IKEv2 and OTNSec configuration.

Configuring an IKEv2 Proposal

To configure an IKEv2 proposal, use the following commands:

config

ikev2 proposal *proposal-name*

encryption {aes-gcm-256} {aes-gcm-128} {aes-cbc-256} {aes-cbc-192} {aes-cbc-128}

integrity {sha-1} {sha-256} {sha-384} {sha-512}

prf {sha-1} {sha-256} {sha-384} {sha-512}

dh {19} {20} {21}

Note Configuring an AES-GCM encryption algorithm does not require configuring an integrity algorithm. AES-GCM and non-GCM algorithms cannot be configured in the same proposal. However, you can configure the AES-GCM and non-GCM algorithms under two different proposals and attach both the proposals to the same IKEv2 policy.

The following sample displays how to configure an IKEv2 proposal.

```
RP/0/RP0/CPU0:ios#configure
Thu Mar  7 19:19:30.259 UTC
RP/0/RP0/CPU0:ios(config)#ikev2 proposal proposal1
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#encryption aes-cbc-256
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#integrity sha-1
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#prf sha-256
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#dh 20
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#commit
Thu Mar  7 19:20:30.916 UTC
RP/0/RP0/CPU0:ios(config-ikev2-proposal-proposal1)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show ikev2 proposal proposal1
Thu Mar  7 19:20:48.929 UTC

Proposal Name              : proposal1
================================================================================
Status                     : Complete
--------------------------------------------------------------------------------
Total Number of Enc. Alg.  : 1
   Encr. Alg.              : CBC-AES-256
--------------------------------------------------------------------------------
Total Number of Hash. Alg. : 1
   Hash. Alg.              : SHA 1
--------------------------------------------------------------------------------
Total Number of PRF. Alg.  : 1
   PRF. Alg.               : SHA 256
--------------------------------------------------------------------------------
Total Number of DH Group   : 1
   DH Group                : Group 20
```

Configuring an IKEv2 Policy

To configure an IKEv2 policy, use the following commands:

config

ikev2 policy *policy-name*

proposal *proposal-name1 proposal-name2 proposal-name3*

match address local *{ ipv4-address}*

The following sample displays how to configure an IKEv2 policy.

```
RP/0/RP0/CPU0:ios#configure
Thu Mar  7 19:26:45.752 UTC
RP/0/RP0/CPU0:ios(config)#ikev2 policy mypolicy
RP/0/RP0/CPU0:ios(config-ikev2-policy-mypolicy)#proposal proposal1
RP/0/RP0/CPU0:ios(config-ikev2-policy-mypolicy)#match address local 10.1.1.1
RP/0/RP0/CPU0:ios(config-ikev2-policy-mypolicy)#commit
Thu Mar  7 19:29:25.043 UTC
RP/0/RP0/CPU0:ios(config-ikev2-policy-mypolicy)#exit
```

```
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show ikev2 policy mypolicy
Thu Mar  7 19:30:30.343 UTC

Policy Name                           : mypolicy
================================================================================
Total number of match local addr      : 1
   Match address local                : 10.1.1.1
--------------------------------------------------------------------------------
Total number of proposal attached     : 1
   Proposal Name                      : proposal1
```

Configuring a Keyring

To configure a keyring, use the following commands:

config

keyring *keyring-name*

peer *peer-block name*

address *{ipv4-address [mask]}*

Note The IP address of the far-end node (remote node) must be used.

pre-shared-key {{*key***} {clear** *clear-text key***} {local** *local key***} {password***encrypted key***}}**

Note The key input can either be in clear text or in type 7 encrypted password format.

The following sample displays how to configure a keyring.

```
RP/0/RP0/CPU0:ios#configure
Thu Mar  7 19:33:14.594 UTC
RP/0/RP0/CPU0:ios(config)#keyring kyr1
RP/0/RP0/CPU0:ios(config-keyring-kyr1)#peer peer1
RP/0/RP0/CPU0:ios(config-keyring-kyr1-peer-peer1)#address 10.1.1.2 255.255.255.0
RP/0/RP0/CPU0:ios(config-keyring-kyr1-peer-peer1)#pre-shared-key password 106D000A064743595F
RP/0/RP0/CPU0:ios(config-keyring-kyr1-peer-peer1)#commit
Thu Mar  7 19:54:33.314 UTC
RP/0/RP0/CPU0:ios(config-keyring-kyr1-peer-peer1)#exit
RP/0/RP0/CPU0:ios(config-keyring-kyr1)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show keyring kyr1
Thu Mar  7 19:58:07.135 UTC

Keyring Name                          : kyr1
================================================================================
Total Peers                           : 1
--------------------------------------------------------------------------------
   Peer Name                          : peer1
   IP Address                         : 10.1.1.2
   Subnet Mask                        : 255.255.255.0
   Identity                           : Not configured
   Local PSK                          : Configured
   Remote PSK                         : Configured
```

```
   PPK Mode                          : Manual
   PPK Mandatory                     : Yes
   Local PSK Hash (Algo:SHA256, Format:base64)   :
D8XsPtnaX4gCSp4bCNLqHkgEWMJs9l6v2aXirCvGK2I=
   Remote PSK Hash (Algo:SHA256, Format:base64)  :
jBFwTDBOK/kT894SrK3T8hJQ5BZtCus/KyEgNj4cJVQ=
   Manual PPK Hash (Algo:SHA256, Format:base64)  :
6c7nGrky/ehjM40Ivk3p3+OeoEm9r7NCzmWexUULaa4=
```

Configuring a IKEv2 Profile

To configure an IKEv2 profile, use the following commands:

config

ikev2 profile *profile-name*

match identity remote address *{ipv4-address [mask]}*

keyring *keyring-name*

lifetime *seconds*

Note The lifetime range, in seconds, is from 120 to 86400.

The following sample displays how to configure an IKEv2 profile.

```
RP/0/RP0/CPU0:ios#configure
Thu Mar  7 20:00:36.490 UTC
RP/0/RP0/CPU0:ios(config)#ikev2 profile profile1
RP/0/RP0/CPU0:ios(config-ikev2-profile-profile1)#match identity remote address 10.1.1.2
255.255.255.0
RP/0/RP0/CPU0:ios(config-ikev2-profile-profile1)#keyring kyr1
RP/0/RP0/CPU0:ios(config-ikev2-profile-profile1)#lifetime 86400
RP/0/RP0/CPU0:ios(config-ikev2-profile-profile1)#commit
Thu Mar  7 20:15:03.401 UTC
RP/0/RP0/CPU0:ios(config-ikev2-profile-profile1)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show ikev2 profile profile1
Thu Mar  7 20:15:25.776 UTC

Profile Name                         : profile1
================================================================================
Keyring                              : kyr1
Lifetime(Sec)                        : 120
DPD Interval(Sec)                    : 10
DPD Retry Interval(Sec)              : 2
Match ANY                            : NO
Total Match remote peers             : 1
   Addr/Prefix                       : 10.1.1.2/255.255.255.0
```

Configuring an OTNSec Policy

To configure an OTNSec policy, use the following commands:

config

otnsec-policy *policy-name*

cipher-suite AES-GCM-256

security-policy must-secure

sak-rekey-interval *seconds*

Note The interval range, in seconds, is from 30 to 1209600. SAK rekey timer does not start by default until it is configured.

The following sample displays how to configure an OTNSec policy.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 11 15:16:58.417 UTC
RP/0/RP0/CPU0:ios(config)#otnsec policy otnsec-policy1
RP/0/RP0/CPU0:ios(config-otnsec-policy)#cipher-suite AES-GCM-256
RP/0/RP0/CPU0:ios(config-otnsec-policy)#security-policy must-secure
RP/0/RP0/CPU0:ios(config-otnsec-policy)#sak-rekey-interval 120
RP/0/RP0/CPU0:ios(config-otnsec-policy)#commit
```

The following is a sample of an OTNSec policy.

```
RP/0/RP0/CPU0:ios#show run otnsec policy otnsec-policy1
Tue Mar 12 11:14:03.591 UTC
otnsec policy otnsec-policy1
 cipher-suite AES-GCM-256
 security-policy must-secure
 sak-rekey-interval 120
!
```

Note When a software upgrade is performed from R.7.0.1 to later releases, traffic is impacted. This happens if the sak-rekey-interval is configured. To prevent traffic loss, disable the sak-rekey-interval before the software upgrade using the following commands:

```
Tue Nov 26 12:41:01.768 IST
RP/0/RP0/CPU0:ios(config)#otnsec policy OP1
RP/0/RP0/CPU0:ios(config-otnsec-policy)#no sak-rekey-interval
```

The sak-rekey-interval can be configured again after the upgrade process is complete.

Configuring the GCC Interface

To configure the GCC interface, use the following commands:

config

interface GCC2 *R/S/I/P*

ipv4 address *ipv4-address*

The following sample displays how to configure the GCC2 interface for 1.2T and 1.2TL cards.

```
RP/0/RP0/CPU0:ios#config
Tue Mar 12 12:06:32.547 UTC
RP/0/RP0/CPU0:ios(config)#controller odu4 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-odu4)#gcc2
RP/0/RP0/CPU0:ios(config-odu4)#commit
RP/0/RP0/CPU0:ios(config-odu4)#exit
```

```
RP/0/RP0/CPU0:ios#config
Tue Mar 12 11:16:04.749 UTC
RP/0/RP0/CPU0:ios(config)#interface GCC2 0/1/0/0/1
P/0/RP0/CPU0:ios(config-if)#ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
Tue Mar 12 11:18:32.867 UTC
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#sh run interface gcc2 0/1/0/0/1
Tue Mar 12 11:19:00.475 UTC
interface GCC20/1/0/0/1
 ipv4 address 10.1.1.1 255.255.255.0
!
```

The following sample displays how to configure the GCC2 interface for the OTN-XP card.

```
RP/0/RP0/CPU0:ios#config
Wed Sep 28 23:10:28.258 UTC
RP/0/RP0/CPU0:ios(config)#controller ODUC4 0/0/0/12
RP/0/RP0/CPU0:ios(config-oduc4)#gcc2
RP/0/RP0/CPU0:ios(config-oduc4)#commit
RP/0/RP0/CPU0:ios(config-oduc4)#exit
```

```
RP/0/RP0/CPU0:ios#config
Wed Sep 28 23:10:29.808 UTC
RP/0/RP0/CPU0:ios(config)#interface GCC2 0/0/0/12
P/0/RP0/CPU0:ios(config-if)#ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
Wed Sep 28 23:10:30.260 UTC UTC
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#sh run interface gcc2 0/0/0/12
Tue Mar 12 11:19:00.475 UTC
interface GCC20/0/0/12
 ipv4 address 10.1.1.1 255.255.255.0
!
```

Configuring OTNSec on ODU4 Controllers

To configure the OTNSec on ODU4 controller, use the following commands:

config

controller ODU4 *rack/slot/instance/port*

otnsec

source ipv4 *ipv4-address*

destination ipv4 *ipv4-address*

session-id *session-id*

policy *policy-name*

ikev2 *profile-name*

Note The session ID ranges 1–65535.

The following sample displays how to configure OTNSec on the ODU4 controller.

```
RP/0/RP0/CPU0:ios#configure
Mon Mar 12 12:10:21.374 UTC
RP/0/RP0/CPU0:ios(config)#controller ODU4 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-odu4)#otnsec
RP/0/RP0/CPU0:ios(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:ios(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:ios(config-otnsec)#session-id 9000
RP/0/RP0/CPU0:ios(config-otnsec)#policy otnsec-policy1
RP/0/RP0/CPU0:ios(config-otnsec)#ikev2 profile1
RP/0/RP0/CPU0:ios(config-otnsec)#commit
Mon Mar 12 12:14:17.609 UTC
RP/0/RP0/CPU0:ios(config-otnsec)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Configuring OTNSec on ODUC4 Controllers for OTN-XP Card

To configure the OTNSec on ODUC4 controller, use the following commands:

config

controller ODUC4 *rack/slot/instance/port*

otnsec

source ipv4 *ipv4-address*

destination ipv4 *ipv4-address*

session-id *session-id*

policy *policy-name*

ikev2 *profile-name*

Note The session ID ranges 1–65535.

The following sample displays how to configure OTNSec on the ODUC4 controller.

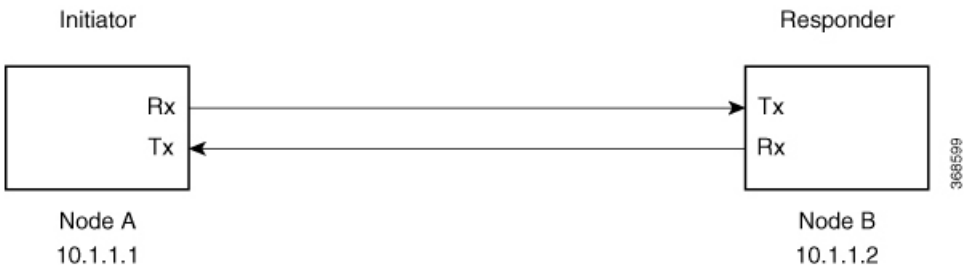
```
RP/0/RP0/CPU0:ios#configure
Wed Sep 28 23:10:48.429 UTC
RP/0/RP0/CPU0:ios(config)#controller oduc4 0/0/0/12
RP/0/RP0/CPU0:ios(config-oduc4)#otnsec
RP/0/RP0/CPU0:ios(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:ios(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:ios(config-otnsec)#session-id 99
RP/0/RP0/CPU0:ios(config-otnsec)#policy otnsec-policy1
RP/0/RP0/CPU0:ios(config-otnsec)#ikev2 profile1
RP/0/RP0/CPU0:ios(config-otnsec)#commit
Wed Sep 28 23:10:48.973 UTC
RP/0/RP0/CPU0:ios(config-otnsec)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Note From Cisco IOS XR Release 7.9.1, in addition to the 10G client, with 40x10G-4x100G-MXP LCMode in OTN-XP card, you can enable encryption on the ODUC1, ODUC2, ODUC3, and ODUC4 controller for trunk rates 100-400G.

Configuration Example

In the following example, there are two nodes. The node with the lower IP address always acts as the initiator. In this case, node A (SITE-A) has the role of an initiator while Node B (SITE-B) has the role of a responder. In this example, the default IKE proposal and policy have been used on both nodes.

Figure 8: Configuration Schema



The configuration on Node A is displayed below.

Node A (Initiator)
Keyring

`RP/0/RP0/CPU0:SITE-A#configure`
`RP/0/RP0/CPU0:SITE-A(config)#keyring KR1`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1)#peer SITE-B`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1-peer-SITE-B)#address 10.1.1.2 255.255.255.0`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1-peer-SITE-B)#pre-shared-key password 106D000A064743595`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1-peer-SITE-B)#commit`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1-peer-SITE-B)#exit`
`RP/0/RP0/CPU0:SITE-A(config-keyring-KR1)#exit`

IKEv2 profile

`RP/0/RP0/CPU0:SITE-A(config)#ikev2 profile IP1`
`RP/0/RP0/CPU0:SITE-A(config-ikev2-profile-IP1)#match identity remote address 10.1.1.2 255.255.255.0`
`RP/0/RP0/CPU0:SITE-A(config-ikev2-profile-IP1)#keyring KR1`
`RP/0/RP0/CPU0:SITE-A(config-ikev2-profile-IP1)#lifetime 86400`
`RP/0/RP0/CPU0:SITE-A(config-ikev2-profile-IP1)#commit`
`RP/0/RP0/CPU0:SITE-A(config-ikev2-profile-IP1)#exit`

OTNSec policy

`RP/0/RP0/CPU0:SITE-A(config)#otnsec policy OP1`
`RP/0/RP0/CPU0:SITE-A(config-otnsec-policy)#cipher-suite AES-GCM-256`
`RP/0/RP0/CPU0:SITE-A(config-otnsec-policy)#security-policy must-secure`
`RP/0/RP0/CPU0:SITE-A(config-otnsec-policy)#sak-rekey-interval 120`
`RP/0/RP0/CPU0:SITE-A(config-otnsec-policy)#commit`
`RP/0/RP0/CPU0:SITE-A(config-otnsec-policy)#exit`

GCC interface for 1.2T and 1.2TL cards

Node A (Initiator)

```
RP/0/RP0/CPU0:SITE-A(config)#controller odu4 0/1/0/0/1
RP/0/RP0/CPU0:SITE-A(config-odu4)#gcc2
RP/0/RP0/CPU0:SITE-A(config-odu4)#commit
RP/0/RP0/CPU0:SITE-A(config-odu4)#exit
RP/0/RP0/CPU0:SITE-A(config)#interface GCC2 0/1/0/0/1
RP/0/RP0/CPU0:SITE-A(config-if)#ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:SITE-A(config-if)#commit
RP/0/RP0/CPU0:SITE-A(config-if)#exit
```

GCC interface for OTN-XP card

```
RP/0/RP0/CPU0:SITE-A(config)#controller oduc4 0/0/0/12
RP/0/RP0/CPU0:SITE-A(config-oduc4)#gcc2
RP/0/RP0/CPU0:SITE-A(config-oduc4)#commit
RP/0/RP0/CPU0:SITE-A(config-oduc4)#exit
RP/0/RP0/CPU0:SITE-A(config)#interface GCC2 0/0/0/12
RP/0/RP0/CPU0:SITE-A(config-if)#ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:SITE-A(config-if)#commit
RP/0/RP0/CPU0:SITE-A(config-if)#exit
```

OTNSec on ODU4 controller

```
RP/0/RP0/CPU0:SITE-A(config)#controller odu4 0/1/0/0/1
RP/0/RP0/CPU0:SITE-A(config-odu4)#otnsec
RP/0/RP0/CPU0:SITE-A(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:SITE-A(config-otnsec)#session-id 9000
RP/0/RP0/CPU0:SITE-A(config-otnsec)#policy OP1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#ikev2 IP1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#commit
RP/0/RP0/CPU0:SITE-A(config-otnsec)#exit
RP/0/RP0/CPU0:SITE-A(config-odu4)#exit
RP/0/RP0/CPU0:SITE-B(config)#exit
```

OTNSec on ODUC4 controller

```
RP/0/RP0/CPU0:SITE-A(config)#controller oduc4 0/0/0/12
RP/0/RP0/CPU0:SITE-A(config-oduc4)#otnsec
RP/0/RP0/CPU0:SITE-A(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:SITE-A(config-otnsec)#session-id 99
RP/0/RP0/CPU0:SITE-A(config-otnsec)#policy OP1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#ikev2 IP1
RP/0/RP0/CPU0:SITE-A(config-otnsec)#commit
RP/0/RP0/CPU0:SITE-A(config-otnsec)#exit
RP/0/RP0/CPU0:SITE-A(config-oduc4)#exit
RP/0/RP0/CPU0:SITE-B(config)#exit
```

The configuration on Node B is displayed below.

Node B (Responder)

Keyring

Node B (Responder)

```
RP/0/RP0/CPU0:SITE-B#configure
RP/0/RP0/CPU0:SITE-B(config)#keyring KR1
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1)#peer SITE-A
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1-peer-SITE-A)#address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1-peer-SITE-A)#pre-shared-key password
14341B180F547B7977
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1-peer-SITE-A)#commit
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1-peer-SITE-A)#exit
RP/0/RP0/CPU0:SITE-B(config-keyring-KR1)#exit
```

IKEv2 profile

```
RP/0/RP0/CPU0:SITE-B(config)#ikev2 profile IP1
RP/0/RP0/CPU0:SITE-B(config-ikev2-profile-IP1)#match identity remote address 10.1.1.1
255.255.255.0
RP/0/RP0/CPU0:SITE-B(config-ikev2-profile-IP1)#keyring KR1
RP/0/RP0/CPU0:SITE-B(config-ikev2-profile-IP1)#lifetime 86400
RP/0/RP0/CPU0:SITE-B(config-ikev2-profile-IP1)#commit
RP/0/RP0/CPU0:SITE-B(config-ikev2-profile-IP1)#exit
```

OTNSec policy

```
RP/0/RP0/CPU0:SITE-B(config)#otnsec policy OP1
RP/0/RP0/CPU0:SITE-B(config-otnsec-policy)#cipher-suite AES-GCM-256
RP/0/RP0/CPU0:SITE-B(config-otnsec-policy)#security-policy must-secure
RP/0/RP0/CPU0:SITE-B(config-otnsec-policy)#sak-rekey-interval 120
RP/0/RP0/CPU0:SITE-B(config-otnsec-policy)#commit
RP/0/RP0/CPU0:SITE-B(config-otnsec-policy)#exit
```

GCC interface for 1.2T and 1.2TL cards

```
RP/0/RP0/CPU0:SITE-B(config)#controller odu4 0/1/0/0/1
RP/0/RP0/CPU0:SITE-B(config-odu4)#gcc2
RP/0/RP0/CPU0:SITE-B(config-odu4)#commit
RP/0/RP0/CPU0:SITE-B(config-odu4)#exit
RP/0/RP0/CPU0:SITE-B(config)#interface GCC2 0/1/0/0/1
RP/0/RP0/CPU0:SITE-B(config-if)#ipv4 address 10.1.1.2 255.255.255.0
RP/0/RP0/CPU0:SITE-B(config-if)#commit
RP/0/RP0/CPU0:SITE-B(config-if)#exit
```

GCC interface for OTN-XP card

```
RP/0/RP0/CPU0:SITE-B(config)#controller oduc4 0/0/0/12
RP/0/RP0/CPU0:SITE-B(config-oduc4)#gcc2
RP/0/RP0/CPU0:SITE-B(config-oduc4)#commit
RP/0/RP0/CPU0:SITE-B(config-oduc4)#exit
RP/0/RP0/CPU0:SITE-B(config)#interface GCC2 0/0/0/12
RP/0/RP0/CPU0:SITE-B(config-if)#ipv4 address 10.1.1.2 255.255.255.0
RP/0/RP0/CPU0:SITE-B(config-if)#commit
RP/0/RP0/CPU0:SITE-B(config-if)#exit
```

OTNSec on ODU4 controller

Node B (Responder)
RP/0/RP0/CPU0:SITE-B(config)#controller odu4 0/1/0/0/1 RP/0/RP0/CPU0:SITE-B(config-odu4)#otnsec RP/0/RP0/CPU0:SITE-B(config-otnsec)#source ipv4 10.1.1.2 RP/0/RP0/CPU0:SITE-B(config-otnsec)#destination ipv4 10.1.1.1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#session-id 9000 RP/0/RP0/CPU0:SITE-B(config-otnsec)#policy OP1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#ikev2 IP1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#commit RP/0/RP0/CPU0:SITE-B(config-otnsec)#exit RP/0/RP0/CPU0:SITE-B(config-odu4)#exit RP/0/RP0/CPU0:SITE-B(config)#exit
OTNSec on ODUC4 controller
RP/0/RP0/CPU0:SITE-B(config)#controller oduc4 0/0/0/12 RP/0/RP0/CPU0:SITE-B(config-oduc4)#otnsec RP/0/RP0/CPU0:SITE-B(config-otnsec)#source ipv4 10.1.1.2 RP/0/RP0/CPU0:SITE-B(config-otnsec)#destination ipv4 10.1.1.1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#session-id 99 RP/0/RP0/CPU0:SITE-B(config-otnsec)#policy OP1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#ikev2 IP1 RP/0/RP0/CPU0:SITE-B(config-otnsec)#commit RP/0/RP0/CPU0:SITE-B(config-otnsec)#exit RP/0/RP0/CPU0:SITE-B(config-oduc4)#exit RP/0/RP0/CPU0:SITE-B(config)#exit

Verification

- Verify that there are no alarms on the ports of the NCS 1004.
- Use the **show** commands listed in the table below to verify the IKEv2 and OTNSec configuration. For details of these commands, see the *Command Reference for Cisco NCS 1004*.

Table 63: Show Commands

Show Commands	Purpose
`show run ikev2`	Displays the running configuration of IKEv2
`show ikev2 session`	Displays the child SAs created for the session
`show ip interface brief`	Displays the status of the GCC interfaces
`show run controller ODU4 0/1/0/0/1`	Displays the running configuration of the ODU4 controller
`show run controller ODUC4 0/0/0/12`	Displays the running configuration of the ODUC4 controller

Show Commands	Purpose
`show controllers ODU4 0/1/0/0/1 otnsec`	Displays the OTNSec configuration on the ODU4 controller
`show controllers ODUC4 0/0/0/12 otnsec`	Displays the OTNSec configuration on the ODUC4 controller
`show controllers ODU4 0/1/0/0/1 pm current 15-min otnsec`	Displays the PM statistics that help verify the encrypted and decrypted blocks.
`show controllers ODUC4 0/0/0/12 pm current 15-min otnsec`	Displays the PM statistics that help verify the encrypted and decrypted blocks.

Troubleshooting

Problem: The IKE session is not established between the two nodes.

Solution: Check the status of the GCC interface using the **show ip interface brief** command.

To gather logs and traces, use the **show tech-support ncs1004 detail**, **show tech-support ikev2**, and **show tech-support otnsec** commands.

IKEv2 Certificate-Based Authentication

IKEv2 can use RSA digital signatures to authenticate peer devices before setting up SAs. RSA signatures employ a PKI-based method of authentication.

Certification Authority (CA) interoperability permits Cisco NCS 1004 devices and CAs to communicate so that your device can obtain and use digital certificates from the CA. A CA is responsible for managing certificate requests and issuing certificates to participating network devices. With a CA, a router authenticates itself to the remote router by sending a certificate to the remote router and performing some public key cryptography. Each router must send its own unique certificate that was issued and validated by the CA. This process works because the certificate of each router encapsulates the public key of the router, each certificate is authenticated by the CA, and all participating routers recognize the CA as an authenticating authority. This scheme is called IKE with an RSA signature.

In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other. In simple terms, a signature is formed when data is encrypted with a user's private key. The receiver verifies the signature by decrypting the message with the sender's public key. The fact that the message could be decrypted using the sender's public key indicates that the holder of the private key, the sender, must have created the message. This process relies on the receiver's having a copy of the sender's public key and knowing with a high degree of certainty that it does belong to the sender and not to someone pretending to be the sender.

Configuring IKEv2 Certificate-Based Authentication

To configure IKEv2 certificate-based authentication, perform the following steps:

1. Configure router hostname and IP domain name—You must configure the hostname and IP domain name of the router if they have not already been configured. The hostname and IP domain name are required because the router assigns a fully qualified domain name (FQDN) to the keys and certificates used by OTNsec, and the FQDN is based on the hostname and IP domain name you assign to the router.

   configure

   hostname name

   domain name *domain-name*

   commit

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:IOS(config)#hostname myhost
RP/0/RP0/CPU0:IOS(config)#domain name mydomain.com
RP/0/RP0/CPU0:IOS(config)#commit
```

2. Generate RSA key pair—The RSA key pair is required before you can obtain a certificate for your router.

   crypto key generate rsa *keypair-label*

```
RP/0/RP0/CPU0:ios#crypto key generate rsa tp
Thu May  7 16:18:44.243 IST
The name for the keys will be: tp
Do you really want to replace them? [yes/no]: yes
  Choose the size of the key modulus in the range of 512 to 4096 for your General
Purpose Keypair. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the [2048]:
Generating RSA keys ...
Done w/ crypto generate keypair
[OK]

RP/0/RP0/CPU0:ios#show crypto key mypubkey rsa
Thu May  7 16:19:06.606 IST
Key label: tp
Type     : RSA General purpose
Size     : 2048
Created  : 16:18:49 IST Thu May 07 2020
Data     :
 30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
 00CAC6E9 737D5ACF 31D0F8F2 281A450C 4F251D95 53587BCA 13592991 0AF2E6AF
 02A89439 1DEDA683 C467C55F 032F05F3 A72DDED9 323F171A FDDEE3C4 DC124439
 A78652F6 BB97BE63 F5AC8E3B 03B9B141 DD5D1AAE E41D7C15 28DE96E4 D3F4CE33
 B12C477A 525CBDF6 17B92A8C E94A816E 7C4BCEFA 0EA7972D A3B0CBF1 A1DED71E
 36CE08B7 3EF477A7 7B875BE1 E1B9E3A8 1E6C2717 6AB6D5AE 3A11D200 B32F4CCB
 0B4163A4 E44D5729 70ECFEE6 4713D1CC 588C8AFB D3EE9891 B27BCA5B 8CD82B76
 C278B32C 9A24B2EA 0CB9F2F3 6D1A1C95 044106F6 7E71520E B0201414 1E15B1C8
 A88E4164 F3474B66 86CF4DFB 8B0DA66C 8C80C8BF EF192CC8 F85FBF71 D1A35B7A
 05020301 0001
```

3. Declare a Certification Authority and configure a trustpoint.

   configure

   crypto ca trustpoint {*ca-name*}

   enrollment url {*ca-url*}

   subject-name {*x.500-name*}

   serial-number

rsakeypair {*keypair-label*}

crl optional

ip-address *ip-address*

commit

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios (config)# crypto ca trustpoint myca

RP/0/RP0/CPU0:ios(config-trustp)# enrollment url http://209.165.200.226

RP/0/RP0/CPU0:ios(config-trustp)# subject-name CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US

RP/0/RP0/CPU0:ios(config-trustp)#serial-number
RP/0/RP0/CPU0:ios(config-trustp)# rsakeypair tp

RP/0/RP0/CPU0:ios(config-trustp)# crl optional
RP/0/RP0/CPU0:ios(config-trustp)# ip-address 10.105.57.100

RP/0/RP0/CPU0:ios(config-trustp)# commit
```

4. Authenticate the CA—The router must authenticate the CA by obtaining the self-signed certificate of the CA, which contains the public key of the CA. Because the certificate of the CA is self-signed (the CA signs its own certificate), manually authenticate the public key of the CA by contacting the CA administrator to compare the fingerprint of the CA certificate.

   crypto ca authenticate ca-name

```
RP/0/RP0/CPU0:ios#crypto ca authenticate myca
Thu May  7 16:20:08.458 IST
  Serial Number  : 01
   CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
  Issued By      :
        CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
  Validity Start : 11:55:46 UTC Wed Jan 08 2020
  Validity End   : 11:55:46 UTC Sat Jan 07 2023
  SHA1 Fingerprint:
         70562AA850DE24B2D94AACF62528042E53C33D23
Do you accept this certificate? [yes/no]: yes
```

5. Request Device Certificates—You must obtain a signed certificate from the CA for each of your router's RSA key pairs.

   crypto ca enroll ca-name

```
RP/0/RP0/CPU0:ios#crypto ca enroll myca
Thu May  7 16:20:34.776 IST
% Start certificate enrollment ...
% Create a challenge password. You will need to verbally provide this
  password to the CA Administrator in order to revoke your certificate.
% For security reasons your password will not be saved in the configuration.
% Please make a note of it.

Password:
Re-enter Password:

% The subject name in the certificate will include:
CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
% The subject name in the certificate will include: myhost.mydomain.com
% The serial number in the certificate will be: 93f379c1
% The IP address in the certificate is 10.105.57.100
    Fingerprint:  41304434 42393333 45314143 42443134
```

6. Verify CA certificate.

   show crypto ca certificates *certificate-name*

```
RP/0/RP0/CPU0:ios#show crypto ca certificates myca
Thu May  7 16:21:24.633 IST

Trustpoint       : myca
================================================
CA certificate
  Serial Number  : 01
  Subject:
        CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
  Issued By      :
        CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
  Validity Start : 11:55:46 UTC Wed Jan 08 2020
  Validity End   : 11:55:46 UTC Sat Jan 07 2023
  SHA1 Fingerprint:
         70562AA850DE24B2D94AACF62528042E53C33D23
Router certificate
  Key usage      : General Purpose
  Status         : Available
  Serial Number  : 08:4D
  Subject:
        serialNumber=93f379c1,unstructuredAddress=10.105.57.100,
unstructuredName=myhost.mydomain.com,CN=ncs,OU=BU,O=Govt,
L=Newyork,ST=NY,C=US
  Issued By      :
        CN=ncs,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
  Validity Start : 10:44:51 UTC Thu May 07 2020
  Validity End   : 10:44:51 UTC Fri May 07 2021

  CRL Distribution Point
        http://7200.cisco.com
  SHA1 Fingerprint:
         C9454B6DD92A057A1DDB60740E0459243B070B24
Associated Trustpoint: myca
```

7. Configure IKEv2 profile.

   config

   ikev2 profile *profile-name*

   match identity remote address *{ipv4-address [mask]}*

   pki trustpoint *trustpoint-label*

   lifetime *seconds*

   authentication local rsa-signature

   authentication remote rsa-signature

   commit

```
RP/0/RP0/CPU0:ios#configure
Thu May  7 16:22:33.804 IST
RP/0/RP0/CPU0:ios(config)#ikev2 profile IP1
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#match identity remote address 10.1.1.2
255.255.255.255
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#pki trustpoint myca
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#lifetime 86400
Router(config)#
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#authentication local rsa-signature
```

```
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#authentication remote rsa-signature
RP/0/RP0/CPU0:ios(config-ikev2-profile-IP1)#commit
```

8. Configure the GCC2 interface. See Configuring the GCC Interface, on page 281.

9. Configure OTNsec on ODU4 controller for 1.2T and 1.2TL line cards.

   config

   controller ODU4 *R/S/I/P*

   otnsec

   source ipv4 *ipv4-address*

   destination ipv4 *ipv4-address*

   session-id *session-id*

   policy *policy-name*

   ikev2 *profile-name*

```
RP/0/RP0/CPU0:ios#configure
Thu May  7 16:27:57.294 IST
RP/0/RP0/CPU0:ios(config)#controller ODU4 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-odu4)#otnsec
RP/0/RP0/CPU0:ios(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:ios(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:ios(config-otnsec)#session-id 1
RP/0/RP0/CPU0:ios(config-otnsec)#policy OP1
RP/0/RP0/CPU0:ios(config-otnsec)#ikev2 IP1
RP/0/RP0/CPU0:ios(config-otnsec)#commit
```

10. Configure OTNsec on ODUC4 controller for OTN-XP card.

    config

    controller ODUC4 *R/S/I/P*

    otnsec

    source ipv4 *ipv4-address*

    destination ipv4 *ipv4-address*

    session-id *session-id*

    policy *policy-name*

    ikev2 *profile-name*

```
RP/0/RP0/CPU0:ios#configure
Thu May  7 16:27:58.394 IST
RP/0/RP0/CPU0:ios(config)#controller ODUC4 0/0/0/12
RP/0/RP0/CPU0:ios(config-odu4)#otnsec
RP/0/RP0/CPU0:ios(config-otnsec)#source ipv4 10.1.1.1
RP/0/RP0/CPU0:ios(config-otnsec)#destination ipv4 10.1.1.2
RP/0/RP0/CPU0:ios(config-otnsec)#session-id 1
RP/0/RP0/CPU0:ios(config-otnsec)#policy OP1
RP/0/RP0/CPU0:ios(config-otnsec)#ikev2 IP1
RP/0/RP0/CPU0:ios(config-otnsec)#commit
```

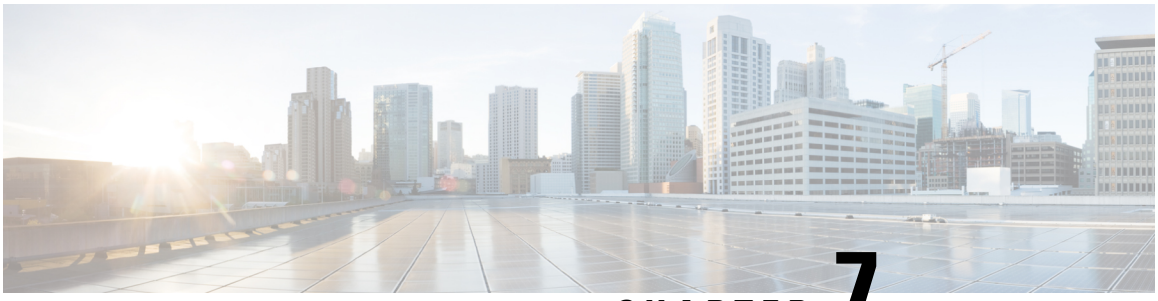
11. Verify the IKEv2 session.

```
RP/0/RP0/CPU0:ios#show ikev2 session
Wed Sep 22 21:09:38.363 IST
```

```
Session ID                              : 4
================================================================================
 Status                                 : UP-ACTIVE
 IKE Count                              : 1
 Child Count                            : 1
 IKE SA ID                              : 373
--------------------------------------------------------------------------------
  Local                                 : 10.1.1.1/500
  Remote                                : 10.1.1.2/500
  Status(Description)                   : READY (Negotiation done)
  Role                                  : Initiator

 Child SA
--------------------------------------------------------------------------------
   Local Selector                       : 10.1.1.1/1 - 10.1.1.1/1
   Remote Selector                      : 10.1.1.2/1 - 10.1.1.2/1
   ESP SPI IN/OUT                       : 0x2803 / 0x2800

RP/0/RP0/CPU0:ios#show ikev2 summary
Wed Sep 22 21:09:42.354 IST

IKEv2 SA Summary
------------------------------------------------
Total SA (Active/Negotiating)         : 1 (1/0)
Total Outgoing SA (Active/Negotiating): 1 (1/0)
Total Incoming SA (Active/Negotiating): 0 (0/0)
```

You May Be Interested In

- For more information about IKEv2, see RFC 7296.
- For more information about NCS 1004, see the NCS 1004 datasheet.



CHAPTER 7

Quantum-Safe Encryption Using Postquantum Preshared Keys

- Postquantum preshared keys, on page 295
- Verify the PPK configuration, on page 305

Postquantum preshared keys

A postquantum preshared key is a security enhancement that

- strengthens IKEv2 encryption by adding additional preshared keys to the key derivation process,
- makes VPN communications resilient against attacks by future quantum computers by incorporating quantum-safe techniques, and
- extends the standard cryptographic protocol to comply with RFC 8784, supporting both manual and dynamic PPK generation.

Table 64: Feature history

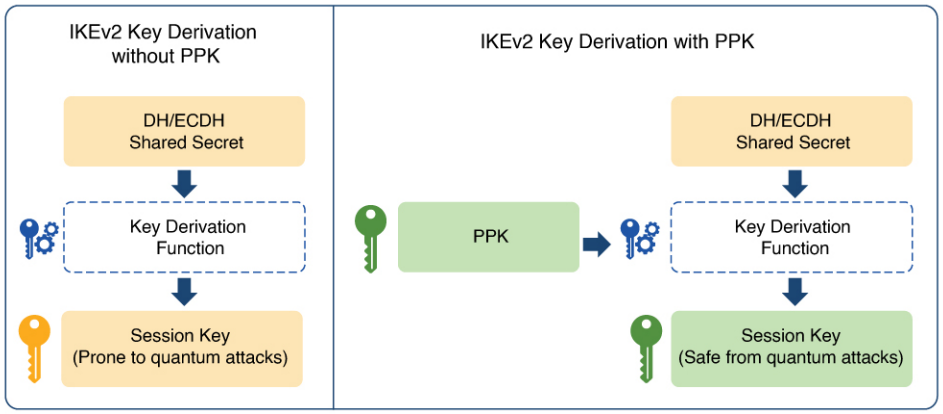
Feature name	Release information	Description
SKIP Protocol Support for Quantum Safe IKEv2 Encryption	Release 24.1.1	Traditionally, the IKEv2 encryption was vulnerable to quantum attacks. Now, IKEv2 encryption complies with RFC 8784, which specifies using postquantum preshared keys (PPK) to make it resilient to quantum attacks. You can generate both manual and dynamic PPKs. The dynamic PPKs are generated using the Cisco Secure Key Integration Protocol (SKIP). The IKEv2 encryption is configured through CLI or by the Cisco-IOS-XR-um-ikev2-cfg Yang model. CLI: • The **ppk manual/dynamic** keyword is introduced in the **keyring** command. • The **keyring ppk** keyword is introduced in the **ikev2 profile** command. • The **sks profile** command is introduced.

Quantum computers have raised significant concerns about the security of traditional cryptographic algorithms. For example, the IKEv2 protocol, which is used to establish VPNs, could become vulnerable to decryption by powerful quantum computers. Postquantum preshared keys address this risk. They extend IKEv2 by using additional keys in the derivation process. This approach ensures that encrypted communications remain secure, even as cryptographic threats increase.

If the preshared keys contain sufficient entropy, session keys derived from them are resistant to quantum attacks. As a result, the system is secure against both modern classical attackers and future quantum attackers.

RFC 8784 (Mixing Preshared Keys in IKEv2 for Postquantum Security) specifies how IKEv2 can use PPKs for quantum resistance, enabling PPK negotiation, PPK ID transmission, integration into session key derivation, and fallback to sessions not using PPKs.

Figure 9: IKEv2 Key Derivation - With and Without PPK



DH: Diffie-Hellman
ECDH: Elliptic-curve Diffie-Hellman
PPK: Postquantum Preshared Key

Dynamic postquantum preshared keys

A dynamic postquantum preshared key is a cryptographic key that

- is imported by encryption devices from external key sources using the Cisco Secure Key Integration Protocol (SKIP),
- enables automated provisioning and periodic updates of preshared keys, and
- provides improved entropy for greater quantum safety compared to static preshared keys.

Dynamic postquantum preshared keys are typically provisioned from sources such as Quantum Key Distribution (QKD) devices, specialized software, or cloud-based key services. These keys support quantum-safe session key generation when used with protocols like IKEv2 and OTNsec.

Cisco Secure Key Integration Protocol

A Cisco Secure Key Integration Protocol is a security protocol that

- uses HTTPS as a transport to securely import preshared keys (PPKs) into Cisco encryption devices,
- acts as a client on encryption devices and a server on external key sources to enable automated and coordinated key provisioning, and
- provides reliable out-of-band synchronization, ensuring both initiator and responder devices receive identical key material.

Externally imported PPKs via SKIP are called dynamic PPKs. For successful integration, encryption devices must implement the SKIP client, and external sources (such as Quantum Key Distribution [QKD] devices or Cisco Session Key Service [SKS] servers) must implement the SKIP server.

SKIP compliance requirements

To be SKIP-compliant, an external key source must:

- implement the SKIP protocol or API as specified in the Cisco SKIP specification,

- provide the same preshared key (PPK) to both the initiator and responder devices using a reliable out-of-band synchronization mechanism,
- contact Cisco for technical guidance during implementation, especially for vendors supplying Quantum Key Distribution (QKD) or third-party solutions.

Workflow for dynamic postquantum preshared keys

Dynamic postquantum preshared keys rely on secure orchestration among encryption devices and their key sources, leveraging SKIP for automated provisioning, synchronization, and quantum-safe session key generation.

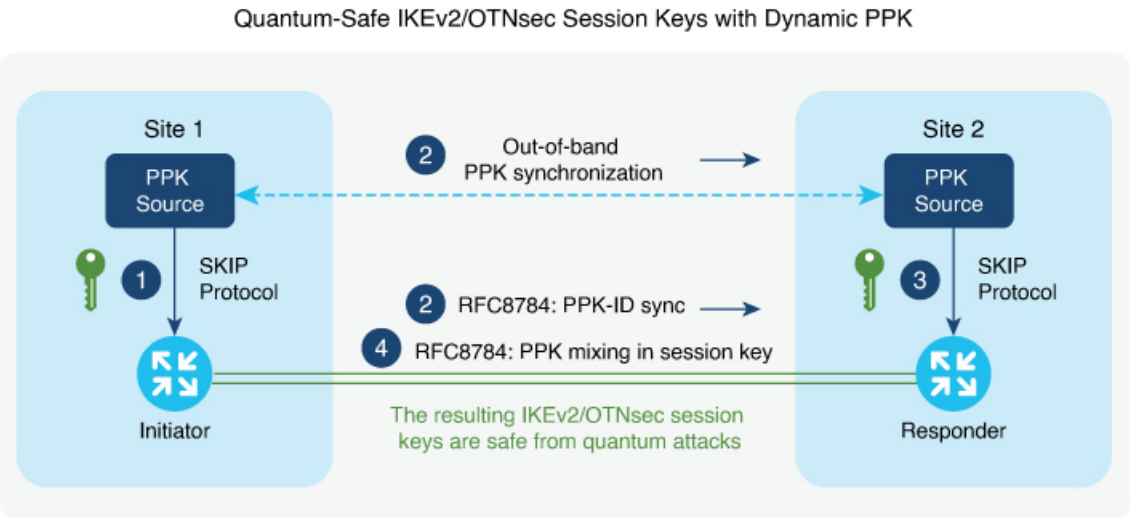
Summary

The key components involved in the process are:

- IKEv2 initiator: Requests and uses PPKs, communicating with the peer during session establishment.
- IKEv2 responder: Uses synchronized PPKs to ensure secure and matching session derivation.
- External key sources (SKIP server): Provide PPKs to devices and synchronize key material out-of-band.
- SKIP client: Runs on encryption devices to interact securely with the key source.

Workflow

Figure 10: Quantum-safe IKEv2 and OTNsec session keys with dynamic PPK



Thiw figure shows quantum-safe IKEv2 and OTNsec session keys using dynamic PPK.

The process involves the following stages:

1. The IKEv2 initiator requests a preshared key (PPK) from its configured external key source using SKIP. The key source replies with a PPK and associated PPK ID.
2. The initiator's key source synchronizes the PPK to the responder's key source out-of-band, based on the type of key source. Simultaneously, the initiator communicates the PPK ID to the responder via IKEv2 using RFC 8784 extensions.

3. The responder requests the correct PPK from its own key source using the received PPK ID and obtains the matching PPK.
4. Both initiator and responder mix the PPK into the key derivation process as specified in RFC 8784, generating quantum-safe IKEv2 and OTNsec session keys.

Result

The process ensures that both participating devices use a synchronized cryptographic key with enhanced entropy, enabling secure, quantum-resistant session establishment.

Configuring Dynamic PPK using SKS SKIP

Use the following commands to configure the dynamic PPK for one or more peers or groups of peers, in the IKEv2 keyring.

configure terminal

keyring *dynamic*

peer *name*

ppk dynamic *sks-profile-name* **[required]**

pre-shared-key *key-string*

address {*ipv4-address mask*}

ikev2 profile *name*

match identity remote address {*ipv4-address mask*}

keyring ppk *keyring-name*

keyring *keyring-name*

sks profile *profile-name* **type remote**

kme server ipv4 *ip-address* **port** *port-number*

exit

exit

Example :

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring dynamic
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#ppk dynamic qkd required
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#pre-shared-key cisco123!cisco123
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#address 10.0.0.1 255.0.0.0
RP/0/1/CPU0:ios(config)#ikev2 profile test
RP/0/1/CPU0:ios(config-ikev2-profile-test)#keyring dynamic
RP/0/1/CPU0:ios(config-ikev2-profile-test)#keyring ppk dynamic
RP/0/1/CPU0:ios(config-ikev2-profile-name)#match address 10.0.0.1 255.255.255.0


RP/0/1/CPU0:ios(config)#sks profile qkd type remote
RP/0/1/CPU0:ios(config-sks-profile)#kme server ipv4 192.0.2.34 port 10001
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Manual postquantum preshared keys

A manual postquantum preshared key is a type of PPK that

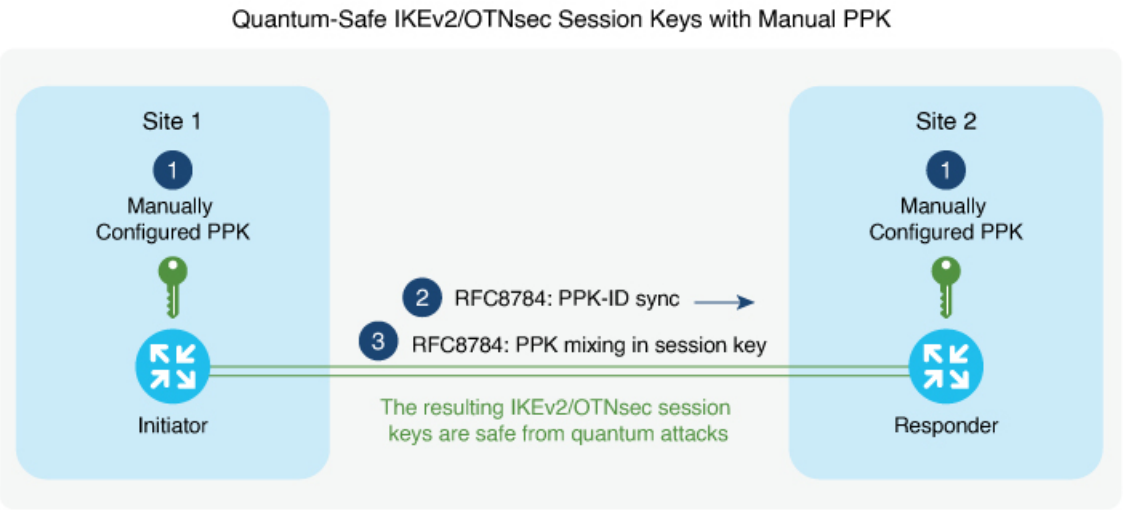
- is manually configured with identical values on both the IKEv2 and OTNsec initiator and responder,
- provides an easier alternative to dynamic PPKs, and
- requires administrator intervention for setup, maintenance, and rotation.

Manual PPK requirements

When using manual PPKs, you can provision the same PPKs on both the IKEv2 and OTNsec initiator and responder by manually configuring the PPKs on both sides. Ensure that a manual PPK is of sufficient size and entropy, and that it is frequently rotated by the administrator to maintain security.

This figure illustrates that the session keys of quantum-safe IKEv2 and OTNsec are obtained through a manual PPK:

Figure 11: Quantum-safe IKEv2 and OTNsec session keys with manual PPK



Configure a manual PPK in an IKEv2 keyring

Define a manual PPK for one or more IKEv2 peers or peer groups to enhance VPN authentication security.

Manually configuring PPKs in IKEv2 keyrings allows for more flexible peer authentication in secure VPN deployments. From Release 24.3.1, Type 6 passwords are supported for preshared keys, providing enhanced protection. See Enable Type 6 password, on page 303.

Follow these steps to configure a manual PPK in an IKEv2 keyring:

Procedure

Step 1 Access global configuration mode and create or enter an IKEv2 keyring using the **keyring** *keyring-name* command.

Example:

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring manual
```

Step 2 Define the peer and configure the manual PPK and preshared key using the keywords **peer** *name* **key [clear | password | password6]** *password* **[required]**.

Example:

```
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#ppk manual id cisco123 key password 060506324F41584B56
 required
```

Step 3 Define the pre-shred-key and peer adress using the keywords **pre-shared-key** *key-string* **address** {*ipv4-address mask* }

Example:

```
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#pre-shared-key cisco123cisco123
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#address 10.0.0.1 255.0.0.0
```

Step 4 Exit to global configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Step 5 Create or modify the IKEv2 profile, associating the keyring and matching remote peer identity using the keywords **ikev2 profile** *name***match identity remote address** {*ipv4-address mask*}**keyring ppk** *keyring-name* **keyring** *keyring-name*

Step 6 Exit to global configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

These are the examples where the PPK and the preshared key are configured with the `clear` keyword, after enabling the Type 6 method:

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring type6_psk
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#pre-shared-key clear cisco123
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#address 10.0.0.1 255.0.0.0
```

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring type6_ppk
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#ppk manual id 123 key clear cisco123 required
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#pre-shared-key
```

These are the examples where the PPK and preshared key are configured with the `password6` keyword, after enabling the Type 6 method:

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring type6_psk
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#pre-shared-key
password6 525548665b4e534660504c54645d63526668604945635a6452604a5f644d605a5c4461644d4e444e6566414142
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#address 10.0.0.1 255.0.0.0
```

```
RP/0/RP0/CPU0:ios#configure terminal
RP/0/RP0/CPU0:ios(config)#keyring type6_ppk
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#peer peer1
```

```
RP/0/RP0/CPU0:ios(config-ikev2-keyring-peer)#ppk manual id 123 key password6
 426447414e60494d48655d434f4749525d69484f434d445850675258544d56444a5d4d5b664b4c55624e414142 required
RP/0/RP0/CPU0:ios(config-ikev2-keyring)#pre-shared-key
```

Note
When using the `password6` keyword for PPK and preshared key, you must use an encrypted key. You can use the encrypted form of the `clear` PPK and preshared key that you previously configured (retrieve the encrypted form using the **show running-config keyring** command), as long as the primary key used to enable the type 6 password remains the same.

Type 6 password support for preshared keys in IKEv2 authentication

A Type 6 password support is an encryption method that

- uses AES256-GCM (Advanced Encryption Standard 256-bit Galois/Counter Mode) encryption,
- applies a user-configured primary key to encrypt preshared keys and postquantum preshared keys (PPK) in IKEv2 keyring configuration, and
- stores the primary key in the Trusted-Anchor-Module (TAM) so that it is never displayed in the device configuration.

Feature history

Table 65: Feature history

Feature name	Release information	Description
Type 6 Password Support for Keyring Configuration used in IKEv2	Release 24.3.1	The keyring configuration for IKEv2 encryption now supports the Type 6 password encryption method. This method uses AES256-GCM encryption and a user-configured primary key to encrypt the preshared key and postquantum preshared keys (PPK). This Type 6 encryption enhances security by storing sensitive information, such as the preshared key, in an encoded format on the device and makes it difficult to decipher. You can enable this feature using the following commands: • **key config-key password-encryption** • **password6 aes encryption** You can verify the status of the encryption using the following command: • **show type6 server**

Type 6 password security benefits

The Type 6 password support feature enhances security by encoding sensitive information, such as the preshared key and PPK, in configurations for IKEv2 encryption. The primary key is securely stored in the Trusted-Anchor-Module and remains inaccessible through configuration display, ensuring the encrypted data is highly protected against unauthorized access.

From Release 24.3.1, Type 6 password support is available for both preshared keys and PPKs in IKEv2 keyring configurations. The use of AES256-GCM encryption, in combination with a user-configured primary key, prevents exposure of sensitive key material by encoding it in the device and never displaying the key in clear text.

Enable Type 6 password

Enable Type 6 password encryption, which uses AES to secure device credentials.

Context:

Type 6 password encryption provides strong protection for preshared keys and pre-shared PPKs (Pairwise Primary Keys) on your device, using the Advanced Encryption Standard (AES). This ensures credentials are not stored as plain text and are not directly visible in the configuration.

Follow these steps to enable Type 6 password.

Procedure

Step 1 Use the **password6 encryption aes** command to enable the AES encryption and save the changes.

Example:

```
RP/0/RP0/CPU0:ios(config)#password6 encryption aes
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Use the **key config-key password-encryption** command to create the primary key.

Example:

```
P/0/0/CPU0:ios#key config-key password-encryption
Thu Jul 11 09:40:47.396 UTC
New password Requirements: Min-length 6, Max-length 64
Enter new key :
Enter confirm key :
Master key operation is started in background
```

Step 3 Use the same command **key config-key password-encryption** again, if you want to update the primary key. When it prompts for an old key, input the old key and then enter the new key that you want to configure.

```
RP/0/RP0/CPU0:ios#key config-key password-encryption
Thu Jul 11 09:40:47.396 UTC
New password Requirements: Min-length 6, Max-length 64
Enter old key :
Enter new key :
Enter confirm key :
Master key operation is started in background
```

If you forget the old key while updating the primary key, use the command **key config-key password-encryption delete** to delete the old key and create a new one. Make sure that you disable **password6 encryption aes** before deleting the primary key.

Example:

```
RP/0/RP0/CPU0:ios#key config-key password-encryption delete
Thu Jul 11 09:42:39.612 UTC
Disable 'password6 encryption aes' before deleting master key, Masterkey delete Failed
RP/0/RP0/CPU0:ios#con
Thu Jul 11 10:01:47.056 UTC
RP/0/RP0/CPU0:ios(config)#no password6 encryption aes
RP/0/RP0/CPU0:ios(config)#commit
Thu Jul 11 10:01:57.755 UTC
RP/0/RP0/CPU0:ios(config)#end
RP/0/RP0/CPU0:ios#key config-key password-encryption delete
Thu Jul 11 10:02:02.238 UTC
WARNING: All type 6 encrypted keys will become unusable
Continue with master key deletion ? [yes/no]:yes
Master key operation is started in background
```

Enable the AES encryption and configure the primary key again.

Step 4 Use the **show type6 server** command to verify the status of the Type 6 password encryption information.

Example:

```
RP/0/RP0/CPU0:ios#show type6 server
Thu Jul 11 09:43:36.103 UTC
Server detail information:
==========================================
AES config State       :             Enabled
Masterkey config State :             Enabled
Type6 feature State    :             Enabled
Master key Inprogress  :             No
Masterkey Last updated/deleted :     Thu Jul 11 09:40:57 2024
```

When the key is created, it is stored internally; not as part of the NCS 1004 device configuration. The device does not display the primary key as part of the running configuration. So, you cannot see or access the primary key when you connect to the device.

Step 5 (Optional) Use the nonvolatile generation (NVGEN) command, **nvgen-default-sanitize passwords**, if you want to mask completely the preshared key in the show command output.

Example:

```
RP/0/0/CPU0:ios(config)#nvgen-default-sanitize passwords
RP/0/0/CPU0:ios(config)#commit
```

Step 6 (Optional) Use the command **show running-config keyring** to view encrypted keys.

The configured primary key and AES encryption encrypt the preshared key and PPK.

Example:

```
RP/0/RP0/CPU0:ios#show running-config keyring
Thu Jul 11 09:42:43.085 UTC
keyring keyring type6_psk
 peer link-1
  pre-shared-key password6 62415744534e564365625c544e5b68594d4158515d4f6768436845556747414142
  address 1.1.1.2 255.255.255.0
```

```
RP/0/RP0/CPU0:ios#show running-config keyring
Thu Jul 11 09:42:43.085 UTC
keyring type6_ppk
peer 1
ppk manual id 123 key password6
4d49525f5848444b535b6252435847646369526 52644e64414952415641554655635e5f4c56424d46455c65414142 required
!
```

```
RP/0/0/CPU0:ios(config)#show running-config keyring
Thu Jul 11 09:45:37.193 UTC
keyring keyring_all_in_one
 peer link-1
  pre-shared-key password6 <removed>
  address 1.1.1.2 255.255.255.0
  !
!
```

Verify the PPK configuration

Ensure that PPK (Post-Quantum Key) features are correctly configured and active on your device for quantum-safe encryption.

Follow these steps to verify the PPK configuration.

Procedure

Step 1 Use the **show ikev2 sa detail** command to display information about the current IKEv2 security associations.

Example:

```
RP/0/1/CPU0:ios#show ikev2 sa detail
IKE SA ID : 866
------------------------------------------------------------------------------
Local : 192.0.2.34/500
Remote : 192.0.2.40/500
Status(Description) : READY (Negotiation done)
Role : Initiator
Fvrf : Default
Encryption/Keysize : AES-CBC/256
PRF/Hash/DH Group : SHA512/SHA512/19
Authentication(Sign/Verify) : PSK/PSK
Life/Active Time(sec) : 86400/21
Session ID : 5
Local SPI : C18D2946B0C4259C
Remote SPI : 5D1BD398AEB3A1E1
Local ID : 192.0.2.34
Remote ID : 192.0.2.40
Quantum resistance : Enabled with manual PPK
```

The **Quantum resistance** parameter in the output of the command indicates that manual PPK-based quantum-safe encryption is enabled.

Note
Both manual and dynamic PPK options can be used for viewing IKEv2 details.

Step 2 Use the **show ikev2 statistics** command to display the statistics and counters related to IKEv2 sessions

Example:

```
RP/0/1/CPU0:ios#show ikev2 statistics
Thu Jun 8 13:30:06.360 IST
.............................
NO_NAT : 2 0 0 0
PPK COUNTERS
=============================================================================================
PPK ERRORS
---------------------------------------------------------------------------------------------
PPK_ID_MISMATCH : 0
PPK_RETRIEVE_FAIL : 0
PPK_AUTH_FAIL : 0
```

Step 3 Use the **show ikev2 summary** command to display the IKEv2 session summary of NCS 1004.

Example:

```
RP/0/1/CPU0:ios#show ikev2 summary
Thu Jun 8 12:54:30.969 IST
IKEv2 SA Summary
-----------------------------------------------
Total SA (Active/Negotiating) : 2 (2/0)
Total Outgoing SA (Active/Negotiating): 2 (2/0)
Total Incoming SA (Active/Negotiating): 0 (0/0)
Total QR SA (Dynamic/Manual) : 2 (1/1)
```

Step 4 Use the **show ikev2 profile** command to display the details for each IKEv2 profile.

Example:

```
RP/0/1/CPU0:ios#show ikev2 profile
Tue Jun 6 18:00:20.277 IST
Profile Name : p4
================================================================================
Keyring : k4
Fvrf : Default
Lifetime(Sec) : 86400
DPD Interval(Sec) : 4
DPD Retry Interval(Sec) : 2
Match ANY : NO
Total Match remote peers : 1
Addr/Prefix : 198.51.100.19/255.255.255.0
Number of Trustpoints : 0
Local auth method : PSK
Number of remote auth methods : 1
Auth Method : PSK
PPK Keyring : Not Configured
Profile Name : ppk_d
================================================================================
Keyring : Not Configured
Fvrf : Default
Lifetime(Sec) : 86400
DPD Interval(Sec) : 4
DPD Retry Interval(Sec) : 2
Match ANY : NO
Total Match remote peers : 0
Number of Trustpoints : 0
Local auth method : NULL
Number of remote auth methods : 0
PPK Keyring : ppk_d
Profile Name : ppk_m
================================================================
Keyring : Not Configured
Fvrf : Default
Lifetime(Sec) : 86400
DPD Interval(Sec) : 4
DPD Retry Interval(Sec) : 2
Match ANY : NO
Total Match remote peers : 0
Number of Trustpoints : 0
Local auth method : NULL
Number of remote auth methods : 0
PPK Keyring : ppk_m
```

Step 5 Use the show keyring command to display the configured keyring details on NCS 1004.

Example:

```
RP/0/1/CPU0:ios#show keyring
Tue Jun 6 18:00:28.272 IST
Keyring Name : k4
================================================================================
Total Peers : 1
--------------------------------------------------------------------------------
Peer Name : init
IP Address : 198.51.100.19
Subnet Mask : 255.255.255.0
Local PSK : Configured
Remote PSK : Configured
PPK Mode : Not Configured
PPK Mandatory : Not Configured
Keyring Name : ppk_m
================================================================================
```

```
Total Peers : 1
--------------------------------------------------------------------------------
Peer Name : init
IP Address : Not Configured
Subnet Mask : Not Configured
Local PSK : Not Configured
Remote PSK : Not Configured
PPK Mode : Manual
PPK Mandatory : No
Keyring Name : ppk_m_req
================================================================================
Total Peers : 1
--------------------------------------------------------------------------------
Peer Name : init
IP Address : Not Configured
Subnet Mask : Not Configured
Local PSK : Not Configured
Remote PSK : Not Configured
PPK Mode : Manual
PPK Mandatory : Yes
Keyring Name : ppk_d
================================================================================
Total Peers : 1
--------------------------------------------------------------------------------
Peer Name : init
IP Address : Not Configured
Subnet Mask : Not Configured
Local PSK : Not Configured
Remote PSK : Not Configured
PPK Mode : Dynamic
PPK Mandatory : No
Keyring Name : ppk_d_req
================================================================================
Total Peers : 1
--------------------------------------------------------------------------------
Peer Name : init
IP Address : Not Configured
Subnet Mask : Not Configured
Local PSK : Not Configured
Remote PSK : Not Configured
PPK Mode : Dynamic
PPK Mandatory : Yes
```

Step 6 Use the show ikev2 session detail command to display information about the current IKEv2 session.

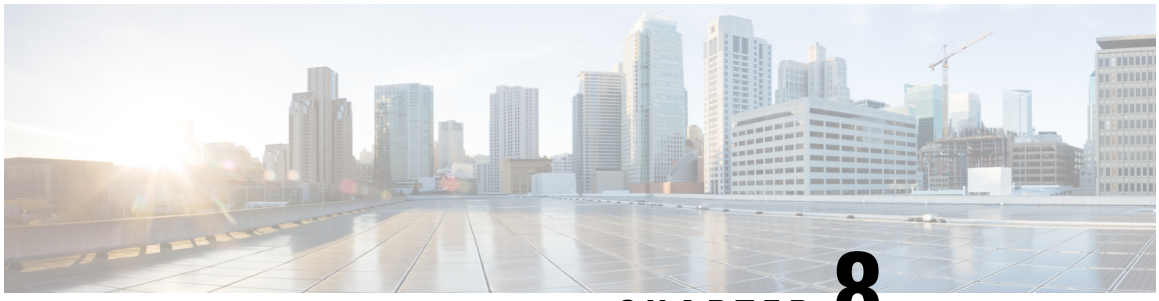
Example:

```
RP/0/1/CPU0:ios#show ikev2 session detail
Fri Feb 2 11:21:09.131 IST
Session ID : 3
================================================================================
Status : UP-ACTIVE
IKE Count : 1
Child Count : 1
IKE SA ID : 11625
--------------------------------------------------------------------------------
Local : 192.0.2.3/500
Remote : 192.0.2.1/500
Status(Description) : READY (Negotiation done)
Role : Initiator
Fvrf : Default
Encryption/Keysize : AES-CBC/256
PRF/Hash/DH Group : SHA512/SHA512/19
Authentication(Sign/Verify) : PSK/PSK
Life/Active Time(sec) : 200/115
```

```
Session ID : 3
Local SPI : E8F0716FF44EA1C3
Remote SPI : B1046E13B805178E
Local ID : 192.0.2.3
Remote ID : 192.0.2.1
Quantum resistance : Enabled with manual PPK
Child SA
--------------------------------------------------------------------------------
Local Selector : 0.0.0.0/0 - 255.255.255.255/65535
Remote Selector : 0.0.0.0/0 - 255.255.255.255/65535
ESP SPI IN/OUT : 0xf5e2a1c2 / 0x12bb94fd
Encryption : AES-CBC
Keysize : 256
ESP HMAC : SHA384
```

What to do next

-



CHAPTER 8

GMPLS UNI for Packet and Optical Integration

- GMPLS UNI, on page 311

GMPLS UNI

Generalized Multiprotocol Label Switching User Network Interface (GMPLS UNI) is a control plane technology that

- enables packet networks to interact directly with the optical transport control plane,
- coordinates packet network resource requirements with the optical transport network, and
- uses open standards to optimize network resource use across packet and optical domains.

Table 66: Feature History

Feature Name	Release Information	Feature Description
GMPLS UNI Support for OTN-XP and 2-QDD-C Cards	Cisco IOS XR Release 7.10.1	Generalized Multiprotocol Label Switching (GMPLS) User Network Interface (UNI) support is enabled for OTN-XP and 2-QDD-C cards in NCS 1004. GMPLS UNI helps in optimizing the utilization of network resources. For OTN-XP card the following data paths are allowed. • 2x100 - 200G MXP • 4x100 - 400G MXP • 40x10 - 400G MXP • 20x10 - 200G MXP For 2-QDD-C card only 200G/300G/400G trunk rates are allowed with 100GE or OTU4 client payloads in both the muxponder and muxponder slice configurations.

Packet and optical integration

As cloud services become central to business operations, packet and optical network services must evolve to be more efficient and dynamic. Closer integration of packet and optical networks is critical, especially in the control plane. GMPLS UNI is a key enabling technology that supports this integration.

Channel spacing

The Dense Wavelength Division Multiplexing (DWDM) grid in the C-band (1530–1569 nm) of the optical spectrum is divided into multiple channels so that each channel can carry traffic independently. The number of channels available from the DWDM grid depends on the channel spacing: lower channel spacing yields more channels, and higher channel spacing yields fewer channels.

GMPLS supports two types of channel spacing.

- Fixed grid channel spacing: The channel spacing is fixed at 50 GHz and supports 100 Gbps and 200 Gbps traffic.
- Flexible grid channel spacing: The channel spacing is 6.25 GHz and supports all data rates. The channels are divided into finer slices so that the channel width can be defined as a multiple of these slices to fit the data bandwidth requirement.

  NCS 1004 supports only flexible grid channel spacing.

  The **neighbor flexi-grid-capable** command enables GMPLS UNI flexible grid channel spacing. This command is run during the LMP configuration.

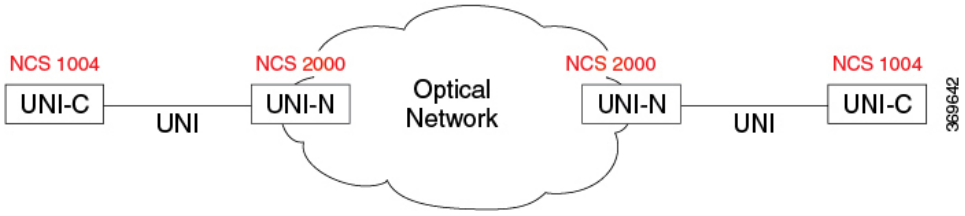
GMPLS UNI tunnels

GMPLS UNI tunnels are end-to-end optical paths between two NCS 1004 nodes that

- connects two nodes,
- efficiently uses the Dense Wavelength Division Multiplexing (DWDM) grid with minimal wastage of spectral bandwidth, and
- carries mixed bit-rate or mixed modulation data in a flexible grid using different channel widths.

To address common packet and optical network requirements, you create a tunnel between two NCS 1004 nodes to carry traffic by using GMPLS UNI technology, as shown in the following figure.

Figure 12: GMPLS UNI reference model



UNI-C is the client, packet, or router node, for example, an NCS 1004 node. UNI-N is the network or optical node, for example, an NCS 2000 node.

A Link Management Protocol (LMP) link is created to establish connectivity between an NCS 2000 node and an NCS 1004 node. The tunnel is then created between the trunk interfaces of the source and destination NCS 1004 nodes to carry traffic. When the tunnel is created between NCS 1004 nodes, a circuit is internally created between the NCS 2000 nodes to perform path computation, restoration, and reversion functions.

The tunnel can be created between the source and destination NCS 1004 nodes without involving NCS 2000 nodes in the middle. However, restoration and reversion capabilities are provided only by NCS 2000 nodes that use GMPLS UNI.

Licensing and configuration prerequisites for GMPLS UNI tunnels

Ensure that all necessary software packages, licenses, network reachability, and operational port conditions are met on NCS 1002, NCS 1004, and NCS 2000 nodes before creating a tunnel using Generalized Multiprotocol Label Switching User Network Interface (GMPLS UNI), enabling successful tunnel deployment and operation.

Before you create a tunnel by using GMPLS UNI, ensure that the following prerequisites are met.

- The NCS 1004 node has both the MPLS and MPLS-TE packages installed. The package names are **ncs1004-mpls** and **ncs1004-mpls-te-rsvp**.
- The NCS 2000 node has a valid license for ROADM and Wavelength Switched Optical Network (WSON) support.
- The management IP addresses of the NCS 1004 and NCS 2000 nodes are reachable.
- The administrative state of the trunk port of the optics controller on the NCS 1004 node is not in the shutdown state.

Limitations for a GMPLS UNI tunnel

This reference enables you to accurately identify the supported trunk port types and software release restrictions for GMPLS UNI tunnels on NCS 1002 nodes, ensuring compliant and optimized tunnel configuration.

The following limitations apply when you create a GMPLS UNI tunnel.

Configuration workflow for a GMPLS UNI tunnel

Summary

Creating a GMPLS UNI tunnel involves the following actors.

- NCS 2000 node (UNI-N): Provides the optical transport network and runs the Link Management Protocol (LMP) endpoint, alien wavelength, and ifindex provisioning.
- NCS 1004 node (UNI-C): Acts as the client router that runs LMP, Resource Reservation Protocol (RSVP), and the MPLS traffic-engineering tunnel toward the destination node.

You complete the NCS 2000 node configuration first, then configure the source and destination NCS 1004 nodes to bring up the tunnel.

Workflow

These stages describe the GMPLS UNI tunnel configuration workflow.

1. On the NCS 2000 node, perform the following configurations:
   - Configure LMP and alien wavelength on the NCS 2000 node by using CTC.
   - Retrieve ifindex from the NCS 2000 node.
2. On the NCS 1004 node, perform the following configurations:
   - Configure LMP on the Cisco NCS 1002 or NCS 1004 node.
   - Configure the MPLS tunnel on the NCS 1002 or NCS 1004 node.
   - Configure RSVP on the NCS 1002 or NCS 1004 node.

Result

The GMPLS UNI tunnel is established between the source and destination NCS 1004 nodes, with the optical path computed by the NCS 2000 nodes.

Configure an LMP link and Alien Wavelength on an NCS 2000 node using CTC for signaled numbered circuit

Establish connectivity and enable advanced opUse this task to set up a static LMP link between a Cisco NCS 2000 node and either an NCS 1004 node. The CTC LMP creation wizard enables you to select source and destination endpoints, configure optical parameters, and set alien wavelength options. tical parameters and alien wavelength configurations for signaled numbered circuits between Cisco optical network nodes.

Use this task to set up a static LMP link between a Cisco NCS 2000 node and either an NCS 1002 or NCS 1004 node. The CTC LMP creation wizard enables you to select source and destination endpoints, configure optical parameters, and set alien wavelength options.

Procedure

Step 1 From the **View** menu, choose **Go to Network View**.

Step 2 Click the **Provisioning > LMP** tabs.

Step 3 Click **Create**.

The LMP Creation window appears.

Step 4 Click **Signaled** in the **Router Not Managed by CTC** area.

A wizard appears with the following options:

LMP Origination, **LMP Termination**, **Optical Parameters**, and **Alien Wavelength**

Step 5 In the LMP Origination screen of the wizard, provision these parameters:

- From the **Originating Node** drop-down list, choose the source node of the LMP.

  If the source node is Cisco NCS 1004, the destination node must be MSTP, and the other way round.
- From the **Local Interfaces** drop-down list, choose an available interface.
- Choose the Type, Shelf, Slot, and Port for Ingress Port Selection and Egress Port Selection.
- Choose **Numbered** interface.
- Enter the IP address of the source node in the **Interface IP** field.
- Set the mode of revertive restoration to either UNI-C or UNI-N. If the mode is set to UNI-C, the reversion of the circuit from the restored path to the original path is initiated by the UNI client that is connected to NCS 1004. If the mode is set to UNI-N, the reversion of the circuit is initiated by the DWDM network and can either be a manual revert or an auto revert.
- Enter the RSVP signaling interval and RSVP signaling missed values in the respective fields.
- Click **Next**.

Step 6 In the LMP Termination screen of the wizard, provision these parameters:

- From the **Terminating Node** drop-down list, choose the destination node of the LMP; for example, MSTP node.
- From the **Rx Port Selection** area, perform the following.
  - Choose the card type from the **Type** drop-down list.
  - Choose a shelf from the **Shelf** drop-down list.
  - Choose a source slot from the **Slot** drop-down list
  - Choose a port from the **Port** drop-down list.
- From the **Tx Port Selection** area, perform the following.

- Choose the card type from the **Type** drop-down list.
- Choose a shelf from the **Shelf** drop-down list.
- Choose a destination slot from the **Slot** drop-down list.
- Choose a port from the **Port** drop-down list

- Enter the IP address of the destination node in the **Interface IP** field.
- Set the mode of revertive restoration to either UNI-C or UNI-N. If the mode is set to UNI-C, the reversion of the circuit from the restored path to the original path is initiated by the UNI client that is connected. If the mode is set to UNI-N, the reversion of the circuit is initiated by the DWDM network and can be either a manual revert or an auto revert.
- Enter the remote Ifindex of NCS 1004 node (in decimals) in the **Remote If Index** field.
- Click **Next**.

Step 7 In the Optical Parameters screen of the wizard, provision these parameters:

- **Allow Regeneration**—When checked, the computed path traverses through the regeneration site only if the optical validation is not satisfied. You can regenerate a circuit that is created from the UNI interface. If a transparent path is feasible, the regenerator is not used.
- **UNI State**—Choose **Enable** or **Disable** from the UNI State drop-down list.

  The Enable state is used to configure the UNI interface for the circuits to pass through, between the router and the DWDM node. In the Disable state, the interface is configured but not active, and so the circuit activation is rejected. When the status is changed from Enable to Disable, all active circuits on the interface are deleted.
- **Description**—Enter the description of the UNI interface. The description can be up to 256 characters.
- **Label**—Enter an alphanumeric string. This label is a unique circuit identifier.
- **Validation**—Sets the optical validation mode.
  - **Full**—The circuit is created when the circuit validation result is greater than or equal to the acceptance threshold value.
  - **None**—The circuit is created without considering the acceptance threshold value. The Opt Valid column in the Circuits tab displays the value as **Not Valid**.
  - **Inherited**—The restoration circuit inherits the validation and acceptance threshold values from the primary circuit.
- **Acceptance threshold**—Sets the acceptance threshold value for the GMPLS circuit. The circuit is created if the actual acceptance threshold value is greater than, or equal to, the value set in this field.
  - Green—Indicates that the channel failure risk is 0%.
  - Yellow—Indicates that the channel failure risk is between 0% and 16%.
  - Orange—Indicates that the channel failure risk is between 16% and 50%.
  - Red—Indicates that the channel failure risk is greater that 50%.
- **Restoration**—Check this check box to enable the restoration of the GMPLS circuits on the UNI interface.

- **Revert**—Check this check box to enable the revert of the GMPLS circuits on the UNI interface.
- **Auto Revert**—Click this radio button to automatically revert the circuit from the restored path to the original path after the failure is fixed, WSON alarms are acknowledged, and the soak time expires.
- **Manual Revert**—Click this radio button to manually revert the circuit from the restored path to the original path after the failure is fixed, the WSON alarms are acknowledged, and the soak time expires.
- **Soak Time**—Enter the time (in hours, minutes, and seconds) in the Soak Time field that the circuit on the restored path waits before moving to the original path after the failure is fixed. The circuit reverts to the original path after the soak time expires. The soak time must be set only if both the **Restoration** and **Revert** check boxes are checked.

Step 8 Click **Next**.

Step 9 In the Alien wavelength screen of the wizard, provision these parameters.

- From the **Alien Wavelength** drop-down list, choose the alien wavelength class.
- From the **Trunk Selection** drop-down list, choose 100G, 200G, or 250G.
- From the **FEC** drop-down list, choose a valid value for forward error correction (FEC) mode. If an invalid FEC value is chosen, LMP link is created; however, the circuit creation fails.
- Click **Finish** to create an LMP link.

The LMP link with specified optical parameters and alien wavelength settings is created and listed in CTC, ready for signaled numbered circuit provisioning.

Configure an LMP link with Alien Wavelength on NCS 2000 using CTC for signaled unnumbered circuits

Establish connectivity between NCS 2000 and NCS 1002/1004 nodes using a signaled, unnumbered LMP link with Alien Wavelength support.

Use the LMP creation wizard in Cisco Transport Controller (CTC) to provision link parameters, optical settings, and Alien Wavelength options for interoperability between managed and unmanaged router nodes.

Procedure

Step 1 From the **View** menu, choose **Go to Network View**.

Step 2 Click the **Provisioning > LMP** tabs.

Step 3 Click **Create**.

The LMP Creation window appears.

Step 4 Click **Signaled** in the **Router Not Managed by CTC** area.

A wizard appears with options for **LMP Origination**, **LMP Termination**, **Optical Parameters**, and **Alien Wavelength**

Step 5 On the **LMP Origination** screen, provision these parameters:

- From the **Originating Node** drop-down list, choose the source node of the LMP.

- From the **Local Interfaces** drop-down list, choose an available interface.
- Choose the Type, Unit, and Port for Ingress Port Selection and Egress Port Selection.
- Choose **Unnumbered** interface.
- The IP address of the source node selected appears in the **IP** field.
- Set the mode of revertive restoration to UNI-N. If the mode is set to UNI-N, the reversion of the circuit is initiated by the DWDM network and can either be a manual revert or an auto revert.
- Click **Next**.

Step 6 On the LMP Termination screen of the wizard, provision these parameters:

- From **Interfaces Configuration**:

  Enter the NCS 1004 system IP address in the **System IP** field.
- Enter the IP address of the source node in the **Communication Channel** field.
- Enter the SNMP Ifindex value of optic trunk in the **Remote If Index** field.
- Click **Next**.

Step 7 In the Optical Parameters screen of the wizard, provision these parameters:

- **Allow Regeneration**—When checked, the computed path traverses through the regeneration site only if the optical validation is not satisfied. You can regenerate a circuit that is created from the UNI interface. If a transparent path is feasible, the regenerator is not used.
- **UNI State**—Choose **Enable** or **Disable** from the UNI State drop-down list.

  The Enable state is used to configure the UNI interface for the circuits to pass through, between the router and the DWDM node. In the Disable state, the interface is configured but not active, and so the circuit activation is rejected. When the status is changed from Enable to Disable, all active circuits on the interface are deleted.
- **Description**—Enter the description of the UNI interface like **Signal Unnumb LMP**. The description can be up to 256 characters.
- **Label**—Enter an alphanumeric string. This label is a unique circuit identifier.
- **Validation**—Sets the optical validation mode.
  - **Full**—The circuit is created when the circuit validation result is greater than or equal to the acceptance threshold value.
  - **None**—The circuit is created without considering the acceptance threshold value. The Opt Valid column in the Circuits tab displays the value as **Not Valid**.
  - **Inherited**—The restoration circuit inherits the validation and acceptance threshold values from the primary circuit.
- **Acceptance Threshold**—Sets the acceptance threshold value for the GMPLS circuit. The circuit is created if the actual acceptance threshold value is greater than, or equal to, the value set in this field.
  - Green—Indicates that the channel failure risk is 0%.
  - Yellow—Indicates that the channel failure risk is between 0% and 16%.

- Orange—Indicates that the channel failure risk is between 16% and 50%.
- Red—Indicates that the channel failure risk is greater than 50%.

- **Restoration**—Check this check box to enable the restoration of the GMPLS circuits on the UNI interface.
- **Revert**—Check this check box to enable the revert of the GMPLS circuits on the UNI interface.
- **Auto Revert**—Click this radio button to automatically revert the circuit from the restored path to the original path after the failure is fixed, WSON alarms are acknowledged, and the soak time expires.
- **Manual Revert**—Click this radio button to manually revert the circuit from the restored path to the original path after the failure is fixed, the WSON alarms are acknowledged, and the soak time expires.
- **Soak Time**—Enter the time (in hours, minutes, and seconds) in the Soak Time field that the circuit on the restored path waits before moving to the original path after the failure is fixed. The circuit reverts to the original path after the soak time expires. The soak time must be set only if both the **Restoration** and **Revert** check boxes are checked.

Step 8 Click **Next**.

Step 9 In the Alien wavelength screen of the wizard, provision these parameters.

- From the **Alien Wavelength** drop-down list, choose the alien wavelength class such as NCS 1004.
- From the **Trunk Selection** drop-down list, choose 100G, 200G, or 250G.
- From the **FEC** drop-down list, choose a valid value for forward error correction (FEC) mode. If an invalid FEC value is chosen, LMP link is created; however, the circuit creation fails.
- Click **Finish** to create an LMP link.

  The newly created signaled LMP unnumbered circuit link appears in the LMP table in CTC.

The LMP link between NCS 2000 and NCS 1002/1004 is established, supporting signaled unnumbered circuits with full optical and Alien Wavelength configuration.

Retrieve ifindex from the NCS 2000 node

You can retrieve the ifindex value of all the LMP ports of the NCS 2000 node by using the Cisco Transport Controller (CTC) or the TL1 interface. Use this value in the **neighbor interface-id unnumbered** command during the LMP configuration on the NCS 1004 node.

Procedure

Retrieve the ifindex by using either CTC or TL1.

- To use CTC, click the **Provisioning** > **LMP** tab and read the ifindex value, in decimal format, under the **Originating Interface Index** column.
- To use TL1, log in to the TL1 interface and run the **rtrv-unicfg ::all:1;** command. The command returns the ifindex of all the LMP ports of the NCS 2000 node in hexadecimal format. Convert the value to decimal format before you use it as the remote ifindex of the NCS 1004 node.

The following sample TL1 output shows the ifindex of two LMP ports.

Example:

```
PSLINE-81-1-9-RX:PSLINE-81-1-9-TX,10.77.142.92,3.3.3.4,3.3.3.3,0.0.0.0,VALMODE=NONE,ADMINSTATE=UP,
RESTTYPE=REVERT,USPWROFS=0.0,
DSPWROFS=0.0,ALLOWREGEN=NO,UNICTRLMODE=CLIENT,REVERTMODE=MANUAL,SOAK=00-01-00,
RESTVALMODE=NONE,TERMINTFDX=0,ORIGINTFIDX=7f000d12,NUMBERED=TRUE,UNIMODE=GMPLS

PSLINE-81-1-10-RX:PSLINE-81-1-10-TX,10.77.142.92,4.4.4.4,4.4.4.3,0.0.0.0,VALMODE=NONE,ADMINSTATE=UP,
RESTTYPE=REVERT,USPWROFS=0.0,DSPWROFS=0.0,ALLOWREGEN=NO,UNICTRLMODE=CLIENT,
REVERTMODE=MANUAL,SOAK=00-01-00,RESTVALMODE=NONE,TERMINTFDX=0,
ORIGINTFIDX=7f000d14,NUMBERED=TRUE,UNIMODE=GMPLS
```

In the sample output, the ifindex of port 81-1-9 is **7f000d12** in hexadecimal and **2130709778** in decimal. The ifindex of port 81-1-10 is **7f000d14** in hexadecimal and **2130709780** in decimal.

Configure LMP on Cisco NCS 1004 node

Set up LMP parameters to enable proper communication and management between optical network devices on a Cisco NCS 1004 node.

Use this task to configure trunk optics, neighbor link parameters, flexible grid capabilities, and associated identifiers for numbered or unnumbered circuits. This configuration ensures seamless interoperability between Cisco NCS 1004 and NCS 2000 nodes.

Procedure

Step 1 Enter the LMP GMPLS optical UNI configuration mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#lmp
RP/0/RP0/CPU0:ios(config-lmp)#gmpls optical-uni
```

Step 2 Specify the trunk optics controller.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-uni)#controller optics 0/1/0/0
```

Step 3 Configure the LMP neighbor name.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)#neighbor VEGA2K-Site-2_47
```

Step 4 Specify the neighbor link ID for numbered circuit.

The neighbor link ID is the IP address of the Multiservice Transport Platform (MSTP) interface on the NCS 2000 node.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)#neighbor link-id ipv4 unicast 192.168.2.3
```

Step 5 Enable GMPLS UNI flexible grid channel spacing.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)#neighbor flexi-grid-capable
```

Step 6 Specify the neighbor optical interface ID, which is the ifindex value of the LMP port of the NCS 2000 node in decimal format.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)#neighbor interface-id unnumbered 19
```

To retrieve the ifindex value, see Retrieve ifindex from the NCS 2000 node.

Step 7 Set the IPCC mode to routed.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-uni)#ipcc routed
```

Step 8 Specify the neighbor router ID for GMPLS UNI, and commit the configuration.

Example:

```
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)# router-id ipv4 unicast 192.168.2.1
RP/0/RP0/CPU0:ios(config-lmp-gmpls-cntl)#commit
```

Step 9 View the configuration.

Example:

```
show running-config lmp
lmp
gmpls optical-uni
  controller optics0/0/0/13
   neighbor NCS1K
   neighbor link-id ipv4 unicast 192.168.2.3
   neighbor flexi-grid-capable
   neighbor interface-id unnumbered 19
   link-id ipv4 unicast 192.168.2.4
  !
  neighbor NCS1K
   ipcc routed
   router-id ipv4 unicast 192.168.2.1
  !
  router-id ipv4 unicast 192.168.2.2
  !
!

Mon Jul  1 14:42:46.856 IST
lmp
 gmpls optical-uni
  controller Optics0/0/0/0
   neighbor ncs1k
   neighbor link-id ipv4 unicast 10.1.1.1
   neighbor flexi-grid-capable
   neighbor interface-id unnumbered 2130706976
   link-id ipv4 unicast 10.0.1.1
  !
  controller Optics0/0/0/1
   neighbor ncs1k
   neighbor link-id ipv4 unicast 10.1.3.3
   neighbor flexi-grid-capable
   neighbor interface-id unnumbered 2130707232
   link-id ipv4 unicast 10.0.3.3
  !
  controller Optics0/1/0/0
   neighbor ncs1k
   neighbor link-id ipv4 unicast 10.1.4.4
   neighbor flexi-grid-capable
   neighbor interface-id unnumbered 2130706964
```

```
   link-id ipv4 unicast 10.0.4.4
  !
  controller Optics0/1/0/1
   neighbor ncs1k
   neighbor link-id ipv4 unicast 10.1.5.5
   neighbor flexi-grid-capable
   neighbor interface-id unnumbered 2130706966
   link-id ipv4 unicast 10.0.5.5
  !
  neighbor ncs1k
   ipcc routed
   router-id ipv4 unicast 10.127.60.48
  !
  router-id ipv4 unicast 10.105.57.101
 !
!

show mpls traffic-eng tunnels optical-uni brief
Wed Sep 22 17:08:13.132 IST

                     TUNNEL NAME         DESTINATION      STATUS  STATE
         GMPLS-UNI-Optics0/3/0/1            10.24.1.1         up  up
         GMPLS-UNI-Optics0/0/0/1            10.34.1.1         up  up
Displayed 2 (of 2) heads, 0 (of 0) midpoints, 0 (of 0) tails
Displayed 2 up, 0 down, 0 recovering, 0 recovered heads
```

commit

The Cisco NCS 1004 node is configured with the required LMP parameters for optical trunk and neighbor link management, supporting both numbered and unnumbered circuits.

Configure the MPLS tunnel on the NCS 1004 node

Before you begin

Ensure that the administrative state of the trunk port of the optics controller on the NCS 1004 node is not in the shutdown state.

You configure the Multiprotocol Label Switching (MPLS) traffic-engineering tunnel under the GMPLS optical UNI controller. The destination IP address is the IP address of the optics controller on the destination NCS 1004 node.

The path option supports either an Explicit Route Object (ERO), which includes the routes to use through a specified list of nodes, or an Exclude Route Object (XRO), which excludes routes through a specified list of nodes.

Procedure

Step 1 Enter the MPLS traffic-engineering GMPLS optical UNI configuration mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#mpls traffic-eng
RP/0/RP0/CPU0:ios(config-mpls-te)#gmpls optical-uni
```

Step 2 Specify the optics controller.

Example:

```
RP/0/RP0/CPU0:ios(config-te-gmpls-uni)#controller optics 0/0/0/6
```

Step 3 Enter the tunnel-properties submode and set the tunnel ID.

Example:

```
RP/0
RP/0/RP0/CPU0:ios(config-te-gmpls-cntl)#tunnel-properties
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#tunnel-id 100
```

Step 4 Specify the destination IP address of the optics controller on the destination NCS 1004 node.

Example:

```
R
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#destination ipv4 unicast 100.20.20.20
```

Step 5 Set the path optiona and commit.

Example:

```
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#path-option 10 no-ero lockdown
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#commit
```

What to do next

The following sample shows the MPLS tunnel configuration on the source NCS 1004 node.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#mpls traffic-eng
RP/0/RP0/CPU0:ios(config-mpls-te)#gmpls optical-uni
RP/0/RP0/CPU0:ios(config-te-gmpls-uni)#controller optics 0/0/0/6
RP/0/RP0/CPU0:ios(config-te-gmpls-cntl)#tunnel-properties
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#tunnel-id 100
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#destination ipv4 unicast 100.20.20.20
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#path-option 10 no-ero lockdown
RP/0/RP0/CPU0:ios(config-te-gmpls-tun)#commit
```

The following sample shows the MPLS tunnel configuration on the destination NCS 1004 node.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#mpls traffic-eng
RP/0/RP0/CPU0:ios(config-mpls-te)#gmpls optical-uni
RP/0/RP0/CPU0:ios(config-te-gmpls-uni)#controller optics 0/0/0/6
RP/0/RP0/CPU0:ios(config-te-gmpls-uni)#commit
```

Configure RSVP on the NCS 1004 node

You configure the Resource Reservation Protocol (RSVP) with appropriate signaling timers on the source and destination NCS 1004 nodes of the tunnel.

Procedure

Step 1 Enter the global configuration mode and the RSVP configuration mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#rsvp
```

Step 2 Specify the optics controller.

Example:

```
RP/0/RP0/CPU0:ios(config-rsvp)#controller optics 0/0/0/6
```

Step 3 Set the out-of-band signaling refresh interval, in seconds.

Example:

```
RP/0/RP0/CPU0:ios(config-rsvp-cntl)#signalling refresh out-of-band interval 3600
```

Step 4 Set the maximum number of missed out-of-band signaling refreshes that are allowed before the session is removed, and commit.

Example:

```
RP/0/RP0/CPU0:ios(config-rsvp-cntl)#signalling refresh out-of-band missed 24
RP/0/RP0/CPU0:ios(config-rsvp-cntl)#commit
```

signalling refresh out-of-band missed *mis-count*

What to do next

The following sample shows the RSVP configuration on the source NCS 1004 node.

GMPLS UNI tunnel, RSVP, and LMP configuration commands and sample outputs

Use the show commands in the following table to verify the GMPLS UNI tunnel, RSVP, and LMP configuration.

Table 67: Show commands

Show command	Description
show mpls traffic-eng link-management optical-uni controller optics	Displays detailed GMPLS information of a specific optics controller.
show mpls traffic-eng link-management optical-uni	Displays detailed GMPLS information of all the optics controllers.
show mpls traffic-eng tunnels	Displays information about tunnels.
show mpls traffic-eng link-management optical-uni tabular	Displays detailed GMPLS information of all the optics controllers in tabular format.
show mpls traffic-eng tunnels tabular	Displays information about all the tunnels in tabular format.
show lmp gmpls optical-uni	Verifies the LMP configuration and state.

Show command	Description
show rsvp neighbors	Displays information about RSVP neighbors.

Sample outputs

This sample out of the commnad **show mpls traffic-eng link-management optical-uni controller optics command** displays detailed GMPLS information of a specific optics controller.

```
Mon Jul  1 20:05:27.209 IST
Optical interface: Optics0/0/0/0
  Overview:
    IM state: Up
    Child interface: : IM state Unknown
    OLM/LMP state: Up
    Optical tunnel state: up
  Connection:
    Tunnel role: Tail
    Tunnel-id: 15, LSP-id 3, Extended tunnel-id 10.105.57.100
    Tunnel source: 10.105.57.100, destination: 11.1.1.1
    Optical router-ids: Local: 10.105.57.101, Remote: 10.127.60.48
    Label source: UNI-N
    Upstream label:
      Optical label:
      Grid                : DWDM
      Channel spacing     : 6.25 GHz
      Identifier          : 0
      Channel Number      : -277
    Downstream label:
      Optical label:
      Grid                : DWDM
      Channel spacing     : 6.25 GHz
      Identifier          : 0
      Channel Number      : -277
    SRLG discovery: Disabled
    SRLG announcement: None
    Switching Type: lsc
    MTU: 9212
  Admission Control:
    Upstream: Admitted (LSP ID: 3)
    Downstream: Admitted (LSP ID: 3)
  OLM/LMP adjacency information:
    Adjacency status: Up
    Local:
      node ID: 10.105.57.101
      link interface ID: 10
      link ID: 11.1.1.1
    Neighbor:
      node ID: 10.127.60.48 (VEGA2K-Site-3_48)
      link interface ID: 2130706976
      link ID: 10.1.1.1
      IPCC: Routed to 10.127.60.48
  Optical capabilities:
    Controller type: DWDM
    Channel spacing: 6.25 GHz
    Default channel: 0
    784 supported channels:
      -303, -302, -301, -300, -299, -298, -297, -296
      -295, -294, -293, -292, -291, -290, -289, -288
      -287, -286, -285, -284, -283, -282, -281, -280
      -279, -278, -277, -276, -275, -274, -273, -272
      -271, -270, -269, -268, -267, -266, -265, -264
```

```
-263, -262, -261, -260, -259, -258, -257, -256
-255, -254, -253, -252, -251, -250, -249, -248
-247, -246, -245, -244, -243, -242, -241, -240
-239, -238, -237, -236, -235, -234, -233, -232
-231, -230, -229, -228, -227, -226, -225, -224
-223, -222, -221, -220, -219, -218, -217, -216
-215, -214, -213, -212, -211, -210, -209, -208
-207, -206, -205, -204, -203, -202, -201, -200
-199, -198, -197, -196, -195, -194, -193, -192
-191, -190, -189, -188, -187, -186, -185, -184
-183, -182, -181, -180, -179, -178, -177, -176
-175, -174, -173, -172, -171, -170, -169, -168
-167, -166, -165, -164, -163, -162, -161, -160
-159, -158, -157, -156, -155, -154, -153, -152
-151, -150, -149, -148, -147, -146, -145, -144
-143, -142, -141, -140, -139, -138, -137, -136
-135, -134, -133, -132, -131, -130, -129, -128
-127, -126, -125, -124, -123, -122, -121, -120
-119, -118, -117, -116, -115, -114, -113, -112
-111, -110, -109, -108, -107, -106, -105, -104
-103, -102, -101, -100, -99, -98, -97, -96
-95, -94, -93, -92, -91, -90, -89, -88
-87, -86, -85, -84, -83, -82, -81, -80
-79, -78, -77, -76, -75, -74, -73, -72
-71, -70, -69, -68, -67, -66, -65, -64
-63, -62, -61, -60, -59, -58, -57, -56
-55, -54, -53, -52, -51, -50, -49, -48
-47, -46, -45, -44, -43, -42, -41, -40
-39, -38, -37, -36, -35, -34, -33, -32
-31, -30, -29, -28, -27, -26, -25, -24
-23, -22, -21, -20, -19, -18, -17, -16
-15, -14, -13, -12, -11, -10, -9, -8
-7, -6, -5, -4, -3, -2, -1, 0
1, 2, 3, 4, 5, 6, 7, 8
9, 10, 11, 12, 13, 14, 15, 16
17, 18, 19, 20, 21, 22, 23, 24
25, 26, 27, 28, 29, 30, 31, 32
33, 34, 35, 36, 37, 38, 39, 40
41, 42, 43, 44, 45, 46, 47, 48
49, 50, 51, 52, 53, 54, 55, 56
57, 58, 59, 60, 61, 62, 63, 64
65, 66, 67, 68, 69, 70, 71, 72
73, 74, 75, 76, 77, 78, 79, 80
81, 82, 83, 84, 85, 86, 87, 88
89, 90, 91, 92, 93, 94, 95, 96
97, 98, 99, 100, 101, 102, 103, 104
105, 106, 107, 108, 109, 110, 111, 112
113, 114, 115, 116, 117, 118, 119, 120
121, 122, 123, 124, 125, 126, 127, 128
129, 130, 131, 132, 133, 134, 135, 136
137, 138, 139, 140, 141, 142, 143, 144
145, 146, 147, 148, 149, 150, 151, 152
153, 154, 155, 156, 157, 158, 159, 160
161, 162, 163, 164, 165, 166, 167, 168
169, 170, 171, 172, 173, 174, 175, 176
177, 178, 179, 180, 181, 182, 183, 184
185, 186, 187, 188, 189, 190, 191, 192
193, 194, 195, 196, 197, 198, 199, 200
201, 202, 203, 204, 205, 206, 207, 208
209, 210, 211, 212, 213, 214, 215, 216
217, 218, 219, 220, 221, 222, 223, 224
225, 226, 227, 228, 229, 230, 231, 232
233, 234, 235, 236, 237, 238, 239, 240
241, 242, 243, 244, 245, 246, 247, 248
```

```
      249, 250, 251, 252, 253, 254, 255, 256
      257, 258, 259, 260, 261, 262, 263, 264
      265, 266, 267, 268, 269, 270, 271, 272
      273, 274, 275, 276, 277, 278, 279, 280
      281, 282, 283, 284, 285, 286, 287, 288
      289, 290, 291, 292, 293, 294, 295, 296
      297, 298, 299, 300, 301, 302, 303, 304
      305, 306, 307, 308, 309, 310, 311, 312
      313, 314, 315, 316, 317, 318, 319, 320
      321, 322, 323, 324, 325, 326, 327, 328
      329, 330, 331, 332, 333, 334, 335, 336
      337, 338, 339, 340, 341, 342, 343, 344
      345, 346, 347, 348, 349, 350, 351, 352
      353, 354, 355, 356, 357, 358, 359, 360
      361, 362, 363, 364, 365, 366, 367, 368
      369, 370, 371, 372, 373, 374, 375, 376
      377, 378, 379, 380, 381, 382, 383, 384
      385, 386, 387, 388, 389, 390, 391, 392
      393, 394, 395, 396, 397, 398, 399, 400
      401, 402, 403, 404, 405, 406, 407, 408
      409, 410, 411, 412, 413, 414, 415, 416
      417, 418, 419, 420, 421, 422, 423, 424
      425, 426, 427, 428, 429, 430, 431, 432
      433, 434, 435, 436, 437, 438, 439, 440
      441, 442, 443, 444, 445, 446, 447, 448
      449, 450, 451, 452, 453, 454, 455, 456
      457, 458, 459, 460, 461, 462, 463, 464
      465, 466, 467, 468, 469, 470, 471, 472
      473, 474, 475, 476, 477, 478, 479, 480
    Controller SRLGs
      None
```

show mpls traffic-eng link-management optical-uni

Displays detailed GMPLS information of all the optics controllers. MPLS tunnels are not created when the optics controller is in the shutdown state. The state is shown as "Admin down". Run the **no shutdown** command under the optics controller to initiate the tunnel creation.

```
Mon Jul  1 20:00:42.108 IST

System Information:
  Optical Links Count: 1 (Maximum Links Supported 100)

Optical interface: Optics0/0/0/0
  Overview:
    IM state: Up
    Child interface: : IM state Unknown
    OLM/LMP state: Up
    Optical tunnel state: up
  Connection:
    Tunnel role: Tail
    Tunnel-id: 15, LSP-id 3, Extended tunnel-id 10.105.57.100
    Tunnel source: 10.105.57.100, destination: 11.1.1.1
    Optical router-ids: Local: 10.105.57.101, Remote: 10.127.60.48
    Label source: UNI-N
    Upstream label:
      Optical label:
      Grid                : DWDM
      Channel spacing     : 6.25 GHz
      Identifier          : 0
      Channel Number      : -277
    Downstream label:
      Optical label:
```

```
    Grid                : DWDM
    Channel spacing     : 6.25 GHz
    Identifier          : 0
    Channel Number      : -277
  SRLG discovery: Disabled
  SRLG announcement: None
  Switching Type: lsc
  MTU: 9212
Admission Control:
  Upstream: Admitted (LSP ID: 3)
  Downstream: Admitted (LSP ID: 3)
OLM/LMP adjacency information:
  Adjacency status: Up
  Local:
    node ID: 10.105.57.101
    link interface ID: 10
    link ID: 11.1.1.1
  Neighbor:
    node ID: 10.127.60.48 (VEGA2K-Site-3_48)
    link interface ID: 2130706976
    link ID: 10.1.1.1
    IPCC: Routed to 10.127.60.48
Optical capabilities:
  Controller type: DWDM
  Channel spacing: 6.25 GHz
  Default channel: 0
  784 supported channels:
    -303, -302, -301, -300, -299, -298, -297, -296
    -295, -294, -293, -292, -291, -290, -289, -288
    -287, -286, -285, -284, -283, -282, -281, -280
    -279, -278, -277, -276, -275, -274, -273, -272
    -271, -270, -269, -268, -267, -266, -265, -264
    -263, -262, -261, -260, -259, -258, -257, -256
    -255, -254, -253, -252, -251, -250, -249, -248
    -247, -246, -245, -244, -243, -242, -241, -240
    -239, -238, -237, -236, -235, -234, -233, -232
    -231, -230, -229, -228, -227, -226, -225, -224
    -223, -222, -221, -220, -219, -218, -217, -216
    -215, -214, -213, -212, -211, -210, -209, -208
    -207, -206, -205, -204, -203, -202, -201, -200
    -199, -198, -197, -196, -195, -194, -193, -192
    -191, -190, -189, -188, -187, -186, -185, -184
    -183, -182, -181, -180, -179, -178, -177, -176
    -175, -174, -173, -172, -171, -170, -169, -168
    -167, -166, -165, -164, -163, -162, -161, -160
    -159, -158, -157, -156, -155, -154, -153, -152
    -151, -150, -149, -148, -147, -146, -145, -144
    -143, -142, -141, -140, -139, -138, -137, -136
    -135, -134, -133, -132, -131, -130, -129, -128
    -127, -126, -125, -124, -123, -122, -121, -120
    -119, -118, -117, -116, -115, -114, -113, -112
    -111, -110, -109, -108, -107, -106, -105, -104
    -103, -102, -101, -100, -99, -98, -97, -96
    -95, -94, -93, -92, -91, -90, -89, -88
    -87, -86, -85, -84, -83, -82, -81, -80
    -79, -78, -77, -76, -75, -74, -73, -72
    -71, -70, -69, -68, -67, -66, -65, -64
    -63, -62, -61, -60, -59, -58, -57, -56
    -55, -54, -53, -52, -51, -50, -49, -48
    -47, -46, -45, -44, -43, -42, -41, -40
    -39, -38, -37, -36, -35, -34, -33, -32
    -31, -30, -29, -28, -27, -26, -25, -24
    -23, -22, -21, -20, -19, -18, -17, -16
    -15, -14, -13, -12, -11, -10, -9, -8
```

```
  -7, -6, -5, -4, -3, -2, -1, 0
  1, 2, 3, 4, 5, 6, 7, 8
  9, 10, 11, 12, 13, 14, 15, 16
  17, 18, 19, 20, 21, 22, 23, 24
  25, 26, 27, 28, 29, 30, 31, 32
  33, 34, 35, 36, 37, 38, 39, 40
  41, 42, 43, 44, 45, 46, 47, 48
  49, 50, 51, 52, 53, 54, 55, 56
  57, 58, 59, 60, 61, 62, 63, 64
  65, 66, 67, 68, 69, 70, 71, 72
  73, 74, 75, 76, 77, 78, 79, 80
  81, 82, 83, 84, 85, 86, 87, 88
  89, 90, 91, 92, 93, 94, 95, 96
  97, 98, 99, 100, 101, 102, 103, 104
  105, 106, 107, 108, 109, 110, 111, 112
  113, 114, 115, 116, 117, 118, 119, 120
  121, 122, 123, 124, 125, 126, 127, 128
  129, 130, 131, 132, 133, 134, 135, 136
  137, 138, 139, 140, 141, 142, 143, 144
  145, 146, 147, 148, 149, 150, 151, 152
  153, 154, 155, 156, 157, 158, 159, 160
  161, 162, 163, 164, 165, 166, 167, 168
  169, 170, 171, 172, 173, 174, 175, 176
  177, 178, 179, 180, 181, 182, 183, 184
  185, 186, 187, 188, 189, 190, 191, 192
  193, 194, 195, 196, 197, 198, 199, 200
  201, 202, 203, 204, 205, 206, 207, 208
  209, 210, 211, 212, 213, 214, 215, 216
  217, 218, 219, 220, 221, 222, 223, 224
  225, 226, 227, 228, 229, 230, 231, 232
  233, 234, 235, 236, 237, 238, 239, 240
  241, 242, 243, 244, 245, 246, 247, 248
  249, 250, 251, 252, 253, 254, 255, 256
  257, 258, 259, 260, 261, 262, 263, 264
  265, 266, 267, 268, 269, 270, 271, 272
  273, 274, 275, 276, 277, 278, 279, 280
  281, 282, 283, 284, 285, 286, 287, 288
  289, 290, 291, 292, 293, 294, 295, 296
  297, 298, 299, 300, 301, 302, 303, 304
  305, 306, 307, 308, 309, 310, 311, 312
  313, 314, 315, 316, 317, 318, 319, 320
  321, 322, 323, 324, 325, 326, 327, 328
  329, 330, 331, 332, 333, 334, 335, 336
  337, 338, 339, 340, 341, 342, 343, 344
  345, 346, 347, 348, 349, 350, 351, 352
  353, 354, 355, 356, 357, 358, 359, 360
  361, 362, 363, 364, 365, 366, 367, 368
  369, 370, 371, 372, 373, 374, 375, 376
  377, 378, 379, 380, 381, 382, 383, 384
  385, 386, 387, 388, 389, 390, 391, 392
  393, 394, 395, 396, 397, 398, 399, 400
  401, 402, 403, 404, 405, 406, 407, 408
  409, 410, 411, 412, 413, 414, 415, 416
  417, 418, 419, 420, 421, 422, 423, 424
  425, 426, 427, 428, 429, 430, 431, 432
  433, 434, 435, 436, 437, 438, 439, 440
  441, 442, 443, 444, 445, 446, 447, 448
  449, 450, 451, 452, 453, 454, 455, 456
  457, 458, 459, 460, 461, 462, 463, 464
  465, 466, 467, 468, 469, 470, 471, 472
  473, 474, 475, 476, 477, 478, 479, 480
Controller SRLGs
  None
```

show mpls traffic-eng link-management optical-uni tabular

Displays detailed GMPLS information of all the optics controllers in tabular format.

```
Mon Jul  1 15:10:50.472 IST

System Information:
  Optical Links Count: 4 (Maximum Links Supported 100)

                  State          LMP          GMPLS tunnel
Interface      Admin  Oper    adjacency    role  tun-id  state
-------------------------------------------------------------
     Op0/0/0/0    up    up        up       Tail      15     up
     Op0/0/0/1    up    up        up       Tail      16     up
     Op0/1/0/0    up    up        up       Tail      17     up
     Op0/1/0/1    up    up        up       Tail      18     up
```

show mpls traffic-eng tunnels

Displays information about tunnels.

```
Mon Jul  1 15:03:58.490 IST


LSP Tunnel 10.105.57.100 15 [5] is signalled, Signaling State: up
  Tunnel Name: ckt0/0/0/0 Tunnel Role: Tail
  Upstream label:
    Optical label:
    Grid                : DWDM
    Channel spacing     : 6.25 GHz
    Identifier          : 0
    Channel Number      : -277
  Downstream label:
    Optical label:
    Grid                : DWDM
    Channel spacing     : 6.25 GHz
    Identifier          : 0
    Channel Number      : -277
  Signalling Info:
    Src 10.105.57.100 Dst 11.1.1.1, Tun ID 15, Tun Inst 5, Ext ID 10.105.57.100
    Router-IDs: upstream   10.127.60.48
                local      10.105.57.101
    Priority:  7  7
    SRLGs: not collected
    Path Info:
      Incoming Address: 10.1.1.1
      Incoming:
      Explicit Route:
        No ERO

      Route Exclusions:
        No XRO
      Record Route: Disabled
      Tspec: avg rate=4294967033 kbits, burst=1000 bytes, peak rate=4294967033 kbits
      Session Attributes: Local Prot: Not Set, Node Prot: Not Set, BW Prot: Not Set
    Resv Info: None
      Record Route: Disabled
      Fspec: avg rate=4294967033 kbits, burst=1000 bytes, peak rate=4294967033 kbits
Displayed 0 (of 0) heads, 0 (of 0) midpoints, 1 (of 1) tails
Displayed 0 up, 0 down, 0 recovering, 0 recovered heads
```

show rsvp neighbors

Displays information about RSVP neighbors.

```
Mon Jul  1 14:58:48.888 IST
Global Neighbor: 10.127.60.48
  Interface Neighbor   Interface
  -------------------- ------------
  10.127.60.48         MgmtEth0/RP0/CPU0/0
```

show lmp gmpls optical-uni

Verifies the LMP configuration and state.

```
Mon Jul  1 14:55:35.492 IST

GMPLS Optical-UNI LMP Router ID: 10.105.57.101

LMP Neighbor
Name: ncs1k, IP: 10.127.60.48, Owner: GMPLS Optical-UNI
LMP: Disabled
 IPCC ID: 1, State Up
 LMP UDP port: 701
  Known via             : Configuration
  Type                  : Routed
  Destination IP        : 10.127.60.48
  Source IP             : 10.105.57.101

    Interface I/F     |  Lcl Interface ID | Lcl Link    ID | Interface LMP state
----------------------+-------------------+----------------+----------------------
         Optics0/1/0/1                   7         11.5.5.5                    Up
         Optics0/1/0/0                   6         11.4.4.4                    Up
         Optics0/0/0/1                  11         11.3.3.3                    Up
         Optics0/0/0/0                  10         11.1.1.1                    Up
```

GMPLS UNI tunnel troubleshooting commands and resolutions

Recommended show commands for troubleshooting GMPLS UNI tunnels

Collect and analyze the output of the following commands for any software issues on the GMPLS UNI tunnel.

- **show tech-support mpls traffic-eng file** *filename*
- **show tech-support mpls rsvp file** *filename*
- **show lmp clients**
- **show rsvp neighbors**
- **show mpls traffic-eng link-management optical-uni controller optics** *Rack/Slot/Instance/Port*
- **show mpls traffic-eng tunnels** *tunnel-id*

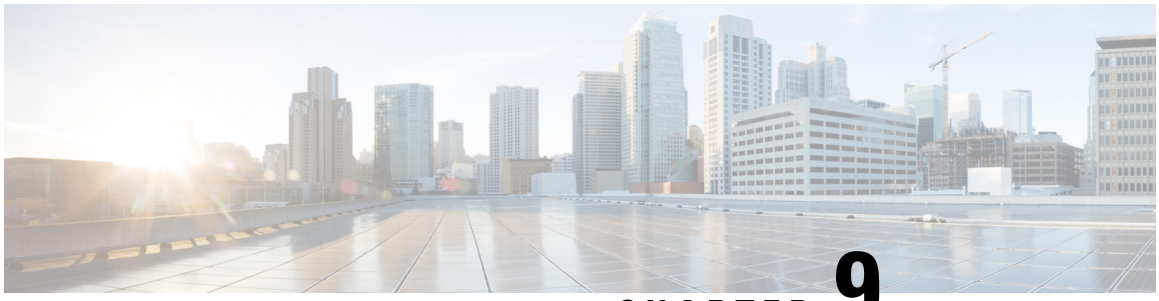
Table 68: GMPLS UNI tunnel issues and recommended resolutions

Problem	Solution
When the NCS 2000 node cannot route the Dense Wavelength Division Multiplexing (DWDM) wavelength to the destination, it displays a generic error message: "No Route to destination".	As a super user, collect and analyze the diagnostic information by entering the following address in the browser: `http://ip-address-of-head-node/diagnostics/wson`

Related information for GMPLS UNI

Refer to the following resources for additional information about GMPLS UNI.

- GMPLS UNI commands: Cisco IOS XR MPLS Command Reference.
- GMPLS Restoration and Reversion.



CHAPTER 9

Remote Node Management

- Remote node management using GCC , on page 333
- iBGP supports over GCC interfaces, on page 340

Remote node management using GCC

Remote node management using GCC is a network management method that

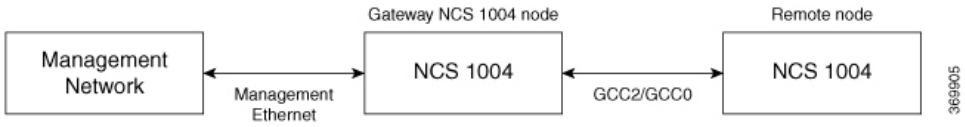
- leverages the General Communication Channel (GCC) embedded in optical transport networks,
- delivers reliable, out-of-band communication between centralized controllers and remote network nodes, and
- enables real-time monitoring, configuration, and maintenance activities without requiring direct physical access to each node.

Table 69: Feature History

Feature Name	Release Information	Feature Description
GCC Support for OTN-XP Card	Cisco IOS XR Release 7.3.2	The node supports a maximum of 48 GCC (GCC0 and GCC1) channels for each OTN-XP card.

From R7.2.1 onwards, the OTN-XP card provides OTU interface that supports communication channels between adjacent network elements or nodes using GCC bytes in the OTN header. Remote node management is supported over the GCC interface.

Figure 13: Remote Node Management in Linear Topology



The remote nodes can be dynamically discovered over the GCC interface using OSPF. The connectivity to the management network can be achieved using OSPF and static routes.

Note The GCC2 and GCC0 interfaces are supported in NCS 1004. The GCC0 interface is supported on the Coherent DSP controller whereas the GCC2 interface is supported on the ODU controller.

Note The GCC0 and GCC2 interfaces are supported in Muxponder and Muxponder slice modes. Only the GCC0 interface is supported in the Regeneration (Regen) mode.

From R7.2.1 onwards, the node supports GCC0 on the corresponding OTU2, OTU2e, and OTU4 interfaces. The node (Cisco FPGA) supports a maximum of 22 GCC channels for each card.

Note The GCC0 and GCC1 interfaces are supported on OTN-XP card and GCC2 interface is not supported.

From R7.3.1 onwards, the node supports GCC0 on the corresponding OTU2, OTU2e, OTU4, and Coherent DSP interfaces, and GCC1 on OTN ODU controller (ODU2, ODU2E, ODU4, and ODUCn).

From R7.3.2 onwards, the node (Cisco FPGA) supports a maximum of 48 GCC (GCC0 and GCC1) channels for each card.

Supported and unsupported features of the GCC interface

This table lists supported and unsupported features for remote node management using the GCC interface.

Table 70: Feature Support Overview:

Feature/Functionality	Supported on GCC Interface?	Notes
gRPC protocol	No	gRPC is not supported over the GCC interface.
Open Config	No	Not supported due to lack of gRPC support.
Streaming telemetry	No	Not supported due to lack of gRPC support.
Tx and Rx packet count statistics	Yes	Only Tx and Rx packet count information is available in GCC
Remote node management (after initial provisioning)	Yes	Devices can be managed over GCC only when connected through the management network using GCC.
Initial provisioning and bring-up via GCC	No	Must use console or management Ethernet interface for initial setup.

Feature/Functionality	Supported on GCC Interface?	Notes
Remote management after headless/HA event	May be impacted	Events like reloads or driver restarts at intermediate nodes may affect management of subsequent nodes.
IP fragmentation for SCP protocol	No	Not supported; reduce packet size to less than 1454 bytes as a workaround.
TCP MSS configuration to avoid fragmentation	Yes	Use `tcp mss <maximum segment size>` in global config mode.
Coherent DSP controller (QXP card) GCC0 interface	Yes	Supported on QXP card.
GCC0 speed on QXP card	Yes	7.7 Mbps.

Note For operations not supported on the GCC interface, use the console or management Ethernet interface as alternatives. For SCP protocol, configure TCP MSS or IPv4 MTU settings to avoid IP fragmentation issues.

Supported protocols

These protocols are supported over the GCC interface:

- PING
- SSH
- TELNET
- SCP
- TFTP
- FTP
- SFTP
- HTTP
- HTTPS
- OSPF

Enable the GCC interface

Use this task to enable GCC0, GCC1, or GCC2 interfaces on various line cards (1.2T, OTN-XP) to support management and communication channels.

Procedure

Step 1 Enter configuration mode.

Step 2 Configure the controller and the GCC interface for a line card.

If you want to configure	Then use the command
GCC2 interface on the 1.2T card	**controller odu4** *R/S/I/P/L* **gcc2**
GCC0 interface on the 1.2T card	**controller CoherentDSP** *R/S/I/P/L* **gcc2**
GCC0 interface for the OTN-XP card	**controller {otu2 | otu2e | otu4}** *R/S/I/P/L* **gcc0**
GCC1 interface for the OTN-XP card	**controller {odu2 | odu2e | odu4 | oducn}** *R/S/I/P/L* **gcc1**

Example:

This sample configuration enables the GCC2 interface for the 1.2T line card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller odu4 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-otu)#gcc2
RP/0/RP0/CPU0:ios(config-otu)#commit
RP/0/RP0/CPU0:ios(config-otu)#exit
```

This sample configuration enables the GCC0 interface for the 1.2T line card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller CoherentDSP0/0/1/1
RP/0/RP0/CPU0:ios(config-otu)#gcc0
RP/0/RP0/CPU0:ios(config-otu)#commit
RP/0/RP0/CPU0:ios(config-otu)#exit
```

This sample configuration enables the GCC0 interface for the OTN-XP line card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller otu2 0/0/0/4/1
RP/0/RP0/CPU0:ios(config-otu)#gcc0
RP/0/RP0/CPU0:ios(config-otu)#commit
RP/0/RP0/CPU0:ios(config-otu)#exit
```

This sample configuration enables the GCC1 interface for the OTN-XP line card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#controller odu2 0/0/0/4/1
RP/0/RP0/CPU0:ios(config-otu)#gcc1
RP/0/RP0/CPU0:ios(config-otu)#commit
RP/0/RP0/CPU0:ios(config-otu)#exit
```

Configure the GCC interface

Use this task to configure the GCC0, GCC1, and GCC2 interfaces on 1.2T and OTN-XP cards using static or loopback IP addresses.

Procedure

Step 1 Enter configuration mode.

Step 2 Specify the GCC2 interface for a line card.

If you want to configure	Then use the command
GCC2 interface on the 1.2T card	**interface gcc2** *R/S/I/P/L*
GCC0 interface on the 1.2T card	**interface gcc0** *R/S/I/P*
GCC0 interface on the OTN-XP card	**interface gcc0** *R/S/I/P*
GCC1 interface on the OTN-XP card	**interface gcc1** *R/S/I/P*

Step 3 Use the command **ipv4 address** *ipv4-address net-mask* to set the IPv4 address for the interface.

Example:

This sample configures the GCC2 interface using the static IP address on the 1.2T line card

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#interface gcc2 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.244 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show run interface gcc2 0/1/0/0/1
interface GCC20/1/0/0/1
ipv4 address 10.1.1.1 255.255.255.0
!
```

This sample configures the GCC2 interface using the loopback IP address on 1.2T card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:R2(config)#interface gcc2 0/1/0/0/1
RP/0/RP0/CPU0:R2(config-if)#ipv4 unnumbered loopback 0
RP/0/RP0/CPU0:ios(config-if)#exit
```

This sample checks the status of GCC2 interface.

```
RP/0/RP0/CPU0:ios#show ipv4 interface brief
Wed Sep 22 17:10:04.190 IST
Interface IP-Address Status Protocol Vrf-Name
GCC20/0/0/0/1 198.51.100.234 Up Up default
GCC20/3/0/1/3 198.51.100.244 Up Up default
Loopback0 198.51.100.224 Up
```

This sample configures the GCC0 interface using the static IP address on 1.2T or OTN-XP card.enables the GCC1 interface for the OTN-XP line card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#interface gcc0 0/1/0/0
P/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.244 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
RP/0/RP0/CPU0:ios#show run interface gcc0 0/1/0/0
interface GCC00/1/0/0
```

```
ipv4 address 198.51.100.244 255.255.255.0
!
```

This sample configures the the GCC0 interface using the loopback IP address on 1.2T or OTN-XP card.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:R2(config)#interface gcc0 0/1/0/0
RP/0/RP0/CPU0:R2(config-if)#ipv4 unnumbered loopback 0
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Configure static routes over the GCC interface

Use this task to configure the router to forward packets for specific networks or hosts via the GCC interface using manually defined routes.

Procedure

Step 1 Enter configuration mode.

Step 2 Enter the router static configuration mode.

Step 3 Use the command **address-family ipv4 unicast** *ip4 address default-gateway* to enter address family configuration mode. This step also configures a routing session using standard IPv4 address prefixes.

Example:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#router static address-family ipv4 unicast 0.0.0.0/0 10.105.57.1
RP/0/RP0/CPU0:ios(config)#exit
```

Configure OSPF routes over the GCC interface

Enable OSPF dynamic routing between gateway and remote nodes using GCC interfaces.

This task enables OSPF over GCC interfaces, which facilitates efficient OSPF communication in specialized network environments.

SUMMARY STEPS

1. Enter configuration mode and enable the OSPF routing process using the command **router ospf** *process-id* **router-id** *ip-address*
2. Assign the OSPF area and specify the interfaces to include using the command **area** *area-id* **interface** *type R/S/I/P/L*
3. On the remote node, redistribute connected routes into OSPF, using the command **redistribute connected**.
4. Exit the configuration mode upon completion.

DETAILED STEPS

Procedure

Step 1 Enter configuration mode and enable the OSPF routing process using the command **router ospf** *process-id* **router-id** *ip-address*

Example:

Step 2 Assign the OSPF area and specify the interfaces to include using the command **area** *area-id* **interface** *type R/S/I/P/L*

Step 3 On the remote node, redistribute connected routes into OSPF, using the command **redistribute connected**.

Step 4 Exit the configuration mode upon completion.

Example:

Gateway Node:

```
configure
router ospf 1
router-id 192.0.2.89 area 0

  interface Loopback0

  !

  interface MgmtEth0/RP0/CPU0/1

  !

  interface GCC20/0/0/0/1

  !

  interface GCC20/0/0/0/2
```

Remote Node:

```
configure
router ospf 1
router-id 192.0.2.92
redistribute connected

 area 0

  interface Loopback0

  !

  interface GCC20/0/0/0/1

  !

  interface GCC20/0/0/0/2
```

OSPF is configured over GCC interfaces, enabling gateway and remote nodes to dynamically exchange routing information.

iBGP supports over GCC interfaces

iBGP support over GCC interfaces is a routing capability that

- allows external devices to exchange BGP routes through the management interfaces of NCS 1004 systems,
- enables NCS 1004 devices to advertise local networks and manage them using BGP-learned paths, and
- establishes iBGP sessions over GCC for exchanging BGP routes.

You can configure VRF on the GCC management interfaces (port 0 and port 1) of the NCS 1004 device to achieve traffic isolation between the two management ports.

NCS 1004 supports GCC0 and GCC2 interfaces for the 1.2T line card, allowing greater flexibility and connectivity options. The device advertises its local networks using BGP and manages them through paths learned from the iBGP sessions established over the GCC interfaces.

Restrictions for iBGP Support Using GCC

- IP fragmentation is not supported on the GCC interface.
- The BGP configuration over Open Config (OC) is not supported.

Note The limitations of Remote Node Management Using GCC are applicable for iBGP Support Using GCC. For more information, see Limitations of Remote Node Management.

Caution: Follow restrictions for iBGP support using GCC

Consider these restrictions when implementing iBGP support using GCC:

- Do not use IP fragmentation on the GCC interface; this function is not supported.
- Do not configure BGP over Open Config (OC) on GCC interfaces; this configuration is not supported.
- Observe all limitations found for Remote Node Management using GCC; these also apply to iBGP support using GCC

Enabling the GCC Interface

To enable the GCC2 interface, use the following commands:

```
configure
controller odu4 R/S/I/P/L
gcc2
commit
exit
```

To enable the GCC0 interface, use the following commands:

```
configure
controller CoherentDSP R/S/I/P
gcc0
commit
exit
```

Configuring the Management Interface

To configure the management Ethernet interface with VRF, use the following commands:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios#interface MgmtEth0/RP0/CPU0/1
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
RP/0/RP0/CPU0:ios(config-if)#ipv4 address ipv4-address
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
```

The following example displays how to configure the management Ethernet interface with VRF.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios#interface MgmtEth0/RP0/CPU0/1
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 192.0.2.1 255.255.255.255
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
```

Configuring the Loopback Interface

To configure the loopback interface 0 with VRF, use the following commands:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios#interface Loopback0
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
RP/0/RP0/CPU0:ios(config-if)#ipv4 address ipv4-address
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
```

The following example displays how to configure the loopback interface 0 with VRF.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios#interface Loopback0
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
RP/0/RP0/CPU0:ios(config-if)#ipv4 address 192.0.2.1 255.255.255.255
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
```

Configuring the GCC interface

To configure the GCC2 interface with VRF and static IP address, use the following commands:

configure

interface gcc2 *R/S/I/P/L*

vrf transport-vrf

ipv4 address *ipv4-address*

commit

exit

To configure the GCC0 interface with VRF and static IP address, use the following commands:

configure

interface gcc0 *R/S/I/P*

vrf transport-vrf

ipv4 address *ipv4-address*

commit

exit

Examples

The following sample displays how to configure the GCC2 interface with VRF and static IP address.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#interface gcc2 0/1/0/0/1
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
P/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.5 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following sample displays how to configure the GCC2 interface using loopback IP address.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:R2(config)#interface gcc2 0/1/0/0/1
RP/0/RP0/CPU0:R2(config-if)#ipv4 unnumbered loopback 0
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following sample displays how to configure the GCC0 interface with VRF and static IP address.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#interface gcc0 0/1/0/0
RP/0/RP0/CPU0:ios(config-if)#vrf transport-vrf
P/0/RP0/CPU0:ios(config-if)#ipv4 address 198.51.100.2 255.255.255.0
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

The following sample displays how to configure the GCC0 interface using the loopback IP address.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:R2(config)#interface gcc0 0/1/0/0
RP/0/RP0/CPU0:R2(config-if)#ipv4 unnumbered loopback 0
RP/0/RP0/CPU0:ios(config-if)#exit
RP/0/RP0/CPU0:ios(config)#exit
```

Verifying iBGP Support Using GCC

To verify BGP support using GCC configuration, use the following **show** commands:

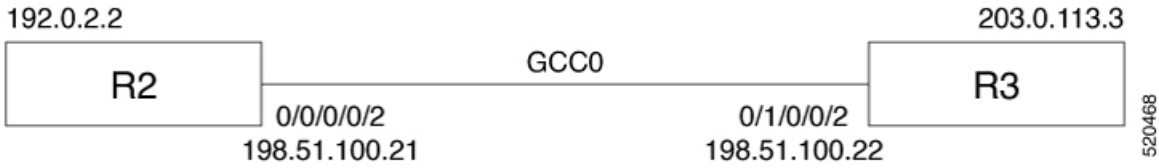
```
RP/0/RP0/CPU0:ios#show bgp vrf transport-vrf neighbors brief
Neighbor        Spk    AS Description                          Up/Down  NBRState
198.51.100.0     0     200                                   00:51:49 Established
198.51.100.1       0    100                                    00:50:32 Established


RP/0/RP0/CPU0:ios#show bgp vrf transport-vrf
BGP VRF transport-vrf, state: Active
BGP Route Distinguisher: 192.0.2.7:0
VRF ID: 0x60000002
BGP router identifier 192.0.2.7, local AS number 100
Non-stop routing is enabled
BGP table state: Active
Table ID: 0xe0000002   RD version: 51
BGP main routing table version 51
BGP NSR Initial initsync version 11 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 192.0.2.7:0 (default for vrf transport-vrf)

*> 209.165.201.30/27        198.51.100.0             0         0 200 i
*> 209.165.201.28/27        0.0.0.0                  0         32768 i
*> 209.165.201.26/27        0    100      0 i
*> 209.165.201.24/27       198.51.100.2             0    100      0 300 i


RP/0/RP0/CPU0:ios#show bgp vrf transport-vrf
BGP VRF transport-vrf, state: Active
BGP Route Distinguisher: 203.0.113.10:0
VRF ID: 0x60000002
BGP router identifier 203.0.113.10, local AS number 100
Non-stop routing is enabled
BGP table state: Active
Table ID: 0xe0000002   RD version: 51
BGP main routing table version 51
BGP NSR Initial initsync version 11 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 203.0.113.10:0 (default for vrf transport-vrf)

*> 209.165.201.30/27        198.51.100.0             0             0 200 i
*> 209.165.201.28/27        0.0.0.0                  0             32768 i
*>i209.165.201.26/27        198.51.100.12              0    100      0 i
*>i209.165.201.24/27       198.51.100.24             0    100      0 300 i
```

iBGP configuration parameters for GCC interfaces

This table summarizes the configuration required for enabling iBGP sessions between two NCS 1004 devices.

Use case:

Consider two NCS 1004 devices, R2 and R3, directly connected through GCC0 interfaces. In this use case, R2 (with IP address 198.51.100.21) and R3 (with IP address 198.51.100.22) are configured with the commands to establish an iBGP session using their respective transport-VRFs.



R2 is connected through GCC0 0/0/0/0 interface with IP address of 198.51.100.21 and R3 is connected through GCC0 0/1/0/0 with IP address of 198.51.100.22. The R2 and R3 devices are connected to external devices through management interfaces.

Table 71: iBGP configuration commands for NCS 1004

Configuration on R2	Configuration on R3

Global Configuration on R2
 	
`hw-module location 0/0`
` mxponder`
`  trunk-rate 600G`
`  client-rate 100GE`
 	
`vrf transport-vrf`
` address-family ipv4 unicast`
	Global Configuration on R3
	
	`hw-module location 0/0`
	` mxponder`
	`  trunk-rate 600G`
	`  client-rate 100GE`
	
	`vrf transport-vrf`
	` address-family ipv4 unicast`

Interface Configuration on R2
 	
`interface Loopback0`
` vrf transport-vrf`
` ipv4 address 192.0.2.2 255.255.255.255`
 	
`interface MgmtEth0/RP0/CPU0/1`
` vrf transport-vrf`
` ipv4 address 198.51.100.25 255.255.255.0`
 	
`controller ODU40/0/0/0/2`
` gcc2`
 	
`interface GCC20/0/0/0/2`
` vrf transport-vrf`
` ipv4 address 198.51.100.21 255.255.255.0`
	Interface Configuration on R3
	
	`interface Loopback0`
	` vrf transport-vrf`
	` ipv4 address 203.0.113.3 255.255.255.255`
	
	`interface MgmtEth0/RP0/CPU0/1`
	` vrf transport-vrf`
	` ipv4 address 198.51.100.32 255.255.255.0`
	
	`controller ODU40/1/0/0/2`
	` gcc2`
	
	`interface GCC20/1/0/0/2`
	` vrf transport-vrf`
	` ipv4 address 198.51.100.22 255.255.255.0`

Route-policy Configuration on R2
 	
`route-policy PASS-ALL`
`  pass`
`end-policy`
	Router Policy Configuration on R3
	
	`route-policy PASS-ALL`
	`  pass`
	`end-policy`

Configuration on R2	Configuration on R3

Static Route Configuration on R2
 	
`router static`
` address-family ipv4 unicast`
`  0.0.0.0/0 198.51.100.28`
` !`
` vrf transport-vrf`
`  address-family ipv4 unicast`
`   198.51.100.0/24 198.51.100.22`
	Static Route Configuration on R3
	
	`router static`
	` address-family ipv4 unicast`
	`  0.0.0.0/0 198.51.100.28`
	
	` vrf transport-vrf`
	`  address-family ipv4 unicast`
	`   198.51.100.0/24 198.51.100.21`

BGP Configurationon R2
 	
`router bgp 100`
` bgp router-id 192.0.2.123`
` address-family vpnv4 unicast`
` !`
` vrf transport-vrf`
`  rd auto`
`  address-family ipv4 unicast`
`   network 203.0.113.1/32`
`  !`
`  neighbor 198.51.100.22`
`   remote-as 100`
`   address-family ipv4 unicast`
`    route-policy PASS-ALL in`
`    route-policy PASS-ALL out`
`    next-hop-self`
`   !`
	BGP Configuration on R3
	
	`router bgp 100`
	` bgp router-id 192.0.2.124`
	` address-family vpnv4 unicast`
	` !`
	` vrf transport-vrf`
	`  rd auto`
	`  address-family ipv4 unicast`
	`   network 203.0.113.3/32`
	`  !`
	`  neighbor 198.51.100.21`
	`   remote-as 100`
	`   address-family ipv4 unicast`
	`    route-policy PASS-ALL in`
	`    route-policy PASS-ALL out`
	`    next-hop-self`
	`   !`

BGP Verification on R2
 	
`RP/0/RP0/CPU0:ios#show bgp sessions`
`Mon Jul 20 14:47:30.378 UTC`
 	
`Neighbor  VRF  Spk`
` AS  InQ  OutQ  NBRState  NSRState`
`198.51.100.22  transport-vrf  0`
`100  0  0  Established  None`
	BGP Verification on R3
	
	`RP/0/RP0/CPU0:regen#show bgp sessions`
	`Tue Jul 21 02:50:14.134 UTC`
	
	`Neighbor  VRF  Spk`
	` AS  InQ  OutQ  NBRState  NSRState`
	`198.51.100.21  transport-vrf  0`
	`100  0  0  Established  None`

CHAPTER 10

Smart Licensing

- Smart Licensing, on page 347
- Generic smart licensing, on page 353
- Create a token, on page 357
- Configure smart licensing, on page 357
- Supported transport modes for Smart Licensing , on page 367
- Smart licensing for OTN-XP line card, on page 371
- Smart license model for QXP line card, on page 376

Smart Licensing

Smart Licensing is a cloud-based licensing solution that simplifies the licensing experience across the enterprise by making it easier to purchase, deploy, track, and renew Cisco software through a single, simple user interface.

- Provides visibility into license ownership and consumption.
- Allows you to track the status of your license and software usage trends.
- Helps you simplify three core functions: purchasing, management, and reporting.

Core functions of Smart Licensing

Smart Licensing helps you simplify the following core functions.

- **Purchasing**: The software that you have installed in your network can be registered, without Product Activation Keys (PAKs).
- **Management**: You can automatically track activations against your license entitlements. Also, there is no need to install the license file on every node. You can create license pools (logical grouping of licenses) to reflect your organization structure. Smart Licensing offers you Cisco Smart Software Manager, a centralized portal that enables you to manage all your Cisco software licenses from one centralized website.
- **Reporting**: Through the portal, Smart Licensing offers an integrated view of the licenses you have purchased and what has been deployed in your network. You can use this data to make better purchasing decisions, based on your consumption.

Smart Licensing features

Smart Licensing provides the following features.

- Your device initiates a call home and requests the licenses it needs.
- Pooled licences - Licences are company account-specific, and can be used with any compatible device in your company. You can activate or deactivate different types of licenses on the device without actually installing a license file on the device.
- Licenses are stored securely on Cisco servers.
- Licenses can be moved between product instances without license transfer. This greatly simplifies the reassignment of a software license as part of the Return Material Authorization (RMA) process.
- It provides a complete view of all the Smart Software Licenses used in the network using a consolidated usage report of software licenses and devices in one easy-to-use portal.

Cisco Smart Account

NCS 1004 integrates with Cisco Smart Accounts to simplify license management.

Smart Accounts provide a centralized, organized, and simple-to-use solution for managing Cisco software licenses across an entire organization. They act as a container that holds all Cisco software assets, allowing customers to view, store, manage, and move these assets as needed. Smart Accounts enable full visibility into license entitlements and usage, helping optimize software management. Smart Accounts support license pooling, portability, and provide compliance reporting.

When creating a Smart Account, you must have the authority to represent the requesting organization. After you submit the request, it goes through a brief approval process. Access http://software.cisco.com to learn about, set up, or manage Smart Accounts.

Cisco Smart Software Manager

Cisco Smart Software Manager enables you to manage all your Cisco Smart software licenses from one centralized website. With Cisco Smart Software Manager, you organize and view your licenses in groups called virtual accounts (collections of licenses and product instances). Use the Cisco Smart Software Manager to do the following tasks:

- Create, manage, or view virtual accounts.
- Create and manage Product Instance Registration Tokens.
- Transfer licenses between virtual accounts or view licenses.
- Transfer, remove, or view product instances.
- Run reports against your virtual accounts.
- Modify your email notification settings.
- View overall account information.

Virtual Accounts

Virtual Accounts are customizable subaccounts within a Smart Account used to organize and optimize Cisco licenses. They can be structured to reflect business units, product types, or geographic locations, allowing

better planning and utilization of assets. They are created and maintained by the Smart Account administrator. Smart Licencing allows you to create multiple license pools or virtual accounts within the Smart Software Manager portal. Using the Virtual Accounts option that you can aggregate licenses into discrete bundles that are associated with a cost center so that one section of an organization cannot use the licenses of another section of the organization. For example, if you segregate your company into different geographic regions, you can create a virtual account for each region to hold the licenses and product instances for that region.

All new licenses and product instances are placed in the default virtual account in the Smart Software Manager, unless you specify a different one during the order process. After you access the default account, you may choose to transfer them to any other account, provided you have the required access permissions.

Use the Smart Software Manager portal to create license pools or transfer licenses.

Product Instance Registration Tokens

A product requires a registration token until you have registered the product. On successful registration, the device receives an identity certificate. This certificate is saved and automatically used for all future communications with Cisco. Registration tokens are stored in the Product Instance Registration Token Table that is associated with your enterprise account. Registration tokens can be valid 1–365 days.

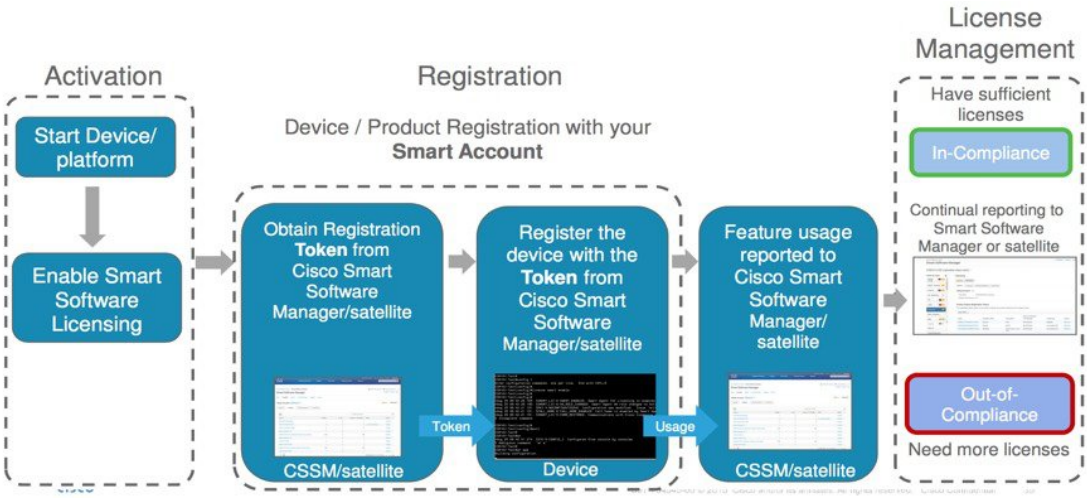
Product Instances

A product instance is an individual device with a unique device identifier (UDI) that is registered using a product instance registration token (or registration token). You can register any number of instances of a product with a single registration token. Each product instance can have one or more licenses residing in the same virtual account. Product instances must periodically connect to the Cisco Smart Software Manager servers during a specific renewal period. If you remove the product instance, its licenses are released and made available within the virtual account.

Smart Licensing workflow

The following figure depicts a working model of smart licensing that involves a three-step procedure.

Figure 14: Smart Licensing Work Flow



1. **Setting up Smart Licensing**: You can place the order for Smart Licensing, to manage licenses on the Cisco.com portal. You agree to the terms and conditions governing the use and access of Smart Licensing in the Smart Software Manager portal.

2. **Enabling and Use Smart Licensing**: Smart Licensing is enabled by default. You can use either of the following options to communicate:

   - **Smart Call Home**: The Smart Call Home feature is automatically configured when Smart Licensing is enabled. Smart Call Home is used by Smart Licensing as a medium for communication with the Cisco license service. Call Home feature allows Cisco products to periodically call-home and perform an audit and reconciliation of your software usage information. This information helps Cisco efficiently track your install base, keep them up and running, and effectively pursue service and support contract renewals. For more information on Smart Call Home feature, see http://www.cisco.com/c/dam/en/us/td/docs/switches/lan/smart_call_home/SCH_Deployment_Guide.pdf.

   - **Smart Software Manager Satellite**: is a component of Cisco Smart Licensing and works with Cisco Smart Software Manager (SSM). It helps customers intelligently manage product licenses, providing near real-time visibility and reporting of the Cisco licenses they purchase and consume.

     For customers who do not want to manage their installed base using a direct Internet connection, the Smart Software Manager satellite is installed on the customer premises and provides a subset of Cisco SSM functionality. After you download the satellite application, deploy it, and register it to Cisco SSM, you can perform the following functions locally:

     - Activate or register a license
     - Get visibility to your company's licenses
     - Transfer licenses between company entities

     Periodically, the satellite must synchronize with Cisco SSM to reflect the latest license entitlements.

     For more information about the Smart Software Manager satellite, see http://www.cisco.com/c/en/us/buy/smart-accounts/software-manager-satellite.html.

3. **Manage and Report Licenses**: You can manage and view reports about your overall software usage in the Smart Software Manager portal. Compliance reporting describes the types of Smart Licensing reports.

Smart licensing benefits for Cisco NCS 1004

Smart Licensing for Cisco NCS 1004 offers several important advantages. These benefits streamline license management, enhance operational flexibility, and improve visibility of software assets across Cisco NCS 1004 deployments.

- License pooling: Licenses can be pooled and reused across the entire organization, enabling flexibility across organizational boundaries.
- No lock-out on over-consumption: Configurations can be performed even if license usage exceeds the purchased limit. When this occurs, out-of-compliance notifications are issued, allowing continued operations while prompting the user to buy additional licenses as needed.
- Asset management visibility: Smart Licensing provides detailed information for software asset management, which helps organizations plan and track their license usage.

Smart Licensing in Cisco NCS 1004

Cisco NCS 1004 supports a set of line card product IDs (PIDs), each with specific licensing characteristics that determine whether smart licensing applies and how the licenses are charged.

- **NCS1K4-1.2T-K9**—High-cost PID. This card does not support smart licensing.
- **NCS1K4-1.2T-L-K9**—Licensed PID for 1.2T line card and the licenses are charged per port.
- **NCS1K4-OTN-XPL**—Licensed PID for OTN-XP line card.
- **NCS1K4-QXP-L-K9**—Licensed PID for QXP line card.
- **NCS1K4-QXP-K9**—Non-licensed PID for QXP line card.
- **NCS1K4-2-QDD-CK9L**—Licensed PID for 800G QDD Line Card - C-band.
- **S-NCS1K4-ULE-400=**—Licensed PID for 400G encryption for OTN-XP Card.

You can use either one or a combination of both types of the 1.2T line card in the NCS 1004.

Software entitlements of Cisco NCS 1004

This reference enables you to identify the smart licensing entitlement for each Cisco NCS 1004 software feature and understand the per-port charging model so that you can plan license procurement.

Software entitlement is a system that consists of a license manager on Cisco NCS 1004. The license manager manages licenses for various software and hardware features. The license manager parses and authenticates the license before accepting it. The following table lists the features and its corresponding entitlements that can be enabled on Cisco NCS 1004 using licenses:

Table 72: Feature History

Feature Name	Release Information	Feature Description
Smart Licensing Support on 2-QDD-C Card	Cisco IOS XR Release 7.9.1	Smart licensing is now supported on the 2-QDD-C card. Being a cloud-based software license management solution, it enables you to automate the licensing process and provides some of these functionalities: • Asset management information to plan and track the licenses. • License pooling across the organization, enabling them to be reused across organizational boundaries. • Notifyes to buy out-of-compliance licenses when the license limit exceeds the paid license limit.

Table 73: Software Entitlements of Cisco NCS 1004

Feature	Software Entitlement
NCS1K4 Smart License-one QSFP28 client	S-NCS1K4-LIC-100G=
NCS1K4 Smart License - one QSFP28 client with encryption	S-NCS1K4-LIC-100X=
NCS1K4 Smart License - 100Gbps of client bandwidth	S-NCS1K4-100G-CL=
NCS1K4 Smart License - Long Haul	S_NCS1K4_LONGHAUL
NCS1K4 Smart License - Subsea	S_NCS1K4_SUBSEA
NCS1K4 Smart License - 100Gbps of client bandwidth	S_NCS1K4_100
NCS1K4 Smart License - 400Gbps of client bandwidth	S_NCS1K4_400

The licenses are charged per port basis and dependent on the number of trunk ports and client ports that you configure. The license count for the configuration of 4 x 100GE client ports or lesser is zero. For configurations greater than 4 x 100GE client ports, the license count is incremented by one for every 100GE client port configured at the slice level. The license count for the trunk port is incremented based on the BPS & optics configuration.

Generic smart licensing

This reference enables you to identify the generic smart license entitlements that span multiple Cisco NCS 1004 cards and verify the license usage with sample show command outputs.

Generic Smart Licensing feature allows you to purchase licenses that can be used on 2-QDD-C card, 1.2T card, and OTN XP card with any client data rate with or without encryption. This feature further simplifies license procurement and management of licensing.

Table 74: Feature History

Feature Name	Release Information	Feature Description
Generic Smart License	Cisco IOS XR Release 7.10.1	Smart licensing functionality is enhanced to enable you to use one common license across cards and functionalities. Hence, a single license will provide entitlement to use 2-QDD-C card, 1.2T card, and OTN XP card, with encryption enabled or disabled, and for different client data rates. This feature reduces license procurement and management effort.

Generic Smart Licensing Software Entitlements of Cisco NCS 1004:

Feature	Software Entitlement
NCS1K4 - 100G without encryption	S_NCS1K4_100
NCS1K4 smart license - 100G with encrypiton	S-NCS1K4-LIC-100X=
NCS1K4 - 400G without encryption	S_NCS1K4_400
NCS1K4 - 100G with encryption	S-NCS1K4-LIC-100X=
NCS1K4 - 400G - with encrypition	S_NCS1K4_400_ENC

Sample output for 100G smart licensing without encryption on the 1.2T card

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
```

```
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC
```

Sample output for 100G smart licensing with encryption on the 1.2T card:

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  NCS1K4 - 100G           (S_NCS1K4_100)                    7 AUTHORIZED
```

Sample output for 100G smart licensing without encryption on OTN-XP card:

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC

License Usage:
  License                                 Entitlement Tag                     Count Status
  -----------------------------------------------------------------------------------------

  NCS1K4 smart license...    (S-NCS1K4-LIC-100X=)              1 AUTHORIZED
```

Sample output for 100G smart licensing without encryption on OTN-XP card.

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
```

```
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  NCS1K4 - 100G           (S_NCS1K4_100)                    1 AUTHORIZED
```

Sample output for 100G smart licensing with encryption on OTN-XP card

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC
Smart Licensing is ENABLED
Registration:
Status: REGISTERED
Smart Account: BU Production Test
Virtual Account: NCS1000
Export-Controlled Functionality: ALLOWED
Last Renewal Attempt: None
Next Renewal Attempt: Aug 14 2023 16:53:02 UTC
License Authorization:
Status: AUTHORIZED
Last Communication Attempt: SUCCEEDED
Next Communication Attempt: Mar 26 2023 14:41:27 UTC
License Usage:
License Entitlement Tag Count Status
-----------------------------------------------------------------------------
NCS1K4 smart license... (S-NCS1K4-LIC-100X=) 1 AUTHORIZED
```

Sample output for 400G smart licensing without encryption on OTN-XP card

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC
Smart Licensing is ENABLED
Registration:
Status: REGISTERED
Smart Account: BU Production Test
Virtual Account: NCS1000
Export-Controlled Functionality: ALLOWED
Last Renewal Attempt: None
Next Renewal Attempt: Aug 14 2023 16:53:02 UTC
License Authorization:
Status: AUTHORIZED
Last Communication Attempt: SUCCEEDED
Next Communication Attempt: Mar 26 2023 14:41:27 UTC
License Usage:
License Entitlement Tag Count Status
-----------------------------------------------------------------------------
NCS1K4 - 400G (S_NCS1K4_400) 1 AUTHORIZED
```

Sample output for 400G smart licensing with encryption on OTN-XP card

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC
Smart Licensing is ENABLED
```

```
Registration:
Status: REGISTERED
Smart Account: BU Production Test
Virtual Account: NCS1000
Export-Controlled Functionality: ALLOWED
Last Renewal Attempt: None
Next Renewal Attempt: Aug 14 2023 16:53:02 UTC
License Authorization:
Status: AUTHORIZED
Last Communication Attempt: SUCCEEDEDNext Communication Attempt: Mar 26 2023 14:41:27 UTC
License Usage:
License Entitlement Tag Count Status
-----------------------------------------------------------------------------
NCS1K4 - 400G - ENC (S_NCS1K4_400_ENC) 1 AUTHORIZED
```

The following sample output displays 100G smart licensing without encryption on OTN-XP card:

The following sample output displays 100G smart licensing with encryption on OTN-XP card:

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC
   License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  NCS1K4 smart license... (S-NCS1K4-LIC-100X=)              1 AUTHORIZED
```

The following sample output displays smart licensing support for 400G without encryption on OTN-XP card:

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC

License Usage:
  License                 Entitlement Tag               Count Status
```

```
  ------------------------------------------------------------------------------
  NCS1K4 - 400G           (S_NCS1K4_400)                    1 AUTHORIZED
```

The following sample output displays smart licensing support for 400G with encryption on OTN-XP card:

```
RP/0/RP0/CPU0:ios#show license summary
Fri Feb 24 14:42:02.029 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 2023 16:53:02 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Mar 26 2023 14:41:27 UTC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  NCS1K4 - 400G - ENC     (S_NCS1K4_400_ENC)                1 AUTHORIZED
```

Create a token

To create a new token using Cisco Smart Software Manager, perform the following tasks:

Procedure

Step 1 Log in to the Cisco Smart Software Manager.

https://software.cisco.com/software/csws/ws/platform/home#SmartLicensing-Inventory

Step 2 Select the token with the name that was provided while creating the token.

Step 3 Click on the blue arrow on the token. The token ID generates.

Step 4 Copy the token and register the NCS1004 with the same token ID.

Configure smart licensing

Enable smart license management on your Cisco NCS 1004 by connecting the device to Cisco Smart Licensing services.

Smart licensing automates license activation and compliance, helping you centrally manage device licenses.

To configure smart licensing in Cisco NCS 1004, perform these tasks:

Procedure

Step 1 Configure the domain name server for the smart license server.

Example:

```
RP/0/RP0/CPU0:ios#configure
Sat Dec 15 15:25:14.385 IST
RP/0/RP0/CPU0:ios(config)#domain name-server 198.51.100.247
```

Step 2 Setup the CiscoTAC-1 profile and destination address for Smart Call Home, using these commands:

call-home

service active

contact smart-licensing

profile CiscoTAC-1

active

destination address http {http|https}://{FQDN}/its/service/oddce/services/DDCEService

destination transport-method http

Note
The destination address must be the fully qualified domain name (FQDN) of the Cisco Smart Software Manager (`tools.cisco.com`) or your Smart Licensing satellite server. Configure the DNS server before setting up the call-home destination. The CiscoTAC-1 profile is the default for smart licensing and must not be deleted.

Example:

```
RP/0/RP0/CPU0:ios#configure
Sat Dec 15 15:25:14.385 IST
RP/0/RP0/CPU0:ios(config)#domain name-server 198.51.100.247
```

```
RP/0/RP0/CPU0:ios(config)#call-home
 RP/0/RP0/CPU0:ios(config)#service active
 RP/0/RP0/CPU0:ios(config)#contact smart-licensing
 RP/0/RP0/CPU0:ios(config)#profile CiscoTAC-1
 RP/0/RP0/CPU0:ios(config)#active
 RP/0/RP0/CPU0:ios(config)#destination address http
https://tools.cisco.com/its/service/oddce/services/DDCEService
 RP/0/RP0/CPU0:ios(config)#destination transport-method http
```

]

Step 3 Configure the crypto ca Trust point profile, if CRL distribution point is not defined in the Satellite server certificate or if the device is not able to reach the host mentioned in the CRL distribution point.

Example:

```
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
```

Step 4 Create and copy the registration token ID using Cisco Smart Software Manager.

For more details about creating a token, see Create a token.

Step 5 Use the **license smart register idtoken** *token-ID* in the privileged EXEC mode, to register the token ID in Cisco NCS 1004.

If registration fails due to communication issues, wait 24 hours for the device to retry or force the registration using the **license smart register idtoken** *token-ID* **force** command.

If the device is removed from inventory, redeployed, or returned to Cisco using Return Merchandise Authorization (RMA), cancel registration and remove entitlements, using the **license smart deregister** command to cancel the registration on your device. All smart licensing entitlements and certificates on the platform are removed.

ID certificates are renewed automatically after six months. In case, the renewal fails, the product instance goes into unidentified state. You can manually renew the ID certificate using the **license smart renew id** command.

Authorization periods are renewed by the Smart Licensing system every 30 days. As long as the license is in an 'Authorized' or 'Out-of-Compliance' (OOC), the authorization period is renewed. Use the **license smart renew auth** command to make an on-demand manual update of your registration. Thus, instead of waiting 30 days for the next registration renewal cycle, you can issue this command to instantly find out the status of your license.

After 90 days, the authorization period expires and the status of the associated licenses display "AUTH EXPIRED". Use the **license smart renew auth** command to retry the authorization period renewal. If the retry is successful, a new authorization period begins.

The Cisco NCS 1004 is registered with Cisco Smart Licensing, license entitlements are tracked automatically, and compliance is maintained.

Verify smart licensing configuration

This reference enables you to verify smart licensing configuration on Cisco NCS 1004 by listing the supported show commands, describing each license authorization status, and providing sample command outputs for confirmation.

After enabling smart licensing, you can use the **show** commands to verify the default smart licensing configuration. If any issue is detected, take corrective action before making further configurations.

- **show license all**
- **show license trace all**
- **show license status**
- **show license summary**
- **show license tech**
- **show license udi**
- **show license usage**
- **show license platform detail**
- **show license platform summary**
- **show license platform trace**
- **show license platform trace all**
- **show tech-support smartlic**
- **show call-home detail**

- **show call-home trace all**
- **show tech-support call-home**

This table defines the available license authorization status in Cisco NCS 1004:

Table 75: License Authorization Status

License Authorization Status	Description
Unconfigured	Smart Software Licensing is not configured.
Unidentified	Smart Software Licensing is enabled but is not registered.
Registered	Device registration is completed and an ID certificate is received that is used for future communication with the Cisco licensing authority.
Authorized	Registration is completed with a valid Smart Account and license consumption has begun. This indicates compliance.
Out of Compliance	Consumption exceeds available licenses in the Smart Account.
Authorization Expired	The device is unable to communicate with the Cisco Smart Software Manager for an extended period. This state occurs after 90 days of expiry. The device attempts to contact the CSSM every hour to renew the authorization until the registration period expires.

Example 1: show license all

The following example shows the sample output of the **show license all** command.

```
RP/0/RP0/CPU0:SIT-5#show license all
Wed Sep 22 17:18:19.761 IST

Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Jun 21 2021 12:40:52 IST
  Last Renewal Attempt: None
  Next Renewal Attempt: Dec 18 2021 12:40:51 IST
  Registration Expires: Jun 21 2022 12:35:49 IST

License Authorization:
  Status: AUTHORIZED on Sep 09 2021 11:45:43 IST
  Last Communication Attempt: SUCCEEDED on Sep 09 2021 11:45:43 IST
  Next Communication Attempt: Oct 09 2021 11:45:42 IST
  Communication Deadline: Dec 08 2021 11:40:41 IST

Export Authorization Key:
```

```
  Features Authorized:
    <none>

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Miscellaneus:
  Custom Id: <empty>

License Usage
==============

NCS1K4 - Subsea - 1.2T Line Card (S_NCS1K4_SUBSEA):
  Description: NCS1K4 - Subsea - 1.2T Line Card
  Count: 3
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED

NCS1K4 smart license - one QSFP28 client (S-NCS1K4-LIC-100G=):
  Description: NCS1K4 smart license - one QSFP28 client
  Count: 18
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED

NCS1K4 smart license - one QSFP28 client with encryption (S-NCS1K4-LIC-100X=):
  Description: NCS1K4 smart license - one QSFP28 client with encryption
  Count: 2
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED

NCS1K4 - Long Haul - 1.2T Line Card (S_NCS1K4_LONGHAUL):
  Description: NCS1K4 - Long Haul - 1.2T Line Card
  Count: 2
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED

Product Information
====================
UDI: SN:CAT2249B009,UUID:default-sdr

Agent Version
=============
Smart Agent for Licensing: 4.13.32_rel/85

Reservation Info
================
License reservation: DISABLED
```

Example 2: show license platform detail

The following example shows the sample output of the **show license platform detail** command.

```
RP/0/RP0/CPU0:ios#show license platform detail
Mon Feb 11 15:59:55.422 IST
Current state:    REGISTERED

Collection: LAST: Mon Feb 11 2019 15:57:53 IST
            NEXT: Mon Feb 11 2019 16:57:53 IST
Reporting:  LAST: Mon Feb 11 2019 15:57:53 IST
            NEXT: Tue Feb 12 2019 15:57:53 IST

Parameters: Collection interval:          60 minute(s)
            Reporting  interval:        1440 minute(s)
            Throughput gauge:        1000000 Kbps

================================================
Feature/Area 'system'
   Name:  System
   Status:   ACTIVE
   Flags: CONFIG

   [ 1] Name:  NCS1K4 smart license - one QSFP28 client
        Entitlement Tag:
regid.2018-05.com.cisco.S-NCS1K4-LIC-100G=,1.0_03df009f-5ac5-48da-af50-4279ddea5e24
        Count: Last reported:    8
               Next report:      0
   [ 2] Name:  NCS1K4 smart license - one QSFP28 client with encryption
        Entitlement Tag:
regid.2018-05.com.cisco.S-NCS1K4-LIC-100X=,1.0_3938b0c5-f635-4426-9f0f-936d930cea9e
        Count: Last reported:    8
               Next report:      0
```

Example 3: show license status

The following example shows the sample output of the **show license status** command.

```
RP/0/RP0/CPU0:ios#show license status
Mon Feb 11 16:02:24.499 IST

Smart Licensing is ENABLED
  Initial Registration: SUCCEEDED on Mon Feb 11 2019 15:51:10 IST
  Last Renewal Attempt: None
  Next Renewal Attempt: Sat Aug 10 2019 15:52:10 IST
  Registration Expires: Tue Feb 11 2020 15:46:59 IST

License Authorization:
  Status: AUTHORIZED on Mon Feb 11 2019 15:53:40 IST
  Last Communication Attempt: SUCCEEDED on Mon Feb 11 2019 15:53:40 IST
  Next Communication Attempt: Wed Mar 13 2019 15:53:39 IST
  Communication Deadline: Sun May 12 2019 15:47:29 IST
```

Example 4: show license usage

The following example shows the sample output of the **show license usage** command.

```
RP/0/RP0/CPU0:ios#show license usage
Mon Feb 11 15:59:29.817 IST

License Authorization:
  Status: AUTHORIZED on Mon Feb 11 2019 15:53:40 IST

NCS1K4 smart license - one QSFP28 client (S-NCS1K4-LIC-100G=):
  Description: NCS1K4 smart license - one QSFP28 client
  Count: 8
  Version: 1.0
  Status: AUTHORIZED

NCS1K4 smart license - one QSFP28 client with encryption (S-NCS1K4-LIC-100X=):
  Description: NCS1K4 smart license - one QSFP28 client with encryption
 Count: 8
  Version: 1.0
  Status: AUTHORIZED
```

Example 5: show license udi

The following example shows the sample output of the **show license udi** command.

```
RP/0/RP0/CPU0:ios#show license udiMon Feb 11 16:02:46.733 IST

Product Information
===================
UDI: SN:CAT2231B18Y,UUID:default-sdr
```

Criteria for license consumption of subsea networks

A subsea license is a trunk-based smart license model that

- is consumed on Cisco NCS 1004 whenever specified subsea trunk-port configurations, bits-per-symbol values, or chromatic dispersion ranges exist in the running configuration,
- supports tracking license status and software usage trends for subsea deployments, and
- allows access to subsea-specific controls such as extended chromatic dispersion and advanced compensation settings.

Table 76: Feature History

Feature Name	Release Information	Feature Description
Licensing Feature Update	Cisco IOS XR Release 7.3.2	Criteria for license consumption in Long Haul and Subsea networks are introduced. Long Haul (LH) and Subsea are two trunk-based license models that are implemented regardless of the line rate, in addition to the existing client-based licenses. These license models allow the user to easily track the status of licenses and software usage trends. The long-haul license is required to enable QPSK and 8QAM modes. The subsea license is required to enable BPSK and subsea specific controls such as extended chromatic dispersion, special non-linear compensation settings and so on.

Criteria for license consumption

The criteria for license consumption of subsea networks are as follows:

- If any of the following trunk-port configurations appear in the running configuration, the subsea license is consumed:
  - `Ios(config-optics)#filter-roll-off-factor <0-1>`
  - `Ios(config-optics)#rx-voa target-power <-190,+30>`
  - `Ios(config-optics)#rx-voa fixed-ratio <+100,+1700>`
  - `Ios(config-optics)#enh-colorless-mode <1-3>`
  - `Ios(config-optics)#enh-sop-tol-mode <1-3>`
  - `Ios(config-optics)#nleq-comp-mode <1-4>`
  - `Ios(config-optics)#cross-pol-gain-mode <1-15>`
  - `Ios(config-optics)#cross-pol-weight-mode <1-7>`
  - `Ios(config-optics)#cpr-win-mode <1-15>`
  - `Ios(config-optics)#cpr-ext-win-mode <1-15>`
- The license is consumed if the bps value is in the range of 1–2 bits per symbol or 1, excluding the boundary value of 2 in the following command:

  ```
  show controller optics <trunk-port>
  ```
- The license is consumed if the cd-min value is less than -10000 or cd-max value is greater than 100000, excluding the boundary values in the output of the following command:

```
show controller optics <trunk-port>
```

Example for license consumption

The following output of **show license usage** command shows the license usage count as one.

```
RP/0/RP0/CPU0:BH-SIT2#show license usage
Tue Aug  3 11:12:42.440 IST
License Authorization:
Status: AUTHORIZED on Aug 03 2021 11:10:12 IST
NCS1K4 - Subsea - 1.2T Line Card (S_NCS1K4_SUBSEA):
Description: NCS1K4 - Subsea - 1.2T Line Card
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
```

Criteria for license consumption of long-haul networks

The long-haul smart license is consumed on Cisco NCS 1004 when the bits-per-symbol value reported by the trunk controller falls within the qualifying range, and the license remains consumed while the line card is up.

Long-haul license consumption criteria

The smart license is consumed if the bps value is in the range of 2 - 4 bits per symbol including the boundary values in the output of the following command:

```
show controller optics 0/1/0/1
```

Example:

show controller optics 0/1/0/1 Bits per symbol = 3.0000000000 bits/symbol

In the above example, the bits per symbol value is 3, so the license is consumed.

Note

- The license is consumed even when the trunk port is in shutdown state.
- The license is consumed only when the line card is up and running.

Example for license consumption

The following output of **show license usage** command shows the license usage count as two:

```
RP/0/RP0/CPU0:SIT-4#show license usage
Tue Aug 3 15:53:48.453 IST
License Authorization:
NCS1K4 - Long Haul - 1.2T Line Card (S_NCS1K4_LONGHAUL):
Description: NCS1K4 - Long Haul - 1.2T Line Card
Count: 2
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
```

Register a smart license on Cisco NCS 1004

Enable smart license functionality on Cisco NCS 1004 and maintain licensing compliance via Cisco Smart Software Manager (CSSM).

Use this process to register a smart license for your Cisco NCS 1004 device, address synchronization issues between the device and CSSM, and perform license renewal and deregistration when needed.

Procedure

Step 1 Register the license using the `license smart register idtoken` <*idtoken*> command:

```
RP/0/RP0/CPU:ios#license smart register idtoken <idtoken>
```

Step 2 Access the Cisco Smart Software Manager portal, https://software.cisco.com/software/csws/ws/platform/home#module/SmartLicensing.

Step 3 Remove the existing product instance in CSSM:

a) Click **Inventory**.
b) Click **Product Instances**.
c) Select the node instance.
d) Click **Actions**, and then **Remove**.

Step 4 Renew the authorization period on the device.

```
RP/0/RP0/CPU0:ios#license smart renew auth
```

```
RP/0/RP0/CPU0:ios#show logging | i "Data and signature"
Thu May 27 09:57:02.237 UTC
RP/0/RP0/CPU0:May 27 09:54:57.783 UTC: smartlicserver[311]:
LICENSE-SMART_LIC-3-AUTH_RENEW_FAILED : Authorization renewal with the Cisco Smart Software Manager
 (CSSM) :
Error received from Smart Software Manager: Data and signature do not match  for udi
PID:8812,SN:FOX2202WIVM
```

Note
The error message in the output of the **show logging** command is expected and is due to loss of synchronization between the CSSM server and the device after removing the product instance directly from the CSSM server.

Step 5 Deregister the device from CSSM.

```
RP/0/RP0/CPU0:ios#license smart deregister
```

```
RP/0/RP0/CPU0:ios#show logging | i DEREG
Thu May 27 14:48:58.170 UTC
RP/0/RP0/CPU0:May 27 09:58:58.464 UTC: smartlicserver[311]:
%LICENSE-SMART_LIC-3-AGENT_DEREG_FAILED : Smart Agent for Licensing DeRegistration with Cisco Smart
 Software Manager (CSSM) failed:
Agent received a failure status in a response message. Please check the Agent log file for the detailed
 message.
```

Note
The error message in the output of the **show logging** command is expected.

The Cisco NCS 1004 device is registered, synchronized, and deregistered as needed with CSSM. Expected error messages confirm the resolution of device and CSSM synchronization issues.

Supported transport modes for Smart Licensing

Smart Licensing software management enables you to choose from several transport modes, depending on your network connectivity and operational requirements. The default transport mode is Cisco Smart Licensing Utility (CSLU), but you can select Call-Home, Smart Transport, or Offline modes based on your needs.

The supported transport modes and their key features are:

- Call-Home mode: Directly communicates with Cisco through a Call-Home feature to register and manage licenses. Suitable for networks with internet access.
- Smart Transport mode: Utilizes Smart Licensing Transport to relay license information securely to Cisco servers. Recommended for environments that require secure, automated reporting.
- CSLU mode: Uses Cisco Smart Licensing Utility as an intermediary for managing licenses locally, offering flexibility and local control. This mode is the default.
- Offline mode: Operates without an active internet connection by generating and transferring files manually for license management. Designed for isolated or secure environments where connectivity is restricted.

You can change the transport mode to best match your network and management preferences.

Configure Call Home for Smart Licensing

Enable Call Home to establish secure communication with CSSM, which is required for smart licensing and license reporting on your device.

Call Home connects your device with CSSM and automates license registration, reporting, and updates. Proper configuration ensures compliance and full access to Cisco licensing features.

Procedure

Step 1 Enable Call Home and configure Call Home mode settings:

Example:

```
RP/0/RP0/CPU0:ios#call-home
RP/0/RP0/CPU0:ios(config-call-home)#service active
RP/0/RP0/CPU0:ios(config-call-home)#contact smart-licensing
RP/0/RP0/CPU0:ios(config-call-home)#profile CiscoTAC-1
RP/0/RP0/CPU0:ios(config-call-home-profile)#active
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination address http
https://tools.cisco.com/its/service/oddce/services/DDCEService
RP/0/RP0/CPU0:ios(config-call-home-profile)#reporting smart-call-home-data
RP/0/RP0/CPU0:ios(config-call-home-profile)#reporting smart-licensing-data
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination transport-method email disable
RP/0/RP0/CPU0:ios(config-call-home-profile)#destination transport-method http
RP/0/RP0/CPU0:ios(config-call-home-profile)#commit
RP/0/RP0/CPU0:ios(config-call-home-profile)#end
```

Step 2 Configure the domain name and DNS name server.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#domain name cisco.com
RP/0/RP0/CPU0:ios(config)#domain name-server 64.102.6.247
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Enable Certificate Revocation List (CRL) configuration, if you have a Trustpool CA.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 4 Set Call Home as the smart license transport mode.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport callhome
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Verify whether the Callhome is Configured.

```
RP/0/RP0/CPU0:ios#show license all
Transport:  Type: Callhome
```

Step 5 Establish trust with smart licensing using an ID token.

Example:

```
license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Call Home is enabled and configured on your device. The device will connect to CSSM for automated license management, compliance, and reporting.

Configure Smart Transport

Set up Smart Transport to connect your device to the CSSM for license management and establish a trusted network connection.

Smart Transport serves as an alternative to Call Home for connecting to CSSM. It enables secure network-based license registration and trust establishment.

Procedure

Step 1 Configure the Smart proxy and hostname for communication with CSSM.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#license smart proxy port 80
RP/0/RP0/CPU0:ios(config)#license smart proxy hostname proxy.esl.cisco.com
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Enable Certificate Revocation List (CRL) configuration to manage certificate trust.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Enable Smart Transport for license registration.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport smart
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Verify Smart Transport is configured correctly.

```
RP/0/RP0/CPU0:ios#show license all
Transport:
  Type: Smart
  URL: https://smartreceiver.cisco.com/licservice/license
  Proxy:
    Address: proxy.esl.cisco.com
    Port: 80
    Username: <empty>
    Password: <empty>
  VRF:
    Not Supported
```

Step 4 Establish trust with CSSM using an id-token.

Example:

```
license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Smart Transport is configured for secure communication and trust establishment with CSSM. The device can now use CSSM for license registration and management tasks.

Configure CSLU transport mode

Establish CSLU as the transport mode for license communication between your device and an on-premise CSLU server.

CSLU (Cisco Smart License Utility) is the default transport mode for software license management. Configuring CSLU enables secure license communications and compliance reporting for your device.

Procedure

Step 1 Use the **license smart url cslu http://<cslu-local>:8182/cslu/v1/pi** command to configure the CSLU URL.

Example:

In this sample configuration, the **10.127.59.44** is the CSLU URL that the on-premise server provided. This URL changes for each on-premise servers.

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#license smart url cslu http://10.127.59.44:8182/cslu/v1/pi
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Use this sample configuration to enable CRL Configuration.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool crl optional
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 3 Use the **license smart transport cslu** command to enable CSLU.

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#license smart transport cslu
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 4 Use the **show license all** command to verify the CSLU configuration.

Example:

The sample output shows the configuration transport type with the on-prem server address.

```
RP/0/RP0/CPU0:ios#show license all
Transport:
  Type: cslu
  Cslu address: http://10.127.59.44:8182/cslu/v1/pi
  Proxy:
    Not Configured
  VRF:
    Not Supported
```

Step 5 (Optional) Use the **license smart trust idtoken**<*token-id*> command to establish trust using id-token.

Example:

The command uses **Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n** as a sample token id.

```
RP/0/RP0/CPU0:ios#license smart trust idtoken Zesdf3243u48329fdfhsfhsfkjs1233j4h1j1j4j41n
```

Step 6 Use the **license smart sync all** command to send the license usage reports to server and receive the compliance status for the licenses consumed by the node.

Example:

```
RP/0/RP0/CPU0:ios#license smart sync all
Tue Aug 26 13:03:28.287 IST
```

Note
If you are using on-prem version 9-202407 or later, this command retrieves the on-prem account details and the compliance information.

The device uses CSLU transport for license communications, and all required trust relationships are established.

What to do next

- Review device status in Cisco Smart Software Manager.
- Ensure device reports successful license communication

Configure offline license mode

Enable offline license operation for devices that cannot directly connect to Cisco Smart Licensing servers.

Offline license mode is necessary when your device cannot communicate with Cisco Smart Licensing servers over the network. This process allows you to manage licenses through manual file operations.

Procedure

Step 1 Disable transport to switch the device to offline license mode

Example:

```
RP/0/RP0/CPU0:ios#config
RP/0/RP0/CPU0:ios(config)#License smart transport off
RP/0/RP0/CPU0:ios(config)#commit
RP/0/RP0/CPU0:ios(config)#end
```

Step 2 Save the license usage report.

Example:

```
RP/0/RP0/CPU0:ios#license smart save usage unreported /misc1/disk1/usage.txt
```

Step 3 Import the acknowledgment report.

Example:

```
RP/0/RP0/CPU0:ios#license smart import /misc/disk1/ACK_usage.txt
```

The device operates in offline license mode. License usage is recorded and acknowledgments are managed manually without network transport to Cisco Smart Licensing.

Smart licensing for OTN-XP line card

Smart licensing on the Cisco NCS 1004 OTN-XP line card is calculated based on 100G client bandwidth and is independent of the client type, with the license charged per 100G client bandwidth used.

- The license calculation is based on 100G client bandwidth and is independent of the client type.
- The licensed OTN-XP line card PID is **NCS1K4-OTN-XPL**.
- The license is charged based on the usage of 100G client bandwidth.

License usage count for Cisco NCS 1004 OTN-XP line card

This reference enables you to verify license usage on the OTN-XP line card by showing a sample slice configuration in muxponder mode and the related show command outputs that confirm the license count.

License usage overview

The Cisco NCS 1004 OTN-XP line card uses smart licensing to track client bandwidth utilization. The S-NCS1K4-100G-CL= license represents 100Gbps of client bandwidth.

License count calculation:

Using 1–10 client ports requires 1 license.

- Using 1–10 client ports requires 1 license.
- Using 11–20 client ports requires 2 licenses.
- For each additional set of 10 ports, one more license is needed.

Sample slice configuration in muxponder mode

```
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
  trunk-rate 100G
  client-port-rate 2 lane 3 client-type OTU2
  client-port-rate 2 lane 4 client-type OTU2E
  client-port-rate 4 lane 1 client-type 10GE
  client-port-rate 4 lane 2 client-type OTU2
  client-port-rate 4 lane 3 client-type OTU2E
  client-port-rate 4 lane 4 client-type 10GE
  client-port-rate 5 lane 1 client-type OTU2E
  client-port-rate 5 lane 2 client-type 10GE
  client-port-rate 5 lane 3 client-type OTU2
  client-port-rate 5 lane 4 client-type OTU2E
```

Slice state output example

The following sample shows the status of the configured slice as **Provisioned**.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder
Fri Dec  6 02:50:32.858 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       MIXED
Trunk  Bitrate:       100G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
Client Port                     Mapper/Trunk Port    Peer/Trunk Port        OTU40/1/0/0
                                Traffic Split Percentage

OTU20/1/0/2/3                         NONE           ODU20/1/0/0/2/3           100
OTU20/1/0/4/2                         NONE           ODU20/1/0/0/4/2           100
OTU20/1/0/5/3                         NONE           ODU20/1/0/0/5/3           100
OTU2E0/1/0/2/4                        NONE           ODU2E0/1/0/0/2/4          100
OTU2E0/1/0/4/3                        NONE           ODU2E0/1/0/0/4/3          100
OTU2E0/1/0/5/1                        NONE           ODU2E0/1/0/0/5/1          100
OTU2E0/1/0/5/4                        NONE           ODU2E0/1/0/0/5/4          100
TenGigECtrlr0/1/0/4/1           ODU2E0/1/0/0/4/1          NONE                100
TenGigECtrlr0/1/0/4/4           ODU2E0/1/0/0/4/4          NONE                100
TenGigECtrlr0/1/0/5/2           ODU2E0/1/0/0/5/2          NONE                100
```

License count output example

The following sample shows the license usage count as "Count: 1":

```
RP/0/RP0/CPU0:ios#show license all
Fri Dec  6 02:58:39.906 UTC

Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Dec 06 2019 02:54:27 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Jun 03 2020 02:54:26 UTC
  Registration Expires: Dec 05 2020 02:49:42 UTC

License Authorization:
  Status: AUTHORIZED on Dec 06 2019 02:56:50 UTC
  Last Communication Attempt: SUCCEEDED on Dec 06 2019 02:56:50 UTC
  Next Communication Attempt: Dec 06 2019 14:56:49 UTC
  Communication Deadline: Mar 05 2020 02:52:06 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED
Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
=============

NCS 1K4 smart License -  100Gbps of client bandwidth (S-NCS1K4-100G-CL=):
  Description: 100G Client bandwidth - Smart License
  Count: 1
  Version: 1.0
  Status: Authorized
  Export status: NOT RESTRICTED

Product Information
===================
UDI: SN:CAT2217B025,UUID:default-sdr

Agent Version
=============
Smart Agent for Licensing: 4.10.4_rel/21

Reservation Info
================
License reservation: DISABLED
```

License count calculation logic

For client port types 10GE, OTU2, and OTU2E:

- If the number of client ports used is between 1 and 10, the "S-NCS1K4-100G-CL=" license count is 1.
- If the number of client ports used is between 11 and 20, the license count is 2.
- This pattern continues in increments of 10 (for each set of 10 ports, an additional license is required).

Smart licensing for OTN-XP card in regen mode

The OTN-XP line card supports smart licensing in regen mode. Regen is a signal regenerator that sits between two nodes to regenerate the signal, and smart licensing automates the licensing process and tracks software usage trends.

Table 77: Feature History

Feature Name	Release Information	Feature Description
Smart Licensing for OTN-XP Card in Regen Mode	Cisco IOS XR Release 7.8.1	Now the OTN-XP Line Card supports the smart licensing feature in Regen mode. Regen is a signal regenerator and it sits between two nodes to regenerate the signal. it enables you to automate the time-consuming manual licensing tasks and allows you to easily track the status of your license and software usage trends. Supported modes: • 200G and 400G

Check the license count for OTN-XP line card in regen mode

This reference enables you to verify the smart license count consumed by the OTN-XP line card in regen mode by showing card details, datapath status, and license verification command outputs.

You can verify the smart license consumption of the OTN-XP line card in regen mode by checking the card details, the configured datapath status, and the license usage details.

Checking card details

The following sample shows how to check the card details:

```
RP/0/RP0/CPU0:regen#show platform
Wed Nov 16 15:11:51.088 UTC
Node              Type                       State             Config state
--------------------------------------------------------------------------------
0/0               NCS1K4-OTN-XPL             OPERATIONAL       NSHUT
0/1               NCS1K4-LC-FILLER           PRESENT           NSHUT
```

Configuring datapath

The following sample shows the status of the configured slice as **Provisioned**.

```
RP/0/RP0/CPU0:regen#show hw-module location 0/0 regen
Wed Nov 16 15:11:01.258 UTC

Location:             0/0
Trunk  Bitrate:       400G
Status:               Provisioned
East Port                  West Port
---------                  ---------
CoherentDSP0/0/0/12         CoherentDSP0/0/0/13
```

Checking the license count

```
RP/0/RP0/CPU0:regen#show license all
Wed Nov 16 15:12:59.353 UTC

Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test 1
  Virtual Account: N1K4-DT-PROD-TEST
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Nov 16 2022 15:06:56 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: May 15 2023 15:06:56 UTC
  Registration Expires: Nov 16 2023 09:40:34 UTC

License Authorization:
  Status: AUTHORIZED on Nov 16 2022 15:08:48 UTC
  Last Communication Attempt: SUCCEEDED on Nov 16 2022 15:08:48 UTC
  Next Communication Attempt: Dec 16 2022 15:08:47 UTC
  Communication Deadline: Feb 14 2023 09:42:29 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Miscellaneous:
  Custom Id: <empty>

License Usage
=============
```

```
NCS1K4 - 400G RG (S_NCS1K4_400G_RG):
  Description: NCS1K4 - 400G REGEN License
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED

Product Information
===================
UDI: SN:CAT2311B0BN,UUID:default-sdr

Agent Version
=============
Smart Agent for Licensing: 5.4.21_rel/77

Reservation Info
================
License reservation: DISABLED
```

Checking license count and usage

The following sample shows the license count and usage details:

```
RP/0/RP0/CPU0:regen#show license usage
Wed Nov 16 15:14:08.648 UTC

License Authorization:
  Status: AUTHORIZED on Nov 16 2022 15:08:48 UTC

NCS1K4 - 400G RG (S_NCS1K4_400G_RG):
  Description: NCS1K4 - 400G REGEN License
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED
```

Smart license model for QXP line card

A smart license model for QXP card is a licensing mechanism that

- uses a specific licensed product ID, **NCS1K4-QXP-L-K9** for the Cisco NCS 1004 QXP line card, and
- calculates license consumption based on the number of trunk slices provisioned.

Examples

Sample configuration of a slice in 400GE TXP modeS

```
RP/0/RP0/CPU0:ios#configure
Thu Oct 21 23:12:12.906 UTC
RP/0/RP0/CPU0:ios(config)#hw-module location 0/1 mxponder-slice 0
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#trunk-rate 400G
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#client-port-rate 1 client-type 400GE
RP/0/RP0/CPU0:ios(config-hwmod-mxp)#commit
```

Checking the slice state

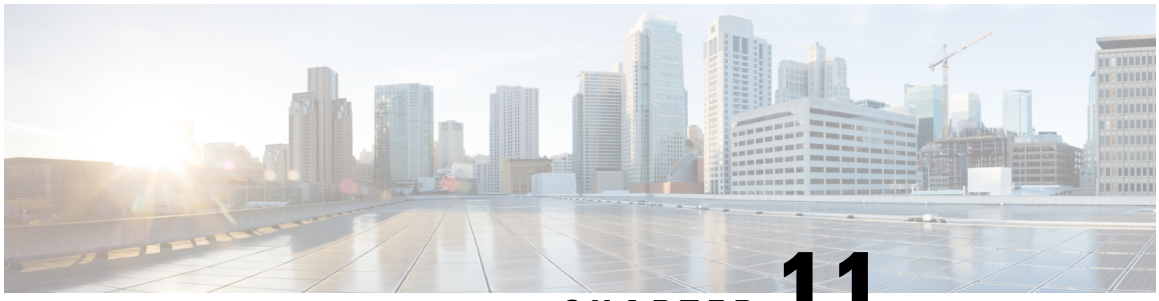
The following sample shows the status of the configured slice as Provisioned.

```
RP/0/RP0/CPU0:ios#show hw-module location 0/1 mxponder-slice 0
Thu Oct 21 23:14:08.700 UTC

Location:             0/1
Slice ID:             0
Client Bitrate:       400GE
Trunk  Bitrate:       400G
Status:               Provisioned
LLDP Drop Enabled:    FALSE
ARP Snoop Enabled:    FALSE
Client Port                     Mapper/Trunk Port          CoherentDSP0/1/0/0
                                Traffic Split Percentage

FourHundredGigECtrlr0/1/0/1              -                            100
```

Checking the license count

The following sample shows the license usage count as 2.

```
RP/0/RP0/CPU0:ios#show license summary
Tue Dec 28 22:35:29.824 UTC

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Jun 26 2022 22:19:25 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Jan 27 2022 22:33:09 UTC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  NCS1K4 - QDD TXP Trunk  (S_N1K4_LIC_TRK)                  2 AUTHORIZED
```



CHAPTER 11

USB Device Automount

- USB automounts, on page 379
- Mount a USB device in virtual machine environments, on page 379
- Unmount a USB device, on page 380

USB automounts

A USB automount is a system feature that

- automatically mounts a USB device in the sysadmin-vm with read and write permissions,
- supports automount only when the USB device file system is FAT or FAT32, and
- allows you to access files and folders on the mounted USB device through the disk2: file system, without needing to explicitly mount the device.

Operational details

- When you insert a USB device into NCS 1004, it is automatically mounted in sysadmin-vm with read and write permissions, and, by default, is unmounted in XR-vm.
- If the USB file system is not FAT or FAT32, the device will not be automounted in sysadmin-vm.
- To access or transfer files, you can use the disk2: file system reference.
- You must unmount the USB device from sysadmin-vm before physically removing it from NCS 1004.
- If you need to remount the USB device after unmounting but before removing it, use the `mount` command.
- You can mount the USB device in both XR-vm and sysadmin-vm at the same time. Before physically removing the USB device from NCS 1004, unmount it from both sysadmin-vm and XR-vm

Mount a USB device in virtual machine environments

Use this task to mount a USB device so you can access, copy, and manage files within your virtual machine environment.

Mounting a USB device integrates external storage into your virtual machine, allowing you to move essential files between systems or for backup purposes. Procedures differ slightly between sysadmin-vm and XR-vm.

Procedure

Step 1 Use the command **usb device operation mount** to mount the USB device in sysadmin-vm.

Example:

```
sysadmin-vm:0_RP0#usb device operation mount
Fri Jul  13 09:26:00.821 UTC success usb mounted
```

Step 2 Use the command **unmount disk2: undo** to mount the USB device in XR-vm.

Example:

```
RP/0/RP0/CPU0:ios#unmount disk2: undo
Fri Jul 13 14:56:34.326 IST
disk2: mounted successfully.
```

Step 3 Use the command **scp** to copy the file to the USB device.

Example:

```
[sysadmin-vm:0_RP0:~/showtech]$scp showtech-envmon-admin-2018-Jul-04.171400.IST.tgz /disk2\:
[sysadmin-vm:0_RP0:~/showtech]$cd /disk2\:
[sysadmin-vm:0_RP0:/disk2:]$ls -lrt
total 122424
drwxr-xr-x 2 root root      8192 Jul 12  2017 System Volume Information
drwxr-xr-x 2 root root      8192 Jun 11 16:16 boot
drwxr-xr-x 3 root root      8192 Jun 11 16:17 EFI
-rwxr-xr-x 1 root root 125306880 Jul 10 13:50 calvVarLog.tar
-rwxr-xr-x 1 root root     23023 Jul 13 05:23 showtech-envmon-admin-2018-Jul-04.171400.IST.tgz
```

The USB device is mounted successfully in the target environment, and files can be copied and managed as needed.

Unmount a USB device

Use this task to safely disconnect a USB device from a sysadmin or XR virtual machine by issuing the appropriate command.

Procedure

Step 1 If you are working in sysadmin-vm, enter the command **usb device operation unmount** to unmount the USB device.

Example:

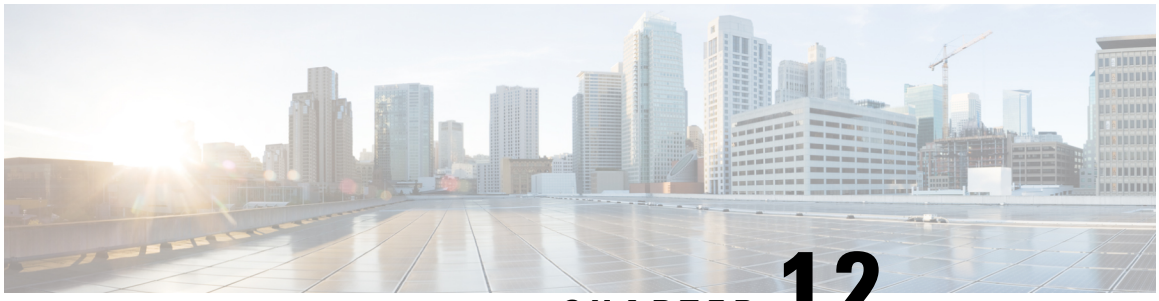
```
sysadmin-vm:0_RP0#usb device operation unmount
Fri Jul  13 09:25:24.531 UTC success usb unmounted
```

Step 2 If you are working in XR-vm, enter the command **unmount disk2:** to unmount the USB device.

Example:

```
RP/0/RP0/CPU0:ios#unmount disk2:
Fri Jul 13 14:56:46.393 IST
disk2: unmounted successfully.
```

The USB device is safely disconnected from the selected virtual machine and ready for physical removal.



CHAPTER 12

Fault Profiles

- Fault profiles, on page 383
- Configure a fault profile, on page 384

Fault profiles

A fault profile is a configuration construct that

- groups fault types generated in a system,
- assigns user-defined severity to each fault, and
- specifies actions such as creating, modifying, or deleting fault profiles and associated alarms

In a system, the default fault list captures all possible faults and their default severity values. The default severity applies when no custom fault profile is attached. Fault profiles can be created for specific scopes, such as the system, line card, node, or port, to override default severity and actions.

If a system profile is attached but you need a different fault profile for a node, you can create a node profile and attach it. The node then inherits its attached node profile properties.

Available severity levels:

- Major
- Minor
- Critical
- Non Faulted
- Non Reported

The defined set of actions for a fault profile are:

- Create and delete a fault profile
- Add alarms to a fault profile
- Remove alarms from a fault profile
- Modify severity of alarm in an existing profile

Best practice for configuring fault profiles

Follow these best practices when configuring fault profiles on Cisco NCS 1004.

- Configure fault profiles only for data path alarms such as Optics, Coherent DSP, Ethernet, and ODU alarms.
- Avoid attempting fault profiling at the port level, as it is not supported.
- Limit the number of fault profiles to a maximum of 61 to prevent configuration errors.

Configure a fault profile

Create a fault profile and apply it to specific hardware or software subsystems so that alarms are triggered with the desired severity and behavior.

Fault profiles allow you to tailor fault management for individual systems, nodes, or propagation domains within Cisco IOS XR devices. By customizing alarm tags and severity levels, you can control how the system responds to faults in various components.

Follow these steps to configure and apply a fault profile:

Procedure

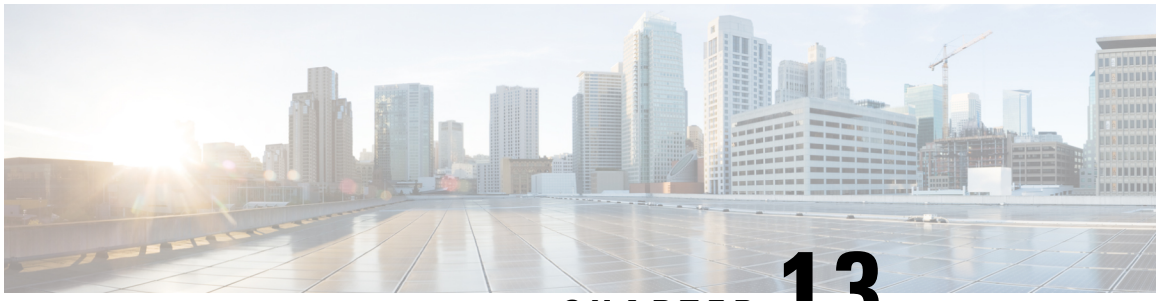
Step 1 Enter the configuration mode.

Step 2 Define a fault profile with a desired name and attributes using the command **fault-profile** *fault_name* **fault identifier subsystem XR fault-type** { **ethernet** | **sdh_controller** | **sonet** | **HW_OPTICS** | **G709** | **CPRI** | **OTS** } **fault-tag** *alarm_name* **severity** { **sas** | **nsas** } *severity_level*. and commit the configuration.

Step 3 Apply the fault profile at system or node level using the command **fault-profile** *fault-name* **apply rack** *rack_id* **slot** { **ALL** | **LC** }.

Step 4 Commit the application and exit the configuration mode.

The fault profile is configured and applied, and alarms are generated according to the severity and locations specified.



CHAPTER 13

OC Support for AAA User

This chapter describes the implementation of the administrative model of *task-based authorization* used to control user access in the software system.

Table 78: Feature History

Feature Name	Release Information	Feature Description
OC support for AAA user	Cisco IOS XR Release 7.3.2	This feature allows all authorized users on XR VM to access administration data on the router through NETCONF or gRPC interface, similar to accessing the CLI. This functionality works by internally mapping the task group of the user on XR VM to a predefined group on System Admin VM. Therefore, the NETCONF and gRPC users can access the administrative information on the router even if their user profiles do not exist on System Admin VM. Command added: • aaa authorization (System Admin-VM)

- OC Support for AAA User, on page 385
- AAA services, on page 386
- Admin access methods for NETCONF and gRPC, on page 387

OC Support for AAA User

This chapter describes the implementation of the administrative model of task-based authorization used to control user access in the software system.

Table 79: Feature History

Feature Name	Release Information	Feature Description
OC support for AAA user	Cisco IOS XR Release 7.3.2	This feature allows all authorized users on XR VM to access administration data on the router through NETCONF or gRPC interface, similar to accessing the CLI. This functionality works by internally mapping the task group of the user on XR VM to a predefined group on System Admin VM. Therefore, the NETCONF and gRPC users can access the administrative information on the router even if their user profiles do not exist on System Admin VM. Command added: • **aaa authorization (System Admin-VM)**

AAA services

AAA is part of the software base package and is available by default.

To configure authentication, authorization, and accounting (AAA) authentication at login, use the **aaa authentication login** command in global configuration mode.

A AAA service is a security framework that

- authenticates users or principals to verify their identities,
- authorizes permissions so authenticated users can perform specific tasks, and
- accounts for user and system activities by recording selected actions and sessions.

AAA (Authentication, Authorization, and Accounting) services are included by default in the software base package. Administrators can use the command set to configure user groups and task groups, ensuring that only authorized individuals have access and that all actions are tracked for regulatory or security purposes.

- Authentication commands verify the identity of users or principals.
- Authorization commands determine if authenticated users or principals are allowed to carry out specific actions.
- Accounting commands log sessions and create audit trails by recording certain user- or system-generated activities.

To configure AAA authentication at login, use the following command in global configuration mode:

```
aaa authentication login
```

Admin access methods for NETCONF and gRPC

A NETCONF/gRPC admin access method is a feature that

- enables all authorized users on the XR VM to securely access administration data on Cisco IOS XR routers using NETCONF or gRPC interfaces,
- internally maps XR VM user task groups to predefined groups on the System Admin VM, and
- allows admin access through NETCONF or gRPC even when user profiles do not exist on the System Admin VM.

NETCONF is an XML-based protocol used over Secure Shell (SSH) transport to configure and manage network devices. gRPC is an open-source remote procedure call framework that allows client applications to request information from the router and make configuration changes.

Before Cisco IOS XR Software Release 7.3.2, users who accessed administrative data through NETCONF, gRPC, or any configuration interface other than the CLI needed to belong to user groups configured directly on the System Admin VM. Otherwise, the router denied access and issued an "UNAUTHORIZED access" error. By default, XR VM only synchronized the first configured user profile to the System Admin VM. If the first user was deleted, the system synchronized the next user in the root-lr group to the System Admin VM only if no other user existed on the System Admin VM. Subsequent XR VM users were not automatically synchronized.

Beginning with Cisco IOS XR Software Release 7.3.2, the system automatically maps authorized users on XR VM to the System Admin VM based on the user's task table. This means NETCONF and gRPC users can access admin-related information on the router, even if their profiles are absent from the System Admin VM. This feature is enabled by default.

Suppose a network administrator creates several user accounts on the XR VM but does not add them explicitly to the System Admin VM. With this feature enabled, all authorized users can use NETCONF or gRPC to access administrative data on the router, streamlining operations and minimizing user account duplication across VMs.

User profile mappings between XR VM and System Admin VM

User privileges to execute commands and access data elements on the router are usually specified using certain command rules and data rules that are created and applied on the user groups.

When the internal process for AAA starts or when you create the first user, the system creates the following set of predefined groups, command rules and data rules in System Admin VM. These configurations are prepopulated to allow users of different groups (such as root-system, admin-r and aaa-r) in System Admin VM

You can use the **show running-configuration aaa** command to view the AAA configurations.

The admin CLI for the user works based on the above configurations. The root-system is the group with the highest privilege in System Admin VM. The admin-r group has only read and execute access to all data. The aaa-r group has access only to AAA data. With the introduction of the admin access feature for all users, the NETCONF and gRPC applications can also access the admin data based on the above rules and groups.

A user profile mapping is an authentication mechanism that

- establishes a direct relationship between user profiles in XR VM and System Admin VM,
- assigns user privileges based on predefined command and data rules for different user groups, and

- streamlines authentication and access control on the system.

User privileges to execute commands and access data elements on the router are defined and enforced by command and data rules assigned to user groups. When the AAA (Authentication, Authorization, and Accounting) process starts, or when you create the first user, the system automatically creates and populates a set of predefined groups, command rules, and data rules in the System Admin VM. These default group assignments, such as root-system, admin-r, and aaa-r, are designed to provide appropriate privilege levels to users in System Admin VM.

You can use the `show running-configuration aaa` command to view the AAA configurations and verify current group and rule assignments.

Example configuration

```
aaa authentication groups group aaa-r gid 100 users %%__system_user__%%
!
aaa authentication groups group admin-r gid 100 users %%__system_user__%%
!
aaa authentication groups group root-system gid 100 users "%%__system_user__%% "
!
aaa authorization cmdrules cmdrule 1 context * command * group root-system ops rx action
accept
!
aaa authorization cmdrules cmdrule 2 context * command "show running-config aaa" group aaa-r
ops rx action accept
!
aaa authorization cmdrules cmdrule 3 context * command "show tech-support aaa" group aaa-r
ops rx action accept
!
aaa authorization cmdrules cmdrule 4 context * command "show aaa" group aaa-r ops rx
action accept
!
aaa authorization cmdrules cmdrule 5 context * command show group admin-r ops rx action
accept
!
aaa authorization datarules datarule 1 namespace * context * keypath * group root-system
ops rwx action accept
!
aaa authorization datarules datarule 2 namespace * context * keypath /aaa group aaa-r ops
r action accept
!
aaa authorization datarules datarule 3 namespace * context * keypath /aaa group admin-r ops
rwx action reject
!
aaa authorization datarules datarule 4 namespace * context * keypath / group admin-r ops r
action accept
```

These key points summarize how user profile mappings and group rules affect system access and privileges:

- The admin CLI operates according to these predefined groups and their rules.
- The root-system group holds the highest privilege level within the System Admin VM.
- The admin-r group has read and execute privileges for all data, while the aaa-r group has access restricted to AAA-related data.
- With admin access available to all users, applications such as NETCONF and gRPC can also access admin data according to these group-based rules and configurations.

Administration data access rules

An administration data access rule is a security policy that

- specifies which users or groups can access administration data on a router,
- defines the permitted operations (such as read-only access) for supported management protocols like NETCONF and gRPC, and
- is implemented through command rules in the AAA authorization framework.

Administration data includes configuration and status information required for monitoring or managing routers via protocols like NETCONF and gRPC. Proper configuration of access rules ensures users can securely retrieve information without having the ability to alter device settings.

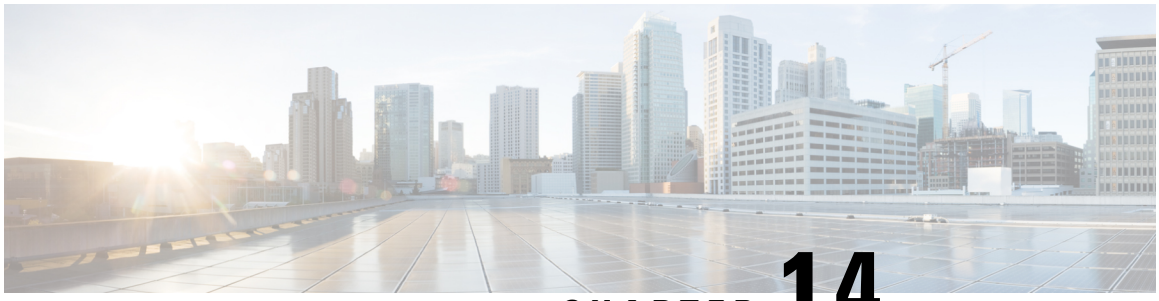
Configuration example

To grant read access to administration data for users in the `admin-r` group via NETCONF or gRPC, configure the following AAA command rule on your router:

```
Router#admin
sysadmin-vm:0_RP0#configure
sysadmin-vm:0_RP0(config)#aaa authorization cmdrules cmdrule 6
sysadmin-vm:0_RP0(config-cmdrule-6)#context netconf
sysadmin-vm:0_RP0(config-cmdrule-6)#command get
sysadmin-vm:0_RP0(config-cmdrule-6)#group admin-r
sysadmin-vm:0_RP0(config-cmdrule-6)#ops rx
sysadmin-vm:0_RP0(config-cmdrule-6)#action accept
sysadmin-vm:0_RP0(config)#commit
```

After configuration, the rule appears in the running configuration as follows:

```
aaa authorization cmdrules cmdrule 6
context netconf
command get
group admin-r
ops rx
action accept
!
```



CHAPTER 14

Implementing Host Services and Applications

Cisco IOS XR software Host Services and Applications features on the router are used primarily for checking network connectivity and the route a packet follows to reach a destination, mapping a hostname to an IP address or an IP address to a hostname, and transferring files between routers and UNIX workstations.

Prerequisites for implementing Host Services and Applications

Ensure to install the relevant optional RPM package before using the host services or applications.

- HTTP client application, on page 391
- TCP , on page 391

HTTP client application

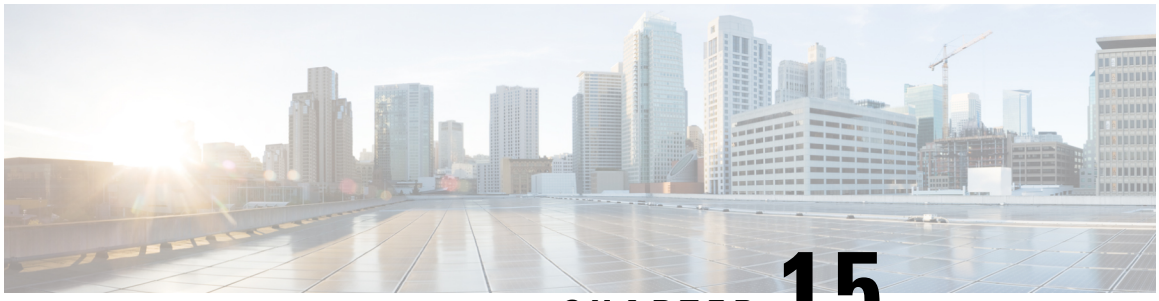
A HTTP client application is a network application that

- allows files to be transferred from HTTP servers to other devices over a network,
- uses the HTTP protocol for communication, and
- provides configurable parameters for connection management and security.

TCP

TCP is a connection-oriented protocol that

- specifies the format of data and acknowledgments that two computer systems exchange to transfer data,
- defines the procedures computers use to ensure that data arrives correctly, and
- allows multiple applications on a system to communicate concurrently by handling all demultiplexing of incoming traffic among the application programs.



CHAPTER 15

Implementing Certification Authority Interoperability

- Certification authority interoperability, on page 393
- How to implement CA interoperability, on page 399
- Authenticate certification authority, on page 405

Certification authority interoperability

Certification authority (CA) interoperability is a security capability on Cisco NCS 1004 that

- supports IPSec, SSL, and SSH protocols.
- provides manageability and scalability for IPSec deployments.
- allows devices to obtain and use digital certificates issued by a CA.

CA interoperability permits devices and CAs to communicate so that a device can obtain and use digital certificates from the CA. Although IPSec can be implemented in a network without the use of a CA, using a CA provides manageability and scalability for IPSec.

Note IPSec is not currently supported on Cisco NCS 1004.

Prerequisites for CA interoperability

Ensure that the user permissions and the certification authority (CA) requirements are in place on Cisco NCS 1004 before you configure CA interoperability.

The prerequisites for implementing CA interoperability on Cisco NCS 1004 include the following requirements:

- You must be in a user group that is associated with a task group that includes the proper task IDs. The command reference guides include the task IDs that are required for each command. If you suspect that user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

- You must have a CA available to your network before you configure this interoperability feature. The CA must support the Cisco Systems PKI protocol and the Simple Certificate Enrollment Protocol (SCEP), which was formerly called Certificate Enrollment Protocol (CEP).

Maximum supported CA key size for interoperability on Cisco NCS 1004

The following restriction applies when configuring CA interoperability on Cisco NCS 1004:

- The restriction for CA interoperability on Cisco NCS 1004 is that the software does not support CA server public keys that are greater than 2048 bits.

Enroll RSA key pairs for device certificates

Secure the Cisco NCS 1004 device by obtaining signed certificates from the certification authority for each RSA key pair.

You must obtain a signed certificate from the CA for each of the RSA key pairs on the Cisco NCS 1004 device. If you generated general-purpose RSA keys, the Cisco NCS 1004 device has only one RSA key pair and needs only one certificate. If you previously generated special usage RSA keys, the Cisco NCS 1004 device has two RSA key pairs and needs two certificates.

Procedure

Step 1 Request certificates for all of your RSA key pairs using the command **crypto ca enroll** *ca-name*.

- This command causes Cisco NCS 1004 to request as many certificates as there are RSA key pairs. Run this command only once, even if you have special usage RSA key pairs.
- This command requires you to create a challenge password that is not saved with the configuration. This password is required if your certificate must be revoked, so you must remember this password.
- A certificate may be issued immediately, or Cisco NCS 1004 sends a certificate request every minute until the enrollment retry period is reached and a timeout occurs. If a timeout occurs, contact your system administrator to get your request approved, and then enter this command again.

Example:

```
RP/0/RP0/CPU0:ios# crypto ca enroll myca
```

Step 2 (Optional) Display information about the CA certificate using the command **show crypto ca certificates**.

Example:

```
RP/0/RP0/CPU0:ios# show crypto ca certificates
```

The Cisco NCS 1004 device receives signed certificates for each RSA key pair from the CA. If the request times out, contact your administrator and retry the enrollment.

Configure certificate enrollment using cut-and-paste

Set up the trustpoint CA that Cisco NCS 1004 uses, and enable manual certificate enrollment through terminal-based cut-and-paste.

Manual cut-and-paste certificate enrollment lets you import the CA certificate and your own certificate by pasting them at the terminal, instead of using a network-based enrollment protocol.

Procedure

Step 1 Enter the XR Config mode.

Example:

```
RP/0/RP0/CPU0:ios# configure
```

Step 2 Declare the CA that Cisco NCS 1004 must use, and enter trustpoint configuration mode. Use the *ca-name* argument to specify the name of the CA.

Example:

```
RP/0/RP0/CPU0:ios(config)# crypto ca trustpoint myca
```

Step 3 Specify manual cut-and-paste certificate enrollment.

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# enrollment terminal
```

Step 4 Save the configuration changes by using the **commit** or **end** command.

Step 5 Authenticate the CA by obtaining the certificate of the CA. Use the *ca-name* argument to specify the name of the CA. Use the same name that you entered in step 2.

Example:

```
RP/0/RP0/CPU0:ios# crypto ca authenticate myca
```

Step 6 Obtain the certificates for Cisco NCS 1004 from the CA. Use the *ca-name* argument to specify the name of the CA. Use the same name that you entered in step 2.

Example:

```
RP/0/RP0/CPU0:ios# crypto ca enroll myca
```

Step 7 Import a certificate manually at the terminal. Use the *ca-name* argument to specify the name of the CA. Use the same name that you entered in step 2.

Note
You must enter the **crypto ca import** command twice if usage keys (signature and encryption keys) are used. The first time the command is entered, one of the certificates is pasted into Cisco NCS 1004; the second time the command is entered, the other certificate is pasted into Cisco NCS 1004. The order in which the certificates are pasted is not significant.

Example:

```
RP/0/RP0/CPU0:ios# crypto ca import myca certificate
```

Step 8 Display information about your certificate and the CA certificate.

Example:

```
RP/0/RP0/CPU0:ios# show crypto ca certificates
```

The following example shows how to configure CA interoperability. Comments are included within the configuration to explain various commands.

```
RP/0/RP0/CPU0:ios# configure
RP/0/RP0/CPU0:ios(config)# hostname mydevice
RP/0/RP0/CPU0:ios(config)# domain name mydomain.com
RP/0/RP0/CPU0:ios(config)# end

Uncommitted changes found, commit them? [yes]:yes

RP/0/RP0/CPU0:ios(config)# crypto key generate rsa mykey

The name for the keys will be:mykey
Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keypair
Choosing a key modulus greater than 512 may take a few minutes.
How many bits in the modulus [1024]:
Generating RSA keys ...
Done w/ crypto generate keypair
[OK]

! The following commands declare a CA and configure a trusted point.

RP/0/RP0/CPU0:ios# configure
RP/0/RP0/CPU0:ios(config)# crypto ca trustpoint myca
RP/0/RP0/CPU0:ios(config)# enrollment url http://xyz-ultra5
RP/0/RP0/CPU0:ios(config)# enrollment retry count 25
RP/0/RP0/CPU0:ios(config)# enrollment retry period 2
RP/0/RP0/CPU0:ios(config)# rsakeypair mykey
RP/0/RP0/CPU0:ios(config)# end

Uncommitted changes found, commit them? [yes]:yes

! The following command authenticates the CA to your device.

RP/0/RP0/CPU0:ios(config)# crypto ca authenticate myca

! The following command requests certificates for all of your RSA key pairs.

RP/0/RP0/CPU0:ios(config)# crypto ca enroll myca

! The following command displays information about your certificate and the CA certificate.

RP/0/RP0/CPU0:ios# show crypto ca certificates
```

Cisco NCS 1004 is successfully configured to use the specified trustpoint CA and manual certificate enrollment via cut-and-paste. Certificates are authenticated, imported, and validated.

Public key pair capabilities in XR Config mode

A public key pair capability in XR Config mode is a security mechanism on Cisco NCS 1004 that

- supports the generation of crypto key pairs in both FIPS (Federal Information Processing Standard) and non-FIPS modes,
- enables selection of multiple key types and key sizes, and
- controls overwriting, deleting, and viewing keys based on XR Config and XR EXEC mode distinctions.

Supported key types for non-FIPS and FIPS modes

The following table lists the key types and key sizes that are supported in non-FIPS and FIPS modes.

Table 80: Supported Key Types for non-FIPS and FIPS modes

Key Types	Non-FIPS mode	FIPS mode
RSA	Supported for all key sizes from 512 to 4096	Supported for key sizes 2048, 3072, and 4096
DSA	Supported for key sizes 512, 768, and 1024	Supported for key size 2048
ECDSA	Supported for key sizes nistp256, nistp384, and nistp512	Supported for key sizes nistp256, nistp384, and nistp512
ED25519	Supported	Not supported

Guidelines and restrictions

When you generate crypto key pairs in XR Config mode, the following guidelines and restrictions apply:

- This feature does not support the generation of **system-root-key** and **system-enroll-key**.
- The key pairs that are generated in XR Config mode overwrite any previously generated key pairs in XR EXEC mode.
- Cisco NCS 1004 does not support overwriting key pairs that are generated in XR Config mode from XR EXEC mode.
- When you run the **no** form of the **crypto key generate** command in XR Config mode, only the keys that are generated in XR Config mode are deleted.
- Cisco NCS 1004 does not support deleting key pairs that are generated in XR Config mode from XR EXEC mode.
- When you run the **crypto key generate** command in XR EXEC mode, the keys that are generated in XR Config mode are not overwritten or deleted.
- The **show crypto key mypubkey** command displays the keys that are generated in XR EXEC mode first, followed by the keys that are generated in XR Config mode.

Configuration exampl

This example shows the creation of key pairs in XR Config mode:

```
RP/0/RP0/CPU0:ios# conf t
RP/0/RP0/CPU0:ios(config)# crypto key generate dsa 512
RP/0/RP0/CPU0:ios(config)# crypto key generate rsa user1 general-keys 2048
RP/0/RP0/CPU0:ios(config)# crypto key generate rsa user2 usage-keys 2048
RP/0/RP0/CPU0:ios(config)# crypto key generate rsa 2048
RP/0/RP0/CPU0:ios(config)# crypto key generate ecdsa nistp256
RP/0/RP0/CPU0:ios(config)# crypto key generate ecdsa nistp384
RP/0/RP0/CPU0:ios(config)# crypto key generate ecdsa nistp521
RP/0/RP0/CPU0:ios(config)# crypto key generate ed25519
RP/0/RP0/CPU0:ios(config)# commit
```

Use the **no** form of the command in XR Config mode to delete any of the key pairs.

System logs and error messages

Cisco NCS 1004 generates the following system logs on successful creation of key pairs:

```
cepki[287]: %SECURITY-CEPKI-6-KEY_INFO : crypto key DSA generated, label:the_default,
modBits:1024
cepki[287]: %SECURITY-CEPKI-6-KEY_INFO : crypto key ECDSA_NISTP256 generated,
label:the_default, modBits:256
```

Cisco NCS 1004 generates the following system logs on deletion of key pairs:

```
cepki[287]: %SECURITY-CEPKI-6-KEY_INFO : crypto key RSA zeroized, label:user1
cepki[287]: %SECURITY-CEPKI-6-KEY_INFO : crypto key DSA zeroized, label:the_default
```

Cisco NCS 1004 generates the following error messages if you try to overwrite key pairs that are generated in XR Config mode from XR EXEC mode:

```
RP/0/RP0/CPU0:ios# conf t
RP/0/RP0/CPU0:ios(config)# crypto key generate ed25519
RP/0/RP0/CPU0:ios(config)# commit
RP/0/RP0/CPU0:ios(config)# crypto key generate ed25519
Cannot execute the command : Operation not permitted
ce_cmd[68727]: %SECURITY-CEPKI-6-ERR_2 : Cannot      execute the command :   Operation not
 permitted
ce_cmd[68736]: %SECURITY-CEPKI-6-ERR : Key is added as part of config mode, key deletion
is not allowed , delete key from config mode
```

Cisco NCS 1004 generates the following error messages if you try to delete key pairs that are generated in XR Config mode from XR EXEC mode:

```
RP/0/RP0/CPU0:ios# conf t
RP/0/RP0/CPU0:ios(config)# crypto key generate ed25519
RP/0/RP0/CPU0:ios(config)# commit
RP/0/RP0/CPU0:ios(config)# crypto key zeroize ed25519
Cannot execute the command : Operation not permitted
ce_cmd[68736]: %SECURITY-CEPKI-6-ERR_2 : Cannot execute the command :   Operation not
permitted
```

Viewing the generated key pairs

You can view the key pairs that are generated in XR Config mode, listed under **Public keys from config sysdb** in the following command output:

```
RP/0/RP0/CPU0:ios# show crypto key mypubkey ecdsa
Key label: the_default
Type     : ECDSA General Curve Nistp256
Degree   : 256
Created  : 11:49:22 IST Wed Apr 21 2021
Data     :
 04D6D132 2253ABD0 81449E3F 9D5CEA3A 1107950A 829E9090 8960FBD5 ABA039B7
 24A4E217 7EA47475 91C60AC7 013DBC2E EA8434D9 0BD5B0FC 694913AE 0098A4F5
 77

Key label: the_default
Type     : ECDSA General Curve Nistp521
Degree   : 521
Created  : 22:44:22 IST Thu Mar 18 2021
Data     :
 04017798 4369F493 8D0E57D1 1975FC46 CDC03A78 03A9F90E B38CA504 17DB9A64
 D1DEA6A6 D23E7E20 4D8D4D31 C7878BDB BF5EEE40 1978A889 70C5D703 BB033B77
 0FFD9201 366A9AC8 35E69BB3 97FF4E91 6B498510 39425971 C5E43858 83286088
 A6A7BF92 0EA2B416 BD4E81CE DCEB65F1 15CC75B5 91204E89 3339A168 2382CAB6
```

```
 40170131 8F

----------------------------
 Public keys from config sysdb:
 ----------------------------
 Key label: the_default
Type     : ECDSA General Curve Nistp384
Degree   : 384
Created  : 11:51:52 IST Wed Apr 21 2021
Data     :
 045F7C14 1A88C27E 9CED3FF1 7FEDFA03 B49575FA 7AD88370 BC9C7D7F F99C8917
 33620916 758BDEFC 7187E33A 2D3CCD33 14FF3267 9855A5E9 E3BD166C CE838462
 40742231 6198EE12 3E189F42 22A8149A 8E7B186D 88E728D4 7F47D565 53441061
 79
```

How to implement CA interoperability

This process enables Cisco NCS 1004 devices to obtain digital certificates from a certification authority (CA) by ensuring device identity, key generation, and trustpoint configuration before requesting certificates.

Summary

The key components involved in the process are:

- Network administrator: Configures the device identity, generates RSA key pairs, and declares the trustpoint that points to the CA.
- Cisco NCS 1004: Stores the keys and trustpoint configuration, and uses them to interact with the CA.
- Certification authority (CA): Issues the digital certificate that the device uses to establish trust with peers.

The process configures the prerequisites that the device needs before it can authenticate the CA and request certificates.

Workflow

The process involves the following stages:

1. Configure the hostname and IP domain name of Cisco NCS 1004 so that the device can build the fully qualified domain name (FQDN) that is used by IPSec keys and certificates.

   For details, see Configure hostname and IP domain name.

2. Generate an RSA key pair on Cisco NCS 1004. The device uses RSA keys to sign and encrypt IKE key management messages and to obtain a certificate from the CA.

   For details, see Generate RSA key pair.

3. Import the public key into Cisco NCS 1004 so that the device can authenticate the user.

   For details, see Import public key to Cisco NCS 1004.

4. Declare the CA and configure the trusted point so that Cisco NCS 1004 can verify certificates issued to peers.

   For details, see Declare certification authority and configure trusted point.

Result

Cisco NCS 1004 is ready to authenticate the CA and request its own certificates.

What's next

After you complete this process, authenticate the CA on Cisco NCS 1004. For details, see Authenticate certification authority.

Configure the hostname and IP domain name on Cisco NCS 1004

Use this task to configure the hostname and IP domain name of Cisco NCS 1004.

You must configure the hostname and IP domain name of Cisco NCS 1004 if they are not already configured. The hostname and IP domain name are required because Cisco NCS 1004 assigns a fully qualified domain name (FQDN) to the keys and certificates that are used by IPSec, and the FQDN is based on the hostname and IP domain name that you assign to the Cisco NCS 1004 device. For example, a certificate that is named *ncs1k.example.com* is based on the Cisco NCS 1004 hostname of *ncs1k* and a device IP domain name of *example.com*.

Procedure

Step 1 Enter the XR Config mode.

Example:

```
RP/0/RP0/CPU0:ios#configure
```

Step 2 Configure the hostname of the Cisco NCS 1004 device.

Example:

```
RP/0/RP0/CPU0:ios(config)# hostname myhost
```

Step 3 Configure the IP domain name of Cisco NCS 1004.

Example:

```
RP/0/RP0/CPU0:ios(config)# domain name mydomain.com
```

Step 4 Save the configuration changes by using the **commit** or **end** command.

Example:

```
RP/0/RP0/CPU0:ios(config)# commit
RP/0/RP0/CPU0:ios(config)# end
```

The hostname and IP domain name are configured, enabling Cisco NCS 1004 to generate the FQDN required for IPSec keys and certificates.

Generate an RSA key pair on Cisco NCS 1004

Enable secure key management and certificate enrollment for Cisco NCS 1004 devices by generating RSA key pairs.

RSA key pairs are required to sign and encrypt IKE key management messages and to obtain certificates from a certification authority. Cisco NCS 1004 automatically generates RSA keys at boot, but you should manually generate keys if none are present. Details for both RSA and DSA keys appear in the running configuration.

Note

- RSA keys are auto-generated when Cisco NCS 1004 boots. Generate the key pair only if the RSA key pair is missing on the device.
- The details of RSA and DSA keys are displayed in the running configuration.

Procedure

Step 1 Generate the RSA key pairs using the command **crypto key generate rsa** [**usage-keys** | **general-keys**] [*keypair-label*]

- Use the **usage-keys** keyword to specify special usage keys, or use the **general-keys** keyword to specify general-purpose RSA keys.
- The *keypair-label* argument is the RSA key pair label that names the RSA key pairs.
- You can also configure this command from XR Config mode. For details about generating key pairs in XR Config mode, see Public key pair generation in XR Config mode.

To delete the RSA keys, use the **no** form of the command: **no crypto key generate rsa**.

Example:

```
RP/0/RP0/CPU0:ios# crypto key generate rsa general-keys
```

Step 2 (Optional) Delete all RSA keys from Cisco NCS 1004.

crypto key zeroize rsa [*keypair-label*]

- Run the **crypto key zeroize** command only in EXEC mode.
- Delete all RSA keys from Cisco NCS 1004 if you suspect that the keys are compromised and must no longer be used.
- To remove a specific RSA key pair, use the *keypair-label* argument.
- You can also delete key pairs with the **no** form of the command from XR Config mode. For details about deleting key pairs in XR Config mode, see Public key pair generation in XR Config mode.

Example:

```
RP/0/RP0/CPU0:ios# crypto key zeroize rsa key1
```

Step 3 (Optional) Display the RSA public keys for Cisco NCS 1004.

show crypto key mypubkey rsa

The **show running-config** command also displays the RSA keys. The keys in the following example are in OpenSSL format.

Note
Only those keys that are generated in XR Config mode are visible in the running configuration.

Example:

```
RP/0/RP0/CPU0:ios# show crypto key mypubkey rsa
Fri Mar 27 14:00:20.954 IST
Key label: system-root-key
Type : RSA General purpose
Size : 2048
Created : 01:13:10 IST Thu Feb 06 2020
Data :
 30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
 00A93DE0 1E485EE3 0E7F0964 C48361D1 B6014BE7 A303D8D6 F7790E92 88E69C4B
 B97B7A9C D1B277E3 1569093C 82BD3258 7F67FB49 94860ECD 34498F1F 59B45757
 F32C8E8F 7CEE23EC C36A43D1 9F85C0D9 B96A14DD DD3BBD4C A1FB0888 EED210A7
 39D9A403 7ACE0F6E 39107226 CA621AD8 6E8102CA 9761B86F D33F2871 9DD16559
 AFCB4729 EFCEDBAF 83DF76E4 9A439844 EE3B1180 4022F575 99E11A2C E25BB23D
 9DD74C81 4E5C1345 D9E3CC79 1B98B1AA 6C06F004 22B901EC 36C099FE 10DE2622
 EB7CE618 9A555769 12D94C90 D9BEE5EA A664E7F6 4DF8D8D4 FE7EAB07 1EF4FEAB
 22D9E55F 62BA66A0 72153CEC 81F2639F B5F2B5C5 25E10364 19387C6B E8DB8990
 11020301 0001
Key label: system-enroll-key
Type : RSA General purpose
Size : 2048
Created : 01:13:16 IST Thu Feb 06 2020
Data :
 30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
 009DBC14 C83604E4 EB3D3CF8 5BA7FDDB 80F7E85B 427332D8 BBF80148 F0A9C281
 49F87D5C 0CEBA532 EBE797C5 7F174C69 0735D13A 493670CB 63B04A12 4BCA7134
 EE0031E9 047CAA1E 802030C5 6071E8C2 F8ECE002 CC3B54E7 5FD24E5C 61B7B7B0
 68FA2EFA 0B83799F 77AE4621 435D9DFF 1D713108 37B614D3 255020F9 09CD32E8
 82B07CD7 01A53896 6DD92B5D 5119597C 98D394E9 DBD1ABAF 6DE949FE 4A8BF1E7
 851EB3F4 60B1114A 1456723E 063E50C4 2D410906 BDB7590B F1D58480 F3FA911A
 6C9CD02A 58E68D04 E94C098F 0F0E81DB 76B40C55 64603499 2AC0547A D652412A
 BCBBF69F 76B351EE 9B2DF79D E490C0F6 92D1BB97 B905F33B FAB53C20 DDE2BB22
 C7020301 0001
```

RSA key pairs are generated on Cisco NCS 1004, enabling secure communication and certificate management. You can delete or display keys as needed for maintenance or troubleshooting.

Import a public key to Cisco NCS 1004

Import a public key to Cisco NCS 1004 to enable user authentication prior to digital certificate requests.a public key to Cisco NCS 1004.

Use this task when you need Cisco NCS 1004 to authenticate a user with a public key before requesting a certificate from the certification authority.

Procedure

Step 1 Import the RSA public key using the command **crypto key import authentication rsa** [**usage-keys** | **general-keys**] [*keypair-label*].

- Use the **usage-keys** keyword to specify special usage keys, or use the **general-keys** keyword to specify general-purpose RSA keys.
- The *keypair-label* argument is the RSA key pair label that names the RSA key pairs.

Example:

```
RP/0/RP0/CPU0:ios# crypto key import authentication rsa general-keys
```

Step 2 (Optional) Display the RSA public keys for Cisco NCS 1004 using the **show crypto key mypubkey rsa** command.

The **show running-config** command also displays the RSA keys. The keys in the following example are in OpenSSL format.

Note
Only those keys that are generated in XR Config mode are visible in the running configuration.

Example:

```
RP/0/RP0/CPU0:ios# show crypto key mypubkey rsa
Fri Mar 27 14:00:20.954 IST
Key label: system-root-key
Type : RSA General purpose
Size : 2048
Created : 01:13:10 IST Thu Feb 06 2020
Data :
 30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
 00A93DE0 1E485EE3 0E7F0964 C48361D1 B6014BE7 A303D8D6 F7790E92 88E69C4B
 B97B7A9C D1B277E3 1569093C 82BD3258 7F67FB49 94860ECD 34498F1F 59B45757
 F32C8E8F 7CEE23EC C36A43D1 9F85C0D9 B96A14DD DD3BBD4C A1FB0888 EED210A7
 39D9A403 7ACE0F6E 39107226 CA621AD8 6E8102CA 9761B86F D33F2871 9DD16559
 AFCB4729 EFCEDBAF 83DF76E4 9A439844 EE3B1180 4022F575 99E11A2C E25BB23D
 9DD74C81 4E5C1345 D9E3CC79 1B98B1AA 6C06F004 22B901EC 36C099FE 10DE2622
 EB7CE618 9A555769 12D94C90 D9BEE5EA A664E7F6 4DF8D8D4 FE7EAB07 1EF4FEAB
 22D9E55F 62BA66A0 72153CEC 81F2639F B5F2B5C5 25E10364 19387C6B E8DB8990
 11020301 0001
Key label: system-enroll-key
Type : RSA General purpose
Size : 2048
Created : 01:13:16 IST Thu Feb 06 2020
Data :
 30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
 009DBC14 C83604E4 EB3D3CF8 5BA7FDDB 80F7E85B 427332D8 BBF80148 F0A9C281
 49F87D5C 0CEBA532 EBE797C5 7F174C69 0735D13A 493670CB 63B04A12 4BCA7134
 EE0031E9 047CAA1E 802030C5 6071E8C2 F8ECE002 CC3B54E7 5FD24E5C 61B7B7B0
 68FA2EFA 0B83799F 77AE4621 435D9DFF 1D713108 37B614D3 255020F9 09CD32E8
 82B07CD7 01A53896 6DD92B5D 5119597C 98D394E9 DBD1ABAF 6DE949FE 4A8BF1E7
 851EB3F4 60B1114A 1456723E 063E50C4 2D410906 BDB7590B F1D58480 F3FA911A
 6C9CD02A 58E68D04 E94C098F 0F0E81DB 76B40C55 64603499 2AC0547A D652412A
 BCBBF69F 76B351EE 9B2DF79D E490C0F6 92D1BB97 B905F33B FAB53C20 DDE2BB22
 C7020301 0001
```

The RSA public key is imported to Cisco NCS 1004, enabling the device to authenticate the user before requesting a digital certificate.

Declare a certification authority and configure a trusted point

Set up the certificate infrastructure by declaring a certification authority and configuring a trusted point on Cisco NCS 1004, allowing secure verification of certificates issued to peers.

Configuring a trusted point that references the CA enables Cisco NCS 1004 to request and verify certificates issued to peer devices for secure communication.

Procedure

Step 1 Enter the XR Config mode.

Example:

```
RP/0/RP0/CPU0:ios# configure
```

Step 2 Declare the CA, configure a trusted point with a selected name so that Cisco NCS 1004 can verify the certificates that are issued to peers, and enter trustpoint configuration mode.

Example:

```
RP/0/RP0/CPU0:ios(config)# crypto ca trustpoint myca
```

Step 3 Specify the URL of the CA. The URL must include any non-standard cgi-bin script location.

enrollment url *CA-URL*

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# enrollment url http://ca.domain.com/certsrv/mscep/mscep.dll
```

Step 4 (Optional) Specify the location of the LDAP server if your CA system supports the LDAP protocol.

query url *LDAP-URL*

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# query url ldap://my-ldap.domain.com
```

Step 5 (Optional) Specify a retry period usnig the command **enrollment retry period** *minutes*.

- After Cisco NCS 1004 requests a certificate, the device waits to receive a certificate from the CA. If the device does not receive a certificate within the retry period, it sends another certificate request.
- The range is from 1 to 60 minutes. The default is 1 minute.

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# enrollment retry period 2
```

Step 6 (Optional) Specify how many times Cisco NCS 1004 continues to send unsuccessful certificate requests before giving up. The range is from 1 to 100.

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# enrollment retry count 10
```

Step 7 (Optional) Use the **rsakeypair** *keypair-label* command to specify a named RSA key pair that is generated by using the **crypto key generate rsa** command for this trustpoint. If you do not set this key pair, the trustpoint uses the default RSA key in the current configuration.

Example:

```
RP/0/RP0/CPU0:ios(config-trustp)# rsakeypair mykey
```

Step 8 Save the configuration changes by using the **commit** or **end** command.

Authenticate certification authority

Use this task to authenticate the CA to your Cisco NCS 1004 device.

The Cisco NCS 1004 device must authenticate the CA by obtaining the self-signed certificate of the CA, which contains the public key of the CA. Because the certificate of the CA is self-signed (the CA signs its own certificate), manually authenticate the public key of the CA by contacting the CA administrator to compare the fingerprint of the CA certificate.

Procedure

Step 1 Authenticate the CA to your Cisco NCS 1004 device by obtaining a CA certificate that contains the public key for the CA.

crypto ca authenticate *ca-name*

Example:

```
RP/0/RP0/CPU0:ios# crypto ca authenticate myca
```

Step 2 (Optional) Display information about the CA certificate.

show crypto ca certificates

Example:

```
RP/0/RP0/CPU0:ios# show crypto ca certificates
```

Multi-tier certificate authority for trustpoint authentication

Multi-tier certificate authority for trustpoint authentication is a security capability on Cisco NCS 1004 that lets a complete CA hierarchy, from the Root CA to the subordinate CA that issues the certificate, be imported as part of a single terminal-based authentication request.

- Supports a maximum of 8 tiers, that is, a chain of CAs with one Root CA and seven subordinate CAs, for trustpoint authentication.
- Lets you import the complete CA hierarchy in a single authentication request.
- Provides flexibility and security in network topologies that use a multi-tier CA hierarchy for enrollment.

Feature history for multi-tier certificate authority for trustpoint authentication

Table 81: Feature History Table

Feature Name	Release Information	Description
Multi-Tier Certificate Authority for Trustpoint Authentication	Cisco IOS XR Release 7.10.1	Apart from the Root certificate authority (CA), you can now use a subordinate CA to issue certificates and authenticate your network devices. This feature is beneficial when you have an existing CA hierarchy where it is not the Root CA but the subordinate CA that issues the leaf or certificates. In earlier releases, you could associate only a single CA, not a multi-tier CA, to a trustpoint. And, you could use only the Root CA certificate to enroll the certificates. This feature modifies the show crypto ca certificates command to display the **Trusted Certificate Chain** field.

Need for multi-tier certificate authority for trustpoint authentication

During terminal-based enrollment of a CA trustpoint, Cisco IOS XR network devices accepted only Root CA certificates. Some network topologies use a multi-tier CA hierarchy for enrollment because it provides more flexibility and security. From Cisco IOS XR Release 7.10.1 and later, as part of terminal-based authentication, you can import a complete CA hierarchy (from the Root CA to the subordinate CA that issues the certificate) as part of a single authentication request. With this feature, you can provide a certificate chain that includes the Root CA and intermediate subordinate CAs as part of the terminal-based enrollment process. This feature is useful if you have an existing multi-tier CA hierarchy where the Root CA does not issue any certificates directly, and if you want only subordinate CAs to issue certificates to authenticate your network devices.

Using multi-tier CA for trustpoint authentication

The **crypto ca authenticate** command is used to authenticate a trustpoint with a multi-tier CA hierarchy. You must use only Privacy Enhanced Mail (PEM)-encoded certificates for trustpoint authentication that uses multi-tier CAs. The enrollment process is the same as enrollment that uses a single-tier CA, except that Cisco NCS 1004 displays a console message that prompts you to use only PEM-encoded certificates.

Prerequisite for using multi-tier CA for trustpoint authentication

You must generate a key pair, import a public key, and configure a trustpoint on Cisco NCS 1004 as detailed in the previous sections.

Configuration example for multi-tier CA trustpoint authentication

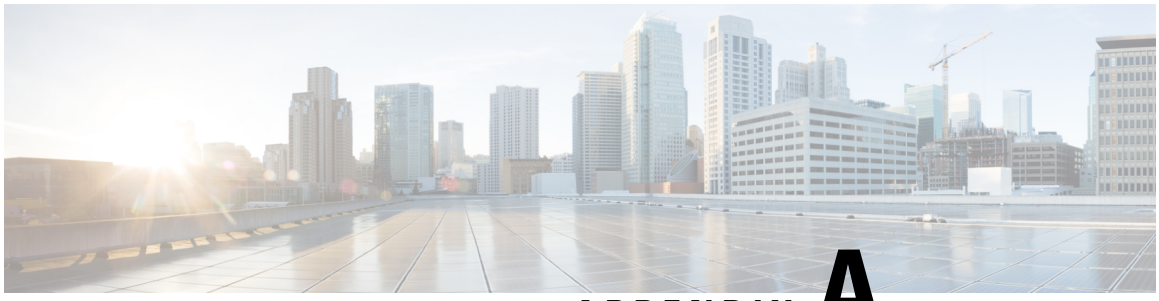
The following example shows the trustpoint authentication for a multi-tier CA on Cisco NCS 1004.

```
RP/0/RP0/CPU0:ios#crypto ca authenticate test-ca
Mon Feb  6 08:17:48.943 UTC


Enter the base 64/PEM encoded certificate/certificates.
Please note: for multiple certificates use only PEM
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIF5TCCA82gAwIBAgICEAEwDQYJKoZIhvcNAQELBQAwXTELMAkGA1UEBhMCSU4x
CzAJBgNVBAgMAktBMQwwCgYDVQQHDANCR0wxDTALBgNVBAoMBENTQ08xDTALBgNV
.
.
.
/4UzeeX6ll0gGJVbDwGeIZTH00artqxHquKQ2P7eXQ1pg0PRNRqWN90SvT5yE33N
eHgbtvdHg1K6K6IAj/NGnd7xUrA1TQ4bdmouCNkgbXM/G9DwgkOOvZ8KYRP9JW57
LYIv2ZcRS2vdnZRD9JPGVig2EgcfVPtj+Q==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIF9TCCA92gAwIBAgIUD6AGesleqedhorkrJ9HWjz1RQzswDQYJKoZIhvcNAQEL
BQAwXTELMAkGA1UEBhMCSU4xCzAJBgNVBAgMAktBMQwwCgYDVQQHDANCR0wxDTAL
.
.
.
+6rMWd6BmfSy2PT3Qz5AjO2+3N1dd67qRRrX7skklkX4JXY42n5/l9PQtSp0wTBh
uy5yUAagynu0z07GczE7E9V+tJHRmNTbnd8pxLk4lTwqtiCIXwQLZA75SkwCS5wh
fn7OrV7uFjMaggNkvj0kSSOkWxqJ+j/KqMAA2zQMUV+qdvT6i+ZV44U=
-----END CERTIFICATE-----
Serial Number  : 10:01
  Subject:
        CN=SUB_CA_CERT,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Issued By      :
        CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Validity Start : 12:31:40 UTC Sun Jun 14 2020
  Validity End   : 12:31:40 UTC Wed Jun 12 2030

  CRL Distribution Point
        http://10.105.236.78/crl_akshath_two_level_ca/crl.der
  SHA1 Fingerprint:
         D8E0C11ECED96F67FDBC800DB6A126676A76BD62
  Serial Number  : 0F:A0:06:7A:C9:5E:A9:E7:61:A2:B9:2B:27:D1:D6:8F:3D:51:43:3B
  Subject:
        CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Issued By      :
        CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Validity Start : 13:12:32 UTC Sun Jun 07 2020
  Validity End   : 13:12:32 UTC Sat Jun 02 2040

  CRL Distribution Point
        http://10.105.236.78/crl_akshath_two_level_ca/crl.der
  SHA1 Fingerprint:
         08E71248FB7578614442E713AC87C461D173952F

CA Certificate validated using issuer certificate.
RP/0/RP0/CPU0:ios#
```

Verification of multi-tier CA trustpoint authentication

Use the **show crypto ca certificates test-ca** command to view the CA certificate chain. The command output displays the **Trusted Certificate Chain** field if one or more subordinate CAs are involved in the hierarchy.

```
RP/0/RP0/CPU0:ios#show crypto ca certificates test-ca
Mon Feb  6 09:03:53.019 UTC

Trustpoint       : test-ca
==================================================
CA certificate
  Serial Number  : 10:01
  Subject:
            CN=SUB_CA_CERT,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Issued By      :
            CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Validity Start : 12:31:40 UTC Sun Jun 14 2020
  Validity End   : 12:31:40 UTC Wed Jun 12 2030

  CRL Distribution Point
            http://10.105.236.78/crl_akshath_two_level_ca/crl.der
  SHA1 Fingerprint:
             D8E0C11ECED96F67FDBC800DB6A126676A76BD62
Trusted Certificate Chain
  Serial Number  : 0F:A0:06:7A:C9:5E:A9:E7:61:A2:B9:2B:27:D1:D6:8F:3D:51:43:3B
  Subject:
            CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Issued By      :
            CN=TWO-LEVEL-CA,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Validity Start : 13:12:32 UTC Sun Jun 07 2020
  Validity End   : 13:12:32 UTC Sat Jun 02 2040

  CRL Distribution Point
            http://10.105.236.78/crl_akshath_two_level_ca/crl.der
  SHA1 Fingerprint:
             08E71248FB7578614442E713AC87C461D173952F
certificate
  Key usage      : General Purpose
  Status         : Available
  Serial Number  : 28:E5
  Subject:
            CN=test
  Issued By      :
            CN=SUB_CA_CERT,OU=SPBU,O=CSCO,L=BGL,ST=KA,C=IN
  Validity Start : 08:49:54 UTC Mon Feb 06 2023
  Validity End   : 08:49:54 UTC Wed Mar 08 2023
  SHA1 Fingerprint:
             6C8644FA67D9CEBC7C5665C35838265F578835AB
Associated Trustpoint: test-ca
```



APPENDIX A

Supported SNMP MIBs on NCS 1004

Supported SNMP MIBs

NCS 1004 supports the following SNMP MIBs:

- CISCO-AM-SNMP-MIB
- CISCO-CONFIG-MAN-MIB
- CISCO-FLASH-MIB
- CISCO-ENTITY-REDUNDANCY-MIB
- CISCO-SYSTEM-MIB
- CISCO-ENTITY-ASSET-MIB
- EVENT-MIB
- DISMAN-EXPRESSION-MIB
- CISCO-FTP-CLIENT-MIB
- NOTIFICATION-LOG-MIB
- CISCO-RF-MIB
- RADIUS-AUTH-CLIENT-MIB
- RADIUS-ACC-CLIENT-MIB
- IEEE8023-LAG-MIB
- CISCO-TCP-MIB
- UDP-MIB
- CISCO-BULK-FILE-MIB
- CISCO-CONTEXT-MAPPING-MIB
- CISCO-OTN-IF-MIB
- CISCO-ENHANCED-MEMPOOL-MIB
- CISCO-PROCESS-MIB

Supported SNMP MIBs on NCS 1004

- CISCO-SYSLOG-MIB
- ENTITY-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-IF-EXTENSION-MIB
- RMON-MIB
- HC-RMON-MIB
- CISCO-OPTICAL-MIB
- CISCO-ENTITY-SENSOR-MIB
- LLDP-MIB
- CISCO-ALARM-MIB
- IEEE8021-SECY-MIB (only SNMP read-only operations are supported for this MIB)

The following table provides more information about SNMP MIBs and related documentation links.

Task	Link
Determine the MIB definitions	SNMP Object Navigator
Configure SNMP	Configure SNMP
Understand SNMP best practices about the recommended order of SNMP query, maximum cache hit, and SNMP retry and timeout recommendation	SNMP Best Practices

Make sure that you configure snmp-server community as the SystemOwner to have the admin-plane parameters to appear to entity MIB. The parameters of fans and power supply units are examples of admin-plane parameters.