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

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

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

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

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

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

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

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

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

© 2019 Cisco Systems, Inc. All rights reserved.
CONTENTS

PREFACE

New and Changed Information  vii

CHAPTER 1

Configuring Slices  1

- Understanding Cisco NCS 1002  1
- Slice and Port Numbering  2
- Supported Configurations  2
- Configure the Slice  3
- Verify Slice Configuration  5
- Illustrations for Supported Configurations  8

CHAPTER 2

Configuring Controllers  11

- Optics Controllers  11
- Maintenance Mode  12
- Configure Optics Controller  12
- Configure Wavelength  13
- Breakout Mode  13
- Laser Squelching  14
- LLDP Snooping  14
- LLDP Snoop and Drop  16
  - Configuring Slices and LLDP Drop at Slice Level  16
  - Verifying the Status of LLDP Drop  17
- Disabling LLDP Drop  19
- Configure Ethernet Controller  19
- Configure the Coherent DSP Controller  20
- Configure Loopback  20
- Restore Factory Settings  21
# Configuration Guide for Cisco NCS 1002, IOS XR Release 6.3.x

## Headless Mode

- View the Headless Statistics

## Chapter 3
### Configuring IP Access List
- IP Access List
  - Configure IP Accesss List
  - Verify IPv6 ACL

## Chapter 4
### Configuring Performance Monitoring
- Configure PM Parameters
- View PM Parameters
- Configuring Pseudo Random Binary Sequence
  - Performance Monitoring Pseudo Random Binary Sequence
  - Verifying PRBS

## Chapter 5
### Configuring MACsec Encryption
- MACsec Frame Format
- MACsec SECTag Format
- MACsec Key Agreement
- MACsec in NCS 1002
- Supported Configurations in Encrypted Mode
- Illustrations for Supported Configurations in Encrypted Mode
- Configure MACsec Encryption Using PSK Authentication
- MACsec Key Chain
- Configure MACsec Key Chain
- Verify MACsec Key Chain
- MACsec Policy
- Configure MACsec Policy
- Verify MACsec Policy
- MACsec Controllers
- Configure the Slice
- Verify Slice Configuration
- Apply MACsec Configuration on MACsec Controller
- Verify MACsec Configuration on MACsec Controller
CHAPTER 6

Configuring GMPLS UNI 79

Configuring GMPLS UNI 81

Configure LMP and Alien Wavelength in NCS 2000 Series Node 82

Configure Unnumbered LMP in NCS 2000 Series Node 83

Retrieve Ifindex from NCS 2000 Series Node 83

Configure LMP in Cisco NCS 1002 84

Configure RSVP in NCS 1002 84

Configure MPLS Tunnel in NCS 1002 85

Headless Mode and GMPLS UNI 85

Display GMPLS UNI Tunnel, RSVP, and LMP Information 85

Example of MPLS Tunnel Creation Without ERO 90

Example of MPLS Tunnel Creation with ERO 91

Example of MPLS Tunnel Creation with XRO 92

Example of MPLS Tunnel Creation with Explicit Signaled Wavelength 93
New and Changed Information

See the Workflow document to refer the other guides of NCS 1002.

This table summarizes new and changed information for configuration guide for Release 6.3.2, and lists where the features are documented.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Where Documented</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMPLS UNI Flexible Grid</td>
<td>The user can create a GMPLS optical channel trail (OCH Trail) in a network where the NCS 1002 node is connected to a NCS 2000 series node. GMPLS UNI flexible grid is supported from R6.3.2 that supports 250G channels and 6.25 GHz channel spacing.</td>
<td>Configuring GMPLS UNI, on page 79</td>
</tr>
<tr>
<td>IPv6 ACL</td>
<td>NCS 1002 supports the following IP Access List (ACL):</td>
<td>Configure IP Accesss List, on page 24</td>
</tr>
<tr>
<td></td>
<td>• Ingress ACL for both IPv4 and IPv6.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Egress ACL: Self-Originated Packet is not supported by ACL, as this is already controlled by user. Only forwarded packets or traffic is classified under ACL. This rule is applicable for both IPv4 and IPv6 ACL.</td>
<td></td>
</tr>
<tr>
<td>MACsec SNMP</td>
<td>The following MIB is supported in NCS 1002.</td>
<td>Configuring SNMP, on page 111</td>
</tr>
<tr>
<td></td>
<td>IEEE8021-SECY-MIB (only SNMP read-only operations are supported for this MIB).</td>
<td></td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
<td>Where Documented</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>MACsec Threshold Crossing Alerts</td>
<td>The user can configure MACsec Threshold Crossing Alerts (TCA) at mac-sec ether, secy-if (interface), secy-tx, and secy-tx. There is no default threshold, minimum, or maximum threshold to configure MACsec TCA. The user must enable MACsec controllers to view MACsec performance. To configure MACsec threshold crossing alerts and the performance monitoring parameters, see the Configuring MACsec Encryption chapter in the Configuration Guide for Cisco NCS 1002.</td>
<td>Configuring MACsec Threshold Crossing Alerts, on page 61</td>
</tr>
<tr>
<td>MACsec MKA Using EAP-TLS</td>
<td>Using IEEE 802.1X port-based authentication with Extensible Authentication Protocol (EAP-TLS), MACsec MKA can be configured between two NCS 1002 device ports. EAP-TLS allows mutual authentication and obtains MSK (master session key). Both Connectivity Association Key Name (CKN) and connectivity association key (CAK) are derived from MSK for MKA operations. The device certificates are carried for authentication to the external AAA server using EAP-TLS.</td>
<td>MACsec MKA Using EAP-TLS Authentication, on page 67</td>
</tr>
<tr>
<td>Mixed Mode Configuration</td>
<td>The first three client ports of a slice can be configured at 100G bitrate and the last two client ports can be configured at 10G bitrate per lane. This feature is called mixed mode configuration.</td>
<td>Configuring Slices, on page 1</td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
<td>Where Documented</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>------------------</td>
</tr>
<tr>
<td>PRBS</td>
<td>Pseudo Random Binary Sequence (PRBS) feature allows the user to perform data integrity checks between the trunk links of NCS 1002 without enabling the client traffic. PRBS generator generates a bit pattern on the device and sends it to the peer device, where PRBS analyzer detects if the transmitted bit pattern is preserved. The user can configure the trunk port in one of the following modes for PRBS.</td>
<td>Configuring Pseudo Random Binary Sequence, on page 31</td>
</tr>
<tr>
<td></td>
<td>• Source Mode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Sink Mode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Source-Sink Mode</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Slices

This chapter describes the procedures to configure slices and supported configurations on the client and trunk ports of each slice.

- Understanding Cisco NCS 1002, on page 1
- Slice and Port Numbering, on page 2
- Supported Configurations, on page 2
- Configure the Slice, on page 3
- Verify Slice Configuration, on page 5
- Illustrations for Supported Configurations, on page 8

Understanding Cisco NCS 1002

Cisco NCS 1002 is a 2Tbps muxponder that addresses the growing bandwidth needs of data center DWDM applications. The muxponder is 2 RU. It provides dense, low power, and cost-optimized DWDM transport for 10G, 40G, and 100G clients. The trunk ports can operate at 100G, 200G, and 250G traffic. NCS 1002 is ROHS6 compliant.

NCS 1002 has four independent slices. A slice is a group of client and trunk ports operating homogeneously. Each slice contains five QSFP+/QFSP28 client optical ports and two CFP2 DWDM trunk ports. Each slice delivers up to 500 Gbps traffic. NCS 1002 has one controller card, two Solid State Disks (SSD), 1+1 redundant 1800W AC power supply modules, and 2+1 redundant fans.

NCS 1002 delivers the following benefits:

- Supports up to 2 Tbps capacity
- Transports 100, 200, or 250Gbps per wavelength on the same platform through software provisioning
- Transports 10 GE, 40 GE, and 100 GE on the same platform through software provisioning
- Supports grid-less tuning for flex-grid dense wavelength-division multiplexing (DWDM)
- Supports different modulation formats (PM-QPSK or PM-16QAM)
- Supports 7% or 20% Soft Decision (SD) FEC for maximum optical performance
- Allows for automated installation, configuration and monitoring
- Supports machine-to-machine (M2M) APIs based on YANG models for ease of configuration
- Supports a telemetry agent for a pub-sub model of device monitoring
Slice and Port Numbering

Figure 1: Slice and Port Numbering

The slices are numbered from 0 to 3. The ports are numbered across the different slices from 0 to 27. The port numbers against blue background represent client ports; port numbers against green background represent trunk ports.

Supported Configurations

The following configurations are supported on client and trunk ports in each slice. Each slice contains up to twenty Ethernet client ports operating at 10G, or five Ethernet client ports operating at 100G, or five Ethernet client ports operating at 40G. The client ports map to two trunk ports operating at 100G, 200G, or 250G that provides muxponder functionality.

In mixed mode configuration, each slice contains up to three Ethernet client ports operating at 100G and eight Ethernet client ports operating at 10G. The client ports map to two trunk ports operating at 200G.

40G client ports are supported from R6.0.1.

<table>
<thead>
<tr>
<th>Client Ports</th>
<th>Trunk Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 x 10G</td>
<td>2 x 100G</td>
</tr>
<tr>
<td>20 x 10G</td>
<td>1 x 200G</td>
</tr>
<tr>
<td>4 x 40G</td>
<td>2 x 100G</td>
</tr>
<tr>
<td>5 x 40G</td>
<td>1 x 200G</td>
</tr>
<tr>
<td>2 x 100G</td>
<td>2 x 100G</td>
</tr>
<tr>
<td>4 x 100G</td>
<td>2 x 200G</td>
</tr>
<tr>
<td>5 x 100G</td>
<td>2 x 250G</td>
</tr>
<tr>
<td>3 x 100G + 8 x 10G</td>
<td>2 x 200G</td>
</tr>
</tbody>
</table>

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

You can configure the slice with traffic on client and trunk ports. All five client ports of the slice need to be configured at the same bitrate except for mixed mode configuration. Both the trunk ports are always set with the same FEC mode. The slice can be configured to send encrypted traffic from R6.1.1.

See the Supported Configurations in Encrypted Mode, on page 38 section to determine the supported configurations on the client and trunk ports in each slice configured in encrypted mode.

---

**Note**

When the slice is configured in encrypted mode, the drop-lldp cannot be enabled.

---

**Note**

When NCS 1002 is installed in a system where both the trunk interfaces in a slice are used, the two 250Gb 16QAM signals need to be co-routed on the same fiber (mandatory when the 5x100Gb client port is provisioned). Also, it is recommended to use adjacent wavelengths when the line modulation is set to 250Gb 16QAM. The reason for this is that the chromatic dispersion generates skew between wavelengths. Assuming a Dispersion of 10000 ps/nm, a span of 500 km, and using adjacent channel, the skew is evaluated in less than 200 ns and it is compensated by the deskew capability of NCS 1002. If the delta between the used channels is increased, the skew increases and it might exceed the skew compensation done by NCS 1002.

---

To configure the slice with unencrypted traffic, use the following commands.

```plaintext
configure
hw-module location location slice [slice_number | all] client bitrate { 10G | 40G | 100G } trunk bitrate { 100G | 200G | 250G } fec { softdecision7 | softdecision20 }
commit
```

To configure the slice with mixed mode, use the following commands.

```plaintext
configure
hw-module location location slice [slice_number | all] client bitrate 10G-100G trunk bitrate 200G fec { softdecision7 | softdecision20 }
commit
```

To configure the slice with encrypted traffic, use the following commands.

```plaintext
configure
hw-module location location slice [slice_number | all] client bitrate { 10G | 40G | 100G } trunk bitrate { 100G | 200G } fec { softdecision7 | softdecision20 } [encrypted]
commit
```

**Examples**

The following is a sample in which slice 0 is configured in mixed mode, and FEC on the trunk ports is set to softdecision7.

```plaintext
configure
```
The following is a sample in which slice 0 is configured to send encrypted traffic with 100G client rate, 200G trunk rate, and FEC on the trunk ports is set to sofdecision7.

configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 10G-100G trunk bitrate 200G fec
SoftDecision7
commit

The following is a sample in which slice 0 is configured to send encrypted traffic with 10G client rate, 100G trunk rate, and FEC on the trunk ports is set to sofdecision20. When a slice is configured with 10G client rate in encrypted mode, ten MACsec controllers are created for each slice. When all the four slices are configured with 10G client rate in encrypted mode, forty MACsec controllers are created for NCS 1002. Two MACsec controllers are created for the middle port, four controllers for the fourth port, and four controllers for the fifth port per slice.

configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 10G trunk bitrate 100G softdecision20
encrypted
commit

The following is a sample in which slice 0 is configured to send encrypted traffic with 40G client rate, 100G trunk rate, and FEC on the trunk ports is set to sofdecision20.

configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 40G trunk bitrate 100G softdecision20
encrypted
commit

The following is a sample to configure all the slices with a specific client rate and trunk rate.

configure
hw-module location 0/RP0/CPU0 slice all client bitrate 10G trunk bitrate 100G fec
SoftDecision7
commit

configure
hw-module location 0/RP0/CPU0 slice all client bitrate 40G trunk bitrate 100G fec
SoftDecision7
commit

configure
hw-module location 0/RP0/CPU0 slice all client bitrate 100G trunk bitrate 200G fec
SoftDecision7
commit

The following is a sample to remove the configuration from all the slices.

configure
no hw-module location 0/RP0/CPU0 slice all client bitrate 10G trunk bitrate 100G fec
SoftDecision7
commit

configure
no hw-module location 0/RP0/CPU0 slice all client bitrate 40G trunk bitrate 100G fec
Verify Slice Configuration

Use this procedure to verify whether the slice is correctly configured.

show hw-module { slice [ slicenumber | all ] }

Example:

RP/0/RP0/CPU0:ios# show hw-module slice 0

Thu Aug 11 16:16:58.935 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage
HundredGigECtrlr0/0/0/3 100
HundredGigECtrlr0/0/0/4 100

RP/0/RP0/CPU0:ios# show hw-module slice 0

Sun Dec 18 13:59:18.805 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 40
Trunk Bitrate: 100
DP FPGA FW Type: MM40
DP FPGA FW Version: 03.00
HW Status: CURRENT
Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage
FortyGigECtrlr0/0/0/3 100
FortyGigECtrlr0/0/0/4 100

RP/0/RP0/CPU0:ios# show hw-module slice 1

Tue Jan 1 06:55:12.293 UTC
Slice ID: 1
Status: Provisioned
Client Bitrate: 10
Trunk Bitrate: 100
DP FPGA FW Type: MM10
DP FPGA FW Version: 03.00
HW Status: CURRENT

Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/13
Traffic Split Percentage

TenGigECtrlr0/0/0/9/1 100
TenGigECtrlr0/0/0/9/2 100
TenGigECtrlr0/0/0/10/1 100
TenGigECtrlr0/0/0/10/2 100
TenGigECtrlr0/0/0/10/3 100
TenGigECtrlr0/0/0/10/4 100
TenGigECtrlr0/0/0/11/1 100
TenGigECtrlr0/0/0/11/2 100
TenGigECtrlr0/0/0/11/3 100
TenGigECtrlr0/0/0/11/4 100

RP/0/RP0/CPU0:ios# show hw-module slice 2

Slice ID: 2
Status: Provisioned
Client Bitrate: 10,100
Trunk Bitrate: 200
DP FPGA FW Type: RMM
DP FPGA FW Version: 04.00
HW Status: CURRENT

Encryption Supported: FALSE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/19  CoherentDSP0/0/0/20
Traffic Split Percentage

HundredGigECtrlr0/0/0/14 100 0
HundredGigECtrlr0/0/0/15 100 0
HundredGigECtrlr0/0/0/16 0 100
TenGigECtrlr0/0/0/17/1 0 100
TenGigECtrlr0/0/0/17/2 0 100
TenGigECtrlr0/0/0/17/3 0 100
TenGigECtrlr0/0/0/17/4 0 100
TenGigECtrlr0/0/0/18/1 0 100
TenGigECtrlr0/0/0/18/2 0 100
TenGigECtrlr0/0/0/18/3 0 100
TenGigECtrlr0/0/0/18/4 0 100

Displays the details of the slice such as the slice ID, client rate, trunk rate, and the traffic percentage carried on the trunk ports. The Encryption Supported field indicates whether the slice is provisioned with firmware that supports encryption or not.
### Configuring Slices

**Note** The HW Status field might display "Need Upgrade" when the user needs to use the MACsec feature and upgrades from R6.0.1 to 6.1.1. Hence, the control FPGA (CTRL_BKP_UP, CTRL_BKP_LOW, CTRL_FPGA_UP, and CTRL_FPGA_LOW) needs to be upgraded to the latest firmware version provided by R6.1.1. See [Verify Firmware Version](#) for more information.

The Provisioned status does not indicate that the traffic can flow immediately. For example, use the `show controllers macSecCtrlr 0/0/0/3` command output to view the provisioning information of the port after the slice is provisioned.

**Example:**

```plaintext
RP/0/RP0/CPU0:ios# show hw-module slice all

Thu Aug 11 16:16:58.935 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage

HundredGigECtrlr0/0/0/3 100
HundredGigECtrlr0/0/0/4 100

Slice ID: 1
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/13
Traffic Split Percentage

HundredGigECtrlr0/0/0/10 100
HundredGigECtrlr0/0/0/11 100

Slice ID: 2
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/20
Traffic Split Percentage

HundredGigECtrlr0/0/0/17 100
HundredGigECtrlr0/0/0/18 100

Slice ID: 3
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
```
DP FPGA FW Version: 02.00  
HW Status: CURRENT  
Encryption Supported: TRUE  
Client Port - Trunk Port  
CoherentDSP0/0/0/27  
Traffic Split Percentage  

Table: Client and Trunk Ports in Slice 0

<table>
<thead>
<tr>
<th>Client Data Rate</th>
<th>Trunk Data Rate</th>
<th>Client Port 0</th>
<th>Client Port 1</th>
<th>Client Port 2</th>
<th>Client Port 3</th>
<th>Client Port 4</th>
<th>Client Port 5</th>
<th>Trunk Port 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>100G</td>
<td>100G</td>
<td>E</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>100G</td>
<td>200G</td>
<td>E</td>
<td>E</td>
<td>D</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>100G</td>
<td>250G</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>10G</td>
<td>100G</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>10G</td>
<td>200G</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>40G</td>
<td>100G</td>
<td>E</td>
<td>E</td>
<td>D</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>40G</td>
<td>200G</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>D</td>
<td>E</td>
<td>E</td>
</tr>
</tbody>
</table>

E indicates that the port is enabled; D indicates that the port is disabled.

The following illustrations describe the mapping of traffic from client to trunk ports for certain configurations.  
**Client: 5 x 100G Trunk: 2 x 250G**  

**Illustrations for Supported Configurations**

Associated Commands:
- hw-module
- show hw-module
Client: 4 x 100G Trunk: 2 x 200G

Client: 20 x 10G Trunk: 2 x 100G
The following illustration describes the mapping of traffic from client to trunk ports for mixed mode configuration.

```
100 GE 0/0/0/0
  |
  v
100 GE 0/0/0/1
  |
  v
100 GE 0/0/0/2
  |
  v
40 GE 0/0/0/3
  |
  v
40 GE 0/0/0/4
  |
  v
200 G WDM 0/0/0/5
  |
  v
200 G WDM 0/0/0/6
```
CHAPTER 2

Configuring Controllers

This chapter describes controllers such as Optics controller, Ethernet controller, and Coherent DSP Controller. This chapter also describes the procedures used to configure the controllers.

- Optics Controllers, on page 11
- Maintenance Mode, on page 12
- Configure Optics Controller, on page 12
- Configure Wavelength, on page 13
- Breakout Mode, on page 13
- Laser Squelching, on page 14
- LLDP Snooping, on page 14
- LLDP Snoop and Drop, on page 16
- Configure Ethernet Controller, on page 19
- Configure the Coherent DSP Controller, on page 20
- Configure Loopback, on page 20
- Restore Factory Settings, on page 21
- Headless Mode, on page 21
- View the Headless Statistics, on page 21

Optics Controllers

Controllers are represented in the Rack/Slot/Instance/Port format; for example, 0/0/0/1. Each port has an optics controller that is created on startup. When the slice is provisioned using the hw-module command, client layer controllers are created on the associated client and trunk ports. For example, HundredGig Ethernet controllers and TenGig Ethernet controllers are created on the associated client ports depending on the slice configuration; the CoherentDSP controller is created on the associated trunk ports.

The following figure describes the controller representation when the client rate is 100G and the trunk rate is 250G.
Maintenance Mode

Controllers can be placed in maintenance mode. Use the `controller optics sec-admin-state maintenance` command to place controllers in maintenance mode.

Use the `show controllers controllertype Rack/Slot/Instance/Port` command to view client and trunk alarms. In maintenance mode, all alarms are suppressed and the `show alarms` command does not display alarm details. However, traffic is not affected in maintenance mode.

Configure Optics Controller

You can configure parameters such as high power threshold, maximum and minimum chromatic dispersion, and wavelength for Optics controller. To configure Optics controller, use the following commands.

```
configure
controller controllertype Rack/Slot/Instance/Port
rx-high-threshold rx-high
tx-high-threshold tx-high
cd-max cd-max
cd-min cd-min
dwdm-carrier {100MHz-grid frequency frequency} | {50GHz-grid [frequency frequency | wavelength wavelength | itu-ch channel-number]}
commit
```

**Note**

To view wavelength and channel mapping for optics controllers, use the `show controllers optics R/S/I/P dwdm-carrier-map` command.

**Example**

The following is a sample in which the high power threshold is configured at the receive and transmit side, maximum and minimum chromatic dispersion is configured, and wavelength is configured in 50GHz grid spacing.

```
configure
controller optics 0/0/0/1
rx-high-threshold 200
tx-high-threshold 300
cd-max 10000
cd-min 2000
dwdm-carrier 50GHz-grid wavelength 1560200
commit
```

**Associated Commands**

- `controller optics`
• show controllers

Configure Wavelength

You can configure the wavelength on trunk ports. Before configuring wavelength, use the following command to determine the valid range of wavelength.

**show controllers optics Rack/Slot/Instance/Port dwdm-carrier-map**

Displays the wavelength and channel mapping for trunk optics controllers. See Show Controllers command to view the DWDM carrier map table.

To configure wavelength, use the following commands.

`configure
controller optics Rack/Slot/Instance/Port
dwdm-carrier {100MHz-grid frequency frequency} | {50GHz-grid [frequency frequency | wavelength wavelength | itu-ch channel-number]}
commit`

In 50GHz grid spacing, enter the 7-digit wavelength value in the range of 1528773 to 1568362 nm. For example, enter 1532290 to specify 1532.29 nm. In 100MHz grid spacing, enter the 8-digit wavelength value in the range of 15667227 to 15287730 nm. For example, enter 15667227 to specify 1566.7227 nm.

**Example**

The following is a sample in which the wavelength is configured on the trunk port in 50GHz grid spacing.

```
show controllers optics 0/0/0/11 dwdm-carrier-map
configure
controller optics 0/0/0
    dwdm-carrier 50GHz-grid wavelength 1560200
commit
```

**Associated Commands**

• dwdm-carrier

• show controllers

Breakout Mode

The client port can be enabled in normal mode or breakout mode. When the client bitrate is 10G, the mode is breakout mode.

```
RP/RP0/CPU0:ios(config)# hw-module location 0/RP0/CPU0 slice 1 client bitrate 10g trunk bitrate 100g
```

The client ports can operate at 10G mode using the breakout cable or the breakout patch panel. All five client ports of the slice need to be configured at the same bitrate. The controllers are represented in the
Rack/Slot/Instance/Port/Lanenumber format; for example, 0/0/0/1/3. The range of Lanenumber is from 1 to 4.

Four TenGig Ethernet controllers are created for each client port in breakout mode. The following figure describes the controller representation when the client rate is 10G and the trunk rate is 100G.

When the optics controller is shutdown, all four TenGig Ethernet controllers are shut down. Individual 10G port can be turned off from the TenGig Ethernet controller.

**Laser Squelching**

Ethernet controllers can be configured to enable laser squelching so that laser is brought down in the event of trunk faults (LOS, LOF) and a SQUELCHED alarm is raised. For 10G Ethernet controllers, laser squelching is supported only on LR4 and QSFP+ pluggables. For more information on SQUELCHED alarm, see the Troubleshooting Guide for Cisco NCS 1000 Series, IOS XR Release 6.0.x.

**LLDP Snooping**

Link Layer Discovery Protocol (LLDP) Snooping is enabled by default on all Ethernet controllers. The user can use LLDP snooping to troubleshoot problems in the client ports.

```
show controllers controllertype Rack/Slot/Instance/Port lldp-snoop
```

**Note**

LLDP snoop and drop is not supported for VLAN-tagged LLDP packets.

**Note**

If mandatory TLVs (Chassis ID, Port ID and TTL) are invalid or not available, then the LLDP neighbor information does not populate the LLDP packet details. The hardware drops the LLDP packet if LLDP drop is enabled.

Verify that the MAC address displayed is same as the MAC address of the traffic generating port. In Release 6.0.1, you can view more details about the LLDP neighbor.

```
RP/0/RP0/CPU0:ios# show controllers hundredGigECtrlr 0/0/0/8 lldp-snoop
Mon Apr  2 04:37:25.603 UTC

LLDP Neighbor Snoop Data
---------------------------------------------
Capability codes:
    (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
```
Local Controller: HundredGigECtrlr0/0/0/8
Source MAC Address: 0010.9400.0002
Chassis ID: ABCD
Port ID: 192.0.2.254
Port Description: descr:ABCD Port-iter2
System Name: Name:ABCD
System Description: descr:ABCD-iter2
Hold Time(TTL): 120 seconds
System Capabilities: P,B,W,R,T,C
Enabled Capabilities: P,R,T
Management Address:
IPv4 address: 192.0.2.254

To verify the LLDP neighbor entries, use the following command:

```
RP/0/RP0/CPU0:ios# show lldp neighbors
Thu Jul 26 15:08:27.943 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
Hav EC0/0/0/4 120 B,R Ethernet1/1
```

Total entries displayed: 1

EC refers to Ethernet Controller.

To display detailed information about LLDP neighbor entries, use the following command:

```
RP/0/RP0/CPU0:ios# show lldp neighbors detail
Thu Jul 26 15:08:03.836 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: HundredGigECtrlr0/0/0/4
Chassis id: a89d.21f8.4aa8
Port id: Ethernet1/1
Port Description: Ethernet1/1
System Name: Hav
System Description:
Cisco Nexus Operating System (NX-OS) Software 7.0(3)I4(7)
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2017, Cisco Systems, Inc. All rights reserved.
Time remaining: 103 seconds
Hold Time: 120 seconds
System Capabilities: B,R
Enabled Capabilities: B,R
LLDP Snoop and Drop

LLDP Snoop and Drop feature is implemented in Release 6.1.2. Cisco NCS 1002 snoops the LLDP packets at each client controller port of a slice and drops the LLDP frame in the same slice without forwarding it to the peer slice.

Limitation:

LLDP Drop functionality with MACSEC encryption on NCS 1002 is not supported in Release 6.1.2 or earlier. Hence, it is not recommended to configure LLDP Drop functionality with MACSEC encryption in these releases. The LLDP snoop does not work for VLAN tagged LLDP packets, and hence the LLDP drop functionality does not occur.

If LLDP drop is disabled, slices perform legacy snoop and forward functionality by simply snooping and forwarding the LLDP frames to their peer slice.

Prerequisites:

Slices should be provisioned with client/trunk rate and should indicate Provisioned as status.

Configuring Slices and LLDP Drop at Slice Level

You can configure the slices and LLDP drop at a single slice, or over all slices. When the LLDP drop is enabled at slice then its corresponding client controller ports will also be enabled. By default, LLDP drop status is set to False. On enabling the LLDP Drop, its status is set to True.
To enable LLDP drop at single slice, use the following commands:

```
configure
hw-module location location slice [slice_number | all] client bitrate {10G | 40G | 100G} trunk bitrate {100G | 200G | 250G} fec {softdecision7 | softdecision20}
```

```
hw-module location location slice slice_number drop-lldp
```

```
commit
```

**Example:**

To enable LLDP drop at slice 0, use the following commands.

```yaml
RP/0/RP0/CPU0:ios(config)# hw-module location 0/RP0/CPU0 slice 0 client bitrate 40G trunk bitrate 200G fec softDecision7
RP/0/RP0/CPU0:ios(config)# hw-module location 0/RP0/CPU0 slice 0 drop-lldp
RP/0/RP0/CPU0:ios(config)# commit
```

To enable LLDP drop over all slices, use the following commands.

```
configure
hw-module location location slice [slice_number | all] client bitrate {10G | 40G | 100G} trunk bitrate {100G | 200G | 250G} fec {softdecision7 | softdecision20} [encrypted]
```

```
hw-module location location slice all drop-lldp
```

```
commit
```

**Example:**

To enable LLDP drop over all slices, use the following commands.

```yaml
RP/0/RP0/CPU0:ios(config)# hw-module location 0/RP0/CPU0 slice all client bitrate 40G trunk bitrate 200G fec softDecision7
RP/0/RP0/CPU0:ios(config)# hw-module location 0/RP0/CPU0 slice all drop-lldp
RP/0/RP0/CPU0:ios(config)# commit
```

**Note**

You can configure LLDP drop for either on a single slice or over all slices. Both configuration commands cannot be executed together.

**Associated Commands**

- `hwmodule`
- `show controllers`

**Verifying the Status of LLDP Drop**

To verify the LLDP drop status of a slice, use the following command.

```
show hw-module {slice slicenumber | all | fpd}
```

**Example:**
The following is a sample in which the slice 0 is configured with 40G client bitrate, 200G trunk bitrate and LLDP drop is enabled.

```
RP/0/RP0/CPU0:ios(config)# show hw-module slice 0
Thu Sep 22 10:55:35.985 UTC
Slice ID: 0
Status: Provisioned
Client Bitrate: 40
Trunk Bitrate: 200
DP FPGA FW Type: XMG4
DP FPGA FW Version: 01.01
HW Status: CURRENT
Encryption Supported: FALSE
LLDP Drop Enable: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage
FortyGigECtrlr0/0/0/0 100
FortyGigECtrlr0/0/0/1 100
FortyGigECtrlr0/0/0/2 100
FortyGigECtrlr0/0/0/3 100
FortyGigECtrlr0/0/0/4 100
```

To verify the LLDP drop status at the client controller level, use the following command.

```
show controllers  controllertype Rack/Slot/Instance/Port  lldp-snoop
```

**Example**:

The following is a sample in which the LLDP Drop is enabled for Forty GigE controller.

```
RP/0/RP0/CPU0:ios(config)# show controllers FortyGigECtrlr 0/0/0/0 1lldp-snoop
Thu Apr 28 09:49:20.684 UTC
Capability codes:R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
(W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
LLDP Neighbor Snoop Data
Local Controller: FortyGigECtrlr0/0/0/0
Source MAC Address: 0010.9400.0044
Chassis ID: 50.1.1.10
Port ID: 0010.9400.0044
Port Description: Spirent Port ROSCO
System Name: Spirent Test CenterROSCO1111111111111
System Description: Spirent Test Center dddddddddddd
Hold Time(TTL): 20 seconds
System Capabilities: R
Enabled Capabilities: R
Management Address:
IPv4 address: 50.1.1.10
LLDP Packet Drop enabled: TRUE
RX LLDP Packet Count : 1
```

**Note**

RX LLDP Packet count represents the total number of packets received at the ingress of the Ethernet controller.
Disabling LLDP Drop

To disable LLDP drop, use the following commands:

```
configure
hw-module location location slice [slice_number | all] drop-lldp
commit
```

**Example:**

To disable LLDP drop at slice 0, use the following commands.

```
RP/0/RP0/CPU0:ios(config)# no hw-module location 0/RP0/CPU0 slice 0 drop-lldp
RP/0/RP0/CPU0:ios(config)# commit
```

Once you disable LLDP Drop, show hw-module slice X and show controllers commands would display LLDP DROP ENABLED as FALSE.

---

**Note**

When you disable LLDP Drop at slice level its corresponding client controller ports will also be disabled.

Configure Ethernet Controller

You can configure parameters such as performance monitoring, administrative state, laser squelching, and FEC for the Ethernet controller. To configure the Ethernet controller, use the following commands:

```
configure
controller controllertype Rack/Slot/Instance/Port
perf-mon { enable | disable }
sec-admin-state admin-state
laser-squelch
fec { none | standard }
commit
```

**Example**

The following is a sample in which the performance monitoring is enabled for HundredGig Ethernet controller, administrative state is placed under maintenance, laser squelching is enabled, and standard FEC is enabled.

```
configure
controller HundredGigECtrlr 0/0/0/0
perf-mon enable
sec-admin-state maintenance
laser-squelch
fec standard
commit
```
Configure the Coherent DSP Controller

You can configure parameters such as performance monitoring, administrative state, and trail trace identifier (TTI) for the Coherent DSP controller. In Release 6.0, the Coherent DSP is provisioned per slice, from Release 6.0.1, the Coherent DSP is provisioned per port. To configure the Coherent DSP controller, use the following commands.

```
configure
ccontroller controllertype Rack/Slot/Instance/Port
perf-mon { enable | disable }
sec-admin-state admin-state
tti { sent | expected } ascii string
commit
```

Example

The following is a sample in which the performance monitoring is enabled for Coherent DSP controller, administrative state is placed under maintenance, and tti is configured.

```
configure
ccontroller coherentDSP 0/0/0/12
perf-mon enable
sec-admin-state maintenance
tti sent ascii joy
commit
```

Configure Loopback

You can configure the loopback on the Coherent DSP and Ethernet controllers. The loopback can be performed only in maintenance mode. Use the `controller optics sec-admin-state maintenance` command to place the controllers in maintenance mode. The line loopback on the tenGig Ethernet controller does not work when the port is squelched. To configure the loopback, use the following commands.

```
configure
ccontroller controllertype Rack/Slot/Instance/Port loopback [ line | internal ]
commit
```
**Example**

The following is a sample in which the line loopback is configured on the Ethernet controller.

```
configure
ccontroller HundredGigECtrlr 0/0/0/0 loopback line
commit
```

**Associated Commands**

- controller optics
- controller GigECtrlr
- controller coherentDSP

**Restore Factory Settings**

You can restore factory settings. The `commit replace` command replaces the entire running configuration with the contents of the target configuration. If the target configuration is empty, all existing configurations are removed and NCS 1002 will be restored to factory configuration. To restore NCS 1002 to factory settings, use the following commands.

```
configure
commit replace
```

**Example**

```
configure
commit replace
```

**Headless Mode**

NCS 1002 can carry traffic with a non-functional CPU (headless mode) for up to 72 hours. The data path and cumulative statistics are maintained for up to 72 hours with a non-functional CPU. The fault propagation continues to operate for failures on client and trunk ports. However, the provisioning operations cannot be performed and operational data cannot be viewed with a non-functional CPU. Performance monitoring data based on 15 minutes and 24 hour intervals is not supported with a non-functional CPU. In case of CPU OIR, the CPU needs to be physically replaced within 10 minutes.

**View the Headless Statistics**

Use this procedure to display the statistics collected during the last headless operation.

```
show controllers  controllertype R/S/I/P  headless-stats
```

**Example:**
RP/0/RP0/CPU0:ios# show controllers hundredGigECtrlr 0/0/0/11 headless-stats

Displays the statistics collected during the last headless operation. The collected statistics is preserved for a slice until the hw-module configuration is removed or changed on that slice or until the next headless operation. The statistics is also preserved across process restarts.

**Associated Commands**

- `show controllers`
CHAPTER 3

Configuring IP Access List

This chapter describes how to configure IPv4 and IPv6 ACL.

• IP Access List, on page 23

IP Access List

How an IP Access List Works

An access list is a sequential list consisting of permit and deny statements that apply to IP addresses and possibly the upper-layer IP protocols. ACLs are used to permit or deny the flow of packets based on matching criteria of access list parameters and information contained in packets. An access list must be created and applied to an interface for it to be in effect.

An access list can control traffic arriving or leaving the system, but not traffic originating at the system.

IP Access List Process and Rules

There are two paths for interface packet filtering for ACL configuration:

• Hardware programming path: Hardware programming path is the fast path ACL configuration. The fast path ACL configuration requires Ternary Content Addressable Memory (TCAM) through packet filter Execution Agent.

• Software programming path: Software programming path is the slow path ACL configuration. Slow path ACL configuration requires adding caps to Interface Manager and NetIO.

Use the following process and rules when configuring an IP access list:

• The software tests the source or destination address or the protocol of each packet being filtered against the conditions in the access list, one condition (permit or deny statement) at a time.

• If a packet does not match a statement in the access list, it is then tested against the next statement in the list.

• If a packet matches an access list statement, the remaining statements in the list are skipped and the packet is permitted or denied as specified in the matched statement.

• If the access list denies the address or the protocol, the software discards the packet and returns an Internet Control Message Protocol (ICMP) Host Unreachable message. ICMP is configurable in the Cisco IOS XR software.
• If no conditions match, the software drops the packet because each access list ends with an unwritten or implicit deny statement.

• The access list should contain at least one permit statement otherwise all packets will be denied.

• The software stops testing the conditions after the first match, so the order of the conditions is critical. The same permit or deny statements specified in a different order could result in a packet being passed under one circumstance and denied in another circumstance.

• Only one access list per interface, per protocol, per direction is allowed.

• Inbound access lists process packets arriving at the system. Incoming packets are processed before being routed to an outbound interface. An inbound access list is efficient as it saves the overhead of routing lookups if the packet is to be discarded because it is denied by the filtering tests. If the packet is permitted by the tests, it is then processed for routing. For inbound lists, permit means continue to process the packet after receiving it on an inbound interface; deny means discard the packet.

• Outbound access lists process packets before they leave the system. Incoming packets are routed to the outbound interface and then processed through the outbound access list. For outbound lists, permit means send it to the output buffer; deny means discard the packet.

• An access list can not be removed if that access list is being applied by an access group in use. To remove an access list, remove the access group that is referencing the access list and then remove the access list.

• An access-list must be created first before it can actually be applied on the management interface using access-group command.

• ACLs are applied only on management interface and not on any other type of interfaces or controllers.

Statistics collections are also divided into fast path packets and slow path packets. ACLs information is stored as a global data on the route processor.

**Support of IP Access list in NCS 1002:**

NCS 1002 supports the following:

• Ingress ACL for both IPv4 and IPv6.

• Slow packet path for Management Interface.

• Egress ACL: Self-Originated Packet is not supported by ACL, as this is already controlled by user. Only forwarded packets or traffic is classified under ACL. This rule is applicable for both IPv4 and IPv6 ACL.

### Configure IP Access List

To configure the ACL, use the following configuration at the IPv4 or IPv6 interface:

```
configure
interface interface-type Rack/Slot/Instance/Port
ipv4 | ipv6 access-group access-list-name {ingress | egress}
commit
```

**Example**

```
interface MgmtEth0/RP0/CPU0/0
ipv4 address 10.1.1.1 255.255.255.0
```
ipv6 address 1000::1/64
ipv4 access-group IPV4_ICMP_DENY ingress
ipv4 access-group IPV4_ROUTER_FWD_TELNET_TRAFFIC_DENY egress
ipv6 access-group IPV6_SSH_DENY ingress
ipv6 access-group IPV6_ROUTER_FWD_TELNET_TRAFFIC_DENY egress

Sample Configuration for IPv4 Access Lists

ipv4 access-list IPV4_ICMP_DENY
10 deny icmp any any
20 permit ipv4 any any
!
ipv4 access-list IPV4_ROUTER_FWD_TELNET_TRAFFIC_DENY
10 deny tcp any any eq telnet
20 permit ipv4 any any
!

Sample Configuration for IPv6 Access Lists

ipv6 access-list IPV6_SSH_DENY
10 deny tcp any any eq ssh
20 permit ipv6 any any
!
ipv6 access-list IPV6_ROUTER_FWD_TELNET_TRAFFIC_DENY
10 deny tcp any any eq telnet
20 permit ipv6 any any
!

Verify IPv6 ACL

The following examples verify the number of packets filter by respective ACLs:

Examples to check statistics

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)

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)
RP/0/RP0/CPU0:PROD_20#
Verify IPv6 ACL
Configuring Performance Monitoring

Performance monitoring (PM) parameters are used by service providers to gather, store, set thresholds for, and report performance data for early detection of problems. The user can retrieve both current and historical PM counters for the various controllers in 15 minutes and 1 day intervals.

PM for optical parameters include laser bias current, transmit and receive optical power, mean polarization mode dispersion, accumulated chromatic dispersion, and received optical signal-to-noise ratio (OSNR). These parameters simplify troubleshooting operations and enhance data that can be collected directly from the equipment.

The supported MTU of data plane is as follows:

- Range: 60 bytes to 10 kilobytes
- Jumbo: 10 kilobytes
- Undersize: 60 to 64 bytes

For descriptions of optics, Ethernet, fec, and otn parameters, see the Command Reference for Cisco NCS 1000 Series.

- Configure PM Parameters, on page 27
- View PM Parameters, on page 29
- Configuring Pseudo Random Binary Sequence, on page 31

Configure PM Parameters

You can configure the performance monitoring parameters for the Optics, Ethernet, and coherent DSP controllers. The coherent DSP controller is created on the trunk port when the slice is provisioned using the hw-module command. To configure PM parameters, use the following commands.

```
configure
controller controllertype R/S/I/P { pm {15-min | 30-sec | 24-hour} { optics | ether | fec | otn} { report | threshold } value }
commit
```
Examples

The following is a sample in which the performance monitoring parameters of Optics controller is configured in 24 hour intervals.

configure  
controller optics 0/0/0/0 pm 24-hour optics report cd max-tca enable  
commit

The following is a sample in which the performance monitoring parameters of Ethernet controller is configured in 15 minute intervals.

configure  
controller HundredGigE 0/0/0/1 pm 15-min ether report 1024-1518-octets enable  
commit

The following is a sample in which the performance monitoring parameters of Coherent DSP controller is configured in 15 minute intervals.

configure  
controller coherentDSP 0/0/0/12 pm 15-min otn threshold es-ne  
commit

Configure FEC BER Thresholds

Pre-forward error correction (FEC) bit error rate (BER) or post-FEC BER values are represented in numerical values. BER value is multiplied by 1E+15 to derive numerical value. For example, 2.1e-4 is displayed as 210000000000 (2.1e+11).

The following is a sample to enable minimum and maximum TCAs for pre-FEC BER.

configure  
controller coherentDSP 0/0/0/6 pm 30-sec fec report pre-fec-ber min-tca enable  
controller coherentDSP 0/0/0/6 pm 30-sec fec report pre-fec-ber max-tca enable  
commit

The following is a sample to enable minimum and maximum TCAs for post-FEC BER.

configure  
controller coherentDSP 0/0/0/6 pm 30-sec fec report post-fec-ber min-tca enable  
controller coherentDSP 0/0/0/6 pm 30-sec fec report post-fec-ber max-tca enable  
commit

The following is a sample to configure pre-FEC BER thresholds of Coherent DSP controller in 30 second intervals.

configure  
controller coherentDSP 0/0/0/6 pm 30-sec fec threshold pre-fec-ber max 320000000000  
commit

The following is a sample to configure post-FEC BER thresholds of Coherent DSP controller in 30 second intervals.

configure  
controller coherentDSP 0/0/0/6 pm 30-sec fec threshold post-fec-ber max 320000000000  
commit

The following is a sample of the show controllers command.
show controllers coherentDSP 0/0/0/6 pm current 30-sec fec

Mon Feb 25 05:29:20.980 UTC

g709 FEC in the current interval [05:29:00 - 05:29:21 Mon Feb 25 2019]

FEC current bucket type : Valid
EC-BITS : 1196208549 Threshold : 903330 TCA(enable) : NO
UC-WORDS : 0 Threshold : 5 TCA(enable) : YES

<table>
<thead>
<tr>
<th>MIN</th>
<th>AVG</th>
<th>MAX</th>
<th>Threshold (min)</th>
<th>TCA (enable)</th>
<th>Threshold (max)</th>
<th>TCA (enable)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreFEC BER : 0E-15 0E-15 0E-15 0</td>
<td>NO</td>
<td>32000000000000</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PostFEC BER : 0E-15 0E-15 0E-15 0</td>
<td>NO</td>
<td>32000000000000</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Associated Commands

- pm
- controller optics
- controller GigECtrlr
- controller coherentDSP
- show controllers

View PM Parameters

Use this procedure to view the performance monitoring parameters for Optics, Ethernet, and coherent DSP controllers.

show controllers controllertype R/S/I/P { pm { current | history } { 15-min | 24-hour } { optics | ether | fec | otn } linenumber }

Example:
RP/0/RP0/CPU0:ios# show controllers optics 0/0/0/1 pm current 15-min optics 1

Displays the current performance monitoring parameters of the Optics controller in 15 minute intervals.

Client optic has four lanes and trunk optic has one lane.

Fri Aug 21 09:28:57.608 UTC

Optics in the current interval [ 9:15:00 - 09:28:57 Fri Aug 21 2015]

Optics current bucket type : Valid

<table>
<thead>
<tr>
<th>MIN</th>
<th>AVG</th>
<th>MAX</th>
<th>Threshold (min)</th>
<th>TCA (enable)</th>
<th>Threshold (max)</th>
<th>TCA (enable)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBC[% ] : 0.0 0.0 0.0 0.0</td>
<td>NO</td>
<td>0.00</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPT[dBm] : -inf -inf -inf 0.00</td>
<td>NO</td>
<td>0.00</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPR[dBm] : -inf -inf -inf 0.00</td>
<td>NO</td>
<td>0.00</td>
<td>NO</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Last clearing of "show controllers OPTICS" counters never
Example:

RP/0/RP0/CPU0:ios# show controllers hundredGigECtrlr 0/0/0/3 pm current 15-min ether

Displays the current performance monitoring parameters of the Ethernet controller in 15 minute intervals.

Mon Jan 28 07:20:28.170 IST

ETHER in the current interval [07:15:00 - 07:20:29 Mon Jan 28 2019]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Threshold</th>
<th>TCA(enable)</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>RX-UTIL[]</td>
<td>2.90</td>
<td>0.00</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX-UTIL[]</td>
<td>2.84</td>
<td>0.00</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>RX-PKT</td>
<td>78662810</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STAT-PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OCTET-STAT</td>
<td>117994199787</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OVERSIZE-PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>FCS-ERR</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>LONG-FRAME</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>JABBER-STATS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>64-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>65-127-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>128-255-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>256-511-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>512-1023-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>1024-1518-OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN-UCAST</td>
<td>78662799</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN-MCAST</td>
<td>11</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN-BCAST</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OUT-UCAST</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OUT-BCAST</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OUT-MCAST</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX-PKT</td>
<td>76889333</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>OUT-OCTET</td>
<td>11533999500</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IFIN-ERRORS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IFIN-OCTETS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STAT-MULTICAST-PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STAT-BROADCAST-PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STAT-UNDERSIZED-PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_GOOD_BYTES</td>
<td>117994199787</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_GOOD_PKTS</td>
<td>78662810</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_DROP_OTHER</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_ERROR_FRAGMENTS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKT_64_OCTET</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKTTS_65_127_OCTETS</td>
<td>11</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKTTS_128_255_OCTETS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKTTS_256_511_OCTETS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKTTS_512_1023_OCTETS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>IN_PKTTS_1024_1518_OCTETS</td>
<td>78662799</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX_UNDERSIZED_PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX_OVERSIZED_PKT</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX_FRAGMENTS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX_JABBER</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>TX_BAD_FCS</td>
<td>0</td>
<td>0</td>
<td>NO</td>
<td></td>
</tr>
</tbody>
</table>

Last clearing of "show controllers ETHERNET" counters never

Example:

RP/0/RP0/CPU0:ios# show controllers coherentDSP 0/0/0/13 pm current 15-min ottn

Displays the current performance monitoring parameters of the Coherent DSP controller in 15 minute intervals.
g709 OTN in the current interval [15:30:00 - 15:43:00 Tue Feb 13 2001]

OTN current bucket type : Valid
ES-NE : 0  Threshold : 500  TCA(enable) : YES
ESR-NE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
SES-NE : 0  Threshold : 500  TCA(enable) : YES
SES-NE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
UAS-NE : 0  Threshold : 500  TCA(enable) : YES
BBE-NE : 0  Threshold : 10000  TCA(enable) : YES
BBER-NE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
FC-NE : 0  Threshold : 10  TCA(enable) : YES

ES-FE : 0  Threshold : 500  TCA(enable) : YES
ESR-FE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
SES-FE : 0  Threshold : 500  TCA(enable) : YES
SES-FE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
UAS-FE : 0  Threshold : 500  TCA(enable) : YES
BBE-FE : 0  Threshold : 10000  TCA(enable) : YES
BBER-FE : 0.00000  Threshold : 0.00000  TCA(enable) : NO
FC-FE : 0  Threshold : 10  TCA(enable) : YES

Last clearing of "show controllers OTU" counters never

 Associated Commands
- pm
- show controllers
- controller optics
- controller GigECtrlr
- controller coherentDSP

**Configuring Pseudo Random Binary Sequence**

Pseudo Random Binary Sequence (PRBS) feature allows you to perform data integrity checks between the NCS1002 trunk links without enabling the actual client traffic.

PRBS generator on the device generates a bit pattern and sends it to the peer device, where PRBS analyzer detects if the transmitted bit pattern is intact.

You can operate NCS 1002 trunk port in any one of the following modes for PRBS:
- **Source mode** — ETNA device at trunk port generates PRBS signal on the line continuously as per the configured PRBS pattern.
- **Sink mode** — ETNA device at trunk port gets locked to the ingress signal according to the configured pattern, analyzes and reports the errors.
- **Source-Sink mode** — ETNA device at trunk port acts both as PRBS generator and analyzer i.e. it generates PRBS signal as per the configured pattern and also gets locked to the ingress signal with the same pattern, analyzes and reports the errors.
NCS 1002 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.
- **PRBS11** — Sequence length is from $2^{11} - 1$ bits.

The secondary admin state of the CoherentDSP 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:

```bash
controller coherentDSP R/S/I/P prbs mode {source | sink | source-sink} pattern {pn31 | pn23 | pn15 | pn11}
```

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 overridden by PRBS pattern at the generator.
- Remote fault is raised on the client ports nearer to the PRBS Sink.
- Client ports on both the sides are squelched when PRBS is enabled on the trunks and when laser-squelch is configured on the clients.
- Line Loopback on the client ports works without any issues.
- Internal Loopback on the client ports does not work when PRBS is enabled on the trunk ports.

There are following limitations with the PRBS feature:

- Alarms are not visible on the trunk ports when PRBS is enabled.
- There is no SNMP support to fetch the PRBS status or Performance Monitoring (PM).
- 30-secs current/historic PMs are not supported for PRBS feature.
- PRBS PM TCAs are not supported.
- Apply PRBS config only after coherentDsp upgrade is complete.

### Performance Monitoring Pseudo Random Binary Sequence

To monitor the performance of Pseudo Random Binary Sequence (PRBS) on the CoherentDSP controller, use the following command:

```bash
show controllers coherentDSP R/S/I/P pm {current | history} {15-min|24-hour} prbs
```

```
RP/0/RP0/CPU0:PROD15#sh controllers coherentDSP 0/0/0/6 pm current 15-min prbs
Sat Jun 24 14:04:25 UTC
PRBS in the current interval [14:00:00 - 14:04:25 Sat Jun 24 2017]
PRBS current bucket type : Valid
EBC : 306
FOUND-COUNT : 5  FOUND-AT-TS : 14:04:16 Sat Jun 24 2017
LOST-COUNT : 5  LOST-AT-TS : 14:04:10 Sat Jun 24 2017
CONFIG-PTRN : PRBS_PATTERN_PN31
```
Verifying PRBS

You can monitor the status of Pseudo Random Binary Sequence (PRBS) on the CoherentDSP controller using the following command:

```
show controllers coherentDSP R/S/I/P prbs-details
```

```
RP/0/RP0/CP0:PROD15#sh controllers coherentDSP 0/0/0/6 prbs-details
Sat Jun 24 13:28:57.549 UTC
-------------------------PRBS details----------------------------------
PRBS Test : Enable
PRBS Mode : Source-Sink
PRBS Pattern : PN31
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.
Verifying PRBS
CHAPTER 5

Configuring MACsec Encryption

MAC Security (MACsec) is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec capable devices.

Security breaches can occur at any layer of the OSI model. Some of the common breaches at Layer 2 are MAC address spoofing, ARP spoofing, Denial of Service (DoS) attacks against a DHCP server, and VLAN hopping.

MACsec secures the data on physical media, making it impossible for data to be compromised at higher layers. As a result, MACsec encryption takes priority over any other encryption method for higher layers, such as IPsec and SSL.

MACsec provides encryption at the Layer 2, which is provided by the Advanced Encryption Standard (AES) algorithm that replaces the DES algorithm. MACsec uses the MACsec Key Agreement protocol (MKA) to exchange session keys, and manage encryption keys.

- MACsec Frame Format, on page 36
- MACsec SECTag Format , on page 37
- MACsec Key Agreement, on page 37
- MACsec in NCS 1002, on page 37
- Supported Configurations in Encrypted Mode, on page 38
- Illustrations for Supported Configurations in Encrypted Mode, on page 39
- Configure MACsec Encryption Using PSK Authentication, on page 40
- MACsec Key Chain, on page 40
- Configure MACsec Key Chain, on page 40
- Verify MACsec Key Chain, on page 42
- MACsec Policy, on page 43
- Configure MACsec Policy, on page 43
- Verify MACsec Policy, on page 45
- MACsec Controllers, on page 46
- Configure the Slice, on page 46
- Verify Slice Configuration, on page 48
- Apply MACsec Configuration on MACsec Controller, on page 51
- Verify MACsec Configuration on MACsec Controller, on page 52
- Verify State of MACSec Controller, on page 56
MACsec Frame Format

The MACsec header in a frame consists of three fields.

Table 2: Fields in MACsec Frame

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECTag</td>
<td>8 or 16 bytes</td>
<td>Identifies the Security Association Key (SAK) to be used to validate the received frame. The security tag also provides replay protection when frames are received out of sequence. With Secure Channel Identifier (SCI) encoding, the security tag is 16 bytes in length, and without the encoding, 8 bytes in length (SCI encoding is optional).</td>
</tr>
<tr>
<td>Secure Data</td>
<td>2+ octets</td>
<td>Data in the frame that is encrypted using MACsec.</td>
</tr>
<tr>
<td>ICV</td>
<td>128 bit</td>
<td>Integrity Check Value (ICV) that provides the integrity check for the frame. Frames that do not match the expected ICV are dropped at the port.</td>
</tr>
</tbody>
</table>

Figure 2: MACsec Frame Format
MACsec SECTag Format

The MACsec SECTag header in a frame consists of the following fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ET</td>
<td>16 bit</td>
<td>MACsec EtherType value (0x88E5) for MACsec packet.</td>
</tr>
<tr>
<td>TCI</td>
<td>6 bit</td>
<td>Tag control information that indicates how frame is protected.</td>
</tr>
<tr>
<td>AN</td>
<td>2 bit</td>
<td>Association number.</td>
</tr>
<tr>
<td>SL</td>
<td>8 bit</td>
<td>Short length of MAC service data unit (MSDU).</td>
</tr>
<tr>
<td>PN</td>
<td>32 bit</td>
<td>Packet sequence number.</td>
</tr>
<tr>
<td>SCI</td>
<td>64 bit</td>
<td>(optional) Secure channel identifier.</td>
</tr>
</tbody>
</table>

MACsec Key Agreement

The MACsec Key Agreement (MKA) Protocol, defined in IEEE 802.1X-2010, provides the required session keys and manages the required encryption keys. MKA is a multipoint to multipoint protocol that defines the mechanism to generate and distribute keys for MACsec.

MKA allows authorized multiple devices that possess secret key (CAK) to participate in a CA (Connectivity Association). It defines the election of Key Server (KS) that generates the Security Association Key (SAK) and distributes the SAK to all the participants. MACsec frames across the devices are secured using SAK. MKA also transports MACsec capability such as delay protection and confidentiality offset.

MKA operates in two modes.

- MKA using pre-shared key (PSK) authentication. See Configure MACsec Encryption Using PSK Authentication, on page 40 for configuration steps.

MACsec in NCS 1002

MACsec in NCS 1002 has the following characteristics or limitations.
• Supports 256-bit Extended Packet Numbering (XPN) according to IEEE 802.1AEbn-2011.
• Supports GCM-AES-XPN-256 as the default cipher.
• Supports AES-128-CMAC and AES-256-CMAC cryptographic algorithms.
• Supports SecY function in the data plane as specified by IEEE 802.1 AE-2006 specification.
• Supports only 2 x 100G client and 1 x 200G trunk traffic.
• Supports only cumulative statistics for MACsec counters.
• Supported only with the ncs1k-k9sec package.
• Not supported in the headless mode.
• Recommended to upgrade the nodes to R6.2.1 and bring up the 100G MACsec sessions.
• For 100G MACsec deployed in R6.1.1 and R6.1.2: If the customer migrates from R6.1.2 to R6.2.1, traffic hit occurs. The subsequent headless operations will not have any traffic drops.

### Supported Configurations in Encrypted Mode

The following configurations are supported on client and trunk ports in each slice configured in encrypted mode.

<table>
<thead>
<tr>
<th>Client Ports</th>
<th>Trunk Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 x 100G</td>
<td>1 x 200G</td>
</tr>
<tr>
<td>10 x 10G</td>
<td>1 x 100G</td>
</tr>
<tr>
<td>2 x 40G</td>
<td>1 x 100G</td>
</tr>
</tbody>
</table>

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

The following table describes the client and trunk ports in slice 0 that are enabled or disabled for each supported configuration in encrypted mode.

<table>
<thead>
<tr>
<th>Client Data Rate</th>
<th>Trunk Data Rate</th>
<th>Client Port 0</th>
<th>Client Port 1</th>
<th>Client Port 2</th>
<th>Client Port 3</th>
<th>Client Port 4</th>
<th>Client Port 5</th>
<th>Trunk Port 5</th>
<th>Trunk Port 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>100G</td>
<td>200G</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>D</td>
<td>E</td>
</tr>
</tbody>
</table>
### Illustrations for Supported Configurations in Encrypted Mode

The following illustrations describe the mapping of traffic from client to trunk ports in encrypted mode for the supported configurations.

**Figure 4: Client: 2 x 100G Trunk: 1 x 200G**

![Illustration of 2 x 100G to 1 x 200G mapping in encrypted mode]

**Figure 5: Client: 10 x 10G Trunk: 1 x 100G**

![Illustration of 10 x 10G to 1 x 100G mapping in encrypted mode]

**Figure 6: Client: 2 x 40G Trunk: 1 x 100G**

![Illustration of 2 x 40G to 1 x 100G mapping in encrypted mode]
Configure MACsec Encryption Using PSK Authentication

Configuring MACsec encryption using PSK authentication involves the following tasks:

1. Configure MACsec Key Chain, on page 40
2. Verify MACsec Key Chain, on page 42
3. Configure MACsec Policy, on page 43
4. Verify MACsec Policy, on page 45
5. Configure the Slice, on page 3
6. Verify Slice Configuration, on page 5
7. Apply MACsec Configuration on MACsec Controller, on page 51
8. Verify MACsec Configuration on MACsec Controller, on page 52

MACsec Key Chain

A MACsec key chain is a collection of keys used to authenticate peers needing to exchange encrypted information. While creating a key chain, we define the key(s), key string with password, the cryptographic algorithm, and the key lifetime.

- The key can be up to 64 characters in length.
- The key name must be of even number of characters. Entering an odd number of characters will exit the MACsec configuration mode. The key name must match on both the sides.
- The key string is 64 hexadecimal characters in length when AES 256-bit encryption algorithm is used and 32 hexadecimal characters in length when AES 128-bit encryption algorithm is used. It is recommended to create key name and provide the key-string and lifetime.
- The lifetime period (validity period of the key) can be configured, with a duration in seconds, as a validity period between two dates (for example, Jan 01 2016 to Dec 31 2016), or with infinite validity. The key is valid from the time you configure (in HH:MM:SS format). The duration is configured in seconds. The overlapping time must be configured in two keys to avoid traffic loss.
- The keys roll over to the next key within the same key chain by configuring a second key (key 02) in the key chain and configuring lifetime for the first key. When the lifetime of the first key (key 01) expires, it automatically rolls over to the next key in the list. If the same key is configured simultaneously on both sides of the link, the key rollover is hitless and the key rolls over without interruption in traffic. Based on IEEE 802.1x, the overlapping time between the keys in a key chain can be up to 20 seconds. The re-key operation can take up to 16 seconds.

Configure MACsec Key Chain

```bash
configure
```
key chain key-chain-name macsec
key key-name
key-string password cryptographic-algorithm \{aes-256-cmac \aes-128-cmac\}
lifetime start_time start_date \{ end_time end_date \ duration validity \ infinite \}
exit
commit

Examples
The following is a sample in which the key chain is configured with AES 256-bit encryption algorithm and specific duration for the lifetime period.

```sh
configure
key chain mac_chain macsec
key 1234abcd5678
key-string 1234567812345678123456781234567812345678123456781234567812345678
 cryptographic-algorithm aes-256-cmac
 lifetime 05:00:00 01 july 2016 duration 1800
 exit
 commit
```

The following is a sample in which the key chain is configured with AES 256-bit encryption algorithm and defined period for the lifetime period.

```sh
configure
key chain mac_chain macsec
key 1234abcd5678
key-string 1234567812345678123456781234567812345678123456781234567812345678
 cryptographic-algorithm aes-256-cmac
 lifetime 05:00:00 20 july 2016 12:00:00 30 september 2016
 exit
 commit
```

The following is a sample in which the key chain is configured with AES 256-bit encryption algorithm and infinite duration for the lifetime period.

```sh
configure
key chain mac_chain macsec
key 1234abcd5678
key-string 1234567812345678123456781234567812345678123456781234567812345678
 cryptographic-algorithm aes-256-cmac
 lifetime 05:00:00 01 January 2015 infinite
 exit
 commit
```

The following is a sample in which the key chain is configured with AES 128-bit encryption algorithm and specific duration for the lifetime period.

```sh
configure
key chain mac_chain macsec
key abc1
key-string 12345678123456781234567812345678123456781234567812345678
 cryptographic-algorithm aes-128-cmac
 lifetime 17:30:00 31 August 2016 duration 4000
 exit
 commit
```
The following is a sample in which the key chain is configured with AES 128-bit encryption algorithm and defined period for the lifetime period.

```plaintext
configure
key_chain mac_chain macsec
key abc2
key-string 12345678123456781234567812345678 cryptographic-algorithm aes-128-cmac
lifetime 17:30:00 31 August 2016 12:00:00 30 September 2016
exit
commit
```

The following is a sample in which the key chain is configured with AES 128-bit encryption algorithm and infinite duration for the lifetime period.

```plaintext
configure
key_chain mac_chain macsec
key abc3
key-string 12345678123456781234567812345678 cryptographic-algorithm aes-128-cmac
lifetime 05:00:00 01 January 2015 infinite
exit
commit
```

**Associated Commands**

- `key_chain`
- `key`
- `key-string`
- `cryptographic-algorithm`
- `lifetime`

### Verify MACsec Key Chain

```
show key chain

Wed Aug 17 14:34:00.056 IST
Key-chain: TESTMA -(MacSec)
  Key BDA123
    Key-String -- 08701E1D5D4C53404A5A5E577E727F6B647040534355560E080A00005B554F4E
    Cryptographic-Algorithm -- ALG_AES_256_CMAC
    Send lifetime -- 19:05:00, 16 Aug 2016 - Always valid [Valid now]

Key-chain: mac_chain -(MacSec)
  Key abc1
    Key-String -- 12485744465E5A53327A767B676074445F475152020C0E040C5F514B420C0E000B
    Cryptographic-Algorithm -- ALG_AES_128_CMAC
    Send lifetime -- 17:30:00, 31 Aug 2016 - (Duration) 4000

  Key abc2
    Key-String -- 135445415F9527D73357A60617745504E5253050D0D050356524A450DDD01040A
    Cryptographic-Algorithm -- ALG_AES_128_CMAC
    Send lifetime -- 17:30:00, 31 Aug 2016 - 12:00:00, 30 Sep 2016
```
Key abc3
Key-String -- 101F5B4A5142445C54557878707D65627A425545575400E002065D574D400E00
Cryptographic-Algorithm -- ALG_AES_128_CMAC
Send lifetime -- 05:00:00, 01 Jan 2015 - Always valid [Valid now]

MACsec Policy

You apply a defined MACsec policy to enable MKA on the controller. You can configure these parameters for MACsec policy:

- Policy name, not to exceed 16 ASCII characters.
- Confidentiality (encryption) offset of 0 bytes.
- Replay protection. You can configure MACsec window size, as defined by the number of out-of-order frames that are accepted. This value is used while installing the security associations in the MACsec. A value of 0 means that frames are accepted only in the correct order.
- The cipher suite to be used for MACsec encryption is GCM-AES-XPN-256.
- The range of key server priority parameter is 0 to 255. Lower the value, higher the preference to be selected as the key server.
- The security-policy parameter configures the type of traffic (encrypted traffic or all traffic) that is allowed through the controller configured with MACsec. The default value of security-policy parameter is must-secure that indicates unencrypted packets cannot be transmitted or received except MKA control protocol packets.

Configure MACsec Policy

```
configure
macsec-policy policy-name
cipher-suite encryption-suite
conf-offset offset-value
key-server-priority value
security-policy {should-secure | must-secure}
window-size value
exit
commit
```

Examples

Example 1: The following is a sample of configuring the MACsec policy.

```
configure
macsec-policy mac_policy
cipher-suite GCM-AES-XPN-256
```
Configuring MACsec Encryption

Configure MACsec Policy

Example 2: If a specific setting does not apply to NCS 1002, the setting is rejected during commit.

```
configure
macsec-policy mac_policy
  vlan-tags-in-clear 1
commit
```

Thu Aug 4 19:31:38.033 UTC

% Failed to commit one or more configuration items during a pseudo-atomic operation. All changes made have been reverted. Please issue 'show configuration failed [inheritance]' from this session to view the errors

```
show configuration failed
```

Thu Aug 4 19:31:56.601 UTC

!! SEMANTIC ERRORS: This configuration was rejected by the system due to semantic errors. The individual errors with each failed configuration command can be found below.

```
macsec-policy mac_policy
  !!! A verifier or EDM callback function returned: 'not supported': vlan_tags_in_clear is not supported.

  vlan-tags-in-clear 1
  !!! A verifier or EDM callback function returned: 'not supported': vlan_tags_in_clear is not supported.

! end
```

Example 3: If a specific configuration in the batch operation is not supported, the entire batch is rejected during commit.

```
configure
macsec-policy mac_policy
  cipher-suite GCM-AES-XPN-256
  window-size 64
  conf-offset CONF-OFFSET-0
commit
```

Thu Aug 4 19:37:22.355 UTC

% Failed to commit one or more configuration items during a pseudo-atomic operation. All changes made have been reverted. Please issue 'show configuration failed [inheritance]' from this session to view the errors

```
show configuration failed
```

Thu Aug 4 19:38:29.948 UTC

!! SEMANTIC ERRORS: This configuration was rejected by the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.

macsec-policy mac_policy
!!% A verifier or EDM callback function returned: 'not supported': The only supported
conf_offset is CONF-OFFSET-0

conf-offset CONF-OFFSET-0
!!% A verifier or EDM callback function returned: 'not supported': The only supported
conf_offset is CONF-OFFSET-0

window-size 64
!!% A verifier or EDM callback function returned: 'not supported': The only supported
conf_offset is CONF-OFFSET-0

cipher-suite GCM-AES-XPN-256
!!% A verifier or EDM callback function returned: 'not supported': The only supported
conf_offset is CONF-OFFSET-0

!
end

Associated Commands

- macsec-policy
- cipher-suite
- conf-offset
- key-server-priority
- security-policy
- window-size

Verify MACsec Policy

show macsec policy

Sun Dec 18 14:22:23.587 IST

Total Number of Policies = 3

<table>
<thead>
<tr>
<th>Policy</th>
<th>Cipher Suite</th>
<th>Key-Svr Priority</th>
<th>Window Size</th>
<th>Conf Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>DEFAULT POLICY</em></td>
<td>GCM-AES-XPN-256</td>
<td>16</td>
<td>64</td>
<td>0</td>
</tr>
<tr>
<td>kcp1</td>
<td>GCM-AES-XPN-256</td>
<td>16</td>
<td>128</td>
<td>0</td>
</tr>
<tr>
<td>kcp2</td>
<td>GCM-AES-XPN-256</td>
<td>16</td>
<td>256</td>
<td>0</td>
</tr>
</tbody>
</table>

show macsec policy 5

Wed Mar 30 12:49:29.371 UTC

<table>
<thead>
<tr>
<th>Policy name</th>
<th>Cipher Suite</th>
<th>Key-Svr Priority</th>
<th>Window Size</th>
<th>Conf Offset</th>
</tr>
</thead>
</table>

Configuration Guide for Cisco NCS 1002, IOS XR Release 6.3.x
If the values you see are different from the ones you configured, then check your configuration by running the `show run macsec-policy` command.

### MACsec Controllers

MACsec controllers are created when a slice is provisioned with the `encrypted` keyword. The MACsec controller is used to configure the MACsec parameters. All the MACsec statistics is available on the MACsec controller. The MACsec controller is represented in the `Rack/Slot/Instance/Port` format, for example, 0/0/0/3.

A unique MAC address is generated for each MACsec controller. When software is upgraded to R6.2.2 with traffic, traffic loss occurs for the slice configured in encrypted mode.

### Configure the Slice

You can configure the slice with traffic on client and trunk ports. All five client ports of the slice need to be configured at the same bitrate except for mixed mode configuration. Both the trunk ports are always set with the same FEC mode. The slice can be configured to send encrypted traffic from R6.1.1.

See the Supported Configurations in Encrypted Mode, on page 38 section to determine the supported configurations on the client and trunk ports in each slice configured in encrypted mode.

**Note**

When the slice is configured in encrypted mode, the drop-lldp cannot be enabled.

**Note**

When NCS 1002 is installed in a system where both the trunk interfaces in a slice are used, the two 250Gb 16QAM signals need to be co-routed on the same fiber (mandatory when the 5x100Gb client port is provisioned). Also, it is recommended to use adjacent wavelengths when the line modulation is set to 250Gb 16QAM. The reason for this is that the chromatic dispersion generates skew between wavelengths. Assuming a Dispersion of 10000 ps/nm, a span of 500 km, and using adjacent channel, the skew is evaluated in less than 200 ns and it is compensated by the deskew capability of NCS 1002. If the delta between the used channels is increased, the skew increases and it might exceed the skew compensation done by NCS 1002.

To configure the slice with unencrypted traffic, use the following commands.

```
configure
hw-module location location slice [slice_number | all] client bitrate {10G | 40G | 100G} trunk bitrate {100G | 200G | 250G} fec {softdecision7 | softdecision20}
commit
```

To configure the slice with mixed mode, use the following commands.

```
configure
hw-module location location slice [slice_number | all] client bitrate 10G-100G trunk bitrate 200G fec {softdecision7 | softdecision20}
```
commit

To configure the slice with encrypted traffic, use the following commands.

```
configure
hw-module location location slice [slice_number | all] client bitrate { 10G | 40G | 100G } trunk bitrate { 100G | 200G } fec { softdecision7 | softdecision20 } [ encrypted ]
commit
```

**Examples**

The following is a sample in which slice 0 is configured in mixed mode, and FEC on the trunk ports is set to softdecision7.

```
configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 10G-100G trunk bitrate 200G fec SoftDecision7
commit
```

The following is a sample in which slice 0 is configured to send encrypted traffic with 100G client rate, 200G trunk rate, and FEC on the trunk ports is set to softdecision7.

```
configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 100G trunk bitrate 200G softdecision7 encrypted
commit
```

The following is a sample in which slice 0 is configured to send encrypted traffic with 10G client rate, 100G trunk rate, and FEC on the trunk ports is set to softdecision20. When a slice is configured with 10G client rate in encrypted mode, ten MACsec controllers are created for each slice. When all the four slices are configured with 10G client rate in encrypted mode, forty MACsec controllers are created for NCS 1002. Two MACsec controllers are created for the middle port, four controllers for the fourth port, and four controllers for the fifth port per slice.

```
configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 10G trunk bitrate 100G softdecision20 encrypted
commit
```

The following is a sample in which slice 0 is configured to send encrypted traffic with 40G client rate, 100G trunk rate, and FEC on the trunk ports is set to softdecision20.

```
configure
hw-module location 0/RP0/CPU0 slice 0 client bitrate 40G trunk bitrate 100G softdecision20 encrypted
commit
```

The following is a sample to configure all the slices with a specific client rate and trunk rate.

```
configure
hw-module location 0/RP0/CPU0 slice all client bitrate 10G trunk bitrate 100G fec softDecision7
commit
```

```
configure
hw-module location 0/RP0/CPU0 slice all client bitrate 40G trunk bitrate 100G fec
```
The following is a sample to remove the configuration from all the slices.

```plaintext
configure
no hw-module location 0/RP0/CPU0 slice all client bitrate 10G trunk bitrate 100G fec
softDecision?
commit

configure
no hw-module location 0/RP0/CPU0 slice all client bitrate 40G trunk bitrate 100G fec
softDecision?
commit

configure
no hw-module location 0/RP0/CPU0 slice all client bitrate 100G trunk bitrate 200G fec
softDecision?
commit
```

**Associated Commands**

- `hw-module`
- `show hw-module`

## Verify Slice Configuration

Use this procedure to verify whether the slice is correctly configured.

```plaintext
show hw-module { slice [ slicenumber | all ] }
```

**Example:**

```
RP/0/RP0/CPU0:ios# show hw-module slice 0
Thu Aug 11 16:16:58.935 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT

Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage
```
Verify Slice Configuration

HundredGigECtrlr0/0/0/3 100
HundredGigECtrlr0/0/0/4 100
RP/0/RP0/CP00:ios# show hw-module slice 0

Sun Dec 18 13:59:18.805 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 40
Trunk Bitrate: 100
DP FPGA FW Type: MM40
DP FPGA FW Version: 03.00
HW Status: CURRENT

Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage
FortyGigECtrlr0/0/0/3 100
FortyGigECtrlr0/0/0/4 100

FortyGigECtrlr0/0/0/3 100
FortyGigECtrlr0/0/0/4 100
RP/0/RP0/CP00:ios# show hw-module slice 1

Tue Jan 1 06:55:12.293 UTC
Slice ID: 1
Status: Provisioned
Client Bitrate: 10
Trunk Bitrate: 100
DP FPGA FW Type: MM10
DP FPGA FW Version: 03.00
HW Status: CURRENT

Encryption Supported: TRUE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/13
Traffic Split Percentage
TenGigECtrlr0/0/0/9/1 100
TenGigECtrlr0/0/0/9/2 100
TenGigECtrlr0/0/0/10/1 100
TenGigECtrlr0/0/0/10/2 100
TenGigECtrlr0/0/0/10/3 100
TenGigECtrlr0/0/0/10/4 100
TenGigECtrlr0/0/0/11/1 100
TenGigECtrlr0/0/0/11/2 100
TenGigECtrlr0/0/0/11/3 100
TenGigECtrlr0/0/0/11/4 100
RP/0/RP0/CP00:ios# show hw-module slice 2

Slice ID: 2
Status: Provisioned
Client Bitrate: 10,100
Trunk Bitrate: 200
DP FPGA FW Type: RMM
DP FPGA FW Version: 04.00
HW Status: CURRENT

Encryption Supported: FALSE
LLDP Drop Enabled: FALSE
Client Port - Trunk Port CoherentDSP0/0/0/19 CoherentDSP0/0/0/20
Traffic Split Percentage
Pipe H-400044-0001

Displays the details of the slice such as the slice ID, client rate, trunk rate, and the traffic percentage carried on the trunk ports. The Encryption Supported field indicates whether the slice is provisioned with firmware that supports encryption or not.

Note The HW Status field might display "Need Upgrade" when the user needs to use the MACsec feature and upgrades from R6.0.1 to 6.1.1. Hence, the control FPGA (CTRL_BKP_UP, CTRL_BKP_LOW, CTRL_FPGA_UP, and CTRL_FPGA_LOW) needs to be upgraded to the latest firmware version provided by R6.1.1. See Verify Firmware Version for more information.

The Provisioned status does not indicate that the traffic can flow immediately. For example, use the `show controllers maCSecCtrlr 0/0/0/3` command output to view the provisioning information of the port after the slice is provisioned.

Example:

```
RP/0/RP0/CPU0:ios# show hw-module slice all
Thu Aug 11 16:16:58.935 IST
Slice ID: 0
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/6
Traffic Split Percentage

HundredGigE Ctrlr0/0/0/3 100
HundredGigE Ctrlr0/0/0/4 100

Slice ID: 1
Status: Provisioned
Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT
Encryption Supported: TRUE
Client Port - Trunk Port CoherentDSP0/0/0/13
Traffic Split Percentage

HundredGigE Ctrlr0/0/0/10 100
HundredGigE Ctrlr0/0/0/11 100

Slice ID: 2
Status: Provisioned
Client Bitrate: 100
```
| Trunk Bitrate: | 200 |
| DP FPGA FW Type: | M100 |
| DP FPGA FW Version: | 02.00 |
| HW Status: | CURRENT |

Encryption Supported: TRUE

Client Port - Trunk Port CoherentDSP0/0/0/20
Traffic Split Percentage

| HundredGigECtrlr0/0/0/17 | 100 |
| HundredGigECtrlr0/0/0/18 | 100 |

Slice ID: 3

Status: Provisioned

Client Bitrate: 100
Trunk Bitrate: 200
DP FPGA FW Type: M100
DP FPGA FW Version: 02.00
HW Status: CURRENT

Encryption Supported: TRUE

Client Port - Trunk Port CoherentDSP0/0/0/27
Traffic Split Percentage

| HundredGigECtrlr0/0/0/24 | 100 |
| HundredGigECtrlr0/0/0/25 | 100 |

Associated Commands

- `hw-module`
- `show hw-module`

### Apply MACsec Configuration on MACsec Controller

You can apply the MACsec key chain and policy configuration on the MACsec controller.

```plaintext
configure
ccontroller MACSecCtrl Rack/Slot/Instance/Port
macsec psk-keychain key-chain-name [policy policy-name]
exit
commit
```

**Example**

```plaintext
configure
ccontroller MACSecCtrl 0/0/0/3
macsec psk-keychain mac_chain policy mac_policy
exit
commit
```
Verify MACsec Configuration on MACsec Controller

1. Verify the MACsec configuration on the controller.

   **show macec mka summary**

   Wed Mar 30 13:35:15.497 UTC
   NODE: node0_RP0_CPU0

   Interface Status Cipher-Suite KeyChain
   MS0/0/0/03 Secured GCM-AES-XPN-256 mac_chain

   Total MACSec Sessions : 1
   Secured Sessions : 1
   Pending Sessions : 0

   The **Status** field in the output confirms that the respective controller is **Secured**. If MACsec encryption is not successfully configured, you will see a status such as **Pending** or **Init**.

2. Verify whether the MKA session is secured with MACsec on the respective controller.

   **show macec mka session**

   Sun Dec 18 14:20:50.626 IST
   NODE: node0_RP0_CPU0

   Interface Local-TxSCI # Peers Status Key-Server
   MS0/0/0/3 3820.563b.eacc/0003 1 Secured YES
   MS0/0/0/18 3820.563b.eacc/0012 1 Secured NO
   MS0/0/0/17 3820.563b.eacc/0011 1 Secured NO
   MS0/0/0/4 3820.563b.eacc/0004 1 Secured YES

   **show macec mka session controller MS0/0/0/03 detail**

   Tue Aug 16 14:08:04.927 IST

   MKA Detailed Status for MKA Session
   Status: SECURED - Secured MKA Session with MACsec
   Local Tx-SCI : 3820.563b.eacc/0003
   Local Tx-SSCI : 2
   Interface MAC Address : 3820.563b.eacc
   MKA Port Identifier : 10
   Interface Name : MS0/0/0/03
   CAK Name (CKN) : ABC1000000000000000000000000000000000000000000000000000000000000
   Member Identifier (MI) : 6609E25F8ACC8653301503B
   Message Number (MN) : 39
   Authenticator : NO
   Key Server : YES
   MKA Cipher Suite : AES-128-CMAC
The **Status** field in the output verifies if the MKA session is secured with MACsec encryption. The output also displays information about the controller and other MACsec parameters.

3. **Verify the MACsec session counter statistics.**

```bash
tue Aug 11 16:02:41.330 IST
MKA Global Statistics

MKA Session Totals
  Secured.................... 0
  Reauthentication Attempts.. 0
  Deleted (Secured).......... 0
  Keepalive Timeouts........ 0

CA Statistics
  Pairwise CAKs Derived...... 0
  Pairwise CAK Rekeys........ 0
  Group CAKs Generated...... 0
  Group CAKs Received....... 0

SA Statistics
  SAKs Generated............ 0
  SAKs Rekeyed.............. 0
  SAKs Received............. 0
  SAK Responses Received.... 0

MKPDU Statistics
  MKPDUs Validated & Rx..... 5
```
The counters display the MACsec PDUs transmitted, validated, and received. The output also displays transmission errors, if any. This completes the verification of MACsec encryption on NCS 1002.

1. Verify the status of the MACsec controller.

   show macsec platform status controller MacSecCtrlr 0/0/0/3

   Mon Jun 6 20:57:15.900 UTC

   -------------------------------------
### Interface Status

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReplayWindowSize</td>
<td>64</td>
</tr>
<tr>
<td>MustSecure</td>
<td>TRUE</td>
</tr>
<tr>
<td>SecureMode</td>
<td>2</td>
</tr>
</tbody>
</table>

### Encrypted Secure Channel Status

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProtectionEnabled</td>
<td>TRUE</td>
</tr>
<tr>
<td>SecureChannelID</td>
<td>0x0200d05a57395540</td>
</tr>
<tr>
<td>ConfidentialityOffset</td>
<td>0</td>
</tr>
<tr>
<td>CipherSuite</td>
<td>GCM-AES-XPN-256</td>
</tr>
<tr>
<td>SecureTagLength</td>
<td>16</td>
</tr>
<tr>
<td>InitialPacketNumber</td>
<td>1</td>
</tr>
<tr>
<td>MaxPacketNumber</td>
<td>18446744073709551615</td>
</tr>
<tr>
<td>RecentPacketNumber</td>
<td>364865080</td>
</tr>
</tbody>
</table>

### Encrypted Active Associations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssociationNumber</td>
<td>1</td>
</tr>
<tr>
<td>DeviceAssociationNum</td>
<td>1</td>
</tr>
<tr>
<td>ShortSecureChannelID</td>
<td>1</td>
</tr>
<tr>
<td>ProgrammedTime</td>
<td>2016 Jun 6 20:57:09.690</td>
</tr>
<tr>
<td>KeyCRC</td>
<td>0x6fe6f59c</td>
</tr>
<tr>
<td>XpnSalt</td>
<td>0xffca89c5 0x4a307f93 0xd3df482e</td>
</tr>
</tbody>
</table>

### Decrypted Secure Channel Status

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProtectionEnabled</td>
<td>TRUE</td>
</tr>
<tr>
<td>SecureChannelID</td>
<td>0x0100d05a57395540</td>
</tr>
<tr>
<td>ConfidentialityOffset</td>
<td>0</td>
</tr>
<tr>
<td>CipherSuite</td>
<td>GCM-AES-XPN-256</td>
</tr>
<tr>
<td>InitialPacketNumber</td>
<td>1</td>
</tr>
<tr>
<td>MaxPacketNumber</td>
<td>18446744073709551615</td>
</tr>
<tr>
<td>RecentPacketNumber</td>
<td>370010268</td>
</tr>
</tbody>
</table>

### Decrypted Active Associations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssociationNumber</td>
<td>1</td>
</tr>
<tr>
<td>DeviceAssociationNum</td>
<td>1</td>
</tr>
<tr>
<td>ShortSecureChannelID</td>
<td>2</td>
</tr>
<tr>
<td>ProgrammedTime</td>
<td>2016 Jun 6 20:57:09.550</td>
</tr>
<tr>
<td>KeyCRC</td>
<td>0x6fe6f59c</td>
</tr>
<tr>
<td>XpnSalt</td>
<td>0xffca89c5 0x4a307f93 0xd3df482e</td>
</tr>
</tbody>
</table>

When IOS XR is reloaded, two association numbers are displayed under Decrypted Active Associations. After the reload, key rollover is required. When the key rollover happens, the active association number is associated.
Verify State of MACSec Controller

The state of MACSec controller can be verified using the `show controllers MACSecCtrlr R/S/I/P` command. If the state of MACSec controller is down, the corresponding MKA sessions do not come up.

The state of MACSec controller is down upon one of the following conditions.

- State of the corresponding Ethernet controller is Admin Down. The state can be verified using the `show controllers HundredGigECtrlr R/S/I/P` command.
- State of the optics controller is Admin Down or Operational Down. The state can be verified using the `show controllers optics R/S/I/P` command.
- Client optics is not present. The client optics can be verified using the `show inventory` command.

The state of the Ethernet controller can be changed from Admin Down using the following commands.

```
configure
ccontroller HundredGigECtrlr Rack/Slot/Instance/Port
no shutdown
commit
```

The state of the optics controller can be changed from Admin Down or Operational Down using the following commands.

```
configure
ccontroller optics Rack/Slot/Instance/Port
no shutdown
commit
```

SecY Statistics

SecY statistics is used to identify issues with the encrypted traffic.

Before You Begin

Ensure that MKA sessions are established. See Verify MACsec Configuration on MACsec Controller, on page 52 for more information.

100G MACsec

```
show macsec secy stats controller MACSecCtrlr 0/0/0/3 SC
```

<table>
<thead>
<tr>
<th>Interface Stats</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>InPktsUntagged</td>
<td>0</td>
</tr>
<tr>
<td>InPktsNoTag</td>
<td>0</td>
</tr>
<tr>
<td>InPktsBadTag</td>
<td>0</td>
</tr>
<tr>
<td>InPktsUnknownSCI</td>
<td>0</td>
</tr>
<tr>
<td>InPktsNoSCI</td>
<td>0</td>
</tr>
<tr>
<td>InPktsOverrun</td>
<td>0</td>
</tr>
<tr>
<td>InOctetsValidated</td>
<td>0</td>
</tr>
<tr>
<td>InOctetsDecrypted</td>
<td>121697919056</td>
</tr>
<tr>
<td>OutPktsUntagged</td>
<td>0</td>
</tr>
<tr>
<td>OutPktsTooLong</td>
<td>0</td>
</tr>
<tr>
<td>OutOctetsProtected</td>
<td>0</td>
</tr>
</tbody>
</table>
OutOctetsEncrypted : 194316914428

SC Stats
TxSC Stats
OutPktsProtected : 0
OutPktsEncrypted : 130941317
OutOctetsProtected : 0
OutOctetsEncrypted : 194316914428
OutPktsTooLong : 0

TxSA Stats
TxSA 0:
   OutPktsProtected : 0
   OutPktsEncrypted : 0
   NextPN : 0
TxSA 1:
   OutPktsProtected : 0
   OutPktsEncrypted : 130941317
   NextPN : 130940105
TxSA 2:
   OutPktsProtected : 0
   OutPktsEncrypted : 0
   NextPN : 0
TxSA 3:
   OutPktsProtected : 0
   OutPktsEncrypted : 0
   NextPN : 0

RxSC Stats
RxSC 1:
   InPktsUnchecked : 0
   InPktsDelayed : 0
   InPktsLate : 0
   InPktsOK : 82006684
   InPktsInvalid : 0
   InPktsNotValid : 0
   InPktsNotUsingSA : 0
   InPktsUnusedSA : 0
   InPktsUntaggedHit : 0
   InOctetsValidated : 0
   InOctetsDecrypted : 121697919056

RxSA Stats
RxSA 0:
   InPktsUnusedSA : 0
   InPktsNotUsingSA : 0
   InPktsNotValid : 0
   InPktsInvalid : 0
   InPktsOK : 0
   NextPN : 1
RxSA 1:
   InPktsUnusedSA : 0
   InPktsNotUsingSA : 0
   InPktsNotValid : 0
   InPktsInvalid : 0
   InPktsOK : 82006684
   NextPN : 82004142
RxSA 2:
   InPktsUnusedSA : 0
   InPktsNotUsingSA : 0
   InPktsNotValid : 0
   InPktsInvalid : 0
   InPktsOK : 0
   NextPN : 0
RxSA 3:
   InPktsUnusedSA : 0
The SecY SA counters are displayed as 64 bit values in the CLI.

10G MACsec

show macsec secy stats controller MACSecCtrlr 0/0/0/3/1 SC

Mon Dec 19 17:04:00.467 IST
Interface Stats
   InPktsUntagged : 0
   InPktsNoTag : 0
   InPktsBadTag : 0
   InPktsUnknownSCI : 0
   InPktsNoSCI : 0
   InPktsOverrun : 0
   InOctetsValidated : 0
   InOctetsDecrypted : 3244694362816
   OutPktsUntagged : 0
   OutPktsTooLong : 0
   OutOctetsProtected : 0
   OutOctetsEncrypted : 3225943872072

SC Stats
   TxSC Stats
   OutPktsProtected : 0
   OutPktsEncrypted : 336597056
   OutOctetsProtected : 0
   OutOctetsEncrypted : 3225943872072
   OutPktsTooLong : 0

   RxSC Stats
   RxSC 1: 0
   InPktsUnchecked : 0
   InPktsDelayed : 0
   InPktsLate : 0
   InPktsOK : 338553493
   InPktsInvalid : 0
   InPktsNotValid : 0
   InPktsNotUsingSA : 1320396
   InPktsUnusedSA : 0
   InPktsUntaggedHit : 0
   InOctetsValidated : 0
   InOctetsDecrypted : 3244694362816

Trunk Side Statistics

Trunk side statistics is used to isolate issues with the encrypt and decrypt blocks. In the Tx direction, the trunk side Egress statistics display statistics after the encrypt block. In the Rx direction, the trunk side Ingress statistics display statistics before the decrypt block.

show controllers MACSecCtrlr 0/0/0/3 stats

Tue Jan 22 04:51:40.858 IST
Statistics for interface MACSecCtrlr0/0/0/3 (cached values):

Ingress:
### Trunk Side Statistics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input total bytes</td>
<td>805443936740</td>
</tr>
<tr>
<td>Input good bytes</td>
<td>805443936740</td>
</tr>
<tr>
<td>Input total packets</td>
<td>525746695</td>
</tr>
<tr>
<td>Input 802.1Q frames</td>
<td>0</td>
</tr>
<tr>
<td>Input pause frames</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 64 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 65-127 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 128-255 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 256-511 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 512-1023 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 1024-1518 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input pkts 1519-Max bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input good pkts</td>
<td>525746695</td>
</tr>
<tr>
<td>Input unicast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Input multicast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Input broadcast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Input drop overrun</td>
<td>0</td>
</tr>
<tr>
<td>Input drop abort</td>
<td>0</td>
</tr>
<tr>
<td>Input drop invalid VLAN</td>
<td>0</td>
</tr>
<tr>
<td>Input drop invalid DMAC</td>
<td>0</td>
</tr>
<tr>
<td>Input drop invalid encap</td>
<td>0</td>
</tr>
<tr>
<td>Input drop other</td>
<td>0</td>
</tr>
<tr>
<td>Input error giant</td>
<td>0</td>
</tr>
<tr>
<td>Input error runt</td>
<td>0</td>
</tr>
<tr>
<td>Input error jabbers</td>
<td>0</td>
</tr>
<tr>
<td>Input error fragments</td>
<td>0</td>
</tr>
<tr>
<td>Input error CRC</td>
<td>0</td>
</tr>
<tr>
<td>Input error collisions</td>
<td>0</td>
</tr>
<tr>
<td>Input error symbol</td>
<td>0</td>
</tr>
<tr>
<td>Input error other</td>
<td>0</td>
</tr>
<tr>
<td>Input MIB giant</td>
<td>0</td>
</tr>
<tr>
<td>Input MIB jabber</td>
<td>0</td>
</tr>
<tr>
<td>Input MIB CRC</td>
<td>0</td>
</tr>
</tbody>
</table>

**Egress:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output total bytes</td>
<td>880411742408</td>
</tr>
<tr>
<td>Output good bytes</td>
<td>880411742408</td>
</tr>
<tr>
<td>Output total packets</td>
<td>574681294</td>
</tr>
<tr>
<td>Output 802.1Q frames</td>
<td>0</td>
</tr>
<tr>
<td>Output pause frames</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 64 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 65-127 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 128-255 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 256-511 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 512-1023 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 1024-1518 bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output pkts 1519-Max bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output good pkts</td>
<td>574681294</td>
</tr>
<tr>
<td>Output unicast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Output multicast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Output broadcast pkts</td>
<td>0</td>
</tr>
<tr>
<td>Output drop underrun</td>
<td>0</td>
</tr>
<tr>
<td>Output drop abort</td>
<td>0</td>
</tr>
<tr>
<td>Output drop other</td>
<td>0</td>
</tr>
<tr>
<td>Output error other</td>
<td>0</td>
</tr>
</tbody>
</table>
Control Plane Statistics

show macsec mka statistics controller macSecCtrlr 0/0/0/3

This command displays control plane statistics for the specific MACSec controller.

MKA Statistics for Session on interface (M50/0/0/3)

Reauthentication Attempts.. 0

CA Statistics
  Pairwise CAKs Derived... 0
  Pairwise CAK Rekeys..... 0
  Group CAKs Generated.... 0
  Group CAKs Received..... 0

SA Statistics
  SAKs Generated.......... 1
  SAKs Rekeyed............ 0
  SAKs Received........... 0
  SAK Responses Received.. 1

MKPDU Statistics
  MKPDUs Transmitted...... 3305
    "Distributed SAK".. 1
    "Distributed CAK".. 0
  MKPDUs Validated & RX... 3305
    "Distributed SAK".. 0
    "Distributed CAK".. 0

MKA IDB Statistics
  MKPDUs Tx Success........ 3305
  MKPDUs Tx Fail............ 0
  MKPDUs Tx Pkt build fail... 0
  MKPDUs No Tx on intf down.. 2
  MKPDUs No Rx on intf down.. 0
  MKPDUs Rx CA Not found..... 0
  MKPDUs Rx Error............ 0
  MKPDUs Rx Success......... 3305
  MKPDUs Rx Invalid Length... 0
  MKPDUs Rx Invalid CKN..... 0

Configuration Guide for Cisco NCS 1002, IOS XR Release 6.3.x
MKPDU Failures
MKPDU Rx Validation (ICV).................. 0
MKPDU Rx Bad Peer MN...................... 0
MKPDU Rx Non-recent Peerlist MN........... 0
MKPDU Rx Drop SAKUSE, KN mismatch....... 0
MKPDU Rx Drop SAKUSE, KN Not Set......... 0
MKPDU Rx Drop SAKUSE, Key MI mismatch... 0
MKPDU Rx Drop SAKUSE, AN Not in Use...... 0
MKPDU Rx Drop SAKUSE, KS Rx/Tx Not Set... 0
MKPDU Rx Drop Packet, Ethertype Mismatch.. 0
MKPDU Rx Drop Packet, Source MAC NULL.... 0
MKPDU Rx Drop Packet, Destination MAC NULL 0
MKPDU Rx Drop Packet, Payload NULL....... 0

SAK Failures
SAK Generation........................... 0
Hash Key Generation....................... 0
SAK Encryption/Wrap.................... 0
SAK Decryption/Unwrap................... 0

CA Failures
ICK Derivation............................ 0
KEK Derivation............................ 0
Invalid Peer MACsec Capability......... 0

MACsec Failures
Rx SC Creation........................... 0
Tx SC Creation........................... 0
Rx SA Installation....................... 0
Tx SA Installation....................... 0

clear macsec mka statistics controller macSecCtrlr 0/0/0/3
This command clears control plane statistics for the specific MACSec controller.

Tue Jan 22 04:59:33.830 IST

Configuring MACsec Threshold Crossing Alerts

You can configure MACsec Threshold Crossing Alerts (TCA) at mac-sec ether, secy-if (interface), and secy-tx. There is no default threshold, minimum or maximum threshold for configuring MACsec TCA. You can configure it between the range 1 to 4294967295. By default, you can find TCA in show logging command. You can configure syslog server in MACsec TCA, and view TCA in syslog server such as EPNM specific parameter. You can set TCA for all supported buckets such as 0-sec/15-mins/24-hour for parameters in mac-sec ether/secy-if/secy-tx/secy-rx.

Use the following command to configure the MACsec Threshold Crossing Alerts (TCA) at MACsec ether layer, MACsec-secy-if, MACsec-secy-Tx, and /or MACsec-secy-Rx:

```
controllers macSecCtrlr R/S/P {pm |30 sec | 15-min | 24-hour} {macsec-ether | macsec-secy-if | macsec-secy-tx | macsec-secy-rx} {report | threshold} {in-oct-decrypted | out-oct-decrypted} value enable
```

Examples

The following is a sample to configure the MACsec TCA parameters for rx-pkt at macsec-ether level for MACsec controller in 15 min intervals:

```
controllers macSecCtrlr 0/0/0/11/1
pm 15-min macsec-ether report rx-pkt enable
pm 15-min macsec-ether threshold rx-pkt 1000000
```
The following is a sample to configure the MACsec TCA parameters for rx-util at macsec-ether level for MACsec controller in 15 min intervals:

```
controllers macSecCtrlr 0/0/0/11/1
pm 15-min macsec-ether report rx-util enable
pm 15-min macsec-ether threshold rx-util 10
```

The following is a sample to configure the MACsec TCA parameters for out-octets at macsec-ether level for MACsec controller in 15 min intervals:

```
controllers macSecCtrlr 0/0/0/11/1
pm 15-min macsec-ether report out-octets enable
pm 15-min macsec-ether threshold out-octets 100000
```

The following is a sample to configure the MACsec TCA parameters for rx-pkt at MAC-SECy-If controller in 30 sec interval:

```
controller MACSecCtrlr0/0/0/4
pm 15-min macsec-ether report rx-pkt enable
pm 15-min macsec-ether threshold rx-pkt 1000
```

### Clear Commands

You can use the following commands to clear the PMs across different buckets:

```
clear controller macSecCtrlr 0/0/0/10 pm 15-min
clear controller macSecCtrlr 0/0/0/10 pm 30-sec
clear controller macSecCtrlr 0/0/0/10 pm 24-hour
```

### View MACsec PM Parameters

You must configure MACSec controllers to view MACsec performance. To configure MACSec controllers, see Apply MACsec Configuration on MACsec Controller, on page 51.

Use the following commands to view the MACsec performance monitoring at MACsec ether layer, MACsec-secy-if, MACsec-secy-Tx, and /or MACsec-secy-Rx:

```
show controllers macSecCtrlr R/S/P { pm { current | history } { 30 sec | 15-min | 24-hour } { macsec-ether | macsec-secy-if | macsec-secy-tx | macsec-secy-rx}
```

**RP/0/RP0/CPU0:ios#show controllers macSecCtrlr 0/0/0/9/1 pm current 30-sec macsec-ether**

Displays the current performance monitoring parameters of the Ethernet controller in 30 second interval.

```
ETHER in the current interval [23:10:30 - 23:10:31 Sat Mar 18 2017]
ETHER current bucket type : Valid
RX-UTIL[%] : 92.75 Threshold : 0.00 TCA(enable) : NO
TX-UTIL[%] : 82.26 Threshold : 0.00 TCA(enable) : NO
RX-PKT : 1077504 Threshold : 0 TCA(enable) : NO
STAT-PKT : 0 Threshold : 0 TCA(enable) : NO
OCTET-STAT : 1137844152 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
```
### Configuring MACsec Encryption

| **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** |
| **OUT-UCAST** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **OUT-MCAST** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX-PKT** | **955636** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **OUT-OCTET** | **1009150716** | **Threshold**: **2000** | **TCA(enable)**: **NO** |
| **IFIN-ERRORS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IFIN-OCRTS** | **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-UNDERIZED-PKT** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_GOOD_BYTES** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_GOOD_PKTS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_DROP_OTHER** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_ERROR_FRAGMENTS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_64_OCTET** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_65_127_OCTETS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_128_255_OCTETS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_256_511_OCTETS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_512_1023_OCTETS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **IN_PKT_1024_1518_OCTETS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX_UNDERIZED_PKT** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX_OVERSIZED_PKT** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX_FRAGMENTS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX_JABBER** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |
| **TX_BAD_FCS** | **0** | **Threshold**: **0** | **TCA(enable)**: **NO** |

**Example**

```
ios#show controllers macSecCtrlr 0/0/0/9/1 pm current 15-min macsec-ether
```

Displays the current performance monitoring parameters of the Ethernet controller in 15 minute intervals

```
RP0/RP0/CPU0:ios#show controllers macSecCtrlr 0/0/0/9/1 pm current 15-min macsec-ether
Sat Mar 18 23:10:41.410 IST
ETHER in the current interval [23:00:00 - 23:10:41 Sat Mar 18 2017]
ETHER current bucket type : Valid
RX-UTIL(%) : 92.75 Threshold : 0.00 TCA(enable) : NO
TX-UTIL(%) : 82.26 Threshold : 0.00 TCA(enable) : NO
RX-PKT : 690733237 Threshold : 0 TCA(enable) : NO
```

**Configuration Guide for Cisco NCS 1002, IOS XR Release 6.3.x**
Example

View MACsec PM Parameters

- 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: 612577128, Threshold: 0, TCA(enable): NO
- OUT-OCTET: 64688147168, Threshold: 8000, 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: 0, Threshold: 0, TCA(enable): NO
- IN_GOOD_PKTS: 0, Threshold: 0, TCA(enable): NO
- IN_DROP_OTHER: 0, Threshold: 0, TCA(enable): NO
- IN_ERROR_FRAGMENTS: 0, 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: 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_FRAGMENTS: 0, Threshold: 0, TCA(enable): NO
- TX_JABBER: 0, Threshold: 0, TCA(enable): NO
- TX_BAD_FCS: 0, Threshold: 0, TCA(enable): NO
ios#show controllers macSecCtrlr 0/0/0/9/1 pm current 24-hour macsec-ether

Displays the current performance monitoring parameters of the Ethernet controller in 24 hour intervals.

ios#show controllers macSecCtrlr 0/0/0/9/1 pm current 24-hour macsec-ether
Sat Mar 18 23:10:49.939 IST
ETHER in the current interval [00:00:00 - 23:10:50 Sat Mar 18 2017]
ETHER current bucket type : Invalid

| RX-UTIL[%] | 92.75 | Threshold | 0.00 | TCA(enable) | NO |
| TX-UTIL[%] | 82.26 | Threshold | 0.00 | TCA(enable) | NO |
| RX-PKT     | 1508621424 | Threshold | 0 | TCA(enable) | NO |
| STAT-PKT   | 0 | Threshold | 0 | TCA(enable) | NO |
| OCTET-STAT | 1593104223188 | 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     | 1337921648 | Threshold | 0 | TCA(enable) | NO |
| OUT-OCTET  | 1412845260584 | Threshold | 100000 | 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 | 0 | Threshold | 0 | TCA(enable) | NO |
| IN_GOOD_PKTS | 0 | Threshold | 0 | TCA(enable) | NO |
Example

ios#show controllers macSecCtrlr 0/0/0/16/2 pm current 30-sec macsec-secy-if

Displays the current performance monitoring parameters of the controller in macsec-secy-if mode in 30 sec intervals.

Example

ios#show controllers macSecCtrlr 0/0/0/16/2 pm current 15-min macsec-secy-if

Displays the current performance monitoring parameters of the controller in macsec-secy-if mode in 15 minute intervals.
**Example**

```bash
ios#show controllers macSecCtrlr 0/0/0/16/2 pm current 30-sec macsec-secy-tx
```

Displays the current performance monitoring parameters of the controller in macsec-secy-tx mode in 30 minute intervals.

```bash
Macsec-Secy-Tx in the current interval [10:18:30 - 10:18:59 Sat Apr 22 2017]
Macsec-Secy-Tx current bucket type : Valid
OutPktsProtected : 0 Threshold : 0 TCA(enable) : NO
OutPktsEncrypted : 286527 Threshold : 0 TCA(enable) : NO
OutOctetsProtected : 0 Threshold : 0 TCA(enable) : NO
OutOctetsEncrypted : 430363554 Threshold : 0 TCA(enable) : NO
```

Displays the current performance monitoring parameters of the controller in macsec-secy-tx mode in 24-hour interval.

```bash
RP/0/RP0/CPU0:ios#show controllers macSecCtrlr 0/0/0/24/3 pm current 24-hour macsec-secy-tx Sat Apr 1 15:38:30.158 IST
Macsec-Secy-Tx in the current interval [00:00:00 - 15:38:30 Sat Apr 1 2017]
Macsec-Secy-Tx current bucket type : Valid
OutPktsProtected : 0 Threshold : 0 TCA(enable) : NO
OutPktsEncrypted : 3160983513 Threshold : 0 TCA(enable) : NO
OutOctetsProtected : 0 Threshold : 0 TCA(enable) : NO
OutOctetsEncrypted : 31559259393792 Threshold : 0 TCA(enable) : NO
```

Displays the current performance monitoring parameters of the controller in macsec-secy-rx mode in 24-hour interval.

```bash
RP/0/RP0/CPU0:ios#show controllers macSecCtrlr 0/0/0/10/3 pm current 24-hour macsec-secy-rx Sat Apr 1 15:38:00.820 IST
Macsec-Secy-Rx in the current interval [00:00:00 - 15:38:01 Sat Apr 1 2017]
Macsec-Secy-Rx current bucket type : Valid
InPktsUnchecked : 0 Threshold : 0 TCA(enable) : NO
InPktsDelayed : 0 Threshold : 0 TCA(enable) : NO
InPktsLate : 0 Threshold : 0 TCA(enable) : NO
InPktsInvalid : 0 Threshold : 0 TCA(enable) : NO
InPktsOK : 3159299558 Threshold : 0 TCA(enable) : NO
InPktsNotValid : 0 Threshold : 0 TCA(enable) : NO
InPktsNotUsingSA : 0 Threshold : 0 TCA(enable) : NO
InPktsUnusedSA : 0 Threshold : 0 TCA(enable) : NO
InOctetsUntaggedHit : 0 Threshold : 0 TCA(enable) : NO
InOctetsValidated : 0 Threshold : 0 TCA(enable) : NO
InOctetsDecrypted : 31542446787072 Threshold : 0 TCA(enable) : NO
```

**MACsec MKA Using EAP-TLS Authentication**

Using IEEE 802.1X port-based authentication with Extensible Authentication Protocol (EAP-TLS), MACsec MKA can be configured between two NCS 1002 device ports. EAP-TLS allows mutual authentication and obtains MSK (master session key). Both Connectivity Association Key Name (CKN) and connectivity
association key (CAK) are derived from MSK for MKA operations. The device certificates are carried for authentication to the external AAA server using EAP-TLS.

IEEE 802.1X Device Roles

The devices in the network have the following specific roles with IEEE 802.1X authentication.

- Supplicant - An entity at one end of a point-to-point LAN segment that seeks to be authenticated by an Authenticator attached to the other end of that link.
- Authenticator - An entity that facilitates authentication of other entities attached to the same LAN.
- Authentication Server - An entity that provides an authentication service to an Authenticator. This service determines, from the credentials provided by the Supplicant, whether the Supplicant is authorized to access the services provided by the system in which the Authenticator resides.

Prerequisites for MACsec MKA Using EAP-TLS Authentication

- Ensure that a Certificate Authority (CA) server is configured for the network.
- Ensure a valid CA certificate.
- Ensure that the user has configured Cisco Identity Services Engine (ISE) Release 2.2 onwards or Cisco Secure Access Control Server Release 5.6 onwards as external AAA server.
- It is always good to have the NCS 1002 devices, the CA server, and the external AAA/Radius server synchronized using Network Time Protocol (NTP) Server. If clock is not synchronized, there might be instances where certificate validation will not be successful, due to timing issues.

However, there is no dependency on the timezone between NCS 1002 devices, the CA Server and the external AAA/Radius server.

Configure MACsec Encryption Using EAP-TLS Authentication

Configuring MACsec encryption using EAP-TLS authentication involves the following tasks:

- Configure RADIUS Server, on page 69
- Configure 802.1X Authentication Method, on page 70
- Generate RSA Key Pair, on page 70
- Configure Trust Point, on page 71
- Authenticate Certificate Authority and Request Certificates, on page 72
- Configure EAP Profile, on page 73
- Configure 802.1X Profile, on page 74
- Configure EAP and 802.1X Profile on MACsec Controller, on page 75
Configure RADIUS Server

configure
radius-server host {IPv4 address of RADIUS server} [auth-port port-number] [acct-port port-number] [key string]
radius-server vsa attribute ignore unknown
exit
commit

Examples
The following is a sample of configuring the RADIUS server.

configure
radius-server host 209.165.200.225 auth-port 1645 acct-port 1646 key cisco
radius-server vsa attribute ignore unknown
exit
commit

The following is sample output of show radius command.

Tue Jun 27 10:39:20.851 IST
Global dead time: 0 minute(s)
Number of Servers: 1
Server: 209.165.200.225 is UP
Address family: IPv4
Total Deadtime: 0s Last Deadtime: 0s
Timeout: 5 sec, Retransmit limit: 3
Quarantined: No
Authentication:
42 requests, 0 pending, 0 retransmits
6 accepts, 0 rejects, 0 challenges
0 timeouts, 0 bad responses, 0 bad authenticators
0 unknown types, 0 dropped, 361 ms latest rtt
Throttled: 0 transactions, 0 timeout, 0 failures
Estimated Throttled Access Transactions: 0
Maximum Throttled Access Transactions: 0
Automated TEST Stats:
0 requests, 0 timeouts, 0 response, 0 pending

Accounting:
0 requests, 0 pending, 0 retransmits
0 responses, 0 timeouts, 0 bad responses
0 bad authenticators, 0 unknown types, 0 dropped
0 ms latest rtt
Throttled: 0 transactions, 0 timeout, 0 failures
Estimated Throttled Accounting Transactions: 0
Maximum Throttled Accounting Transactions: 0
Automated TEST Stats:
0 requests, 0 timeouts, 0 response, 0 pending
Configure 802.1X Authentication Method

This procedure allows the user to configure 802.1X authentication method using RADIUS as the protocol. 802.1X authentication configuration allows to configure non-default profiles. However, only default is supported in NCS 1002.

```
configure
aaa authentication dot1x default group radius
exit
commit
```

Examples

The following is a sample of configuring the 802.1X authentication method.

```
configure
aaa authentication dot1x default group radius
exit
commit
```

The following is sample output of `show run aaa` command.

```
Tue Jun 27 10:39:17.437 IST
radius-server vsa attribute ignore unknown
radius-server host 209.165.200.225 auth-port 1645 acct-port 1646 key cisco
aaa authentication dot1x default group radius
```

Generate RSA Key Pair

RSA key pairs are used to sign and encrypt key management messages. This is required before the you can obtain a certificate for the node.

```
configure
crypto key generate rsa [usage-keys | general-keys] [keypair-label]
exit
commit
```
Examples

The following is a sample of generating the RSA key pair.

configure
crypto key generate rsa ncs1k
exit
commit

The following is sample output of `show crypto key key-name rsa` command.

Tue Jun 27 10:39:44.152 IST
Key label: ncs1k
Type : RSA General purpose
Size : 2048
Created : 10:02:32 IST Tue Jun 27 2017
Data :
30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
00B59C0D 005B52ED A0FCB710 6F5C2A0F DF83504F 801B43C4 987D3E6B
1F8981E 3520C20E D7934082 D9BF04D7 07E5824F EA5EA1BB DDDFF6CD 9FCBDF75
F3EFP39DA C0B67C65 40AC89D4 58FF3FD0 1ED2AC8C 770C2339 E8508B48 E648A15D
6F8DE9FA 058E87B2 3094E2F8 8F6280C3 469FF22F 386483FC 5EDE8178 5F7537C6
785C487E 766BC636 2BEC55E4 3A6264CE A113BE64 A20F47F0 E1AA603E D5DB078F
A0B9F3E6 4114C435 2283D93F 40B4FEEE 63C35968 DB3992BC F6DF66D9
2CAD24BD FA85C636 247DE466 E2622D79 B2579D3 FADD9DE2 70474236 2FD585F5A
67D6CC24 38DE7C8F 33923479 E822E92D C9B141FF E576C59C 50BB5CC5 P693A7D4
81020301 0001

Associated Commands

- `crypto key generate rsa`

Configure Trust Point

configure
crypto ca trustpoint {ca-name}
enrollment url {ca-url}
subject-name {x.500-name}
rsakeypair {keypair-label}
crl optional
exit
commit

Examples

The following is a sample of configuring the trust point.

configure
crypto ca trustpoint ncs1k
enrollment url http://209.165.200.226
subject-name CN=ncs1k,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
rsakeypair ncs1k
crl optional
exit
commit

The following is sample output of `show run crypto ca trustpoint ca-name` command.

Tue Jun 27 10:39:40.375 IST
crypto ca trustpoint ncs1k
crl optional
subject-name CN=ncs1k,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
enrollment url http://209.165.200.226
rsakeypair ncs1k
!

Associated Commands

- `crypto ca trustpoint`
- `enrollment url`
- `rsakeypair`

## Authenticate Certificate Authority and Request Certificates

This procedure authenticates the certificate authority (CA) with NCS 1002 and requests certificates from the CA. NCS 1002 must authenticate the CA by obtaining the self-signed certificate of the CA. The self-signed certificate contains the public key of the CA. It is required to manually authenticate the public key of the CA by contacting the CA administrator to compare the fingerprint of the CA certificate.

```
configure
crypto ca authenticate {ca-name}
crypto ca enroll {ca-name}
exit
commit
```

**Examples**

The following is a sample of authenticating the certificate authority and requesting certificates.

```
configure
crypto ca authenticate ncs1k
crypto ca enroll ncs1k
exit
commit
```

The following is sample output of `show crypto ca certificates` command.

```
Tue Jun 27 10:39:47.356 IST
Trustpoint : ncs1k
---------------------------------------------------------------
CA certificate
  Serial Number : 01
```
Subject: CN=ncs1k,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
Issued By: CN=ncs1k,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
Validity Start: 03:50:38 UTC Wed Mar 22 2017
Validity End: 03:50:38 UTC Sat Mar 21 2020
SHA1 Fingerprint: 0B2E1F69BB42CE068AAB67F1FC2C99F3F66

Router certificate
Key usage: General Purpose
Status: Available
Serial Number: 01:1F
Subject: serialNumber=cf302761,unstructuredAddress=209.165.200.226,unstructuredName=ncs1k,
C=US,ST=NY,L=Newyork,O=Govt,OU=BU,CN=ncs1k
Issued By: CN=ncs1k,OU=BU,O=Govt,L=Newyork,ST=NY,C=US
Validity Start: 04:37:11 UTC Tue Jun 27 2017
Validity End: 04:37:11 UTC Wed Jun 27 2018
SHA1 Fingerprint: AF640E4E9E0521217BF3F4770465FD832B2AE90B
Associated Trustpoint: ncs1k

Associated Commands
- crypto ca authenticate
- crypto ca enroll

Configure EAP Profile

You can configure multiple EAP profiles.

configure
eap profile {name}
identity {user-name}
method tls pki-trustpoint {trustpoint-name}
exit
commit

Examples

The following is a sample of configuring the EAP profile.

configure
eap profile ncs1k
identity PRO67
method tls pki-trustpoint ncs1k
exit
commit

The following is sample output of show run eap command.
Configure 802.1X Profile

You can configure multiple 802.1X profiles. The role of the node running 802.1X profile can be supplicant, authenticator, or both.

```
configure
dot1x profile {name}
pae {authenticator | supplicant | both}
authentication timer reauthenticate {seconds | server}
supplicant eap profile {profile-name}
exit
commit
```

Examples

The following is a sample of configuring the 802.1X profile.

```
configure
dot1x profile reauth
pae both
authentication timer reauthenticate 3600
supplicant eap profile ncs1k
exit
commit
```

The following is sample output of `show run dot1x` command.

```
Tue Jun 27 10:39:55.178 IST
dot1x profile both_local_reauth
  pae both
  authenticator
    timer reauth-time 3600
  !
supplicant
  eap profile ncs1k
  !
```
Associated Commands

- dot1x profile
- authentication timer reauthenticate
- dot1x supplicant eap profile

Configure EAP and 802.1X Profile on MACsec Controller

You can attach one of the 802.1X profiles on the MACsec controller.

configure
controller MACSecCtrl Rack/Slot/Instance/Port
dot1x profile profile-name
macsec eap [policy macsec-policy-name]
exit
commit

Example

The following is a sample of configuring MACsec EAP and 802.1X profile on the MACsec controller.

configure
controller MACSecCtrl 0/0/0/24
dot1x profile reauth
macsec eap
exit
commit

Verify EAP and 802.1X Configuration on MACsec Controller

show run controller MACSecCtrl Rack/Slot/Instance/Port

Tue Jun 27 10:39:59.148 IST
ccontroller MACSecCtrl0/0/0/24
dot1x profile both_local_reauth
macsec eap
!

show dot1x controller MACSecCtrl Rack/Slot/Instance/Port detail

Tue Jun 27 10:40:02.648 IST
Dot1x info for MACSecCtrlr0/0/0/24

<table>
<thead>
<tr>
<th>Interface short name</th>
<th>MS0/0/0/24</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface handle</td>
<td>0x8000544</td>
</tr>
<tr>
<td>Interface MAC</td>
<td>2c00.4314.5c6c</td>
</tr>
<tr>
<td>Ethertype</td>
<td>888E</td>
</tr>
<tr>
<td>PAE</td>
<td>Both</td>
</tr>
<tr>
<td>Dot1x Port Status</td>
<td>AUTHORIZED</td>
</tr>
<tr>
<td>Dot1x Profile</td>
<td>both_local_reauth</td>
</tr>
</tbody>
</table>

**Supplicant:**
- **Config Dependency**: Resolved
- **Eap profile**: ncs1k
- **Client List**:
  - **Authenticator**: 2c02.dc14.636c
  - **EAP Method**: EAP-TLS
  - **Supp SM State**: Authenticated
  - **Supp Bend SM State**: Idle
  - **Last authen time**: 2017 Jun 27 209.165.200.227

**Authenticator:**
- **Config Dependency**: Resolved
- **ReAuth**: Enabled, 0 day(s), 01:00:00
- **Client List**:
  - **Supplicant**: 2c02.dc14.636c
  - **Auth SM State**: Authenticated
  - **Auth Bend SM State**: Idle
  - **Last authen time**: 2017 Jun 27 209.165.200.225
  - **Time to next reauth**: 0 day(s), 00:46:09

**MKA Interface**:
- **Dot1x Tie Break Role**: Auth
- **EAP Based Macsec**: Enabled
- **MKA Start time**: 2017 Jun 27 209.165.200.227
- **MKA Stop time**: NA
- **MKA Response time**: 2017 Jun 27 209.165.200.226

```
show macsec mka session controller MACSecCtrlr Rack/Slot/Instance/Port
```

Tue Jun 27 10:40:07.492 IST

**MKA Detailed Status for MKA Session**

<table>
<thead>
<tr>
<th>Status: SECURED - Secured MKA Session with MACsec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Tx-SCI: 2c00.4314.5c6c/0018</td>
</tr>
<tr>
<td>Local Tx-SSCI: 2</td>
</tr>
<tr>
<td>Interface MAC Address: 2c00.4314.5c6c</td>
</tr>
<tr>
<td>MKA Port Identifier: 24</td>
</tr>
<tr>
<td>Interface Name: MS0/0/0/24</td>
</tr>
<tr>
<td>CAK Name (CKN): A3B6509EC0EBCAE8D610139669952E0A</td>
</tr>
<tr>
<td>CA Authentication Mode: EAP</td>
</tr>
<tr>
<td>Keychain: NA (EAP mode)</td>
</tr>
<tr>
<td>Member Identifier (MI): C12DD7CF165438D8B1732211</td>
</tr>
<tr>
<td>Message Number (MN): 422</td>
</tr>
<tr>
<td>Authenticator: YES</td>
</tr>
<tr>
<td>Key Server: YES</td>
</tr>
<tr>
<td>MKA Cipher Suite: AES-128-CMAC</td>
</tr>
<tr>
<td>Latest SAK Status: Rx &amp; Tx</td>
</tr>
<tr>
<td>Latest SAK AN: 0</td>
</tr>
<tr>
<td>Latest SAK KI (KN): C12DD7CF165438D8B173221100000001 (1)</td>
</tr>
<tr>
<td>Old SAK Status: FIRST-SAK</td>
</tr>
<tr>
<td>Old SAK AN: 0</td>
</tr>
<tr>
<td>Old SAK KI (KN): FIRST-SAK (0)</td>
</tr>
</tbody>
</table>

**Configuration Guide for Cisco NCS 1002, IOS XR Release 6.3.x**
SAK Transmit Wait Time : 0s (Not waiting for any peers to respond)
SAK Retire Time : 0s (No Old SAK to retire)
Time to SAK Rekey : NA

MKA Policy Name : *DEFAULT POLICY*
Key Server Priority : 16
Delay Protection : FALSE
Replay Window Size : 64
Include ICV Indicator : FALSE
Confidentiality Offset : 0
Algorithm Agility : 80C201
SAK Cipher Suite : 0080C20001000004 (GCM-AES-XPN-256)
MACsec Capability : 3 (MACsec Integrity, Confidentiality, & Offset)
MACsec Desired : YES

# of MACsec Capable Live Peers : 1
# of MACsec Capable Live Peers Responded : 1

Live Peer List:
<table>
<thead>
<tr>
<th>MI</th>
<th>MN</th>
<th>Rx-SCI (Peer)</th>
<th>SSCI KS-Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>578656B568B072819160DCD4</td>
<td>420</td>
<td>2c02.dc14.636c/0018</td>
<td>1 16</td>
</tr>
</tbody>
</table>

Potential Peer List:
<table>
<thead>
<tr>
<th>MI</th>
<th>MN</th>
<th>Rx-SCI (Peer)</th>
<th>SSCI KS-Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>578656B568B072819160DCD4</td>
<td>420</td>
<td>2c02.dc14.636c/0018</td>
<td>1 16</td>
</tr>
</tbody>
</table>

show macsec mka session

Tue Jun 27 10:40:59.320 IST
NODE: node0_RP0_CPU0

<table>
<thead>
<tr>
<th>Interface</th>
<th>Local-TxSCI</th>
<th># Peers</th>
<th>Status</th>
<th>Key-Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS0/0/0/24</td>
<td>2c00.4314.5c6c/0018</td>
<td>1</td>
<td>Secured</td>
<td>YES</td>
</tr>
</tbody>
</table>
Verify EAP and 802.1X Configuration on MACsec Controller
Configuring GMPLS UNI

The primary function of Generalized Multiprotocol Label Switching (GMPLS) User Network Interface (UNI) is to create circuit connection between two clients (UNI-C) of an optical network. This is achieved by signaling exchanges between UNI Client (UNI-C) and UNI Network (UNI-N) nodes. NCS 1002 node acts as UNI-C and NCS 2000 series node acts as UNI-N in GMPLS-UNI reference model.

The user can create a GMPLS optical channel trail (OCH Trail) in a network where the NCS 1002 node is connected to a NCS 2000 series node. The OCH trail circuit originates from a NCS 1002 trunk interface (UNI-C) on the source NCS 1002 node and terminates on the NCS 2000 series interface (UNI-N) on the destination NCS 2000 series node to create an optical connection. The prerequisite for the OCH trail circuit is to create a Link Management Protocol (LMP) link between the optical channel Add/Drop NCS 2000 series interface on the NCS 2000 series node and the NCS 1002 interface on the NCS 1002 node.

GMPLS UNI is supported only on the 100G and 200G trunk ports of the NCS 1002 node until R6.2.1. GMPLS UNI is supported on the 250G trunk port of the NCS 1002 node from R6.3.2.

Channel Spacing

DWDM grid (C-band: 1530 nm to 1569 nm) in optical spectrum can be divided into multiple channels so that each channel can carry data traffic independently. These channels can be separated with available slot widths called channel spacing.

GMPLS has two types of channel spacing.

- Fixed Grid channel spacing - The channel spacing is fixed to either 100 GHz, 50 GHz, or 12.5 GHz. NCS 1002 supports 50 GHz channel spacing.
- Flexible Grid channel spacing - The channels are divided in finer slices. The flexibility in channel spacing allows to transmit mixed bit-rate or mixed modulation data in a grid with different channel widths. It also helps in effective usage of DWDM grid with minimal wastage of spectral bandwidth. NCS 1002 supports 6.25 GHz channel spacing.

GMPLS UNI is supported only for fixed grid until R6.2.1. Hence, 250G channels from NCS 1002 node cannot pass through the NCS 2000 series node because of spectral issues of 50 GHz spacing. GMPLS UNI flexible grid is supported from R6.3.2 that suports 250G channels.

Prerequisites

- NCS 1002 node must have both the MPLS and MPLS-TE packages.
- NCS 2000 series node must have a valid license for ROADM and WSON support.
• The management IPs of NCS 1002 and NCS 2000 series nodes on both the source and destination must be reachable.

**Limitations of GMPLS Flexible Grid**

Flexible grid interoperability between R6.3.2 of NCS 1002 and R10.8 of NCS 2000 works for LMP discovery, flexible grid based wavelength assignment, and E2E tunnels.

However, there are interoperability issues with NCS 2000 at the following UNI revert and restoration scenarios.

• UNI-revert does not work for tunnels with XRO configurations.
• Restoration does not work due to wrong carrier allocation.

**Configure GMPLS UNI**

Configuring GMPLS UNI involves the following tasks:

The following configurations must be performed on the NCS 2000 series node.

1. Configure LMP and Alien Wavelength in NCS 2000 Series Node, on page 82
2. Retrieve Ifindex from NCS 2000 Series Node, on page 83

The following configurations must be performed on the NCS 1002 node.

1. Configure LMP in Cisco NCS 1002, on page 84
2. Configure RSVP in NCS 1002, on page 84
3. Configure MPLS Tunnel in NCS 1002, on page 85

**GMPLS UNI Command Reference**

For detailed command information about GMPLS UNI commands, see Cisco IOS XR MPLS Command Reference.

**Debuggability**

For any software issues, it is recommended to collect the output of show tech of (mpls, mpls-te, rsvp, cf-mgr, sysdb, ncs1k) for head node and tail node.

• Configuring GMPLS UNI, on page 81
• Configure LMP and Alien Wavelength in NCS 2000 Series Node, on page 82
• Configure Unnumbered LMP in NCS 2000 Series Node, on page 83
• Retrieve Ifindex from NCS 2000 Series Node, on page 83
• Configure LMP in Cisco NCS 1002, on page 84
• Configure RSVP in NCS 1002, on page 84
• Configure MPLS Tunnel in NCS 1002, on page 85
• Headless Mode and GMPLS UNI, on page 85
• Display GMPLS UNI Tunnel, RSVP, and LMP Information, on page 85
• Example of MPLS Tunnel Creation Without ERO, on page 90
• Example of MPLS Tunnel Creation with ERO, on page 91
• Example of MPLS Tunnel Creation with XRO, on page 92
Configuring GMPLS UNI

The primary function of Generalized Multiprotocol Label Switching (GMPLS) User Network Interface (UNI) is to create a circuit connection between two clients (UNI-C) of an optical network. This is achieved by signaling exchanges between UNI Client (UNI-C) and UNI Network (UNI-N) nodes. NCS 1002 node acts as UNI-C and NCS 2000 series node acts as UNI-N in GMPLS-UNI reference model.

The user can create a GMPLS optical channel trail (OCH Trail) in a network where the NCS 1002 node is connected to a NCS 2000 series node. The OCH trail circuit originates from a NCS 1002 trunk interface (UNI-C) on the source NCS 1002 node and terminates on the NCS 2000 series interface (UNI-N) on the destination NCS 2000 series node to create an optical connection. The prerequisite for the OCH trail circuit is to create a Link Management Protocol (LMP) link between the optical channel Add/Drop NCS 2000 series interface on the NCS 2000 series node and the NCS 1002 interface on the NCS 1002 node.

GMPLS UNI is supported only on the 100G and 200G trunk ports of the NCS 1002 node until R6.2.1. GMPLS UNI is supported on the 250G trunk port of the NCS 1002 node from R6.3.2.

Channel Spacing

DWDM grid (C-band: 1530 nm to 1569 nm) in optical spectrum can be divided into multiple channels so that each channel can carry data traffic independently. These channels can be separated with available slot widths called channel spacing.

GMPLS has two types of channel spacing.

- **Fixed Grid channel spacing** - The channel spacing is fixed to either 100 GHz, 50 GHz, or 12.5 GHz. NCS 1002 supports 50 GHz channel spacing.

- **Flexible Grid channel spacing** - The channels are divided in finer slices. The flexibility in channel spacing allows to transmit mixed bit-rate or mixed modulation data in a grid with different channel widths. It also helps in effective usage of DWDM grid with minimal wastage of spectral bandwidth. NCS 1002 supports 6.25 GHz channel spacing.

GMPLS UNI is supported only for fixed grid until R6.2.1. Hence, 250G channels from NCS 1002 node cannot pass through the NCS 2000 series node because of spectral issues of 50 GHz spacing. GMPLS UNI flexible grid is supported from R6.3.2 that supports 250G channels.

Prerequisites

- NCS 1002 node must have both the MPLS and MPLS-TE packages.

- NCS 2000 series node must have a valid license for ROADM and WSON support.

- The management IPs of NCS 1002 and NCS 2000 series nodes on both the source and destination must be reachable.

Limitations of GMPLS Flexible Grid

Flexible grid interoperability between R6.3.2 of NCS 1002 and R10.8 of NCS 2000 works for LMP discovery, flexible grid based wavelength assignment, and E2E tunnels.
However, there are interoperability issues with NCS 2000 at the following UNI revert and restoration scenarios.

- UNI-revert does not work for tunnels with XRO configurations.
- Restoration does not work due to wrong carrier allocation.

**Configure GMPLS UNI**

Configuring GMPLS UNI involves the following tasks:

The following configurations must be performed on the NCS 2000 series node.

1. Configure LMP and Alien Wavelength in NCS 2000 Series Node, on page 82
2. Retrieve Ifindex from NCS 2000 Series Node, on page 83

The following configurations must be performed on the NCS 1002 node.

1. Configure LMP in Cisco NCS 1002, on page 84
2. Configure RSVP in NCS 1002, on page 84
3. Configure MPLS Tunnel in NCS 1002, on page 85

**GMPLS UNI Command Reference**

For detailed command information about GMPLS UNI commands, see Cisco IOS XR MPLS Command Reference.

**Debuggability**

For any software issues, it is recommended to collect the output of show tech of (mpls, mpls-te, rsvp, cf-mgr, sysdb, ncs1k) for head node and tail node.

---

**Configure LMP and Alien Wavelength in NCS 2000 Series Node**

This procedure creates a static LMP link between a NCS 2000 series node and NCS 1002 node. Unnumbered LMP can be configured only through TL1. Numbered LMP can be configured through both CTC and TL1.

In unnumbered LMP, the tunnel is terminated in any of the links reaching the peer NCS 1002 node. The numbered LMP is recommended because NCS 1002 trunk connections must be symmetric.

The alien wavelength must be configured for the NCS 2000 series interface (UNI-N) before creating the MPLS tunnel from NCS 1002 node (UNI-C). As CTC does not manage the NCS 1002 node, the alien wavelength must be separately configured in the Add Drop ports of NCS 2000 series node.

See DLP-G800 Create an LMP Link Using CTC to configure LMP and alien wavelength in NCS 2000 series node.
Configure Unnumbered LMP in NCS 2000 Series Node

In unnumbered LMP, the tunnel is terminated in any of the links reaching the peer NCS 1002 node. The number of LMP must be symmetric.

Use the following command to configure the unnumbered LMP from the TL1 interface.

```
ENT-UNICFG::[<TID>]:<src>:<CTAG>::<rvrsaid>,<rsysip>,<rifcip>,<mstpip>,<commip>,[<remoteifinfo>],[VALMODE=<optval>],[VALZONE=<oprzone>],[ADMINSTATE=<adminstate>],[RESTTYPE=<resttype>],[CKTLABEL=<cktlabel>],[USPWROFS=<upstrmpwr>],[DSPWROFS=<dnstrmpwr>],[ALLOWREGEN=<allowregen>],[UNICTRLMODE=<unictrlmode>],[REVERTMODE=<revertmode>],[SOAK=<HH-MM-SS>],[RESTVALMODE=<restvalmode>],[RESTVALZONE=<restvalzone>],[TERMINTFDX=<termintfdx>],[DIVERSITY=<diversity>],[DIVERSITYTYPE=<diversitytype>],[ISLOOSE=<isloose>],[UNIMODE=<unimode>],[ALIENID=<alienid>],[FECMODE=<fecmode>],[TRUNKMODE=<trunkmode>],[OPTENPOINT=<optendpoint>],[PAIRDEPENPOINT=<pairedoptendpoint>],[REMOTEIFINFOLIST=<remoteifinfolist>],[LMPTYPE=<lmptype>];
```

*rvrsaid* is the destination AID. *rsysip* is the remote system IP address. *rifcip* is the remote interface IP address and enter 0.0.0.0 for unnumbered LMP. *mstpip* is the MSTP interface IP address and enter 0.0.0.0 for unnumbered LMP. *commip* is the destination IP address.

**Example**

```
ENT-UNICFG::PLINE-20-5-RX:111::PLINE-20-5-TX,10.58.229.22,22.22.22.5,90.90.90.5,10.58.229.22,"Unnumb":VALMODE=FULL,VALZONE=RED,DESCR_90_Ports22_250_205,ADMINSTATE=UP,RESTTYPE=NONE,UNICTRLMODE=CLIENT,RESTVALMODE=NONE,UNIMODE=GMPLS,ALIENID=NCS1K,FECMODE=SD-20,TRUNKMODE=250G,LMPTYPE=SIGNALED;
```

Retrieve Ifindex from NCS 2000 Series Node

The Ifindex of all the LMP ports of NCS 2000 series node can be retrieved using CTC or TL1.

**Using CTC**

The Ifindex of all the LMP ports of NCS 2000 series node in decimal format can be retrieved using CTC from the **Originating Interface Index** column under the **Provisioning > LMP tabs**.

**Using TL1**

1. Log in to the TL1 interface and issue the following command.
2. `rtrv-unicfg ::all:1;`

This command retrieves the Ifindex of all the LMP ports of NCS 2000 series node in hexadecimal. This must be converted to decimal and used in remote Ifindex of NCS 1002 node during the LMP configuration.

**TL1 Output**

```
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,RESTVALMODE=MANUAL,SOAK=00-01-00,RESTVALMODE=NONE,TERMINTFDX=0,ORIGINTFIDX=0,NUMBERED=TRUE,UNIMODE=GMPLS
```
Configure LMP in Cisco NCS 1002

Link Management Protocol (LMP) is a logical link that must be created on the trunk optics controller on the source and destination NCS 1002 nodes of the tunnel. Only static LMP is supported.

As CTC does not manage the NCS 1002 node, the Ifindex of Add Drop ports of NCS 2000 series node must be manually retrieved through TL1 or CTC LMP panel and used in LMP configuration in NCS 1002. See Retrieve Ifindex from NCS 2000 Series Node, on page 83 to retrieve the Ifindex. This Ifindex must be converted to decimal and used in the `neighbor interface-id unnumbered` command node during the LMP configuration.

Numbered trail creation requires the management IP address, link IP address (IP address of the optics controller), and the interface index. Unnumbered trail creation requires the management IP address and the interface index.

`link-id ipv4 unicast` IP address is the IP address of the optics controller. `neighbor link-id ipv4 unicast` IP address is the IP address of the MSTP interface. `neighbor flexi-grid-capable` enables GMPLS UNI flexible grid channel spacing.

The following is a sample of configuring the LMP on the source NCS 1002 node.

```
show running-config lmp

lmp
  gmpls optical-uni
    controller Optics0/0/0/13
    neighbor NCS1K
    neighbor link-id ipv4 unicast 192.0.2.3
    neighbor flexi-grid-capable
    neighbor interface-id unnumbered 19
    link-id ipv4 unicast 192.0.2.4
    !
    neighbor NCS1K
    ipcc routed
    router-id ipv4 unicast 192.0.2.1
    !
    router-id ipv4 unicast 192.0.2.2
    !
```

Configure RSVP in NCS 1002

Resource Reservation Protocol (RSVP) configuration with appropriate timeout for optical network must be performed on the source and destination NCS 1002 nodes of the tunnel. The following is a sample of configuring RSVP on the source NCS 1002 node.

```
controller optics 0/0/0/6
```
Configure MPLS Tunnel in NCS 1002

MPLS tunnels can be configured only from the 100G and 200G trunk ports of the NCS 1002 node. The trunk optics controller must be in no shut state.

The following is a sample of configuring the MPLS tunnel on the source NCS 1002 node.

```plaintext
mpls traffic-eng
  gmpls optical-uni
    controller optics 0/0/0/6
    tunnel-properties
      tunnel-id 100
      destination ipv4 unicast 100.20.20.20
      path-option 10 no-ero lockdown
```

Explicit Route Object (ERO) - Includes the route(s) to be used through a list of specified nodes for a tunnel. Exclude Route Object (XRO) - Excludes the route(s) to be used through a list of specified nodes for a tunnel.

The following is a sample to enable the controller to participate in the MPLS tunnel on the destination NCS 1002 node.

```plaintext
mpls traffic-eng
  gmpls optical-uni
    controller optics 0/0/0/6
```

Headless Mode and GMPLS UNI

NCS 1002 can carry traffic with a non-functional CPU (headless mode) for up to 72 hours. The existing GMPLS UNI tunnels are not affected by headless events such as system reload and CPU OIR operation on NCS 1002 node. However, the existing GMPLS UNI tunnels are affected if changes to the tunnel are triggered by the peer device when NCS 1002 node operates in headless mode.

Display GMPLS UNI Tunnel, RSVP, and LMP Information

```plaintext
show mpls traffic-eng link-management optical-uni controller optics 0/0/0/13
```

Mon Sep 25 10:58:02.018 UTC
Optical interface: Optics0/0/0/13
Overview:
  IM state: Up
  Child interface: : IM state Unknown
  OLM/LMP state: Up
  Optical tunnel state: up
Connection:
  Tunnel role: Head
  Tunnel-id: 32, LSP-id 144, Extended tunnel-id 10.77.142.93
  Tunnel source: 10.77.142.93, destination: 6.1.1.1
  Optical router-ids: Local: 10.77.142.93, Remote: 10.77.142.94
  Label source: UNI-N
Upstream label:
  Optical label:
    Grid : DWDM
    Channel spacing : 6.25 GHz
    Identifier : 0
    Channel Number : 88

Downstream label:
  Optical label:
    Grid : DWDM
    Channel spacing : 6.25 GHz
    Identifier : 0
    Channel Number : 88

SRLG discovery: Disabled
SRLG announcement: None

Admission Control:
Upstream: Admitted (LSP ID: 144)
Downstream: Admitted (LSP ID: 144)

OLM/LMP adjacency information:
  Adjacency status: Up
Local:
  node ID: 10.77.142.93
  link interface ID: 19
  link ID: 5.1.1.1
Neighbor:
  node ID: 10.77.142.94 (38-SIT3)
  link interface ID: 2130709792
  link ID: 5.1.1.2
  IPCC: Routed to 10.77.142.94

Optical capabilities:
  Controller type: DWDM
  Channel spacing: 6.25 GHz
  Default channel: 88
  776 supported channels:
  -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
  -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
Display GMPLS UNI Tunnel, RSVP, and LMP Information

-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
show mpls traffic-eng link-management optical-uni

Displays the summary of the GMPLS UNI tunnel state. MPLS tunnels are not created when the optics controller is in shutdown state. IM state is shown as "Admin down". Issue the no shutdown command under the controller interface to initiate the tunnel creation.

Mon Jan 11 04:57:46.220 UTC

System Information:
Optical Links Count: 1 (Maximum Links Supported 100)

Optical interface: Optics0/0/0/13
Overview:
IM state: Up
Child interface: : IM state Unknown
OLM/LMP state: Up
Optical tunnel state: up
Connection:
Tunnel role: Tail
Tunnel-id: 1, LSP-id 2, Extended tunnel-id 10.77.132.158
Tunnel source: 10.77.132.158, destination: 1.1.1.1
Optical router-ids: Local: 10.77.132.156, Remote: 10.77.132.158
Label source: UNI-N
Upstream label:
Optical label:
Grid : DWDM
Channel spacing : 50 GHz
Identifier : 0
Channel Number : -8
Downstream label:
Optical label:
Grid : DWDM
Channel spacing : 50 GHz
Identifier : 0
Channel Number : -8
SRLG discovery: Disabled
SRLG announcement: None
Admission Control:
Upstream: Admitted (LSP ID: 2)
Downstream: Admitted (LSP ID: 2)
OLM/LMP adjacency information:
Adjacency status: Up
Local:
node ID: 10.77.132.156
link interface ID: 6
link ID: 1.1.1.1
Neighbor:
node ID: 10.77.132.158 (RDT_2)
link interface ID: 19
link ID: 1.1.1.2
IPCC: Routed to 10.77.132.158
Optical capabilities:
Controller type: DWDM
Channel spacing: 50 GHz
Default channel: -7
97 supported channels:
-36, -35, -34, -33, -32, -31, -30, -29
-28, -27, -26, -25, -24, -23, -22, -21
Controller SRLGs
None

show mpls traffic-eng link-management optical-uni tabular
Displays the summary of the GMPLS UNI tunnel state in tabular format.

Mon Jan 11 05:27:06.407 UTC
System Information:
Optical Links Count: 1 (Maximum Links Supported 100)

<table>
<thead>
<tr>
<th>Interface</th>
<th>Admin</th>
<th>Oper</th>
<th>adjacency</th>
<th>role</th>
<th>tun-id</th>
<th>state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Op0/0/0/13</td>
<td>up</td>
<td>up</td>
<td>up</td>
<td>Tail</td>
<td>1</td>
<td>up</td>
</tr>
</tbody>
</table>

show mpls traffic-eng tunnels
Displays information about tunnels.

Mon Jan 11 05:30:44.501 UTC
LSP Tunnel 10.77.132.158 1 [8] is signalled, Signaling State: up
Tunnel Name: ios_ot1_1.1.1.1 Tunnel Role: Tail
Upstream label:
Optical label:
Grid: DWDM
Channel spacing: 50 GHz
Identifier: 0
Channel Number: -8
Downstream label:
Optical label:
Grid: DWDM
Channel spacing: 50 GHz
Identifier: 0
Channel Number: -8
Signalling Info:
Src 10.77.132.158 Dest 1.1.1.1, Tun ID 1, Tun Inst 8, Ext ID 10.77.132.158
Router-IDs: upstream 10.77.132.158
local 10.77.132.156
Priority: 7 7
SRLGs: not collected
Path Info:
Incoming Address: 1.1.1.2
Incoming:
Explicit Route:
No ERO
Route Exclusions:
No XRO
Record Route: Disabled
Tspec: avg rate=4294967033 kbits, burst=1000 bytes, peak rate=4294967033 kbits
show rsvp neighbors
Displays information about RSVP neighbors.

Mon Jan 11 05:33:21.483 UTC
Global Neighbor: 10.77.132.158

show lmp gmpls optical-uni
Verifies LMP configuration and state.

GMPLS Optical-UNI LMP Router ID: 10.77.132.156

LMP Neighbor
Name: RDT_2, IP: 10.77.132.158, Owner: GMPLS Optical-UNI
IPCC ID: 1, State Up
Known via: Configuration
Type: Routed
Destination IP: 10.77.132.158
Source IP: 10.77.132.156

Interface I/F | Lcl Interface ID | Lcl Link ID | Interface LMP state
---------------------------------------------------------------
Optics0/0/0/13 6 1.1.1.1 Up

Example of MPLS Tunnel Creation Without ERO
On the source NCS 1002 node:

lmp
gmpls optical-uni
controller Optics0/0/0/5
neighbor HUB-1
neighbor link-id ipv4 unicast 4.4.4.2
neighbor interface-id unnumbered 2130709780
link-id ipv4 unicast 4.4.4.1
neighbor HUB-1
ipcc routed
router-id ipv4 unicast 10.77.142.94
mpls traffic-eng
gmpls optical-uni
controller Optics0/0/0/5
tunnel-properties
tunnel-id 10
destination ipv4 unicast 4.4.4.4
path-option 10 no-ero lockdown
rsvp
controller Optics0/0/0/5
signalling refresh out-of-band interval 3600
signalling refresh out-of-band missed 24

On the destination NCS 1002 node:

lmp
gmpls optical-uni
controller Optics0/0/0/5
neighbor HUB-2
neighbor link-id ipv4 unicast 4.4.4.3
neighbor interface-id unnumbered 2130709780
link-id ipv4 unicast 4.4.4.4
neighbor HUB-2
ipcc routed
router-id ipv4 unicast 10.77.142.46
router-id ipv4 unicast 10.77.142.92
mpls traffic-eng
gmpls optical-uni
controller Optics0/0/0/5
rsvp
controller Optics0/0/0/6
signalling refresh out-of-band interval 3600
signalling refresh out-of-band missed 24

The following is the output of the show mpls traffic-eng tunnels 10 command on the source NCS 1002 node.

Name: GMPLS-UNI-Optics0/0/0/5 Destination: 4.4.4.4
Signalled-Name: HEADNODE.ot10_4.4.4.4
GMPLS UNI tunnel controlling link Optics0/0/0/5, tunnel-id: 10
Status:
  Admin: up Oper: up Path: valid Signalling: connected
  path option 10, (LOCKDOWN) type no-ero (Basis for Setup)
  Last Signalled Error : Tue Feb 14 02:19:01 2017
    Info: [11] PathErr(24,6)-(routing, unacceptable label object) at 4.4.4.2
    G-PID: 0x0800 (derived from egress interface properties)
    Creation Time: Tue Jan 10 15:07:11 2017 (4w6d ago)
Config Parameters:
  Priority: 7 7 Affinity: 0x0/0xffff
  Path Protection: Not Enabled
  BFD Fast Detection: Disabled
  Reoptimization after affinity failure: Enabled
  SRLG discovery: Disabled
History:
  Tunnel has been up for: 00:00:33 (since Tue Feb 14 02:19:02 IST 2017)
  Current LSP:
    Uptime: 00:00:33 (since Tue Feb 14 02:19:02 IST 2017)
Displayed 1 (of 4) heads, 0 (of 0) midpoints, 0 (of 0) tails
Displayed 1 up, 0 down, 0 recovering, 0 recovered heads

Example of MPLS Tunnel Creation with ERO

On the source NCS 1002 node:

mpls traffic-eng
gmpls optical-uni
controller Optics0/0/0/5
tunnel-properties
tunnel-id 10
destination ipv4 unicast 4.4.4.4
path-option 10 explicit name ero-1 lockdown verbatim

explicit-path name ero-1
  index 10 next-address strict ipv4 unicast 4.4.4.2
  index 20 next-address strict ipv4 unicast 10.77.142.66

The following is the output of the `show mpls traffic-eng tunnels 10` command on the source NCS 1002 node.

Name: GMPLS-UNI-Optics0/0/0/5 Destination: 4.4.4.4
Signalled-Name: HEADNODE_ot10_4.4.4.4
GMPLS UNI tunnel controlling link Optics0/0/0/5, tunnel-id: 10
Status:
  Admin: up Oper: up Path: valid Signalling: connected

  path option 10, (LOCKDOWN verbatim) type explicit ero-1 (Basis for Setup)
  Last Signalled Error : Tue Feb 14 01:57:02 2017
    Info: [7] PathErr(24,6)-(routing, unacceptable label object) at 4.4.4.2
    G-PID: 0x0800 (derived from egress interface properties)
    Creation Time: Tue Jan 10 15:07:11 2017 (4w6d ago)
   Config Parameters:
     Priority: 7 7 Affinity: 0x0/0xffffffff
     Path Protection: Not Enabled
     BFD Fast Detection: Disabled
     Reoptimization after affinity failure: Enabled
     SRLG discovery: Disabled
   History:
     Tunnel has been up for: 00:09:19 (since Tue Feb 14 01:57:02 IST 2017)
     Current LSP:
       Uptime: 00:09:19 (since Tue Feb 14 01:57:02 IST 2017)

   Path info (No IGP):
   Hop0: 4.4.4.2
   Hop1: 10.77.142.66
   Displayed 1 (of 4) heads, 0 (of 0) midpoints, 0 (of 0) tails
   Displayed 1 up, 0 down, 0 recovering, 0 recovered heads

Example of MPLS Tunnel Creation with XRO

On the source NCS 1002 node:

```plaintext
mpls traffic-eng
  attribute-set xro xro-1
    exclude strict lsp source 10.77.132.93 destination 3.3.3.4 tunnel-id 22
  extended-tunnel-id 10.77.132.93
    exclude strict srlg value 123123
mpls optical-uni
  controller Optics0/0/0/5
    tunnel-properties
      tunnel-id 10
      destination ipv4 unicast 4.4.4.4
      path-option 10 no-ero xro-attribute-set xro-1 lockdown
  controller Optics0/0/0/6
    tunnel-properties
      tunnel-id 22
      destination ipv4 unicast 3.3.3.4
      path-option 12 no-ero lockdown
```

The following is the output of the `show mpls traffic-eng tunnels 10` command on the source NCS 1002 node.
Example of MPLS Tunnel Creation with Explicit Signaled Wavelength

On the source NCS 1002 node:

```bash
gmpls optical-uni
controller Optics0/0/0/5
  tunnel-properties
    tunnel-id 10
    destination ipv4 unicast 4.4.4.4
    path-option 10 no-ero signaled-label dwdm wavelength 22 lockdown
```

Validate GMPLS Assigned Channel Number (6.25 Ghz) over Coherent DSP

This procedure allows you to validate whether the channel number assigned through GMPLS is programmed in optics controller.

**Step 1**
show mpls traffic-eng link-management optical-uni controller optics 0/0/0/13

This command provides the channel number assigned (say 88) and channel spacing (say 6.25 Ghz).

Mon Sep 25 10:58:02.018 UTC
Optical interface: Optics0/0/0/13

Overview:
- IM state: Up
- Child interface: : IM state Unknown
- OLM/LMP state: Up
- Optical tunnel state: up

Connection:
- Tunnel role: Head
- Tunnel-id: 32, LSP-id 144, Extended tunnel-id 10.77.142.93
- Tunnel source: 10.77.142.93, destination: 6.1.1.1
- Optical router-ids: Local: 10.77.142.93, Remote: 10.77.142.94
- Label source: UNI-N
- Upstream label:
  - Optical label:
    - Grid : DWDM
    - Channel spacing : 6.25 GHz
    - Identifier : 0
    - Channel Number : 88
- Downstream label:
  - Optical label:
    - Grid : DWDM
    - Channel spacing : 6.25 GHz
    - Identifier : 0
    - Channel Number : 88
- SRLG discovery: Disabled
- SRLG announcement: None

Step 2  show controllers optics 0/0/0/13 dwdm-carrrier-map flexi-grid

This command provides the mapping for the channel number (say 88) to respective frequency (say 193.65000) and wavelength (say 1548.115).

DWDM Carrier Band:: OPTICS_C_BAND
MSA ITU channel range supported: 1-97

DWDM Carrier Map table
----------------------------------------------------
| Channel G.694.1 Frequency Wavelength |
| index Ch Num (THz) (nm)               |
----------------------------------------------------
| 393 88 193.65000 1548.115            |

Step 3  show controllers optics 0/0/0/13 inc Wavelength

This command provides the wavelength present in optics controller for the specific trunk port.

Tue Nov  7 04:05:09.148 UTC
Wavelength= 1548.115

Step 4  show hw-module slice 1 coherentDSP 13 config hardware inc Frequency

This command verifies the frequency obtained in step 3 with the frequency programmed in coherent DSP.

Tue Nov  7 04:07:35.449 UTC

**********COHERENT DSP 13**********
Device Configuration:
- Traffic mode: 16-QAM 250G 20%-FEC (0)
- Power Mode: High Power (1)
- Near End Loop: disabled (0)
- Far End Loop: disabled (0)
- Cau Bitrate: Ethernet (1)
- Cau Mode: 0x0006
- TX Laser Frequency: 193.65000
- TX Optical Output Power SetPoint: -1.50 dBm
- TX Cau Ports enabled: 0x00000007
- TX Output Power: enabled
- TX Differential Encoder: enabled
- Tx LaWavlM ConsAction: disabled
- RX Laser Frequency: 193.65000
- RX Cau Ports enabled: 0x00000007
- RX CD Start Value: -70000 ps/nm
- RX CD Stop Value: 70000 ps/nm
- RX PreFec Ber Raise Threshold: 1065353216.000000
- RX PreFec Ber Cease Threshold: 1065353216.000000
- RX Preemphasis: P: 0 R: 0 T: 0 Q: 0 S: 0 U: 0
- RX Differential Decoder: enabled
- RX SD-FEC Decoder: enabled
- RX Lock to Hold-On Clock: disabled
- RX Enable POWALM cons action: disabled
- TX MCLK: disabled
- Tx MCLK Rate: 1/32 of symbolrate (0)
- RX MCLK: disabled
- RX MCLK Rate: 1/32 of symbolrate (0)
- RX PreFec Ber Raise Time Interval: 1.00 s
- RX PreFec Ber Cease Time Interval: 1.00 s
- TX Mapper: A: PortP_L1, B: PortP_L2, C: PortR_L1, D: PortR_L2, E: PortT_L1
- RX Demapper: A: PortP_L1, B: PortP_L2, C: PortR_L1, D: PortR_L2, E: PortT_L1
- TX Laser Fine Tuning: 0.0 GHz
- RX Laser Fine Tuning: 0.0 GHz

Validate GMPLS Assigned Channel Number (6.25 GHz) over Coherent DSP
Validate GMPLS Assigned Channel Number (6.25 Ghz) over Coherent DSP
The client ports can operate at 10G mode using an external breakout patch panel.

- Breakout Patch Panel, on page 97
- Configure Breakout Patch Panel, on page 97

**Breakout Patch Panel**

The key features of the breakout patch panel are as follows:

- Has 20 MPO ports in the back side that can be connected to 20 QSFP+ client ports of NCS 1002.
- Has 4 * 10G client ports in the front side for each MPO port.
- Has dual power supply.

The benefits of using the breakout patch panel are as follows:

- Labels are assigned to each 10G client port and MPO port. 10G client ports are labeled 0-1 0-2 0-3 0-4, 1-1, 1-2, 1-3, 1-4, and so on. MPO ports are labeled 0, 1, 2, 3, and so on.
- Link status LED indication is provided to each 10G client port and MPO port.

The breakout patch panel can be connected to NCS 1002 using the following methods:

- The RJ45 Ethernet port, ETH2, available in the rear side of NCS 1002, is used to connect the breakout patch panel back-to-back with NCS 1002. This port is visible as MgmtEth0/RP0/CPU0/2 in IOS XR. The user must configure the ETH2 interface to bring up the back-to-back IP network.
- Management LAN, ETH0, can be used to connect breakout patch panel with NCS 1002. The user needs to manually bring up the patch panel using the serial port on the patch panel.

**Configure Breakout Patch Panel**

Connect Patch Panel Back-to-back with NCS 1002
Bring up the ETH2 interface in 169.254.1 network.

```
configure
interface interface
ipv4 ipv4 address subnetmask
no shut
exit
patch-panel
exit
commit
```

**Example**

The following is a sample to configure the breakout patch panel by connecting patch panel back-to-back with NCS 1002.

```
configure
interface MgmtEth0/RP0/CPU0/2
ipv4 address 10.1.1.4 255.255.255.0
no shut
exit
patch-panel
exit
commit
```

**Connect Patch Panel with NCS 1002 Using Management LAN**

The user needs to manually configure the ETH0 interface of the patch panel.

Issue the following commands from the patch panel.

```
sudo ifconfig eth0 ipaddress ipaddress netmask ipaddress up
sudo route add default gw ipaddress
```

Issue the following commands from NCS 1002.

```
patch-panel
ipv4 ipv4 address
exit
commit
```

**Example**

The following is a sample to configure the breakout patch panel using the management LAN.

From the patch panel:
sudo ifconfig eth0 169.254.1.1 netmask 255.255.255.0 up
sudo route add default gw 10.77.132.1

From NCS 1002:

patch-panel
ipv4 10.77.132.176
exit
commit

Display Patch Panel Events

show patch-panel events

Mon Oct 24 12:07:19.963 UTC
{u'fimo_alarms_history_header': u'History of Alarms and Events'}
{u'fimo_alarms_history_help': u'(Time) (ID) (Type) (Message)'}
{u'events_800': u'(2016/09/17 18:37:58) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC2 data: port_led_color: off, port_led_mode: solid')}
{u'events_799': u'(2016/09/17 18:37:58) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC3 data: port_led_color: off, port_led_mode: solid')}
{u'events_798': u'(2016/09/17 18:37:57) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC4 data: port_led_color: off, port_led_mode: solid')}
{u'events_797': u'(2016/09/17 18:37:57) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC5 data: port_led_color: off, port_led_mode: solid')}
{u'events_796': u'(2016/09/17 18:37:57) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC8 data: port_led_color: off, port_led_mode: solid')}
{u'events_795': u'(2016/09/17 18:37:57) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC9 data: port_led_color: off, port_led_mode: solid')}
{u'events_794': u'(2016/09/17 18:37:56) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC10 data: port_led_color: off, port_led_mode: solid')}
{u'events_793': u'(2016/09/17 18:37:56) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC11 data: port_led_color: off, port_led_mode: solid')}
{u'events_792': u'(2016/09/17 18:37:56) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC12 data: port_led_color: off, port_led_mode: solid')}
{u'events_791': u'(2016/09/17 18:37:55) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC13 data: port_led_color: off, port_led_mode: solid')}
{u'events_790': u'(2016/09/17 18:37:55) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC14 data: port_led_color: off, port_led_mode: solid')}
{u'events_789': u'(2016/09/17 18:37:55) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC15 data: port_led_color: off, port_led_mode: solid')}
{u'events_788': u'(2016/09/17 18:37:55) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC16 data: portLedColor: off, portLedMode: solid')}
{u'events_787': u'(2016/09/17 18:37:54) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC17 data: portLedColor: off, portLedMode: solid')}
{u'events_786': u'(2016/09/17 18:37:54) (4360) (Event) (Vars were changed; from IP: 10.77.132.125 table: port/b827eba9fb157000/LC18 data: portLedColor: off, portLedMode: solid')

The **patch-panel reset** command is used to reset the patch panel.

### Alarms in Breakout Patch Panel

Two alarms, **NOT ABLE TO COMMUNICATE WITH PATCH-PANEL** and **PATCH-PANEL POWER REDUNDANCY LOST**, are raised for the breakout patch panel. For description and clearing procedures of these alarms, see the Alarm Troubleshooting chapter in the *Troubleshooting Guide for Cisco NCS 1000 Series*. 
Smart Licensing

This chapter describes Smart Licensing Software configuration on Cisco NCS1002.

Smart Licensing is a cloud-based approach to licensing. Smart Licensing simplifies the licensing experience across the enterprise making it easier to purchase, deploy, track and renew Cisco Software. It provides visibility into license ownership and consumption through a single, simple user interface. The solution allows you to easily track the status of your license and software usage trends.

This chapter contains the following topics:

• Prerequisites for Smart Licensing, on page 101
• Information About Smart Licensing, on page 101
• Configuring Smart Licensing, on page 103
• Verifying Smart Licensing Configuration, on page 105

Prerequisites for Smart Licensing

To use the smart licensing feature of Cisco NCS1002, ensure that you have a Cisco Smart Account to launch Cisco Smart Software Manager portal. For more details, see Cisco Smart Software Manager User Guide.

Information About Smart Licensing

Before using the Smart licensing feature, you should understand what is a PID.

PID is the name by which the product can be ordered; it has been historically called the Product Name or Part Number. This is the identifier that you use to order an exact replacement part.

Cisco NCS1002 has two PIDs:

• Fully licensed PID (NCS1002-K9)—This is a high cost PID. By using the fully licensed PID, you can configure and use the software without any explicit licensing.
• Partially licensed PID (NCS1002-LIC-K9)—This is a low cost PID and you have to additionally buy licenses to configure the software.

Benefits of Smart Licensing

• Licenses are not locked to perform configurations using Cisco NCS1002 even if the license limit exceeds the paid license limit. You will be notified with out-of-compliance notification to buy additional licenses.
when the license limit exceeds the paid license limit. This saves time with the ability to transfer licenses across the company.

- Licenses can be pooled across the entire company, enabling them to be reused across organizational boundaries.
- Provides software asset management information so that you can plan and easily track if you have purchased adequate licenses.

Software Entitlements of Cisco NCS1002

Software entitlement is a system that consists of a license manager on Cisco NCS1002 that manages licenses for various software and hardware features. The license manager parses and authenticates a license before accepting it.

Core features are available for use without any license. The following features are available on Cisco NCS1002 only using licenses:

- Configuring a slice with 200G DWDM traffic.
- Configuring a slice with 200G/250G DWDM traffic.
- Configuring a slice with encryption.
- (Only for R6.2.1 and R6.2.2) Configuring streaming telemetry data.

The following table lists the features and its corresponding entitlements that can be enabled on Cisco NCS1002 using licenses:

Table 4: Software Entitlements of Cisco NCS1002

<table>
<thead>
<tr>
<th>Feature</th>
<th>Software Entitlement</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCS1K Smart License - one slice with 200G DWDM</td>
<td>L_NCS1K_200G_LIC</td>
</tr>
<tr>
<td>NCS1K Smart License - one slice with 200G/250G DWDM</td>
<td>L_NCS1K_250G_LIC</td>
</tr>
<tr>
<td>NCS1K Smart License - one slice with encryption</td>
<td>L_NCS1K_ENCR_LIC</td>
</tr>
<tr>
<td>(Only for R6.2.1 and R6.2.2) NCS1K Smart License - streaming telemetry</td>
<td>L-NCS1K-ST-LIC</td>
</tr>
</tbody>
</table>

Note

You can view these entitlements and their usage details in the Cisco Smart Software Manager portal.

Smart Licensing Work Flow

The figure below depicts a working model of Smart Licensing that is a three-step procedure.
Figure 7: Smart Licensing Work Flow

Activation — You can place the order for Smart Licensing to manage licenses on Cisco.com portal.

Note — Smart licensing is enabled by default in Cisco NCS1002.

Registration — You must register Cisco NCS1002 with your smart account using either of the following options to communicate:

- Smart Call Home — The Smart Call Home feature is automatically configured after the Smart Licensing is enabled. Smart Call Home is used by Smart Licensing as a medium to communicate with the Cisco license service. Smart 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 pursue service and support contract renewals more effectively, without much intervention from you.

- Smart Licensing Satellite — Cisco Smart Software Manager satellite is an element of Cisco Smart Software Licensing. It coordinates with the Cisco Smart Software Manager to manage software licenses on premises. The devices register locally to report license ownership and consumption. This creates synchronization between the Cisco Smart Software Manager satellite and the Cisco Smart Software Manager. For more details about the Cisco Smart Software Manager satellite, see www.cisco.com/go/smartsatellite.


License Management — 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.

For more information, see Configuring Smart Licensing, on page 103.

Configuring Smart Licensing

To configure smart licensing in Cisco NCS1002, perform the following tasks:
Step 1
Set up the CiscoTAC-1 profile and destination address for Smart Call Home, using the following 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*  
FQDN must be either Cisco Smart Software Manager FQDN (tools.cisco.com) or Smart Licensing satellite server FQDN. You must configure the DNS server before setting up the call-home destination address as FQDN. Use `domain name-server {DNS server IP}` command for configuring DNS server on device.

*Example:*

domain name-server 64.102.6.247
call-home
service active
contact smart-licensing
profile CiscoTAC-1
active
destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService
destination transport-method http

*Note*  
CiscoTAC-1 profile is the default profile for smart licensing and it should not be deleted.

*Note*  
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, then you must add the following configuration on the device to make CRL check optional as part of registration.

crypto ca trustpoint Trustpool crl optional

Step 2
Create and copy the registration token ID using Cisco Smart Software Manager.

For more details on how to create a token, see *Creating a Token, on page 104.*

Step 3
Register the token ID in Cisco NCS1002, using the following commands:

license smart register idtoken token-ID

---

**Creating a Token**

To create a new token using Cisco Smart Software Manager, perform the following tasks:

**Step 1**
Log into the Cisco Smart Software Manager.

URL: https://software.cisco.com/#SmartLicensing-Inventory
Verifying Smart Licensing Configuration

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 status`
- `show license summary`
- `show license tech`
- `show license udi`
- `show license usage`
- `show license platform detail`
- `show tech-support smartlic`
- `show tech-support call-home`

The following table defines the available license authorization status in Cisco NCS1002:

<table>
<thead>
<tr>
<th>License Authorization Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unconfigured</td>
<td>Smart Software Licensing is not configured</td>
</tr>
<tr>
<td>Unidentified</td>
<td>Smart Software Licensing is enabled but is not registered</td>
</tr>
<tr>
<td>Registered</td>
<td>Device registration is completed and an ID certificate is received that will be used for future communication with the Cisco licensing authority</td>
</tr>
<tr>
<td>Authorized</td>
<td>Registration is completed with a valid Smart Account and license consumption has begun. This indicates compliance</td>
</tr>
<tr>
<td>Out of Compliance</td>
<td>Consumption exceeds available licenses in the Smart Account</td>
</tr>
<tr>
<td>Authorization Expired</td>
<td>The device is unable to communicate with the Cisco Smart Software Manager for an extended period of time. This state occurs after 90 days of expiry. The device will attempt to contact the CSSM every hour in order to renew the authorization until the registration period expires</td>
</tr>
</tbody>
</table>

Example 1:
The following example shows the sample output of the `show license all` command.

```plaintext
RP/0/RP0/CPU0:ios#show license all
Fri Jan  6 01:33:24.230 UTC
Smart Licensing Status
-------------------------------
Smart Licensing is ENABLED
Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000 Initial
  Registration: SUCCEEDED on Thu Jan 05 2017 15:04:18 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Tue Jul 04 2017 15:04:18 UTC
  Registration Expires: Fri Jan 05 2018 09:25:50 UTC
License Authorization:
  Status: AUTHORIZED on Thu Jan 05 2017 15:17:04 UTC
  Last Communication Attempt: SUCCEEDED on Thu Jan 05 2017 15:17:04 UTC
  Next Communication Attempt: Sat Feb 04 2017 15:17:04 UTC
  Communication Deadline: Fri Jan 05 2018 09:25:50 UTC
License Usage
-------------------
No licenses in use
Product Information
-----------------------
UDI: SN:CAT1111B0KR, UUID: default-sdr
Agent Version-------------------Smart Agent for Licensing: 2.2.0_rel/17
```

Example 2:

The following example shows the sample output of the `show license platform detail` command. The output displays telemetry entry only for R6.2.1 and R6.2.2.

```plaintext
RP/0/RP0/CPU0:ios#show license platform detail
Fri Jan 20 12:09:30.729 UTC
Current state: REGISTERED
Collection: LAST: Fri Jan 20 12:03:13 2017
  NEXT: Fri Jan 20 13:03:13 2017
Reporting: LAST: Fri Jan 20 12:03:13 2017
  NEXT: Sat Jan 21 12:03:13 2017
Parameters: Collection interval: 60 minute(s)
  Reporting interval: 1440 minute(s)
  Throughput gauge: 1000000 Kbps
```

Feature/Area 'sys_features'
-------------------------------
Name: sys_features
  DLL: libplat_sl_plugin_sys_features.so
  Flags: CONFIG
  # CLI: 1
  Entitlements (total 1):
    [ 0] Name: NCS1K Smart License - streaming telemetry
      Token:
        regid.2017-01.com.cisco.L-NCS1K-ST-LIC-,1.0_6222b080-6651-4611-a725-fc84b647d614
      Count: Last reported: 1
      Next report: 0

Feature/Area 'system'
-------------------------------
Name: System
  DLL: libplat_sl_plugin_system.so
  Flags: CONFIG
  # CLI: 3
  Entitlements (total 3):
    [ 0] Name: NCS1K Smart License - one slice with 200G DWDM
```
The following examples show the sample output of `show license summary` command.

```
RP/0/RP0/CPU0:ios#show license summary
Thu Feb 2 23:10:46.723 UTC
Smart Licensing is ENABLED
Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: NCS1000
  Last Renewal Attempt: None
  Next Renewal Attempt: Tue Aug 01 2017 23:09:00 UTC
License Authorization:
  Status: AUTHORIZED on Thu Feb 02 2017 23:09:12 UTC
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Thu Feb 02 2017 23:10:53 UTC
License Usage:
  License Entitlement tag     Count Status
  ------------------------------------------
  {L-NCS1K-200G-LIC}           2 PENDING
  {L-NCS1K-250G-LIC}           1 PENDING
  {L-NCS1K-ENCR-LIC-}          1 PENDING
```

The following example shows the sample output of `show license usage` command.

```
RP/0/RP0/CPU0:ios#show license usage
Fri Feb 3 21:08:24.097 UTC
License Authorization:
  Status: No Licenses in Use
RP/0/RP0/CPU0:ios#
RP/0/RP0/CPU0:ios#show license trace ?
  all  Show tracing for both Smart Licensing and client(cisco-support)
  client Show tracing for the agent client code(cisco-support)
  smartlic Show tracing for the Smart Licensing Code(cisco-support)
```

The following table defines the license counting based on the configurations:
### Table 6: License Counting Mechanism

<table>
<thead>
<tr>
<th>License Requested by Customer</th>
<th>License Given from Cisco</th>
<th>Configuration Done by Customer</th>
<th>License Counted in CSSM</th>
<th>License Requested by Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>S-NCS1K-LIC-200G=</td>
<td>S-NCS1K-LIC-200G=</td>
<td>Slice with 200G trunk without encryption</td>
<td>1 - S-NCS1K-LIC-200G=</td>
<td>Only 200G trunk can be used.</td>
</tr>
<tr>
<td>S-NCS1K-LIC-250G=</td>
<td>S-NCS1K-LIC-200G=</td>
<td>Slice with 250G trunk without encryption</td>
<td>1 - S-NCS1K-LIC-200G=</td>
<td>Both the licenses is counted as used. This is for restricting the customer to configure either 200G or 250G trunk at the same time using this license (not both at a time). i.e. whenever there is a 250G configuration performed, both 200G and 250G licenses should be available to use.</td>
</tr>
<tr>
<td></td>
<td>S-NCS1K-LIC-250G=</td>
<td></td>
<td>1 - S-NCS1K-LIC-200G=</td>
<td>1 - S-NCS1K-LIC-250G=</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 - S-NCS1K-LIC-250G=</td>
<td>(Additional 200G license will be given)</td>
</tr>
<tr>
<td>S-NCS1K-LIC-ENCR=</td>
<td>S-NCS1K-LIC-ENCR=</td>
<td>One slice with encryption</td>
<td>1 - S-NCS1K-LIC-ENCR=</td>
<td>This works as 200G license procurement, 250G license is not counted. Still the customer cannot configure 250G trunk, because 250G trunk configuration requires both 200G and 250G licenses.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 - S-NCS1K-LIC-ENCR=</td>
<td>This is counted regardless of trunk rate.</td>
</tr>
<tr>
<td>License Requested by Customer</td>
<td>License Given from Cisco</td>
<td>Configuration Done by Customer</td>
<td>License Counted in CSSM</td>
<td>License Requested by Customer</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------</td>
<td>-------------------------------</td>
<td>--------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>(Only for R6.2.1 and R6.2.2) S-NCS1K-LIC-ST=</td>
<td>S-NCS1K-LIC-ST=</td>
<td>Telemetry configuration</td>
<td>1- S-NCS1K-LIC-ST=</td>
<td>-</td>
</tr>
</tbody>
</table>
Configuring SNMP

The following MIBs are supported in NCS 1002.

- CISCO-CONFIG-MAN-MIB
- CISCO-FLASH-MIB
- CISCO-ENTITY-REDUNDANCY-MIB
- CISCO-SYSTEM-MIB
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-STATE-EXT-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
- CISCO-SYSLOG-MIB
The following table provides more information about SNMP MIBs and the documentation links.

<table>
<thead>
<tr>
<th>Task</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determine the MIB definitions</td>
<td>SNMP Object Navigator</td>
</tr>
<tr>
<td>Configure SNMP</td>
<td>Configure SNMP</td>
</tr>
<tr>
<td>Understand the SNMP best practices regarding the recommended order of SNMP query, maximum cache hit, and SNMP retry and timeout recommendation</td>
<td>SNMP Best Practices</td>
</tr>
</tbody>
</table>

snmp-server community must be configured as SystemOwner for admin-plane parameters to appear to entity mib. The parameters of fans and power supply units are examples of admin-plane parameters.