



7705 Service Aggregation Router Gen 2

Release 26.10.R1

Interface Configuration Guide

3HE 29557 AAAC TQZZA 01

Edition: 01

October 2026

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

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

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

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

No part of this document may be modified.

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

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

© 2026 Nokia.

Table of contents

List of tables.....	9
List of figures.....	10
1 Getting started.....	12
1.1 About this guide.....	12
1.1.1 Platforms and terminology.....	12
1.2 Conventions.....	13
1.2.1 Precautionary and information messages.....	13
1.2.2 Options or substeps in procedures and sequential workflows.....	13
2 Configuration overview.....	15
2.1 Chassis slots and card slots.....	15
2.1.1 7705 SAR-10x.....	15
2.1.2 7705 SAR-Hx and 7705 SAR-Mx.....	16
2.1.3 7705 SAR-1.....	17
2.2 MDAs.....	18
3 Digital Diagnostics Monitoring.....	19
3.1 SFPs and XFPs.....	22
3.2 Statistics collection.....	23
4 Ports.....	24
4.1 Port types.....	24
4.1.1 Ethernet ports.....	26
4.1.1.1 Port speeds on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx.....	27
4.1.1.2 Port speed modes.....	28
4.1.2 TDM ports.....	29
4.1.2.1 8-port T1/E1 MDA.....	30
4.1.2.2 8-port RS-232 and C37.94 combination MDA.....	30
4.1.2.3 8-port C37.94 Teleprotection card.....	30
4.2 Port features.....	30
4.2.1 Port State and Operational State.....	30
4.2.2 802.1x network access control.....	31

4.2.2.1	802.1x basics.....	31
4.2.2.2	802.1x modes.....	33
4.2.2.3	802.1x timers.....	34
4.2.2.4	802.1x tunneling.....	35
4.2.2.5	Per-host authentication.....	35
4.2.2.6	802.1x configuration and limitations.....	38
4.2.3	MACsec.....	39
4.2.3.1	MACsec 802.1AE header — security TAG.....	40
4.2.3.2	MACsec encryption mode.....	40
4.2.3.3	MACsec key management modes.....	41
4.2.3.4	MACsec terminology.....	42
4.2.3.5	MACsec static CAK.....	43
4.2.3.6	SAK rollover.....	45
4.2.3.7	MKA.....	45
4.2.3.8	PSK.....	50
4.2.3.9	Autogenerated PSK.....	54
4.2.3.10	MKA Hello timer.....	56
4.2.3.11	MACsec Capability, Desire, and encryption offset.....	57
4.2.3.12	Key server.....	57
4.2.3.13	SA limits and network design.....	58
4.2.3.14	P2P (switch-to-switch) topology.....	58
4.2.3.15	P2MP (switch-to-switch) topology.....	59
4.2.3.16	SA exhaustion behavior.....	60
4.2.3.17	Clear tag mode.....	60
4.2.3.18	802.1x tunneling and multihop MACsec.....	61
4.2.3.19	EAPoL destination address.....	61
4.2.3.20	Mirroring consideration.....	61
4.2.4	Exponential Port Dampening.....	61
4.2.5	Network synchronization on ports.....	64
4.2.5.1	Network synchronization on T1/E1 ports.....	64
4.2.6	Ethernet port down-when-looped.....	64
4.3	Forward Error Correction.....	65
5	Port Cross-Connect.....	66
5.1	PXC terminology.....	66
5.2	Overview.....	66

5.3	Port-based PXC.....	67
5.4	PXC sub-ports.....	69
5.5	Bandwidth considerations and QoS.....	72
5.5.1	QoS.....	72
5.5.1.1	QoS on PXC sub-ports.....	73
5.5.2	Queue allocation on PXC sub-ports.....	75
5.5.3	Pool allocations on PXC ports.....	75
5.6	Operational states.....	75
5.7	PXC statistics.....	75
5.7.1	Statistics on PXC ports.....	76
5.7.2	Statistics collection on PXC sub-ports and PXC LAG.....	76
5.7.2.1	MIBs.....	81
5.7.2.2	Restrictions.....	81
5.8	PXC LAG.....	82
5.9	Basic PXC provisioning.....	83
5.10	PXC mirroring and LI.....	85
5.11	Configuration example.....	86
6	LAG.....	93
6.1	LACP.....	93
6.1.1	LACP multiplexing.....	94
6.1.2	LACP tunneling.....	95
6.2	LAG sub-group.....	95
6.3	Traffic load balancing options.....	96
6.3.1	Per-flow hashing.....	97
6.3.1.1	LSR hashing.....	98
6.3.1.2	Layer 4 load balancing.....	100
6.3.1.3	System IP load balancing.....	101
6.3.1.4	Source-only/destination-only hash inputs.....	101
6.3.2	LAG port hash weight.....	101
6.3.2.1	Configurable hash weight to control flow distribution.....	102
6.3.3	Consistent per-service hashing.....	104
6.3.4	ESM.....	105
6.3.4.1	Load balancing per subscriber.....	105
6.3.4.2	Load balancing per Vport.....	107
6.3.4.3	Load balancing per secondary shaper.....	108

6.3.4.4	Load balancing per destination MAC.....	108
6.4	QoS consideration for access LAG.....	108
6.4.1	Adapt QoS modes.....	108
6.4.2	Per-fp-ing-queuing.....	111
6.4.3	Per-fp-egr-queuing.....	111
6.4.4	Per-fp-sap-instance.....	112
6.5	LAG hold-down timers.....	112
6.6	Multi-Chassis LAG.....	113
6.6.1	Overview.....	113
6.6.2	MC-LAG and SRRP.....	117
6.6.3	P2P redundant connection across Layer 2/3 VPN network.....	117
6.6.4	DSLAM dual-homing in a Layer 2/3 TPSDA model.....	118
6.7	LAG thresholds.....	119
6.7.1	LAG IGP cost.....	119
6.7.2	Adjusting the operational state of the LAG.....	120
7	Ethernet port monitoring.....	122
8	IEEE 802.3ah OAM.....	126
8.1	OAM events.....	128
8.2	Remote loopback.....	129
8.3	802.3ah OAM PDU tunneling for Epipe service.....	129
9	Serial transport over raw sockets.....	130
9.1	Raw socket configuration.....	131
9.2	Raw socket packet processing.....	131
9.2.1	Raw socket processing for UDP sessions.....	132
9.2.2	Raw socket processing for TCP sessions.....	132
9.3	Raw socket squelch functionality.....	133
10	MTU configuration guidelines.....	134
10.1	Default MTU values.....	134
10.2	Modifying MTU defaults.....	134
10.3	Configuration example.....	135
11	Deploying preprovisioned components.....	136

12	Configuration process overview.....	137
12.1	Configuration notes.....	137
13	Configuring physical ports with CLI.....	138
13.1	Preprovisioning guidelines.....	138
13.1.1	Predefining entities.....	138
13.1.2	Preprovisioning a port.....	138
13.1.3	Maximizing bandwidth use.....	138
13.2	Basic configuration.....	139
13.3	Common configuration tasks.....	140
13.3.1	Configuring cards and MDAs.....	140
13.3.2	Configuring ports.....	141
13.3.2.1	Configuring port pools.....	141
13.3.2.2	Changing hybrid-buffer-allocation.....	143
13.3.2.3	Configuring Ethernet ports.....	144
13.3.2.4	Configuring LAG.....	146
13.3.2.5	Configuring GNSS ports.....	147
13.3.2.6	Configuring TDM ports.....	147
13.3.2.7	Configuring teleprotection ports.....	148
13.3.2.8	Configuring serial ports for raw socket transport.....	149
14	Service management tasks.....	150
14.1	Modifying or deleting an MDA.....	150
14.2	Modifying a card type.....	150
14.3	Deleting a card.....	151
14.4	Deleting port command options.....	151
14.5	Soft IOM reset.....	151
14.5.1	Soft reset.....	152
14.5.2	Deferred MDA reset.....	152
15	Standards and protocol support.....	154
15.1	Bidirectional Forwarding Detection (BFD).....	154
15.2	Border Gateway Protocol (BGP).....	154
15.3	Bridging and management.....	155
15.4	Certificate management.....	156
15.5	Ethernet.....	156

15.6	Ethernet VPN (EVPN).....	156
15.7	gRPC Remote Procedure Calls (gRPC).....	157
15.8	Intermediate System to Intermediate System (IS-IS).....	157
15.9	Internet Protocol (IP) general.....	158
15.10	Internet Protocol (IP) multicast.....	159
15.11	Internet Protocol (IP) version 4.....	160
15.12	Internet Protocol (IP) version 6.....	160
15.13	Internet Protocol Security (IPsec).....	161
15.14	Label Distribution Protocol (LDP).....	162
15.15	Multiprotocol Label Switching (MPLS).....	163
15.16	Network Address Translation (NAT).....	163
15.17	Network Configuration Protocol (NETCONF).....	163
15.18	Open Shortest Path First (OSPF).....	163
15.19	Path Computation Element Protocol (PCEP).....	164
15.20	Pseudowire (PW).....	165
15.21	Quality of Service (QoS).....	166
15.22	Remote Authentication Dial In User Service (RADIUS).....	166
15.23	Resource Reservation Protocol - Traffic Engineering (RSVP-TE).....	166
15.24	Routing Information Protocol (RIP).....	166
15.25	Segment Routing (SR).....	167
15.26	Simple Network Management Protocol (SNMP).....	167
15.27	Storage sanitization.....	169
15.28	Timing.....	169
15.29	Two-Way Active Measurement Protocol (TWAMP).....	169
15.30	Virtual Private LAN Service (VPLS).....	170
15.31	Yet Another Next Generation (YANG).....	170

List of tables

Table 1: Platforms and terminology.....	12
Table 2: Real-time DDM information.....	21
Table 3: DDM alarms and warnings.....	21
Table 4: Supported speeds for pluggable transceivers.....	26
Table 5: Supported Ethernet port and pluggable transceiver types.....	26
Table 6: Ethernet port speeds.....	28
Table 7: Port speed modes on the 7705 SAR-10x.....	29
Table 8: Port speed modes on the 7705 SAR-Hx and 7705 SAR-Mx.....	29
Table 9: Relationship of Port state and Oper state.....	31
Table 10: MACsec key management modes.....	42
Table 11: MACsec terminology.....	43
Table 12: MKA PDU generation.....	46
Table 13: Tags in clear behavior.....	48
Table 14: MKA participant timer values.....	56
Table 15: MACsec basic settings.....	57
Table 16: Port mapping to security zone.....	58
Table 17: MACsec encryption of 802.1Q tags with clear-tag-mode configured.....	60
Table 18: Port types and speeds.....	102
Table 19: Adapt QoS bandwidth/rate distribution.....	110
Table 20: MTU default values.....	134
Table 21: MTU configuration example values.....	135

List of figures

Figure 1: 802.1x architecture.....	32
Figure 2: 802.1x authentication scenario.....	33
Figure 3: 802.1x EAPOL timers (left) and RADIUS timers (right).....	34
Figure 4: 802.1 AE LAN-MODE.....	40
Figure 5: SecTAG format.....	40
Figure 6: 802.1 AE LAN and WAN modes and VLAN encrypted and cleartext.....	41
Figure 7: MACsec concepts for static-CAK.....	42
Figure 8: MACsec generating the CAK.....	44
Figure 9: MACsec control plane and encryption.....	45
Figure 10: Tolerance with three key entries.....	52
Figure 11: Backup key with the tolerance set to forever.....	53
Figure 12: Autogenerated PSK CKN format.....	55
Figure 13: Switch-point to switch-point topology.....	59
Figure 14: Switch-multipoint to switch-multipoint topology.....	59
Figure 15: EPD example.....	62
Figure 16: Traffic preprocessing using PXC.....	67
Figure 17: Port-based PXC.....	67
Figure 18: Two cross-connected external ports versus a single cross-connect.....	70
Figure 19: Interaction between PXC and non-PXC traffic.....	73
Figure 20: Bandwidth management on PXC sub-ports.....	74
Figure 21: Monitor port interval issue.....	82

Figure 22: Logical concept of a LAG on PXC ports.....	82
Figure 23: MD-CLI flow.....	84
Figure 24: Classic CLI flow.....	85
Figure 25: Active/standby LAG operation deployment examples.....	96
Figure 26: LAG on access interconnection.....	96
Figure 27: LAG on access failure switchover.....	96
Figure 28: Same-speed LAG with ports of different hash weight.....	103
Figure 29: MC-LAG Layer 2 dual-homing to remote PE pairs.....	115
Figure 30: MC-LAG Layer 2 dual homing to local PE pairs.....	116
Figure 31: Point-to-Point (P2P) redundant connection through a Layer 2 VPN network.....	118
Figure 32: DSLAM dual-homing using MC-LAG.....	119
Figure 33: Serial transport over raw socket application on the 7705 SAR-Hx.....	131
Figure 34: Raw socket packet processing.....	132
Figure 35: MTU configuration example.....	135
Figure 36: Slot, card, MDA, port configuration, and implementation flow.....	137

1 Getting started

1.1 About this guide

This guide describes system concepts and provides configuration examples to provision Input/Output modules (IOMs), Media Dependent Adapters (MDAs), and ports.

This guide is organized into functional chapters and provides concepts and descriptions of the implementation flow, as well as Command Line Interface (CLI) syntax and command usage.

Unless otherwise indicated, the topics and commands described in this guide apply only to the 7705 SAR Gen 2 platforms listed in [Platforms and terminology](#).

Command outputs shown in this guide are examples only; actual displays may differ depending on supported functionality and user configuration.



Note: Unless otherwise indicated, CLI commands, contexts, and configuration examples in this guide apply for both the classic CLI and the MD-CLI.

The SR OS CLI trees and command descriptions can be found in the following guides:

- *7705 SAR Gen 2 Classic CLI Command Reference Guide*
- *7705 SAR Gen 2 Clear, Monitor, Show, Tools CLI Command Reference Guide* (for both the MD-CLI and classic CLI)
- *7705 SAR Gen 2 MD-CLI Command Reference Guide*



Note: This guide generically covers Release 26.x.Rx content and may contain some content that will be released in later maintenance loads. See the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000x TQZZA, for information about features supported in each load of the Release 26.x.Rx software. For a list of features and CLI commands that are present in SR OS but not supported on the 7705 SAR Gen 2 platforms, see "SR OS Features not Supported on SAR Gen 2" in the *SR OS R26.x.Rx Software Release Notes*.

1.1.1 Platforms and terminology



Note: Unless explicitly noted otherwise, this guide uses the terminology defined in the following table to collectively designate the specified platforms.

Table 1: Platforms and terminology

Platform	Collective platform designation
7705 SAR-10x	7705 SAR Gen 2
7705 SAR-Hx	

Platform	Collective platform designation
7705 SAR-Mx	
7705 SAR-1	

1.2 Conventions

This section describes the general conventions used in this guide.

1.2.1 Precautionary and information messages

The following information symbols are used in the documentation.



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



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



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



Note: Note provides additional operational information.



Tip: Tip provides suggestions for use or best practices.

1.2.2 Options or substeps in procedures and sequential workflows

Options in a procedure or a sequential workflow are indicated by a bulleted list. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform one of the listed options to complete the step.

Example: Options in a procedure

1. User must perform this step.
2. This step offers three options. User must perform one option to complete this step.
 - This is one option.
 - This is another option.
 - This is yet another option.

Substeps in a procedure or a sequential workflow are indicated by letters. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step.

Example: Substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
 - a. This is one substep.
 - b. This is another substep.

Nested substeps within a procedure or a sequential workflow are indicated by roman numerals. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step. At substep b, the user must perform two additional substeps (i. and ii.) to complete the step.

Example: Nested substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
 - a. This is one substep.
 - b. User must perform all nested substeps to complete this action.
 - i. This is a nested substep.
 - ii. This is another nested substep.

2 Configuration overview

**Note:**

- This document uses the term "preprovisioning" in the context of preparing or preconfiguring entities such as chassis slots, cards, MDAs, ports, and interfaces, before initialization. These entities can be installed while remaining administratively disabled (shutdown). When the entity is in a no shutdown state (administratively enabled), then the entity is considered to be provisioned.
- Unless specified otherwise, the term "card" is used generically to refer to IOMs .

Nokia routers provide the capability to configure chassis slots to accept specific card and MDA types and set the relevant configurations before the equipment is actually installed. The preprovisioning capability allows you to plan your configurations as well as monitor and manage your router hardware inventory. Ports and interfaces can also be preprovisioned. When the functionality is needed, the cards can be inserted into the appropriate chassis slots when required.

2.1 Chassis slots and card slots

This section provides information about the chassis slots and cards for 7705 SAR Gen 2 routers.

Use the following command to display the cards provisioned and equipped in the chassis.

```
show card
```

2.1.1 7705 SAR-10x

The 7705 SAR-10x is a chassis-based platform with eight MDA slots and two CPIOM slots; CPIOM cards provide both CPM and IOM functionality. The 7705 SAR-10x chassis supports single-width 7705 SAR-10x MDAs in MDA slots 1 through 6, and double-width MDAs in slots 7 and 8.

The 7705 SAR-10x chassis supports the following removable MDAs.

- 7705 SAR-10x Ethernet MDAs
- 8-port RS-232 and C37.94 Combination MDA
- 8-port T1/E1 MDA
- 8-port C37.94 Teleprotection card

See the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000x TQZZA for a list of supported MDA types. See the *7705 SAR-Hx, SAR-Mx, and SAR-10x MDA Installation Guide* for more information about the installation of 7705 SAR-10x MDAs.

MDA slots must be provisioned on the 7705 SAR-10x. The MDA type must be specified to preprovision a chassis slot. System administrators or network operators can enter MDA type information for each slot. A preprovisioned slot can remain empty without conflicting with populated slots.



Note: The following restrictions apply when provisioning MDAs on the 7705 SAR-10x:

- The following MDAs are supported only in slots 2, 4, and 5 to 8:
 - m8-1g-sfp
 - m8-1g-rj45
 - m4-rs232-rj45+4-c3794-sfp
 - m8-t1e1-rj48
 - m8-c3794-sfp
- The m2-1g-sfp+2-10g-sfp+ MDA is supported only in slots 1 and 3.
- When the 7705 SAR-10x is in the **10g/25g+1g** port speed mode, TDM cards are supported only in slots 2 and 4.

CPIOMs can only be installed in slots A and B of the chassis and do not require provisioning. Both IOM components (card 1 and card 2) are always automatically provisioned, regardless of the number of installed CPIOMs. All MDAs must be configured under card 1; configuration of MDAs under card 2 is blocked.

The 7705 SAR-10x is preprovisioned from the factory with the following permanent configuration:

- CPIOM card type in slot A and B is cpm-sar-x
- IOM card type in slot 1 and 2 is iom-sar-x

Output example: Card information on the 7705 SAR-10x

```
=====
Card Summary
=====
```

Slot	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Comments
1	iom-sar-x	up	up	
2	iom-sar-x (not equipped)	up	provisioned	
A	cpiom-sar-x	up	up/active	
B	cpiom-sar-x (not equipped)	up	down/standby	

```
=====
```

2.1.2 7705 SAR-Hx and 7705 SAR-Mx

On the 7705 SAR-Hx and 7705 SAR-Mx, the system hardware is permanently built into the chassis and has preassigned parameters, with the exception of MDA slot 1, which supports removable MDAs.

The IOM of the 7705 SAR-Hx and 7705 SAR-Mx is a virtual entity, defined in software. The I/O ports are grouped and virtualized into slots (MDAs) for convenience of assignment. As a result of this design, the CLI identifiers for the IOM and MDAs are preset.

The 7705 SAR-Hx and 7705 SAR-Mx replace the CPM with an integrated control and switching functional block that does not need to be provisioned. It is displayed in the CLI as CPM A.

The 7705 SAR-Hx and 7705 SAR-Mx chassis support the following removable MDAs in slot 1:

- 7705 SAR-Hx and 7705 SAR-Mx Ethernet MDAs

- 8-port RS-232 and C37.94 Combination MDA
- 8-port T1/E1 MDA

See the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000x TQZZA for a list of supported MDA types. See the *7705 SAR-Hx, SAR-Mx, and SAR-10x MDA Installation Guide* for more information about the installation of 7705 SAR-Hx and 7705 SAR-Mx MDAs.

MDA slot 1 must be provisioned on the 7705 SAR-Hx and 7705 SAR-Mx. The MDA type must be specified to provision a chassis slot. A preprovisioned slot can remain empty without conflicting with populated slots.

MDA slots 3 and 4 are reserved for virtual ISA MDAs.

The 7705 SAR-Hx and 7705 SAR-Mx are preprovisioned from the factory with the following permanent configuration:

- CPM card type in slot A is cpm-sar-1x
- IOM card type in slot 1 is iom-sar-1x
- MDA type in slot 1/2 is m4-1g-rj+6-10g-sfp++2-25g-sfp28

Output example: Card information on the 7705 SAR-Hx and 7705 SAR-Mx

```
=====
Card State
=====
```

Slot/ Id	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Num Ports	Num MDA	Comments
1	iom-sar-1x	up	up		4	
1/1	(not provisioned)	up	unprovisioned			
1/2	m2-1g-sfp+2-10g-sfp+					
1/3	m4-1g-rj+6-10g-sfp++2-25g-sfp28	up	up	12		
1/4	(not provisioned)	up	unprovisioned	12		
	isa-ms-v					
1/4	(not provisioned)	up	unprovisioned	12		
	isa-ms-v					
A	cpm-sar-1x	up	up			Active

```
=====
```

2.1.3 7705 SAR-1

The 7705 SAR-1 is a fixed chassis; that is, the system hardware is permanently built into the chassis and has preassigned parameters. The IOM of the 7705 SAR-1 is a virtual entity, defined in software. The I/O ports are grouped and virtualized into slots (MDAs) for convenience of assignment. As a result of this design, the CLI identifiers for the IOM and MDAs are preset. The fixed configuration of the 7705 SAR-1 restricts the router to port-level provisioning; however, the IOM and MDA numbering must still be specified in CLI commands.

The 7705 SAR-1 replaces the CPM with an integrated control and switching functional block that does not need to be provisioned. It is displayed in the CLI as CPM A.

The 7705 SAR-1 is provisioned at the factory with the following permanent configuration:

- CPM card type in slot A is cpm-sar
- IOM card type in slot 1 is iom-sar
- MDA types:

- Slot 1/1 is m10-sfp++6-sfp
- Slot 1/2 is isa-tunnel-v
- Slot 1/3 is isa-bb-v

Output example: Card information on the 7705 SAR-1

```
=====
Card State
=====
```

Slot/ Id	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Num Ports	Num MDA	Comments
1	iom-sar	up	up		3	
1/1	m10-sfp++6-sfp	up	up	16		
1/2	isa-tunnel-v	up	up	2		
	isa-ms-v					
1/3	isa-bb-v	up	up	7		
	isa-ms-v					
A	cpm-sar	up	up			Active

```
=====
```

2.2 MDAs

MDAs are pluggable adapter cards that provide physical interface connectivity and are available in a range of interface and density configurations. MDA modules differ by chassis. See the applicable chassis installation guides and the corresponding MDA installation guides for more information about specific MDAs.

The 7705 SAR-10x supports removable MDAs in slots 1 through 8. Most 7705 SAR-10x MDAs can be provisioned in any MDA slot. To provision an MDA on the 7705 SAR-10x, the card slot value must be 1; the card type value cannot be configured.

The 7705 SAR-Hx and 7705 SAR-Mx support removable MDAs only in slot 1. For MDAs in slot 2 to 4, the software internally treats the MDA as a logical entity to represent the ports on the chassis. The MDA type is autoprovisioned on startup.

The 7705 SAR-1 does not support removable MDAs. The software internally treats the MDA as a logical entity to represent the ports on the chassis. The MDA type is autoprovisioned on startup.

In all cases, the card slot and IOM card type must be provisioned before an MDA can be provisioned. A preprovisioned MDA slot can remain empty without interfering with services on populated equipment. When an MDA is installed and enabled, the system verifies that the MDA type matches the provisioned type. If the command options do not match, the MDA remains offline.

3 Digital Diagnostics Monitoring

Some Nokia SFPs, XFPs, QSFPs, CFPs and the MSA DWDM transponder have the Digital Diagnostics Monitoring (DDM) capability where the transceiver module maintains information about its working status in device registers including:

- temperature
- supply voltage
- transmit (TX) bias current
- TX output power
- received (RX) optical power

For QSFPs and CFPs, DDM Temperature and Supply voltage is available only at the Module level as shown in [Table 3: DDM alarms and warnings](#).

See the [Statistics collection](#) section for details about the QSFP and CFP example DDM and DDM Lane information.

For the QSFPs and CFPs, the number of lanes is indicated by DDM attribute "Number of Lanes: 4".

Subsequently, each lane threshold and measured values are shown per lane.

If a lane entry is not supported by the specific QSFP or CFP specific model, then it is shown as "-" in the entry.

Use the following command to show QSFP and CFP lane information.

```
show port port-id detail
```

Output example

```
...
Transceiver Data
Transceiver Type   : QSFP+
Model Number      : 3HE06485AAAA01  ALU  IPUIBM3AA
TX Laser Wavelength: 1310 nm                      Diag Capable      : yes
Number of Lanes   : 4
Connector Code    : LC                               Vendor OUI          : e4:25:e9
Manufacture date  : 2012/02/02                      Media              : Ethernet
Serial Number     : 12050188
Part Number      : DF40GELR411102A
Optical Compliance : 40GBASE-LR4
Link Length support: 10km for SMF
=====
Transceiver Digital Diagnostic Monitoring (DDM)
=====

```

	Value	High Alarm	High Warn	Low Warn	Low Alarm
Temperature (C)	+35.6	+75.0	+70.0	+0.0	-5.0
Supply Voltage (V)	3.23	3.60	3.50	3.10	3.00

```
=====
Transceiver Lane Digital Diagnostic Monitoring (DDM)
=====

```

	High Alarm	High Warn	Low Warn	Low Alarm

Lane Tx Bias Current (mA)	78.0	75.0	25.0	20.0
Lane Rx Optical Pwr (avg dBm)	2.30	2.00	-11.02	-13.01

Lane ID Temp(C)/Alm	Tx Bias(mA)/Alm	Tx Pwr(dBm)/Alm	Rx Pwr(dBm)/Alm	

1	-	43.5	-	0.42
2	-	46.7	-	-0.38
3	-	37.3	-	0.55
4	-	42.0	-	-0.52
=====				
Transceiver Type : CFP				
Model Number : 3HE04821ABAA01 ALU IPUIBHJDAA				
TX Laser Wavelength: 1294 nm		Diag Capable : yes		
Number of Lanes : 4				
Connector Code : LC		Vendor OUI : 00:90:65		
Manufacture date : 2011/02/11		Media : Ethernet		
Serial Number : C22CQYR				
Part Number : FTLC1181RDNL-A5				
Optical Compliance : 100GBASE-LR4				
Link Length support: 10km for SMF				
=====				
Transceiver Digital Diagnostic Monitoring (DDM)				
=====				
	Value	High Alarm	High Warn	Low Warn Low Alarm

Temperature (C)	+48.2	+70.0	+68.0	+2.0 +0.0
Supply Voltage (V)	3.24	3.46	3.43	3.17 3.13
=====				
Transceiver Lane Digital Diagnostic Monitoring (DDM)				
=====				
		High Alarm	High Warn	Low Warn Low Alarm

Lane Temperature (C)		+55.0	+53.0	+27.0 +25.0
Lane Tx Bias Current (mA)		120.0	115.0	35.0 30.0
Lane Tx Output Power (dBm)		4.50	4.00	-3.80 -4.30
Lane Rx Optical Pwr (avg dBm)		4.50	4.00	-13.00 -16.00

Lane ID Temp(C)/Alm	Tx Bias(mA)/Alm	Tx Pwr(dBm)/Alm	Rx Pwr(dBm)/Alm	

1	+47.6	59.2	0.30	-10.67
2	+43.1	64.2	0.27	-10.31
3	+47.7	56.2	0.38	-10.58
4	+51.1	60.1	0.46	-10.37
=====				

The transceiver is programmed with warning and alarm thresholds for low and high conditions that can generate system events. These thresholds are programmed by the transceiver manufacturer.

There are no CLI commands required for DDM operations, however, the **show port port-id detail** command displays DDM information in the Transceiver Digital Diagnostics Monitoring output section.

DDM information is populated into the router's MIBs, so the DDM data can be retrieved by Network Management using SNMP. Also, RMON threshold monitoring can be configured for the DDM MIB variables to set custom event thresholds if the factory-programmed thresholds are not at the wanted levels.

The following are potential uses of the DDM data:

- **optics degradation monitoring**

With the information returned by the DDM-capable optics module, degradation in optical performance can be monitored and trigger events based on custom or the factory-programmed warning and alarm thresholds.

- **link or router fault isolation**

With the information returned by the DDM-capable optics module, any optical problem affecting a port can be quickly identified or eliminated as the potential problem source.

Supported real-time DDM features are summarized in the following table.

Table 2: Real-time DDM information

Fields	User units	SFP/XFP units	SFP	XFP	MSA DWDM
Temperature	Celsius	C	✓	✓	✓
Supply Voltage	Volts	μV	✓	✓	
TX Bias Current	mA	μA	✓	✓	✓
TX Output Power	dBm (converted from mW)	mW	✓	✓	✓
RX Received Optical Power4	dBm (converted from dBm) (Avg Rx Power or OMA)	mW	✓	✓	✓
AUX1	option dependent (embedded in transceiver)			✓	
AUX2	option dependent (embedded in transceiver)			✓	

The factory-programmed DDM alarms and warnings that are supported are summarized in the following table.

Table 3: DDM alarms and warnings

Alarms and Warnings	SFP/XFP units	SFP	XFP	Required?	MSA DWDM
Temperature - High Alarm - Low Alarm - High Warning - Low Warning	C	Yes	Yes	Yes	Yes
Supply Voltage - High Alarm	μV	Yes	Yes	Yes	No

Alarms and Warnings	SFP/XFP units	SFP	XFP	Required?	MSA DWDM
- Low Alarm - High Warning - Low Warning					
TX Bias Current - High Alarm - Low Alarm - High Warning - Low Warning	μ A	Yes	Yes	Yes	Yes
TX Output Power - High Alarm - Low Alarm - High Warning - Low Warning	mW	Yes	Yes	Yes	Yes
RX Optical Power - High Alarm - Low Alarm - High Warning - Low Warning	mW	Yes	Yes	Yes	Yes
AUX1 - High Alarm - Low Alarm - High Warning - Low Warning	option dependent (embedded in transceiver)	No	Yes	Yes	No
AUX2 - High Alarm - Low Alarm - High Warning - Low Warning	option dependent (embedded in transceiver)	No	Yes	Yes	No

3.1 SFPs and XFPs

The availability of the DDM real-time information and the warning and alarm status is based on the transceiver. It may or may not indicate that DDM is supported. Although some Nokia SFPs support DDM,

Nokia has not required DDM support in releases before Release 6.0. Non-DDM and DDM-supported SFPs are distinguished by a specific value in their EEPROM.

For SFPs that do not indicate DDM support in their EEPROM, DDM data is available although the accuracy of the information has not been validated or verified.

For non-Nokia transceivers, DDM information may be displayed, but Nokia is not responsible for formatting, accuracy, and so on.

3.2 Statistics collection

The DDM information and warnings and alarms are collected at one-minute intervals. As such, the minimum resolution for any DDM events when correlating with other system events is one minute.

In the Transceiver Digital Diagnostic Monitoring section of the **show port *port-id* detail** command output:

- If the present measured value is higher than either or both of the High Alarm and High Warn thresholds, an exclamation mark (!) displays along with the threshold value.
- If the present measured value is lower than either or both of the Low Alarm and Low Warn thresholds, an exclamation mark (!) displays along with the threshold value.

Use the following command to show Transceiver Digital Diagnostic Monitoring information.

```
show port 2/1/6 detail
```

Output example

```
...
=====
Transceiver Digital Diagnostic Monitoring (DDM)
=====
```

	Value	High Alarm	High Warn	Low Warn	Low Alarm
Temperature (C)	+33.0	+98.0	+88.0	-43.0	-45.0
Supply Voltage (V)	3.31	4.12	3.60	3.00	2.80

```
=====
Transceiver Lane Digital Diagnostic Monitoring (DDM)
=====
```

	High Alarm	High Warn	Low Warn	Low Alarm
...				
Lane Tx Bias Current (mA)	60.0	50.0	0.1	10.0
Lane Tx Output Power (dBm)	0.00	-2.00	-10.50	-12.50
Lane Rx Optical Pwr (avg dBm)	-3.00!	-4.00	-19.51	-20.51

```
-----
```

4 Ports

4.1 Port types

Before a port can be configured, the slot must be provisioned with a card type and MDA type.

7705 SAR Gen 2 routers support the following port types:

- **Ethernet**

Supported Ethernet port types include:

- Fast Ethernet (10/100BASE-T, 100BASE-FX)
- Gb Ethernet (1GbE, 1000BASE-T)
- 10 Gb Ethernet (10GbE, 10GBASE-X)
- 25 Gb Ethernet (25GBASE-R)

Router ports must be configured as either access, network, or hybrid. The default is network.

- **access**

Access ports are configured with services to carry customer-facing traffic. If a Service Access Port (SAP) is required on a port or channel, it must first be configured as an access port or channel. When a port is configured for access mode, the appropriate encapsulation type must be configured to distinguish the services on the port or channel. After a port has been configured for access mode, one or more services can be configured on the port or channel, depending on the encapsulation value.

- **network**

Network ports are configured for network-facing traffic. These ports participate in the service provider transport or infrastructure network. Dot1q is supported on network ports.

- **hybrid**

Hybrid ports are configured for access and network-facing traffic. While the default mode of an Ethernet port remains network, the port cannot change between access, network, and hybrid modes unless the port is first shut down and any configured SAPs or interfaces are deleted.

Hybrid ports can operate in both access and network modes. The MTU of a port in hybrid mode is the same as in network mode. The default encapsulation for hybrid port mode is dot1q; the port also supports QinQ encapsulation. Null hybrid port mode is not supported. After the port is changed to hybrid, the default MTU is changed to 9212 bytes, the value currently used in network mode (higher than an access port). This ensures that both SAP and network VLANs are accommodated. The only exception applies to 10/100 Fast Ethernet ports operating in hybrid mode. In those cases, this is set to 1522 bytes, which corresponds to the default access MTU with QinQ, and is larger than the network dot1q MTU or the access dot1q MTU for this type of Ethernet port.

All command options in the **access** and **network** contexts can be configured within the port using the same CLI hierarchy as in the existing implementation. The difference is that a port configured in hybrid mode allows both ingress and egress contexts to be configured concurrently.

An Ethernet port configured in hybrid mode can have two encapsulation types: dot1q and QinQ. The NULL value is not supported because a single SAP is allowed. This can be achieved by configuring the

port in access mode, or by allowing a single network IP interface achieved by configuring the port in network mode.

Hybrid mode can be enabled on a LAG port when the port is part of a single-chassis LAG configuration. When the port is part of a multichassis LAG configuration, it can only be configured to access mode because MC-LAG is not supported on network ports and, consequently, not supported on hybrid ports. The same restriction applies to a port that is part of an MC-Ring configuration.

For a hybrid port, use the following commands to split the amount of allocated port buffers for ingress and egress equally between network and access contexts:

- **MD-CLI**

```
configure port hybrid-buffer-allocation ingress-weight access network
configure port hybrid-buffer-allocation egress-weight access network
```

- **classic CLI**

```
configure port hybrid-buffer-allocation ing-weight access network
configure port hybrid-buffer-allocation egr-weight access network
```

Adapting the terminology in buffer-pools, the access active bandwidth and network active bandwidth of the port are derived in each ingress and egress as follows (egress formulas shown only):

- $\text{total-hybrid-port-egress-weights} = \text{access-weight} + \text{network-weight}$
- $\text{hybrid-port-access-egress-factor} = \text{access-weight} / \text{total-hybrid-port-egress-weights}$
- $\text{hybrid-port-network-egress-factor} = \text{network-weight} / \text{total-hybrid-port-egress-weights}$
- $\text{port-access-active-egress-bandwidth} = \text{port-active-egress-bandwidth} \times \text{hybrid-port-access-egress-factor}$
- $\text{port-network-active-egress-bandwidth} = \text{port-active-egress-bandwidth} \times \text{hybrid-port-network-egress-factor}$

- **GNSS receiver**

The 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx are equipped with an integrated Global Navigation Satellite System (GNSS) receiver and GNSS radio frequency (RF) port for retrieval and recovery of GPS and Galileo signals.



Note: Signal recovery must always be enabled in the system configuration when the GNSS receiver is used.

See the *7705 SAR Gen 2 Basic System Configuration Guide* for information about using a GNSS receiver as a timing source for the node.

- **WAN PHY**

25G Ethernet ports on the 7705 SAR-Hx and 7705 SAR-Mx can be configured in WAN PHY mode. Use commands in the following context to configure 25G Ethernet ports in WAN PHY mode.

```
configure port ethernet xgig
```

When configuring the port in WAN mode, you can change specific SONET/SDH command options to reflect the SONET/SDH requirements for this port.

- **TDM**

Time-Division Multiplexing (TDM) ports support access mode only. Supported TDM port types on the 7705 SAR-10x, 7705 SAR-Hx and 7705 SAR-Mx include:

- T1
- E1
- C37.94

- **LAG**

A Link Aggregation Group (LAG) can be used to group multiple ports into one logical link. The aggregation of multiple physical links enables load sharing and provides seamless redundancy. If one link fails, traffic is redistributed over the remaining links.

- **connector**

A connector that can accept transceiver modules, including breakout connectors to multiple physical ports. For example, an SFP28 connector can support 1Gb, 10 Gb, and 25 Gb Ethernet ports, depending on the port speed mode configuration. The connectors themselves cannot be used as ports in other commands; however, the breakout ports can be used as any Ethernet port.

See [Ethernet ports](#) for more information.

4.1.1 Ethernet ports

This section provides information about the supported Ethernet ports and pluggable transceiver types for 7705 SAR Gen 2 platforms.

The following table lists the supported speeds for pluggable transceiver types.

Table 4: Supported speeds for pluggable transceivers

Pluggable transceiver type	Supported speeds
SFP	SFP transceivers support speeds of 1 Gb/s, unless 100 Mb/s support is noted in other table footnotes.
SFP+	SFP+ transceivers support speeds of 10 Gb/s.
SFP28	SFP28 transceivers support speeds of 25 Gb/s.

The following table lists the supported port types and pluggable transceivers for 7705 SAR Gen 2 platforms.

Table 5: Supported Ethernet port and pluggable transceiver types

7705 SAR Gen 2 hardware	Supported platforms	Physical port type	Accepted pluggable transceivers		
			SFP ¹	SFP+	SFP28
m10-sfp++6-sfp	7705 SAR-1	SFP	✓	—	—

¹ SFP ports support speeds of 100 Mb/s or 1 Gb/s. However these ports only advertise their configured speed when **autonegotiate** or **autonegotiate limited** is configured.

7705 SAR Gen 2 hardware	Supported platforms	Physical port type	Accepted pluggable transceivers		
			SFP ¹	SFP+	SFP28
		SFP+	✓	✓	—
m4-1g-rj+6-10g-sfp++2-25g-sfp28	7705 SAR-Hx and 7705 SAR-Mx	SFP+	✓	✓	—
		SFP28	✓	✓	✓
		RJ45 ²	—	—	—
m2-1g-sfp+2-10g-sfp+	7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx	SFP	✓	—	—
		SFP+	✓	✓	—
m8-1g-sfp	7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx	SFP	✓	—	—
m8-1g-rj45 ²	7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx	RJ45	—	—	—

4.1.1.1 Port speeds on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx

The following table lists the supported port speeds on 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx MDAs.



Note: The following features are not supported on fiber SFP ports with 100M speeds:

- MACSec
- Precision Time Protocol (PTP)
- Synchronous Ethernet (SyncE)



Note: On the 7705 SAR Gen 2, 10M port speeds can operate only in full duplex mode.



Note: Jumbo frames (greater than 1500 bytes) are not supported on 10M Ethernet ports on 7705 SAR Gen 2 platforms.

¹ SFP ports support speeds of 100 Mb/s or 1 Gb/s. However these ports only advertise their configured speed when **autonegotiate** or **autonegotiate limited** is configured.

² RJ45 ports support 1000 Mb/s (1 Gb/s), 100 Mb/s, and 10 Mb/s port speeds. These ports do not accept pluggable transceivers.

Table 6: Ethernet port speeds

MDA	Port or pluggable transceiver type	Port number	Supported speeds
m4-1g-rj+6-10g-sfp++2-25g-sfp28	RJ45	1 to 4	10M/100M/1000M
	SFP	c5 to c12	100M/1G ^{3 4}
	SFP+		10G
	SFP28	c11 to c12	25G
m2-1g-sfp+2-10g-sfp+	SFP	c1 to c4	100M/1G ^{5 6}
	SFP+	c3 to c4	10G
m8-1g-sfp	SFP	c1 to c8	10M/100M/1G ^{7 8}
m8-1g-rj45	RJ45	1 to 8	10M/100M/1000M

4.1.1.2 Port speed modes

The 7705 SAR-10x, 7705 SAR-Hx and 7705 SAR-Mx support speed modes to meet the requirements of different deployment models.

Use the following command to configure the speed mode.

- **MD-CLI**

```
admin system port-speed-mode mode
```

- **classic CLI**

```
admin system port-speed-mode
```



Note: The router must be rebooted after changing the speed mode.

On the 7705 SAR-10x, the speed modes allow the SFP interfaces in MDA slots 1, 3, 5, and 6 to be configured as a combination of 1G, 10G, and 25G speeds when provisioned with 25G-capable MDAs.

³ Fiber SFPs are supported on ports 1/2/c5 to 1/2/c12 at 100M and 1G speeds. The connector must be configured as **c1-1g**.

⁴ Copper SFPs are supported on ports 1/2/c5 to 1/2/c12 at 1G speed only.

⁵ Fiber SFPs are supported on ports 1/1/c1 to 1/1/c4 at 100M speed. The connector must be configured as **c1-1g**.

⁶ Copper SFPs are supported on ports 1/1/c1 to 1/1/c4 at 1G speed only.

⁷ Fiber SFPs are supported on ports 1/1/c1 to 1/1/c8 at 100M speed. The connector must be configured as **c1-1g**.

⁸ Copper SFPs are supported on ports 1/1/c1 to 1/1/c8 at 10M, 100M, and 1G speeds.



Caution: On the 7705 SAR-10x with redundant CPIOMs, ensure that both the active and standby CPIOMs are fully up before switching the port speed mode. If the user attempts to switch the port speed mode while one of the CPIOMs is not fully up (for example, when the CPIOM is rebooting after a **force-switchover**), the port speed mode configuration may fail.

The following tables list the supported port speeds on the 7705 SAR-10x.

Table 7: Port speed modes on the 7705 SAR-10x

Speed mode	Port and port type			
	1/1/c1 to 1/1/c2 1/3/c1 to 1/3/c2	1/1/c3 to 1/1/c4 1/3/c3 to 1/3/c4	1/5/c1	1/6/c1
	SFP	SFP, SFP+, or SFP28	SFP, SFP+, or SFP28	SFP, SFP+, or SFP28
Speed mode 1 1G/10G	1G	1G/10G	1G/10G	1G/10G
Speed mode 2 ⁹ 10G/25G + 1G	1G	10G/25G	10G/25G	10G/25G

On the 7705 SAR-Hx and 7705 SAR-Mx, the speed modes allow the SFP interfaces on ports 1/2/c5 to 1/2/c12 to be configured for either 1G/10G or 10G/25G speeds.



Note: See [Port speeds on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx](#) for restrictions on using 100M port speeds.

The following tables list the supported port speeds on the 7705 SAR-Hx and 7705 SAR-Mx.

Table 8: Port speed modes on the 7705 SAR-Hx and 7705 SAR-Mx

Speed mode	Port and port type	
	1/2/c5 to 1/2/c10	1/2/c11 to 1/2/c12
	SFP or SFP+	SFP, SFP+, or SFP28
Speed mode 1 1G/10G	100M/1G/10G	1G/10G
Speed mode 2 10G/25G	10G	10G/25G

4.1.2 TDM ports

TDM ports are supported on the following MDAs:

⁹ This speed mode also restricts provisioning TDM MDAs to slots 2 and 4 only.

- 8-port T1/E1 MDA
- 8-port RS-232 and C37.94 combination MDA
- 8-port C37.94 Teleprotection card

4.1.2.1 8-port T1/E1 MDA

T1/E1 port configuration is available on the 8-port T1/E1 MDA supported on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx.

On T1/E1 ports, channelization is supported down to the DS0 level. To change port types, all ports must first be administratively disabled. The ports can be configured for T1 and E1 operations. When the first port on a card is configured, all other ports on the card must be set to the same type.

T1/E1 ports support only **cem** encapsulation.

4.1.2.2 8-port RS-232 and C37.94 combination MDA

This MDA is supported on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx.

On the 8-port RS-232 and C37.94 combination MDA, channelization is supported down to the DS0 level on the four RS-232 serial ports and four IEEE C37.94 teleprotection interface (TPIF) ports.

The RS-232 and C37.94 ports are used in critical networks for applications like SCADA and teleprotection.

4.1.2.3 8-port C37.94 Teleprotection card

This MDA is supported on the 7705 SAR-10x. Channelization is supported down to the DS0 level.

This MDA supports the same functionality as the 8-port RS-232 and C37.94 Combination MDA, except all 8 ports are for C37.94. This MDA is ideal for installations that require high C37.94 port density.

4.2 Port features

4.2.1 Port State and Operational State

There are two port attributes that are related and similar but have slightly different meanings: Port State and Operational State (or Operational Status).

The following descriptions are based on normal individual ports. Many of the same concepts apply to other objects that are modeled as ports in the router such as APS groups but the show output descriptions for these objects should be consulted for the details.

- **Port State**
 - Displayed in port summaries such as **show port** or **show port 1/1**
 - `tmnxPortState` in the TIMETRA-PORT-MIB
 - Values: None, Ghost, Down (linkDown), Link Up, Up
- **Operational State**

- Displayed in the show output of a specific port such as **show port 2/1/3**
- `tmnxPortOperStatus` in the TIMETRA-PORT-MIB
- Values: Up (inService), Down (outOfService)

The behavior of Port State and Operational State are different for a port with link protocols configured (for example, LACP for Ethernet ports). A port with link protocols configured only transitions to the Up Port State when the physical link is up and all the configured protocols are up. A port with no link protocols configured transitions from Down to Link Up and then to Up immediately after the physical link layer is up.

The linkDown and linkUp log events (events 2004 and 2005 in the SNMP application group) are associated with transitions of the port Operational State. Note that these events map to the RFC 2863, *The Interfaces Group MIB*, (which obsoletes RFC 2233, *The Interfaces Group MIB using SMIv2*) linkDown and linkUp traps as mentioned in the SNMPv2-MIB.

An Operational State of Up indicates that the port is ready to transmit service traffic (the port is physically up and any configured link protocols are up). The relationship between port Operational State and Port State is shown in [Table 9: Relationship of Port state and Oper state](#).

Table 9: Relationship of Port state and Oper state

Port state	Operational state (Oper state or Oper status) (as displayed in "show port x/y/z")	
Port State (as displayed in the show port summary)	For ports that have no link layer protocols configured	For ports that have link layer protocols configured (PPP, LACP, 802.3ah EFM, 802.1ag Eth-CFM)
Up	Up	Up
Link Up (indicates the physical link is ready)	Up	Down
Down	Down	Down

4.2.2 802.1x network access control

Nokia routers support network access control of client devices (PCs, STBs, and so on) on an Ethernet network using the IEEE 802.1x standard. 802.1x is known as Extensible Authentication Protocol (EAP) over a LAN network or EAPOL.

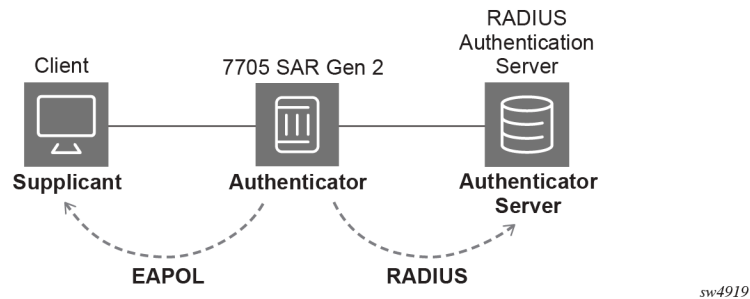
4.2.2.1 802.1x basics

The IEEE 802.1x standard defines three participants in an authentication conversation (see [Figure 1: 802.1x architecture](#) that shows an example with the 7705 SAR Gen 2).

the supplicant	the end-user device that requests access to the network
the authenticator	controls access to the network. Both the supplicant and the authenticator are referred to as Port Authentication Entities (PAEs).

the authentication server performs the actual processing of the user information

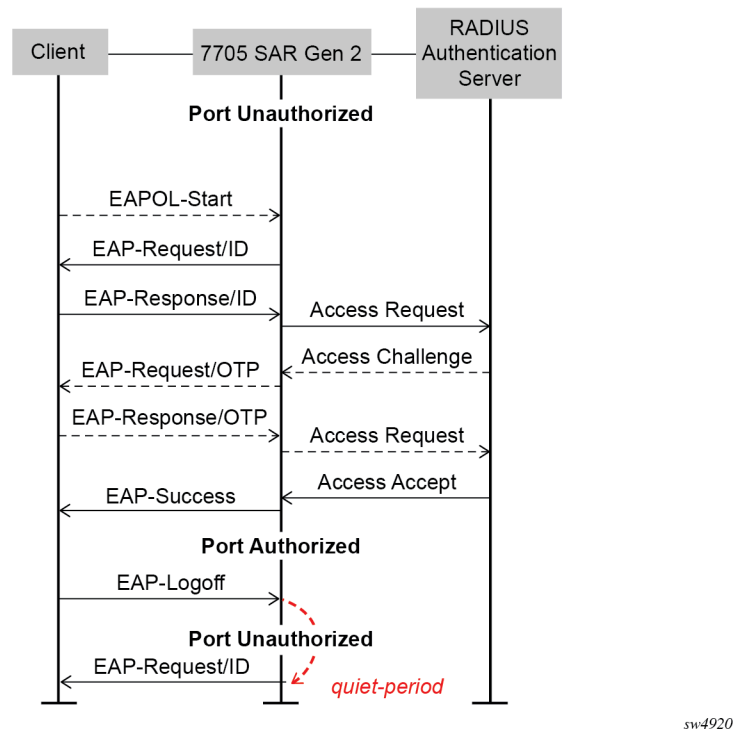
Figure 1: 802.1x architecture



The authentication exchange is carried out between the supplicant and the authentication server, the authenticator acts only as a bridge. The communication between the supplicant and the authenticator is done through the Extended Authentication Protocol (EAP) over LANs (EAPOL). On the back end, the communication between the authenticator and the authentication server is done with the RADIUS protocol. The authenticator is therefore a RADIUS client, and the authentication server a RADIUS server.

The messages involved in the authentication procedure are shown in [Figure 2: 802.1x authentication scenario](#). The router initiates the procedure when the Ethernet port becomes operationally up, by sending a special PDU called EAP-Request/ID to the client. The client can also initiate the exchange by sending an EAPOL-start PDU, if it does not receive the EAP-Request/ID frame during bootup. The client responds on the EAP-Request/ID with a EAP-Response/ID frame, containing its identity (typically username + password).

Figure 2: 802.1x authentication scenario



After receiving the EAP-Response/ID frame, the router encapsulates the identity information into a RADIUS AccessRequest packet, and sends it off to the configured RADIUS server.

The RADIUS server checks the supplied credentials, and if approved returns an Access Accept message to the router. The router notifies the client with an EAP-Success PDU and puts the port in authorized state.

4.2.2.2 802.1x modes

Nokia routers support port-based network access control for Ethernet ports only. Every Ethernet port can be configured to operate in one of three different operation modes, controlled by the **port-control** command:

- **force authorized**

This mode disables 802.1x authentication and causes the port to transition to the authorized state without requiring any authentication exchange. The port transmits and receives normal traffic without requiring 802.1x-based host authentication. This is the default setting.

- **force unauthorized**

This mode causes the port to remain in the unauthorized state, ignoring all attempts by the hosts to authenticate. The switch cannot provide authentication services to the host through the interface.

- **auto**

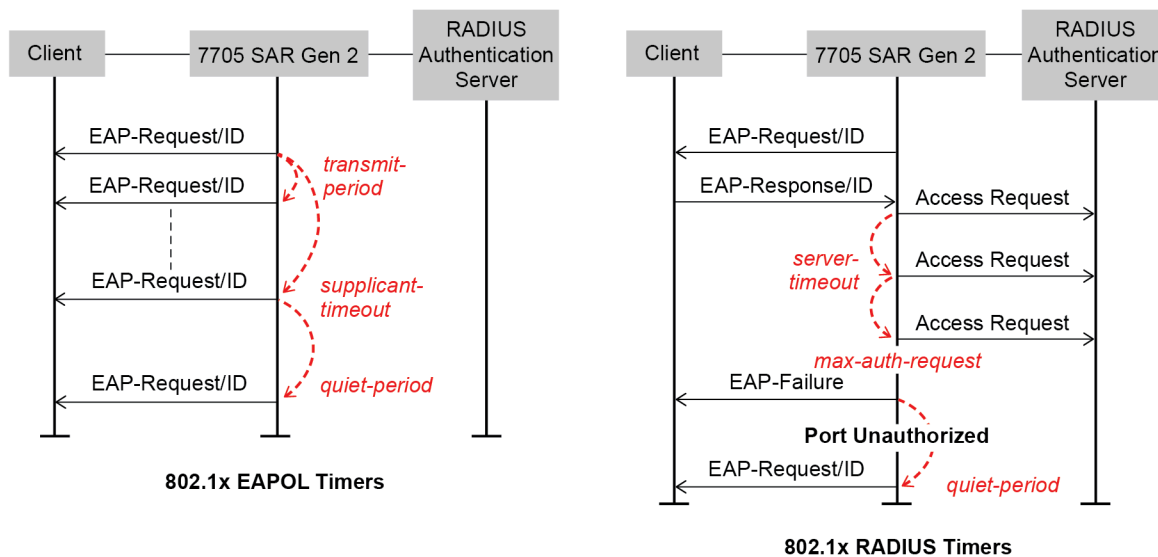
This mode enables 802.1x authentication. The port starts in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. Both the router and the host can initiate an authentication procedure as described below. The port remains in unauthorized state (no traffic except

EAPOL frames is allowed) until the first client is authenticated successfully. After this, traffic is allowed on the port for all connected hosts.

4.2.2.3 802.1x timers

The 802.1x authentication procedure is controlled by a number of configurable timers and scalars. There are two separate sets, one for the EAPOL message exchange and one for the RADIUS message exchange. See [Figure 3: 802.1x EAPOL timers \(left\) and RADIUS timers \(right\)](#) for an example of the timers on the 7705 SAR Gen 2.

Figure 3: 802.1x EAPOL timers (left) and RADIUS timers (right)



sw4921

EAPOL timers:

- **transmit-period**

This timer indicates how many seconds the Authenticator listens for an EAP-Response/ID frame. If the timer expires, a new EAP-Request/ID frame is sent and the timer restarted. The default value is 60. The range is 1 to 3600 seconds.

- **supplicant-timeout**

This timer is started at the beginning of a new authentication procedure (transmission of first EAP-Request/ID frame). If the timer expires before an EAP-Response/ID frame is received, the 802.1x authentication session is considered as having failed. The default value is 30. The range is 1 to 300.

- **quiet-period**

This timer indicates number of seconds between authentication sessions. It is started after logout, after sending an EAP-Failure message or after expiry of the supplicant-timeout timer. The default value is 60. The range is 1 to 3600.

RADIUS timer and scalar:

- **maximum authentication requests**

This scalar indicates the maximum number of times that the router sends an authentication request to the RADIUS server before the procedure is considered as having failed. The default value is value 2. The range is 1 to 10.

- **server timeout**

This timer indicates how many seconds the authenticator waits for a RADIUS response message. If the timer expires, the access request message is sent again, up to maximum authentication request times. The default value is 60. The range is 1 to 3600 seconds.

The router can also be configured to periodically trigger the authentication procedure automatically. This is controlled by enabling re-authentication and re-authentication period. Re-authentication period indicates the period in seconds (since the last time that the authorization state was confirmed) before a new authentication procedure is started. The range of reauth-period is 1 to 9000 seconds (the default is 3600 seconds, one hour). Note that the port stays in an authorized state during the re-authentication procedure.

4.2.2.4 802.1x tunneling

Tunneling of untagged 802.1x frames received on a port is supported for both Epipe and VPLS service using either null or default SAPs (for example 1/1/1.*) when the following command is configured:

- **MD-CLI**

```
configure port ethernet dot1x port-control force-authorized
```

- **classic CLI**

```
configure port ethernet dot1x port-control force-auth
```

When tunneling is enabled on a port, untagged 802.1x frames are treated like user frames and are switched into Epipe or VPLS services which have a corresponding null SAP or default SAP on that port. In the case of a default SAP, it is possible that other non-default SAPs are also present on the port. Untagged 802.1x frames received on other service types, or on network ports, are dropped. Use the following command to enable tunneling on a port.

```
configure port port-id ethernet dot1x tunneling
```

When tunneling is required, it is expected that it is enabled on all ports into which 802.1x frames are to be received. The configuration of dot1x must be configured consistently across all ports in LAG as this is not enforced by the system.

Note that 802.1x frames are treated like user frames, that is, tunneled, by default when received on a spoke or mesh SDP.

4.2.2.5 Per-host authentication

Per-host authentication enables SR OS to authenticate each host individually and allows or disallows the PDUs from this host through the port. Per-host authentication is configurable using the CLI.

When dot1x tunneling is disabled, the port does not allow any PDUs to pass through, with the exception of dot1x packets, which are extracted.

When **per-host-authentication** is configured on the port for dot1x, each host is authenticated individually according to the RADIUS policy and host traffic is allowed or disallowed through the port. After the first successful host authentication, the behavior is the following:

- On downstream (that is, traffic from the network to the host), the port is authorized and allows all traffic to go through.
- On upstream (that is, traffic from the host to the network), the port is authorized, but allows through traffic from the authenticated hosts only. When the host is allowed through the port, all PDUs for that host are allowed to pass through the port, including untagged or tagged packets. The traffic from any unauthenticated host is disallowed.

For per-host authentication, EAPOL packets are sent to the RADIUS server using the RADIUS protocol. The calling station identifier is the source MAC address of the host and is usually present in the packet. The identifier is used to allow or disallow the host source MAC address based on the RADIUS success or failure answer.

The hosts are authenticated periodically. If a host is authenticated and placed on the allow list and a subsequent authentication fails, that host is removed from the allow list.

If a host authenticates unsuccessfully multiple times, that host is put on a disallow list for a specific amount of time. That is, enabling per-host authentication provides per-host (source MAC) DoS mitigation.

Duplicate MAC addresses are not allowed on the port.

All logs display per-host authentication.

4.2.2.5.1 Per-host authentication interaction with dot1x

When per-host authentication is first enabled, all MAC addresses on the port are denied. The user can allow MAC addresses using the static source MAC or dot1x host authentication. The following considerations apply when dot1x authentication is used.

- If the 802.1x authentication mode is configured as force authorized, any host that sends EAPOL frames is authenticated without requiring any exchange with the RADIUS server. Use the following command to configure force authorized:

- **MD-CLI**

```
configure port ethernet dot1x port-control force-authorized
```

- **classic CLI**

```
configure port ethernet dot1x port-control force-auth
```

- If **configure system security dot1x** is administratively disabled, the port behavior is the same as in the force authorized case:

- **MD-CLI**

```
configure port ethernet dot1x admin-state disable
```

- **classic CLI**

```
configure port ethernet dot1x shutdown
```

- If the 802.1x authentication mode is configured as **auto**, the hosts are authenticated using RADIUS. However, if **configure system security dot1x** is administratively disabled, the force authorized behavior takes effect.

4.2.2.5.2 Static allow source MAC

A host can be added to the Allow MAC list statically, without being authenticated using dot1x. In this case, the host source MAC address must be added manually using the CLI.

If the same host is added to the list using dot1x and the CLI, the static configuration takes precedence. If the host is added using the CLI, the host is placed on the Allow list. If the same host tries to authenticate using RADIUS and the authentication fails, the host is still allowed through the port because it was statically added using the following command.

```
configure port ethernet dot1x per-host-authentication allowed-source-macs mac-address
```

4.2.2.5.3 Tagged dot1x authentication

Dot1x packets can arrive tagged or untagged on the authenticator port from the host. SR OS can be configured to tunnel or extract tagged dot1x packets. SR OS forwards tagged dot1x packets only.

The tunneling or extracting of tagged dot1x packets can be enabled for dot1q (**tunnel-dot1q**) and QinQ (**tunnel-qinq**) encapsulation types.

Each of the encapsulation types configured on the port can be configured to tunnel dot1x packets or extract dot1x packets to be authenticated using a configured RADIUS policy.

The extraction or tunneling of tagged packets applies to any tag value.

4.2.2.5.4 Dot1x and LAG

For dot1x authentication support, when the primary port member of the LAG is configured with dot1x, all members inherit the dot1x functionality. Dot1x packets can be extracted on any LAG member and sent to the RADIUS server for processing and authentication. After a successful authentication, the host is allowed on all LAG members. The host dot1x packets can be extracted on one LAG member, while the actual traffic traverses another LAG member. The following is the behavior of dot1x in a LAG bundle.

- When ports are added to the LAG member and dot1x is enabled, all ports inherit the same dot1x configuration as the primary port on the LAG member.
- If a host source address (SA) is authenticated through one of the LAG member ports, all ports on the LAG bundle are authorized and pass traffic.
- When a new port is added to the LAG member, if the LAG bundle has been authenticated and is authorized, the new member is authorized as well.
- Dot1x configuration changes are allowed on the primary LAG member only. A port can be added to a LAG only if its dot1x configuration aligns with that of the primary LAG member. If at least one LAG member is authorized, all LAG members are authorized.

In an upgrade scenario, when an older configuration file (**admin save**) is executed on a new release, a warning is displayed instead of an error for a command that violates the dot1x configuration change behavior; the violating command is ignored.

- If a port is removed from the LAG bundle, the port becomes unauthorized and the EAP negotiation should authorize the port again. This is true for all ports in the LAG bundle, primary or not.
- When Random Early Discard (RED) updates are received during an ISSU on a LAG member in standby, the following updates are ignored:
 - enable dot1x on a LAG member
 - authorize a LAG member

When a port is added to a LAG during ISSU, its dot1x configuration is reset to the default values.

4.2.2.5.5 SR host authentication behavior

SR allows the same MAC source address (MAC SA) on different ports if the MAC address is authenticated. Multiple hosts with the same MAC address can reside and get authenticated on different ports.

4.2.2.5.6 Authentication lists

The following authentication lists are supported:

- **authenticated host list**

This list contains up to 1000 hosts. Only hosts that have been authenticated through RADIUS and are allowed through the port are included in this list.

- **unauthenticated host list**

This list contains up to 2000 hosts. Only hosts that have failed authentication or are in the process of being authenticated are included in this list.

If this list reaches the 2000-host limit and a new host is being authenticated, the new host bumps off the list the first host that has failed authentication. The following sequence shows an example:

Example

Unauthenticated list

Host 1 authenticating

Host 2 failed authentication

...

Host 2000 authenticating

Host 2001 just arrived, this host should bump Host 2 off in the list, not Host 1.

If all hosts are in authenticating state, the new Host 2001 is not allowed on the list.

4.2.2.6 802.1x configuration and limitations

Configuration of 802.1x network access control on the router consists of two parts:

- generic command options, which are configured under **configure system security dot1x**
- port-specific command options, which are configured under **configure port ethernet dot1x**

The following considerations apply:

- If per-host authentication is not configured, the authentication of any host on the port provides access to the port for any device, even if only a single client has been authenticated.
- 802.1x authentication can only be used to gain access to a pre-defined Service Access Point (SAP). It is not possible to dynamically select a service (such as a VPLS service) depending on the 802.1x authentication information.
- If 802.1x access control is enabled and a high rate of 802.1x frames are received on a port, that port is blocked for a period of 5 minutes as a DoS protection mechanism.

4.2.2.6.1 Disabling the 802.1x functionality on a port

By default, the 802.1x functionality consisting of packet extraction and processing on the CPM is enabled on each port.

Use the following command to administratively disable the 802.1x functionality on a port by not extracting the dot1x packets to the CPM.

- **MD-CLI**

```
configure port ethernet dot1x admin-state disable
```

- **classic CLI**

```
configure port ethernet dot1x shutdown
```

4.2.3 MACsec

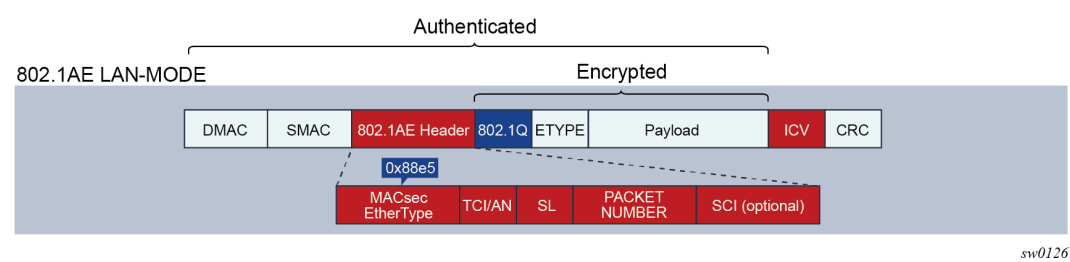
Media Access Control Security (MACsec) is an industry-standard security technology that provides secure communication for almost all types of traffic on Ethernet links. MACsec provides point-to-point and point-to-multipoint security on Ethernet links between directly connected nodes or nodes connected via a Layer 2 cloud. MACsec can identify and prevent most security threats, including:

- denial of service
- intrusion
- man-in-the-middle
- masquerading
- passive wiretapping
- playback attacks

MACsec Layer 2 encryption is standardized in IEEE 802.1AE. MACsec encrypts anything from the 802.1AE header to the end of the payload, including 802.1Q; it leaves the DMAC and SMAC in cleartext.

The following figure shows the 802.1AE LAN-Mode structure.

Figure 4: 802.1 AE LAN-MODE



The destination MAC address, which is in cleartext, is used for MACsec packet forwarding.

4.2.3.1 MACsec 802.1AE header — security TAG

The MACsec 802.1AE header includes a security TAG (SecTAG) field that contains the following information:

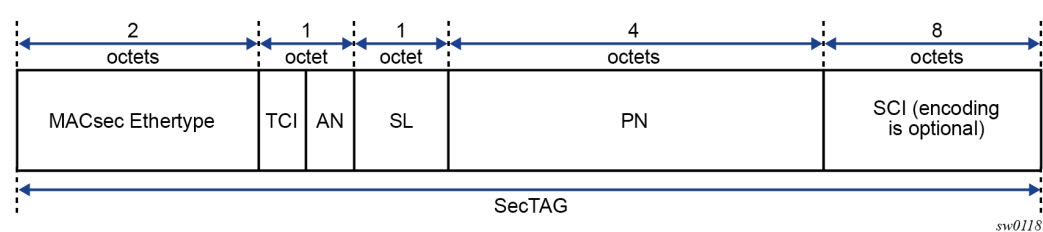
- association number within the channel
- packet number to provide a unique initialization vector for encryption and authentication algorithms, as well as protection against replay attacks
- optional LAN-wide secure channel identifier

The SecTAG field, which is identified by the MACsec EtherType, conveys the following information:

- TAG Control Information (TCI)
- Association Number (AN)
- Short Length (SL)
- Packet Number (PN)
- Optionally-encoded Secure Channel Identifier (SCI)

The following figure shows the format of the SecTAG field.

Figure 5: SecTAG format



4.2.3.2 MACsec encryption mode

The main modes of encryption in MACsec are:

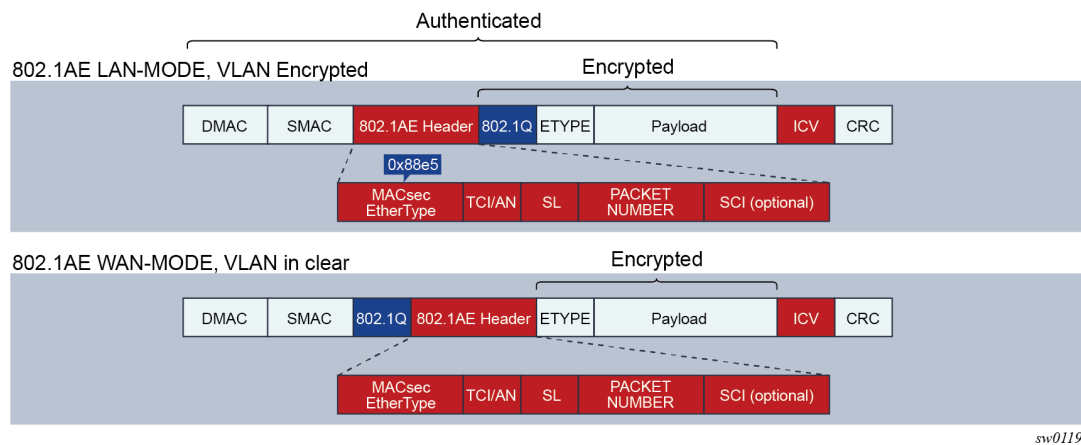
- VLAN in cleartext (WAN Mode)
- VLAN encrypted

802.1AE dictates that the 802.1Q VLAN must be encrypted. Some vendors provide the option of configuring the MACsec on a port with VLAN in cleartext.

SR OS supports both modes.

The following figure shows the encrypted VLAN and the VLAN in cleartext.

Figure 6: 802.1 AE LAN and WAN modes and VLAN encrypted and cleartext



Note: On 25G ports on the 7705 SAR-Hx and 7705 SAR-Mx , MACsec is supported in LAN mode and WAN mode. In WAN mode on the 25G ports, when a VLAN is encrypted, no other VLAN can be in clear text. Multiple VLANs can be encrypted using their own Connectivity Associations and security channel keys, but as soon as one VLAN is encrypted, the system does not allow clear traffic on any other VLAN on that port.

4.2.3.2.1 MACsec encryption per traffic flow encapsulation matching

MACsec can be applied to a selected subset of the port traffic, based on the type and value of the packet encapsulation. The SR OS can be configured to match and encrypt the following traffic encapsulation types:

- all encapsulated traffic arriving on the port, including untagged, single-tag, and double-tag. This is the default MACsec behavior.
- untagged only traffic
- single-tag or dot1q traffic. In this mode, MACsec can apply to a specific tag or a wildcard tag where all single-tag traffic is matched.
- double-tag or QinQ traffic. In this mode, MACsec can apply to a specific service tag, a specific service and customer tag, or a wildcard for any QinQ traffic.

MKA PDUs are generated specifically for the traffic-encapsulation type that is being matched.

4.2.3.3 MACsec key management modes

The following table describes the main key management modes in MACsec.

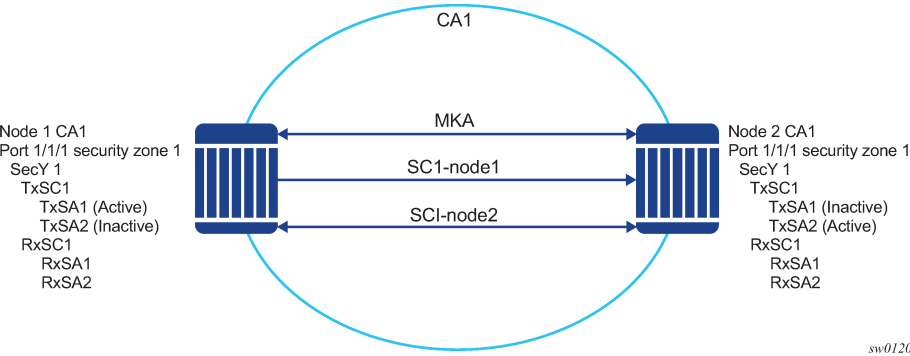
Table 10: MACsec key management modes

Keying	Explanation	SR OS support	Where used
Static SAK	Manually configures each node with a static SAK, SAM, or CLI		Switch to switch
Static CAK PRE SHARED KEY	Uses a dynamic MACsec Key Management (MKA) and a configured pre-shared key to drive the CAK. The CAK encrypts the SAK between two peers and authenticates the peers.	✓	Switch to switch
Dynamic CAK EAP Authentication	Uses a dynamic MKA and an EAP Master System Key (MSK) to drive the CAK. The CAK encrypts the SAK between two peers and authenticates the peers.		Switch to switch
Dynamic CAK MSK distribution via RADIUS and EAP-TLS	Stores the MSKs in the Radius server and distributes to the hosts via EAP-TLS. This is typically used in the access networks where a large number of hosts use MACsec and connect to an access switch. MKA uses MSK to drive the CAK. The CAK encrypts the SAK between 2 peers and authenticates the peers.		Host to switch

4.2.3.4 MACsec terminology

The following figure shows the main concepts used in MACsec for the static-CAK scenario.

Figure 7: MACsec concepts for static-CAK



The following table describes MACsec terminology.

Table 11: MACsec terminology

MACsec term	Description
Connectivity Association (CA)	Provides a security relationship, established and maintained by key agreement protocols (MKA), that comprises a fully connected subset of the SAPs in stations attached to a single LAN that are to be supported by MACsec.
MACsec Key Agreement (MKA) Protocol	Provides a control protocol between MACsec peers, which is used for peer aliveness and encryption key distribution. MKA is responsible for discovering, authenticating, and authorizing the potential participants in a CA.
MAC Security Entity (SecY)	Operates the MAC security protocol within a system. Manages and identifies the SC and corresponding active SA.
Security Channel (SC)	Provides a unidirectional point-to-point or point-to-multipoint communication. Each SC contains a succession of SAs, and each SC has a different SAK.
Security Association (SA)	<p>In the cases of SR OS with two SAs per SC, each with a different SAK, each SC comprises a succession of SAs. Each SA is identified by the SC identifier, concatenated with a two-bit association number. The Secure Association Identifier (SAI) that is created allows the receiving SecY to identify the SA and the SAK used to decrypt and authenticate the received frame. The AN, and consequently the SAI, is only unique for the SAs that can be used or recorded by participating SecYs at any time.</p> <p>The MKA creates and distributes SAKs to each of the SecYs in a CA. This key creation and distribution is independent of the cryptographic operation of each of the SecYs. The decision to replace one SA with its successor is made by the SecY that transmits using the SC, after the MKA has informed it that all the other SecYs are prepared to receive using that SA. No notification, other than receipt of a secured frame with a different SAI is sent to the receiver. A SecY must always be capable of storing SAKs for two SAs for each inbound SC, and of swapping from one SA to another without notice. Certain LAN technologies can reorder frames of different priority, so reception of frames on a single SC can use interleaved SA.</p>
Security Association Key (SAK)	Provides the encryption key used to encrypt the MACsec datapath.

4.2.3.5 MACsec static CAK

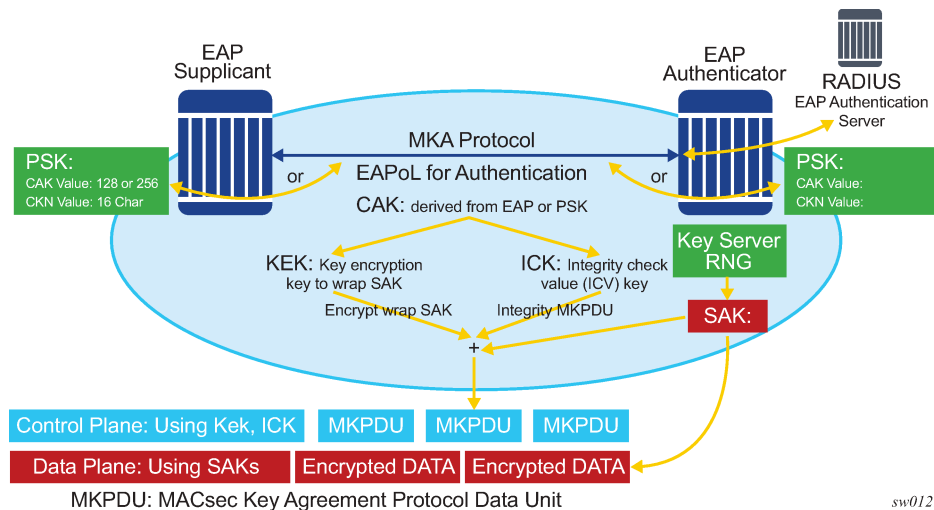
MACsec uses SAs to encrypt packets. SA is a security relationship that provides security guarantees for frames transmitted from one member of a CA to the others. Each SA contains a single secret key (SAK) used to perform the cryptographic operations that encrypt the datapath PDUs.

SAK is the secret key used by an SA to encrypt the channel.

When enabled, MACsec uses a static CAK security mode. Two security keys, a connectivity association key (CAK) that secures control plane traffic, and a randomly generated secure association key (SAK) that secures data plane traffic, are used to secure the point-to-point or point-to-multipoint Ethernet link. Both keys are regularly exchanged between both devices on each end of the Ethernet link to ensure link security.

The following figure shows MACsec generating the CAK.

Figure 8: MACsec generating the CAK



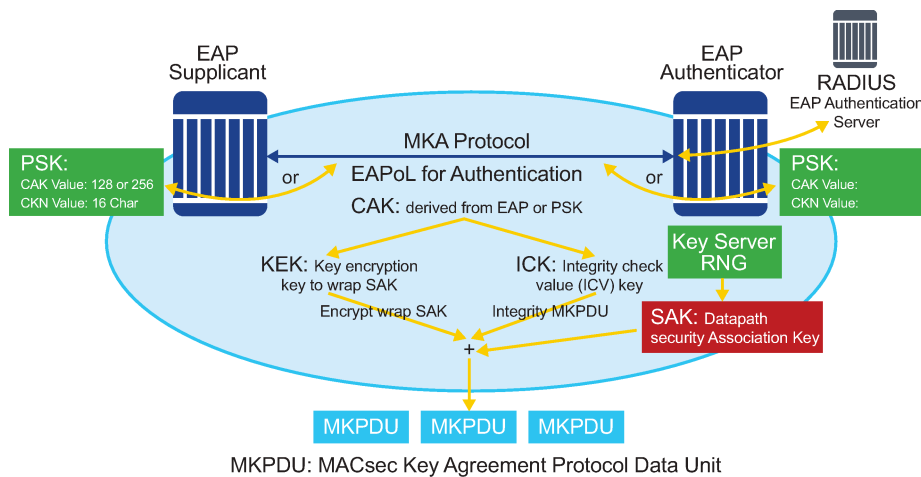
The node initially needs to secure control-plane communication to distribute SAKs between two or more members of a CA domain.

The CAK is used to secure the control plane. The following main methods are used to generate the CAK:

- EAPoL (SR OS does not support EAPoL)
- pre-shared key (CAK and CKN values are configured manually using the CLI). The following CAK and CKN rules apply:
 - CAK has 32 hexadecimal characters for a 128-bit key and 64 hexadecimal characters for a 256-bit key, depending on which algorithm is used for control plane encryption (for example, **aes-128-cmac** or **aes-256-cmac**).
 - CKN has 32 octets char (64 hex) and is the connectivity association key name, which identifies the CAK. This allows each MKA participant to select which CAK to use to process a received MKPDU. MKA places no restriction on the format of the CKN, except that it must comprise an integral number of octets, between 1 and 32 (inclusive), and that all potential members of the CA use the same CKN.
 - CKN and CAK must match on peers to create a MACsec secure CA.

The following figure shows the MACsec control plane authentication and encryption.

Figure 9: MACsec control plane and encryption



sw0122

A generated CAK can obtain the following additional keys:

- Key Encryption Key (KEK) – key used to wrap and encrypt the SAKs
- Integrity Connection Value (ICV) Key (ICK) – key used for an integrity check of each MKPDU sent between two CAs

The key server then creates a SAK, which is shared with the CAs of the security domain, and that SAK secures all data traffic traversing the link. The key server continues to periodically create and share a randomly created SAK over the point-to-point link for as long as MACsec is enabled.

The SAK is encrypted via the AES-CMAC, using the KEK as the encryption key, and ICK as the integration key.

4.2.3.6 SAK rollover

SR OS regenerates the SAK after the following events:

- when a new host has joined the CA domain and MKA hellos are received from this host
- when the sliding window is reaching the end of its 32-bit or 64-bit length
- when a new preshared key (PSK) is configured and a rollover of PSK is executed

4.2.3.7 MKA

Each MACsec peer operates the MKA. Each node can operate multiple MKAs, based on the number of CAs the node belongs to. Each MKA instance is protected by a distinct secure CAK, which allows each PAE to ensure that information for an MKA instance is accepted only from other peers that also possess that CAK, which identifies the peers as members or potential members of the same CA. See [MACsec static CAK](#) for information about the CAK identification process performed via CKN.

4.2.3.7.1 MKA PDU generation

The following table describes the MKA PDUs generated for different traffic encapsulation matches.

Table 12: MKA PDU generation

Configuration	Configuration example (<s-tag>.<c-tag>)	MKA packet generation	Traffic pattern match/behavior
All-encap	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 10 ca-name 10 encap-match all-match true</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 10 encap-match all-encap ca-name 10</pre>	untagged MKA packet	Matches all traffic on the port, including untagged, single-tag, and double-tag. Default behavior; only available behavior in releases before 16.0.
UN-TAG	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 10 ca-name 2 encap-match untagged true</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 10 encap-match untagged ca-name 2</pre>	untagged MKA packet	Matches only untagged traffic on port
802.1Q single S-TAG (specific S-TAG)	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 2 ca-name 3 encap-match single-tag 1</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 2 encap-match single-tag 1 ca-name 3</pre>	MKA packet generated with S-TAG=1	Matches only single-tag traffic on port with tag ID of 1
802.1Q single S-TAG (any S-TAG)	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 3 ca-name 4 encap-match single-tag *</pre>	untagged MKA packet	Matches any dot1q single-tag traffic on port

Configuration	Configuration example (<s-tag>.<c-tag>)	MKA packet generation	Traffic pattern match/behavior
	classic CLI <pre>configure port ethernet dot1x macsec sub-port 3 encap-match single-tag * ca-name 4</pre>		
802.1ad double tag (both tag have specific TAGs)	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 4 ca-name 4 encap-match double-tag 1.1</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 4 encap-match double-tag 1.1 ca-name 4</pre>	MKA packet generated with S-tag=1 and C-TAG=1	Matches only double-tag traffic on port with service tag of 1 and customer tag of 1
802.1ad double tag (specific S-TAG, any C-TAG)	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 6 ca-name 7 encap-match double-tag 1.*</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 6 encap-match double-tag 1.* ca-name 7</pre>	MKA packet generated with S-TAG=1	Matches only double-tag traffic on port with service tag of 1 and customer tag of any
802.1ad double tag (any S-TAG, any C-TAG)	MD-CLI <pre>configure port ethernet dot1x macsec sub-port 7 ca-name 8 encap-match double-tag *.*</pre> classic CLI <pre>configure port ethernet dot1x macsec sub-port 7 encap-match double-tag *.* ca-name 8</pre>	untagged MKA packet	Matches any double-tag traffic on port

4.2.3.7.2 Tags in clear behavior by traffic encapsulation type

[Table 13: Tags in clear behavior](#) describes how single or double tags in clear configuration under a connectivity association affect different traffic flow encryptions.

By default, all tags are encrypted in CA. An MKA can be generated without any tags (untagged), but the data being matched can be based on dot1q or QinQ.

Table 13: Tags in clear behavior

Configuration	Traffic pattern match/behavior	Sub-port CA configuration: no tag in cleartext	Sub-port CA configuration: single tag in cleartext	Sub-port CA configuration: double tag in cleartext
PORT All-encap	Matches all traffic on port, including untagged, single-tag, double-tag traffic (Release 15.0 default behavior)	MKA PDU: untagged Untagged traffic: encrypted Single-tag traffic: encrypted, no tag in clear Double-tag traffic: encrypted, no tag in clear	MKA PDU: untagged Untagged traffic: in clear Single-tag traffic: encrypted, single-tag in clear Double-tag traffic: encrypted, single-tag in clear	MKA PDU: untagged Untagged traffic: in clear Single-tag traffic: in clear Double-tag traffic: encrypted, double-tag in clear
untagged	Matches only untagged traffic on port	MKA PDU: untagged Untagged traffic: encrypted Single-tag traffic: not matched by this MACsec policy Double-tag traffic: not matched by this MACsec policy	N/A	N/A
802.1Q single tag (specific tag)	Matches only single-tag traffic on port with the configured tag value	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: tag is encrypted Double-tag traffic: not matched by this MACsec policy	MKA PDU: same tag as the one configured under encap-match Untagged traffic: not matched by this MACsec policy Single-tag traffic: tag is in clear Double-tag traffic: not matched by this MACsec policy	N/A
802.1Q single tag (any tag)	Matches all single-tag traffic on port	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: encrypted	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: encrypted with single tag in clear	N/A

Configuration	Traffic pattern match/behavior	Sub-port CA configuration: no tag in cleartext	Sub-port CA configuration: single tag in cleartext	Sub-port CA configuration: double tag in cleartext
		Double-tag traffic: not matched by this MACsec policy	Double-tag traffic: not matched by this MACsec policy	
802.1ad double tag (both tag have specific values)	Matches only double-tag traffic on port with both configured tag values	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching both configured tags: encrypted, no tag in clear	MKA PDU: single tag, equal to S-TAG Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching both configured tags: single S-TAG in clear	MKA PDU: double tag, equal to the values configured under the encaps-match Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching both configured tags: encrypted, both tags in clear
802.1ad double tag (specific S-TAG, any C-TAG)	Matches only double-tag traffic on port with the configured S-TAG	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching the configured S-TAG: encrypted, no tag in clear	MKA PDU: single tag, equal to S-TAG Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching the configured S-TAG: S-TAG tag in clear	MKA PDU: single tag, equal to S-TAG Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic matching the configured S-TAG: both tags in clear
802.1ad double tag (any S-TAG, any C-TAG)	Matches all double-tag traffic on port	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic: S-TAG tag in clear	MKA PDU: untagged Untagged traffic: not matched by this MACsec policy Single-tag traffic: not matched by this MACsec policy Double-tag traffic: both tags in clear

Configuration	Traffic pattern match/behavior	Sub-port CA configuration: no tag in cleartext	Sub-port CA configuration: single tag in cleartext	Sub-port CA configuration: double tag in cleartext
		Double-tag traffic: encrypted, no tag in clear		

4.2.3.8 PSK

MKA protects the integrity of MKPDUs and secures the SAK using keys that are derived from the CAK. The CAK and CKN are configured as part of a PSK entry under the static CAK configuration.

SR OS supports the following methods for configuring a static CAK:

- **active PSK index**

Two PSK entries identified by indexes 1 and 2. The active PSK selects the PSK entry that is used for the MKA session. This method uses a single MKA session. Each time the active PSK is rotated, the CKN and the derived keys are replaced for the same MKA session. A new PSK does not instantiate a new MKA session.

- **a keychain where the SR OS keychain infrastructure is used**

A keychain contains multiple key entries, where each key entry corresponds to a PSK that is active from its configured begin time up to the next key entry's begin time plus a configurable tolerance. With this method, every key entry rotation (PSK rotation) instantiates a new MKA session that is named by the CKN and secured by keys derived from the CAK configured in the key entry (PSK). The previous MKA session bound to the old key entry is deleted when its key entry tolerance expires. See [Keychain interaction with MACsec](#) for more information about MACsec usage with keychains.

4.2.3.8.1 Active PSK configuration

With an active PSK, an MKA participant can support one or more PSKs. These PSKs secure the same MKA session; PSK rotation does not instantiate a new MKA session.

A PSK may be created by NSP or entered manually using the CLI.

Each PSK is configured with the following fields:

- CAK name (CKN)
- encryption type

The CKN must be unique per port for the configured sub-ports, and can be used to identify the key in subsequent management operations.

Each static CAK configuration can have two PSK entries for rollover. The active PSK index dictates which CAK is used to derive the keys for encrypting the SAKs and verifying the integrity of the MKPDUs.

NSP provides additional functionality to roll over and configure the PSK. The rollover using NSP can be based on a configured timer.

4.2.3.8.2 Keychain

With a keychain, up to 64 PSK entries (keys) can be configured. Each PSK creates a new dedicated MKA session, active for the duration of the key lifetime plus the configured tolerance.

The active PSK and keychain configurations do not operate simultaneously. When a keychain is configured, it takes precedence over the active PSK configuration. When a keychain is not configured, the active PSK is used. To configure the active PSK or keychain, first administratively disable the CA.

SR OS can use the keychain for MACsec PSK rotation. A new MACsec container holds the keychain's key entries, where the PSK, CAK, and CKN values can be configured as hexadecimal characters. A keychain contains multiple key entries. Each entry is activated at its begin time; it remains valid up to the begin time of the next key entry, plus its own tolerance time.

A MACsec keychain supports the same ciphers as an active PSK. The **aes-128-cmac** and **aes-256-cmac** ciphers can be configured per keychain key entry for MACsec.

The following MACsec container is used to configure the MACsec keychain entries:

- **MD-CLI**

```
configure system security keychains keychain macsec
```

- **classic CLI**

```
configure system security keychain macsec
```

When a keychain is configured with a MACsec key entry, other key entry types, such as bidirectional, received, or send, can no longer be added to the keychain. All MACsec key entries in the keychain must be removed to repurpose it for other key entry types.

Conversely, when a keychain is configured with any bidirectional, received, or send key entries, MACsec key entries can no longer be added. All existing key entries in the keychain must be removed to repurpose it for MACsec use.

Only a keychain with at least one MACsec key entry can be assigned to a MACsec static CAK. When a keychain is assigned to a MACsec static CAK, deleting the MACsec key entry is rejected.

4.2.3.8.2.1 Transitioning from an active PSK to a keychain

For greenfield deployments, the user must choose between an "active PSK" and a "keychain" method for PSK rotation, as the two methods behave differently.

Users can upgrade from an active PSK implementation to a keychain. The following procedures are recommended when upgrading:

1. To configure either a keychain or an active PSK, the CA must be administratively disabled first.
2. Add the keychain after the CA is administratively disabled, and then administratively enable the CA.
3. When the CA's static CAK contains a keychain, the keychain takes precedence over an active PSK configuration under the static CAK, even if the keychain key entries are not active or the MKA sessions for the keychain are not operationally up. That is, only one method is used at a time, with a keychain having precedence over an active PSK when configured.
4. To remove the keychain, the CA must be administratively disabled again.



Note: When a CA is administratively disabled, all MKA sessions using that CA are deleted.

4.2.3.8.2.2 Keychain interaction with MACsec

In other applications, the tolerance of a keychain key entry can be configured as 0, with a default tolerance of 300 seconds. The MACsec application requires a minimum tolerance value of 20 seconds; a 0 tolerance value is rejected if the key entry is a MACsec entry.

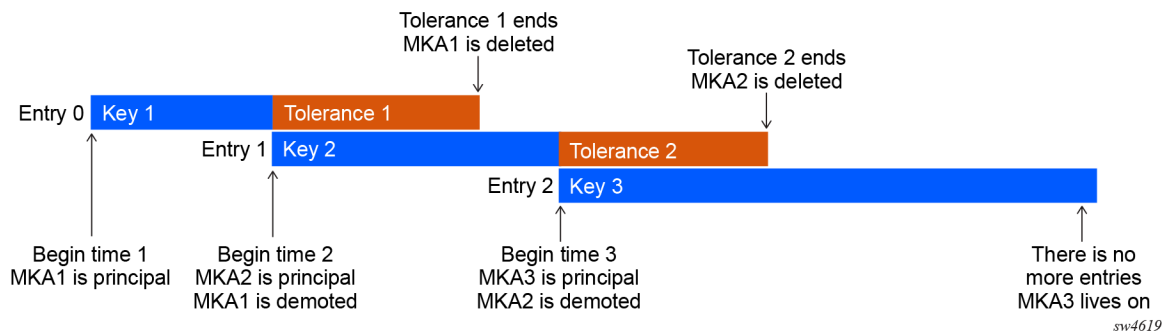
The MKA instance created by the first active key entry serves as the principal session from its begin time and remains functional until the end of its tolerance period. When the next key entry becomes active at its begin time, a new MKA instance is created. When the new MKA instance becomes operationally up, it becomes the principal session, and the MKA instance created by the first key entry becomes the backup session until its tolerance timer expires. If the new MKA session does not become operationally up by the time the tolerance of the first key entry expires, the MACsec sub-port becomes operationally down because neither the first nor second key entry has an operational MKA instance.

4.2.3.8.2.3 Tolerance in a keychain

On SR OS, the default tolerance value for a MACsec key is 300 seconds (5 minutes). A tolerance value below 20 seconds is rejected.

The following figure shows an example of tolerance with three key entries.

Figure 10: Tolerance with three key entries



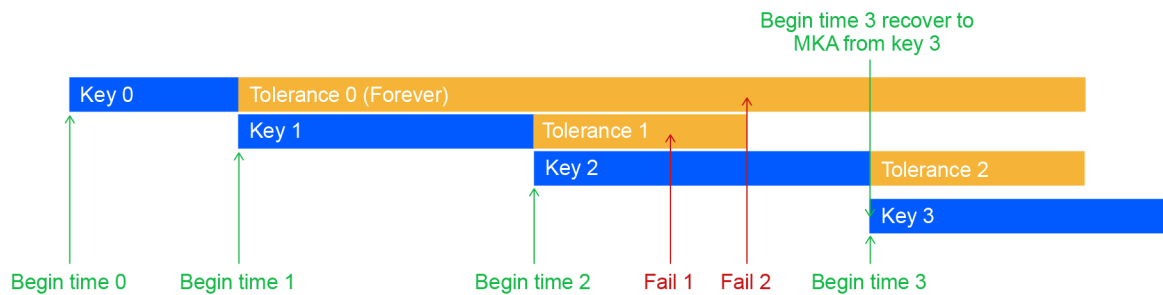
Note: SR OS supports a maximum of three¹⁰ MKA sessions per sub-port at any one time. Therefore, the user must configure the begin times and tolerance periods of each entry in the keychain accordingly to ensure that no more than three MKA sessions exist at the same time.

4.2.3.8.2.3.1 Backup key

The following figure shows an example where key 0 is a backup key with the tolerance set to **forever**.

¹⁰ One of the three concurrent MKA sessions must be from key entry 0 (backup key).

Figure 11: Backup key with the tolerance set to forever



sw4620

In Figure 11: Backup key with the tolerance set to forever, the following also applies:

- The MKA session for Key 0 can live forever using the tolerance set to **forever**. This means the MKA session for Key 0 exchanges MKA hellos, but it is not the principal session.
- Only Key 0 is intended to be a long-lived backup key. To effectively use it as a backup key, its tolerance should be set to **forever**. However, no restrictions are placed on its begin time and tolerance values to allow flexibility in use cases.



Note: If a backup key is not required, it is recommended not to configure Key 0.

- If the MKA session of Key 1 or Key 2 fails, the MKA session of Key 0 takes over as the principal MKA session.
- At Fail 1 point of Key 2, the fallback is Key 1 because its tolerance has not expired.
- At Fail 2 point of Key 2, the fallback key is the Key 0 backup key entry.
- With the exception of Key 0, any configuration that results in more than two overlapping active keys (and therefore MKA sessions) is rejected.
- If at the end of Key 1, Tolerance 1 and Key 2 have not established an MKA session (for example, Key 2 has a mismatch in its CKN between the peers), the Key 0 MKA session takes over and becomes the principal MKA session. The switch from Key 1 to Key 0 after the Key 1 tolerance expires is seamless.

4.2.3.8.2.4 Unique CKN per port

MACsec requires a unique CKN for each port. Identical CKNs are not allowed for consecutive keychain entries. Identical CKNs can be programmed on multiple keychains that are used for multiple subports on a MACsec port.

For example:

- Port 1/1/1, subport 1 is using keychain 1 and has one entry with CKN “1234”
- Port 1/1/1, subport 2 is using keychain 2 and also has one entry with CKN “1234”
- Port 1/1/1, subport 3 is using static CAK PSK 1 that has CKN “1234”



Note: Users must avoid configuring multiple keychains and static CAK PSKs with the same CKN on the same port.

4.2.3.9 Autogenerated PSK

SR OS supports autogenerating PSKs and automatically distributing them to all peers in a CA via the MKA. SR OS uses the local RNG and entropy, which is certified by NIST, to generate the PSK (CAK and CKN). PSKs are generated and distributed based on a user-configured interval.

SR OS uses the key entries in a MACsec keychain to manually to bring up the initial keychain MKA session. After this initial keychain MKA session is established, SR OS generates a PSK and distributes it to create an autogenerated MKA session. The autogenerated MKAs is used to distribute SAKs and all consecutive autogenerated PSKs.



Note: SR OS does not distribute any SAK using the initial keychain MKA session.

Every time SR OS autogenerates a new PSK, it creates a new autogenerated MKA session for this new PSK. This behavior is same as described in [Keychain](#) and differs from the active PSK method where a single MKA session can use multiple PSKs to secure the SAK.

4.2.3.9.1 PSK autogeneration details

Use the following command to enable PSK autogeneration for a specific CA.

```
configure macsec connectivity-association psk-type auto-gen-psk
```

Use the commands in the following context to configure PSK autogeneration.

```
configure macsec connectivity-association auto-gen-psk
```

When MACsec is initialized with PSK autogeneration enabled, SR OS uses an initial MACsec keychain (also known as the recovery keychain) to establish an initial keychain MKA session to the peer. SR OS then locally autogenerates a PSK and distributes it to peers using the initial keychain MKA session. This first autogenerated PSK is used to establish the autogenerated MKA session. The autogenerated MKA session is used to distribute the first SAK and also the next autogenerated PSK at the user-configured interval.

The initial keychain (also known as the recovery keychain) can have a single key entry with the **forever** option, or multiple key entries that activate at a specified **begin-time**. The rules and behavior for using a keychain to establish MKA sessions, as described in [Keychain](#), also apply for PSK autogeneration. SR OS always keeps this initial keychain MKA session active to use as a fallback and recovery option.

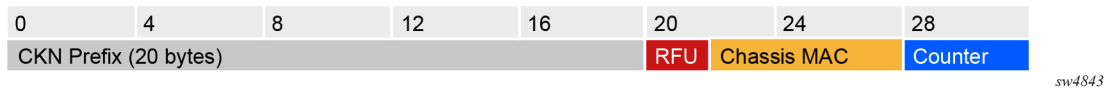
In the event of a system reboot, or if the autogenerated MKA session fails, the system falls back to the initial keychain MKA session and uses it as the primary MKA session. The system distributes a new autogenerated PSK over this initial keychain MKA session. This establishes a new autogenerated MKA session and resumes the PSK autogeneration process. When the system is operating in the autogeneration MKA cycle, after a new autogeneration MKA session becomes active, the previous autogenerated MKA session is removed.

The system autogenerates PSKs using the key server ESV-approved RNG. The following conditions trigger the system to autogenerate and distribute a new PSK:

- the user-configured **generate-interval** timer expires
- a new active peer joins the CA

The following figure shows the CKN format of an autogenerated PSK.

Figure 12: Autogenerated PSK CKN format



The CKN of the autogenerated PSK consists of the following:

- 20-byte user-configured CKN prefix string in hexadecimal format (strings of less than 20 bytes are padded with zeros)
- 2-byte RFU which is reserved for future use
- 6-byte chassis MAC which is used to differentiate the CKN on different chassis
- 4-byte counter that the system increments by one for each new PSK generated, which is used to differentiate CKNs within the same CA

PSK autogeneration and MKA session information can be displayed using the following **show** commands.

```
show macsec connectivity-association
show macsec mka-session
```

Use the following command to force the system to immediately autogenerate and distribute a new PSK.

```
tools perform macsec ca auto-gen-psk force-rollover-now
```

Use the following command to delete the current autogenerated PSK and fall back to the initial keychain recovery process to reestablish PSK autogeneration.

```
tools perform macsec ca auto-gen-psk delete-generated-psk
```

4.2.3.9.2 ISSU and CPM redundancy considerations

When performing an ISSU on supported hardware, there is a period of time during which the MDA is unavailable for SAK installation. Nokia recommends performing the following steps to prevent potential SAK installation issues during an ISSU:

- When using a keychain with multiple entries, administratively disable all entries on both ends of the MACsec session that have a **begin-time** within the ISSU window. This ensures that the keychain does not force a PSK and SAK rotation during the ISSU.
- Configure the **generate-interval** command to a value that covers the entire ISSU window to prevent the system from autogenerating a PSK during the ISSU.
- Use the **tools perform macsec ca auto-gen-psk force-rollover-now** command to force the system to immediately generate a new PSK and SAK.

During the CPM redundancy switchover, the MDA may not be reachable by the new CPM for approximately 10 seconds or more. The system automatically adds 10 minutes to the **generate-interval** command to ensure that PSK and SAK autogeneration does not occur until after the CPM switchover is complete and the MDA is reachable.

4.2.3.10 MKA Hello timer

MKA uses a member identifier (MI) for each node in the CA domain.

A participant proves liveness to each of its peers by including its MI, together with an acceptably recent message number (MN), in an MKPDU.

To avoid a new participant having to respond to each MKPDU from each partner as it is received, or trying to delay its reply until it is likely that MI MN tuples have been received from all potential partners, each participant maintains and advertises both of the following:

- **live peers list**
This list includes all the peers that have included the participant MI and a recent MN in a recent MKPDU.
- **potential peers list**
This list includes all the other peers that have transmitted an MKPDU that was directly received by the participant or that were included in the live peers list of an MKPDU transmitted by a peer that has proved liveness.

Peers are removed from each list when an interval of between MKA Life Time and MKA Life Time plus MKA Hello Time has elapsed since the participant's recent MN was transmitted. This time is sufficient to ensure that two or more MKPDUs have been lost or delayed prior to the incorrect removal of a live peer.



Note: The specified use of the live and potential peers lists allows rapid removal of participants that are no longer active or attached to the LAN while reducing the number of MKPDUs transmitted during group formation. For example, a new participant is admitted to an established group after receiving and then transmitting one MKPDU.

The following table describes the MKA participant timer values used on SR OS.

Table 14: MKA participant timer values

Timer use	Timeout (option)	Timeout (option)
Per participant periodic transmission, initialized for each transmission on expiry	MKA Hello Time or MKA Bounded Hello Time	2.0 0.5
Per peer lifetime, initialized when adding to or refreshing the potential peers list or live peers list; expiry causes removal from the list	MKA Life Time	6.0
Participant lifetime, initialized when participant is created or following the receipt of an MKPDU; expiry causes the deletion of the participant		
Delay after last distributing a SAK, before the Key Server distributes a fresh SAK following a change in the live peer list while the potential peer list is still not empty		

4.2.3.11 MACsec Capability, Desire, and encryption offset

802.1x-2010 had identified the following fields in the MKA PDU:

- MACsec Capability
- Desire

MACsec Capability signals whether MACsec is capable of integrity and confidentiality. The following table describes the basic settings for MACsec Capability.

Table 15: MACsec basic settings

Setting	Description
0	MACsec is not implemented
1	Integrity without confidentiality
2	The following are supported: <ul style="list-style-type: none"> • integrity without confidentiality • integrity and confidentiality with a confidentiality offset of 0
3 ¹¹	The following are supported: <ul style="list-style-type: none"> • integrity without confidentiality • integrity and confidentiality with a confidentiality offset of 0, 30, or 50

An encryption offset of 0, 30, or 50 starts from the byte after the SecTAG (802.1ae header). Ideally, the encryption offset should be configured for IPv4 (offset 30) and IPv6 (offset 50) to leave the IP header in cleartext. This allows routers and switches to use the IP header for LAG or ECMP hashing.

4.2.3.12 Key server

The participants in an MKA instance agree on a Key Server and are responsible for the following:

- deciding on the use of MACsec
- cipher suite selection
- SAK generation and distribution
- SA assignment
- identifying the CA when two or more CAs merge

Each participant in an MKA instance uses the Key Server Priority (an 8-bit integer) encoded in each MKPDU to agree on the Key Server. Each participant selects the live participant advertising the highest priority as its Key Server whenever the live peers list changes, provided that the highest-priority participant has not selected another as its Key Server or is unwilling to act as the Key Server. If a Key Server cannot be selected, SAKs are not distributed. In the event of a tie for the highest-priority Key Server, the

¹¹ SR OS supports setting (3).

member with the highest-priority SCI is chosen. For consistency with other uses of the SCI MAC address component as a priority, numerically lower values of the Key Server Priority and SCI are afforded the highest priority.



Note: For SCI, each SC is identified by an SCI, which comprises a globally unique MAC address and a Port Identifier unique within the system allocated to that address.

4.2.3.13 SA limits and network design

Each MACsec device supports 64 Tx-SAs and 64 Rx-SAs. An SA is the key to encrypt or decrypt the data.

In accordance with the IEEE 802.1AE standard, each SecY contains an SC, which is a unidirectional concept; for example, Rx-SC or Tx-SC. Each SC contains at least one SA for encryption on Tx-SC and decryption on Rx-SC. For extra security, each SC should be able to roll over the SA. The system allocates resources for two SAs on each SC for rollover purposes, as defined in the standard.

Each MACsec phy, referred to as a MACsec security zone, supports up to 64 Tx-SAs and 64 Rx-SAs. Assuming two SAs per SC for SA rollover, each security zone supports up to 32 Rx-SCs and 32 Tx-SCs.

The following table describes the port mapping to security zones.

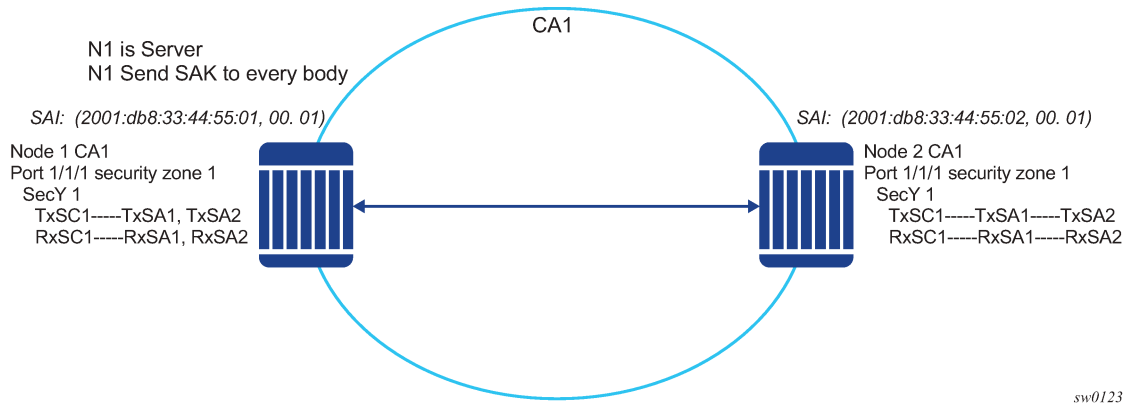
Table 16: Port mapping to security zone

MDA	Ports mapped to security zones								SA limit per security zone
	Zone 1	Zone 2	Zone 3	Zone 4	Zone 5	Zone 6	Zone 7	Zone 8	
m4-1g-rj+6-10g-sfp+ +2-25g-sfp28	1, 2, 3, 4	c5, c6, c7, c8	c9, c10, c11, c12	—	—	—	—	—	Zone 1: Rx-SA = 16 Tx-SA = 16 Zone 2, 3: Rx-SA = 64 Tx-SA = 64
m8-1g-rj45	1	2	3	4	5	6	7	8	Rx-SA = 64 Tx-SA = 64
m8-1g-sfp	c1/1	c2/1	c3/1	c4/1	c5/1	c6/1	c7/1	c8/1	Rx-SA = 64 Tx-SA = 64
m2-1g-sfp+2-10g-sfp+	c1, c2, c3, c4	—	—	—	—	—	—	—	Rx-SA = 64 Tx-SA = 64

4.2.3.14 P2P (switch-to-switch) topology

In a point-to-point (P2P) topology, each router needs a single security zone and a single Tx-SC for encryption, and a single Rx-SC for decryption. Each SC has two SAs. In total, for P2P topology, four SAs are needed: two RxSAs for RxSC1 and two TXSAs for TxSC1. The following figure shows the P2P topology.

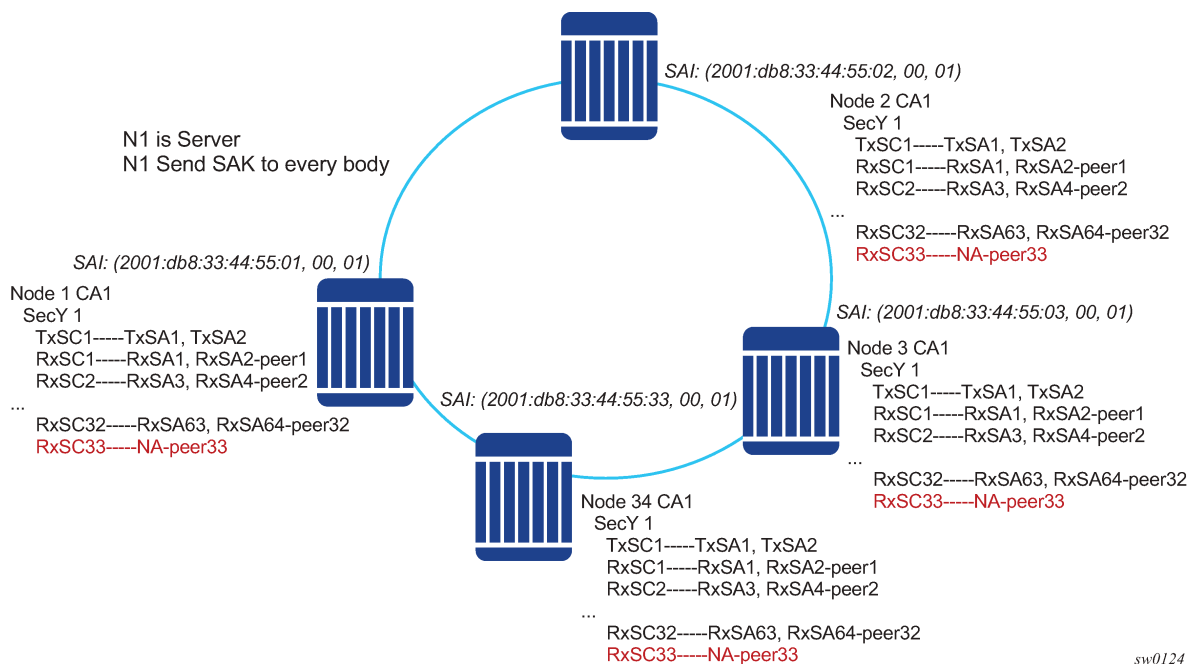
Figure 13: Switch-point to switch-point topology



4.2.3.15 P2MP (switch-to-switch) topology

In a multipoint topology with N nodes, each node needs a single TxSC and N RxSC, one for each of the peers. For example, 64 maximum RX-SAs per security zone translates to 32 Rx-SCs, which breaks down to only 32 peers (only 33 nodes in the multipoint topology per security zone, where each node has one TxSC and 32 RxSC).

Figure 14: Switch-multipoint to switch-multipoint topology



In the preceding figure, when the 34th node joins the multipoint topology, the other 33 nodes that are already part of this domain do not have SAs to create an RxSC for the 34th node. However, the 34th node has a TxSC and accepts 32 peers. The 34th node starts transmitting and encrypting the PDUs based on its TxSC. However, because the other nodes do not have an SC for this SAI, they drop all Rx PDUs.



Note: To ensure that a multicast domain for a single security zone does not exceed 32 peers or the summation of all the nodes in a security zone CA domain, Nokia recommends not exceeding 33. This is the same as if a security zone has four CAs; the sum of all nodes across the four CAs must be 33 or less.

4.2.3.16 SA exhaustion behavior

[SA limits and network design](#) describes that a security zone has 64 RxSAs and 64 TxSAs. Two RxSAs are used for each RxSC for rollover purposes, and two TxSAs are used for TxSC for rollover purposes. This translates to 32 peers per security zone.

Under each port, users can configure the maximum number of peers allowed on that port.



Caution: Nokia strongly recommends that the user ensure the maximum peer value does not exceed the limit of maximum peers per security zone or maximum peers per port values in the following command:

- **MD-CLI**

```
configure port ethernet dot1x macsec sub-port max-peers
```

- **classic CLI**

```
configure port ethernet dot1x macsec sub-port max-peer
```

If the maximum peer limit is exceeded, peer connectivity may be random in the event of a node failure or packet loss. Peers may join the CA randomly, on a first-come, first-served basis.

4.2.3.17 Clear tag mode

In most Layer 2 networks, MAC forwarding is performed via the destination MAC address. The 802.1AE standard dictates that any field after the source and destination MAC address and after the SecTAG must be encrypted. This includes the 802.1Q tags. In some VLAN switching networks, it may be necessary to leave the 802.1Q tag in cleartext.

SR OS supports configuring 802.1Q tags in cleartext by placing the 802.1Q tag before the SecTAG, or in encrypted text by placing it after the SecTAG.

The following table lists the MACsec encryption of 802.1Q tags when **clear-tag-mode** is configured on the SR OS.

Table 17: MACsec encryption of 802.1Q tags with **clear-tag-mode** configured

Unencrypted format	clear-tag-mode configuration	Pre-encryption (Tx)	Pre-decryption (Rx)
Single tag (dot1q)	Single-tag	DA, SA, TPID, VID, Etype	DA, SA, TPID, VID, SecTAG
Single tag (dot1q)	Double-tag	DA, SA, TPID, VID, Etype	DA, SA, TPID, VID, SecTAG
Double tag (QinQ)	Single-tag	DA, SA, TPID1, VID1, IPID2, VID2, Etype	DA, SA, TPID1, VID1, SecTAG

Unencrypted format	clear-tag-mode configuration	Pre-encryption (Tx)	Pre-decryption (Rx)
Double tag (QinQ)	Double-tag	DA, SA, TPID1, VID1, IPID2, VID2, Etype	DA, SA, TPID1, VID1, IPID2, VID2, SecTAG

4.2.3.18 802.1x tunneling and multihop MACsec

MACsec is an Ethernet packet and, as with any other Ethernet packet, can be forwarded through multiple switches via Layer 2 forwarding. Packet encryption and decryption is performed via 802.1x (MKA) capable ports.

To ensure that MKA is not terminated on any intermediate switch or router, the user can enable 802.1x tunneling on the corresponding port.

The following example shows how to check if tunneling is enabled.

Example: MD-CLI

```
[ex:/configure port 1/1/12 ethernet dot1x]
A:admin@node-2# info
    tunneling true
```

Example: classic CLI

```
A:node-2>config>port>ethernet>dot1x# info
-----
    tunneling
```

By enabling tunneling, the 802.1x MKA packets transit the port without being terminated. Therefore, MKA negotiation does not occur on a port that has 802.1x tunneling enabled.

4.2.3.19 EAPoL destination address

The MKA packets are transported over EAPoL with a multicast destination MAC address. If it is required for the MKA have a point-to-point connection to a peer node over a Layer 2 multihop cloud, the EAPoL destination MAC address can be set to the peer MAC address. This forces the MKA to traverse multiple nodes and establish an MKA session with the specific peer.

4.2.3.20 Mirroring consideration

Mirroring is performed before the MACsec encryption engine. Therefore, if a port is MACsec-enabled and mirrored, all mirrored packets are in cleartext.

4.2.4 Exponential Port Dampening

Exponential Port Dampening (EPD) provides the ability to automatically block a port from reuse for a period of time after physical link-down and physical link-up events. If a series of down-up events occur close together, EPD keeps the port's operational state down for a longer period than if only one down-up event

has occurred. The router avoids using that port if external events are causing the link state to fluctuate. The more events that occur, the longer the port is kept down and avoided by the routing protocols.

EPD behavior uses a fixed penalty amount per link-down event and a half-life decay equation to reduce these penalties over time. The following equation defines exponential decay:

$$N(t) = N_0 \left(\frac{1}{2}\right)^{\frac{t}{t_{1/2}}}$$

sw0109

where:

$N(t)$ is the quantity that still remains after a time t

N_0 is the initial quantity

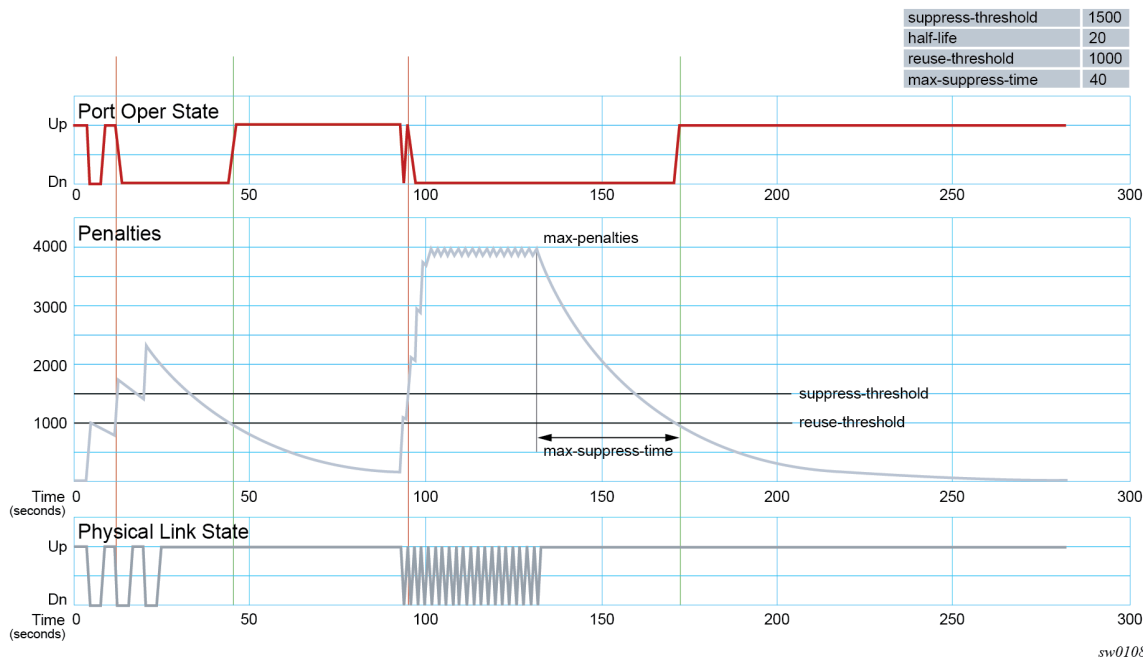
$t_{1/2}$ is the half-life

In dampening, N_0 refers to the starting penalties from the last link-down event. The quantity $N(t)$ refers to the decayed penalties at a specific time, and is calculated starting from the last link-down event (that is, from the time when N_0 last changed).

This equation can also be used on a periodic basis by updating the initial quantity value N_0 each period and then computing the new penalty over the period (t).

The following figure shows an example usage of the EDP feature.

Figure 15: EPD example



At time ($t = 0$) in the preceding figure, the initial condition has the link up, the accumulated penalties are zero, the dampening state is idle, and the port operational state is up. The following series of events and actions occur.

1. $t = 5$: link-down event

- the accumulated penalties are incremented by 1000
 - the accumulated penalties now equal 1000, which is less than the suppress threshold (of 1500), so the dampening state is idle
 - because the dampening state is idle, link-down is passed to the upper layer
 - link-down triggers the port operational state to down
2. $t = 9$: link-up event
- the accumulated penalties equal 869, which is less than the suppress threshold, so the dampening state remains as idle
 - because the dampening state is idle, link-up is passed to the upper layer
 - link-up triggers the port operational state to up
3. $t = 13$: link-down event
- the accumulated penalties are incremented by 1000
 - the accumulated penalties now equal 1755, which is greater than the suppress threshold, so the dampening state is changed to active
 - because the dampening state just transitioned to active, link-down is passed to the upper layer
 - link-down triggers the port operational state to down
4. $t = 17$: link-up event
- the accumulated penalties equal 1527, which is above the reuse threshold (of 1000) and greater than the suppress threshold, so the dampening state remains as active
 - because the dampening state is active, link-up is not passed to the upper layer
 - the port operational state remains down
5. $t = 21$: link-down event
- the accumulated penalties are incremented by 1000
 - the accumulated penalties now equal 2327, which is above the reuse threshold, so the dampening state remains as active
 - because the dampening state is active, link-down is not passed to the upper layer
 - the port operational state remains down
6. $t = 25$: link-up event
- the accumulated penalties equal 2024, which is above the reuse threshold, so dampening state remains as active
 - because the dampening state is active, link-up is not passed to the upper layer
 - the port operational state remains down
7. $t = 46$: accumulated penalties drop below the reuse threshold
- the accumulated penalties drop below the reuse threshold, so the dampening state changes to idle
 - because the dampening state is idle and the current link state is up, link-up is passed to the upper layer
 - the port operational state changes to up
8. $t = 94$ to 133: link-down and link-up events every second

- similar to previous events, the accumulated penalties increment on every link-down event
 - the dampening state transitions to active at $t = 96$, and link state events are not sent to the upper layer after that time
 - the upper layer keeps the port operational state down after $t = 96$
 - the accumulated penalties increment to a maximum of 4000
9. $t = 133$: final link event of link-up
- the accumulated penalties equal 3863
 - the dampening state remains active and link state events are not sent to the upper layer
 - the upper layer keeps the port operational state down
10. $t = 172$: accumulated penalties drop below the reuse threshold
- the accumulated penalties drop below the reuse threshold, so the dampening state changes to idle
 - because the dampening state is idle and the current link state is up, link-up is passed to the upper layer
 - the port operational state changes to up

4.2.5 Network synchronization on ports

4.2.5.1 Network synchronization on T1/E1 ports

Each T1/E1 port can be independently configured for loop-timing (recovered from an Rx line) or node-timing (recovered from the system synchronization unit (SSU) in the active control processing module (CPM)).

A T1/E1 port can be configured to be a timing source for the node.

4.2.6 Ethernet port down-when-looped

Newly provisioned circuits are often looped back using a physical loopback cable for testing and to ensure the ports meet the SLA. If loopbacks are not cleared, or physically removed, by the operator when the testing is completed, they can adversely affect the performance of all other SDPs and customer interfaces (SAPs). This is especially problematic for point-to-multipoint services such as VPLS, because Ethernet does not support TTL, which is essential in terminating loops.

The down-when-looped feature is used on the 7705 SAR Gen 2 to detect loops within the network and to ensure the continued operation of other ports. When this feature is activated, a keepalive loop PDU is periodically transmitted toward the network. The Ethernet port then listens for returning keepalive loop PDUs. In unicast mode, a loop is detected if any received PDUs have an Ethertype value of 9000, which indicates a loopback (Configuration Test Protocol), and the source and destination MAC addresses are identical to the MAC address of the Ethernet port. In broadcast mode, a loop is detected if any received PDUs have an Ethertype value of 9000 and the source MAC address matches the MAC address of the Ethernet port, and the destination MAC address matches the broadcast MAC address. When a loop is detected, the Ethernet port is immediately brought down.

EFM OAM and the down-when-looped feature cannot be enabled on the same port.

4.3 Forward Error Correction

Users can use Forward Error Correction (FEC) on some ports to improve either the transmission reliability or reach, or both. FEC must always be used on some interface types while it is optional for other interface types. Also, some interface types allow more than one type of FEC. No matter what the setting of the FEC attributes, the transmitter and the receiver must have the same configuration, or the link will not work. The setting of FEC on a specific port is dependent on the interface type and the specific optical transceiver in use.

For coherent optics, the FEC (host and media) do not need to be configured and are automatically inherited and enabled based on the specific module and configured coherent mode of operation.

5 Port Cross-Connect

5.1 PXC terminology

The following describes Port Cross-Connect (PXC) terminology:

- **PXC**

PXC is a software concept representing a pair of logical ports interconnecting egress and ingress forwarding paths within the same forwarding complex.

The physical underpinning of a PXC can be either of the following:

- **a faceplate (physical) port in a loopback mode**

The PXC is referred to as a port-based PXC. Multiple PXCs can be created per a faceplate port.

- **a loopback configuration in the MAC chip**

The PXC is referred to as an internal or MAC-based PXC. Multiple PXCs can be created per MAC loopback.

- **PXC sub-port**

PXC sub-port is a logical port that is created under the PXC. Two interconnected PXC sub-ports are created per PXC. This is further described in [Port-based PXC](#).

- **Forwarding Complex (FC)**

FC is a chipset connected to a set of faceplate ports that processes traffic in the ingress direction (the ingress path) and the egress direction (the egress path). A line card can contain multiple FCs for increased throughput, while the inverse is not true, a single FC cannot be distributed over multiple line cards.

The terms cross-connect and loopback can be used interchangeably.

5.2 Overview



Note: On 7705 SAR Gen 2 platforms, PXC is supported on 10G interfaces only.

This section describes the PXC feature implementation. PXC is a software concept representing a pair of logical ports interconnecting egress and ingress forwarding paths within the same forwarding complex (FC). In cross-connect functionality, an egress forwarding path is looped back to the ingress forwarding path on the same forwarding complex instead of leading out of the system.

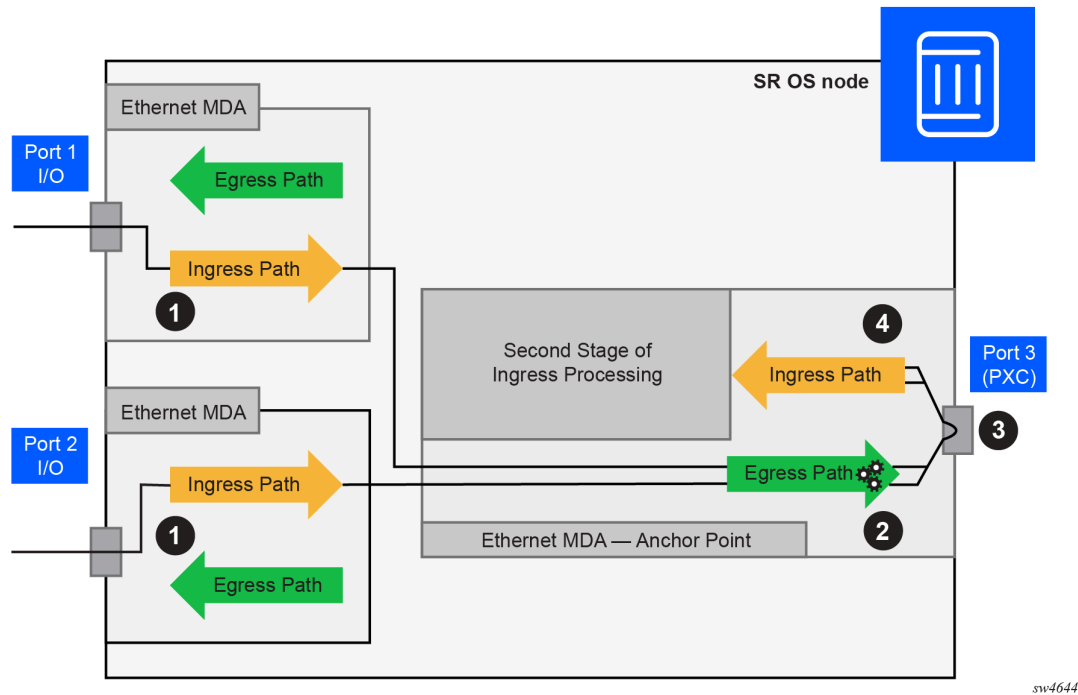
The cross-connect is modeled in the system and in the CLI as a port, appropriately naming the feature Port Cross-Connect (PXC) software concept representing a pair of logical ports interconnecting egress and ingress forwarding paths within the same forwarding complex.

Conceptually, PXC functionality is similar to the functionality provided by two externally interconnected faceplate ports where traffic exits the system through one port (the egress path) and is immediately looped back into another port (the ingress path) through a cable.

Figure 16: Traffic preprocessing using PXC shows the traffic flow from the first to the second stage through a cross-connect in a system with PXC:

1. Traffic entering a node through a faceplate port is processed by the local ingress forwarding path (1) on the line cards 1 and 2. Traffic is then directed toward the PXC (3) on the line card 3.
2. The PXC (3) loops the traffic from the local egress path (2) into the local ingress forwarding path (4) where it is further processed.

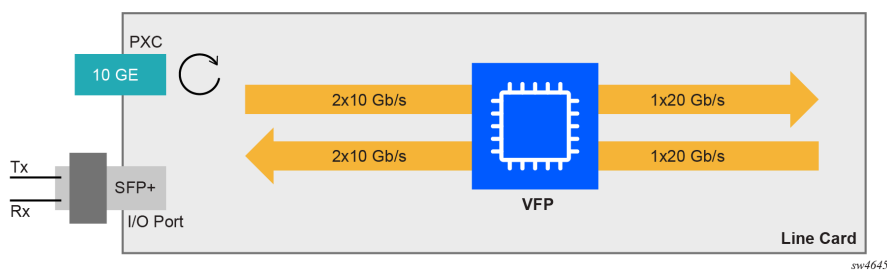
Figure 16: Traffic preprocessing using PXC



5.3 Port-based PXC

The concept of a port-based PXC (a PXC based on a faceplate port in loopback mode) is shown in [Figure 17: Port-based PXC](#). This PXC does not require an optical transceiver.

Figure 17: Port-based PXC



Example: Place the faceplate port into a cross-connect mode (MD-CLI)

```
[ex:/configure]
A:admin@node-2# info
port-xc {
    pxc 1 {
        admin-state enable
        port-id 1/1/c1/1
    }
}
```

Example: Place the faceplate port into a cross-connect mode (classic CLI)

```
A:node-2>config>port-xc# info
-----
    pxc 1 create
        port 1/1/c1/1
        no shutdown
    exit
exit
-----
```

Example: Multiple PXCs on the same underlying cross-connect configuration (MD-CLI)

```
[ex:/configure]
A:admin@node-2# info
port-xc {
    pxc 1 {
        admin-state enable
        port-id 1/1/c1/1
    }
    pxc 2 {
        admin-state enable
        port-id 1/1/c2/1
    }
    pxc 3 {
        admin-state enable
        port-id 1/1/c3/1
    }
}
```

Example: Multiple PXCs on the same underlying cross-connect configuration (classic CLI)

```
A:node-2>config>port-xc# info
-----
    pxc 1 create
        port 1/1/c1/1
        no shutdown
    exit
    pxc 2 create
        shutdown
        port 1/1/c2/1
    exit
    pxc 3 create
        shutdown
        port 1/1/c3/1
    exit
exit
```

A faceplate port that has been placed in the loopback mode for PXC use, supports only hybrid mode of operation and dot1q encapsulation. The recommendation is that the MTU value be configured to the maximum value. dot1x tunneling is enabled and cannot be changed.

The pre-set dot1q Ethernet encapsulation on the faceplate port is irrelevant from the user's perspective and there is no need to change it. The relevant encapsulation carrying service tags defined on PXC subports and that encapsulation is configurable. For more information, see [PXC sub-ports](#).

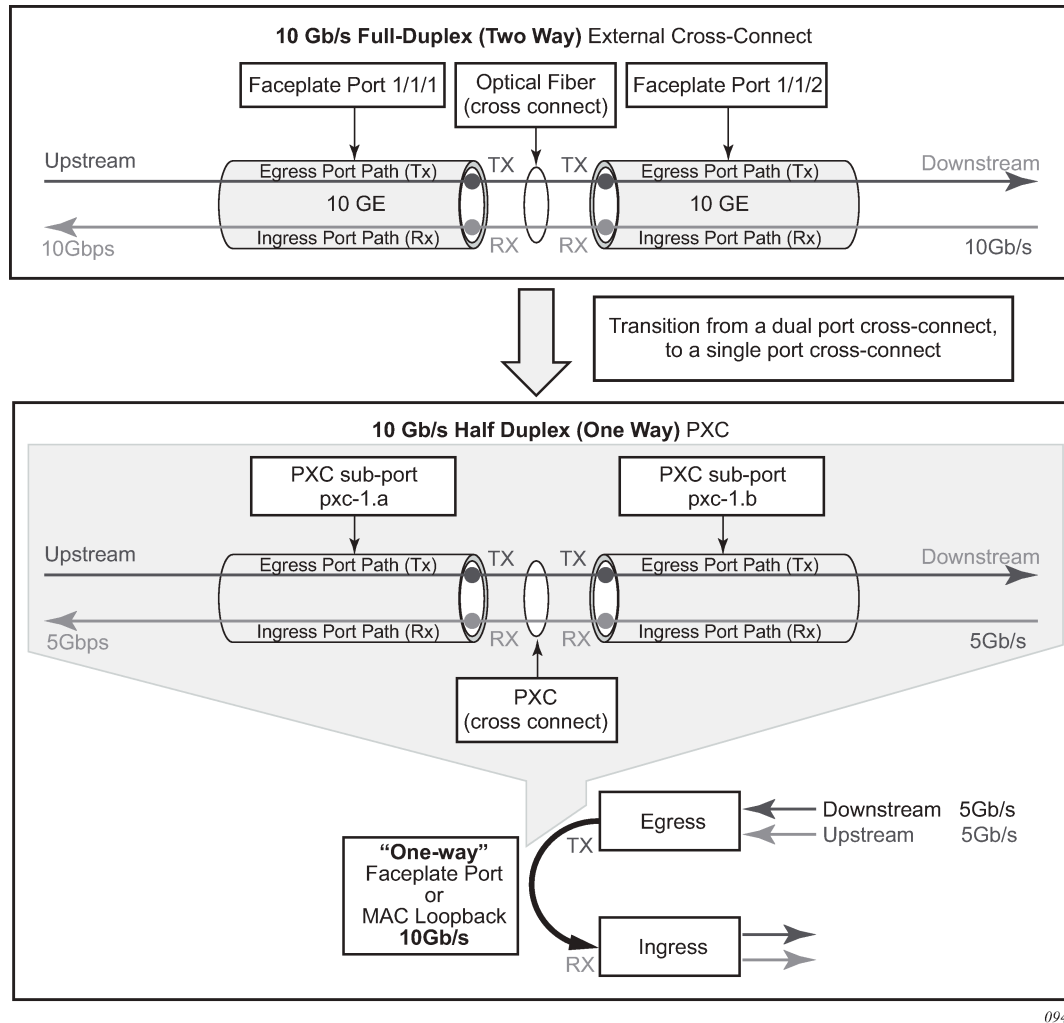
The following guidelines apply to a PXC configuration based on faceplate ports:

- Only unused faceplate ports (not associated with an interface or SAP) can be referenced within a PXC ID configuration.
- When the faceplate port is allocated to a PXC, it cannot be used outside of the PXC context. For example, an IP interface cannot use the faceplate port directly, or a SAP under a such port cannot be associated with an Epipe or VPLS service.

5.4 PXC sub-ports

[Figure 18: Two cross-connected external ports versus a single cross-connect](#) displays the benefit of PXC sub-ports on top of the cross-connect, which is analogous to two distinct faceplate ports that are connected by a fiber cable.

Figure 18: Two cross-connected external ports versus a single cross-connect



Bidirectional connectivity provided by PXC requires two sub-ports, one in each direction. The router uses these PXC sub-ports as logical configurations to transmit traffic in both directions over a half-duplex (one-way) cross-connect created in the system. As a result, the total bandwidth capacity supported by the mated PXC sub-ports is limited by the bandwidth capacity of the underlying cross-connect (a single faceplate port or a MAC loopback).

For example, if a 10 Gb/s faceplate port is allocated for PXC functions, the sum of downstream and upstream traffic on the mated PXC sub-ports is always less than or equal to 10 Gb/s. The bandwidth distribution is flexible; it can be symmetric (5 Gb/s downstream and 5 Gb/s upstream), or asymmetric (9 Gb/s downstream and 1 Gb/s upstream, 8 Gb/s downstream and 2 Gb/s upstream, or any other downstream and upstream distribution combination). Therefore, the faceplate port speed from the PXC perspective is half-duplex.

Similar logic can be followed for MAC-based PXC, with two key differences:

- The bandwidth (for example, 100 Gb/s) is configured under the MAC loopback and there is no need to allocate an additional faceplate port.

- PXC traffic is not reserved as part of the faceplate port bandwidth, as it is in the port-based PXC where a faceplate port is reserved only for PXC traffic. Instead, the PXC traffic is added to the traffic from the faceplate ports even in situations where all faceplate ports are 100% used, potentially oversubscribing the forwarding complex.

After the faceplate port or the port based on MAC loopback is associated with a PXC ID, a pair of mated PXC sub-ports is automatically created in the classic CLI by the SR OS.

In MD-CLI, the user must manually create the sub-ports.

The sub-ports must be explicitly enabled. Use the following commands to enable the subports:

- **MD-CLI**

```
admin-state enable
```

- **classic CLI**

```
no shutdown
```

The two PXC sub-ports are distinguishable by ".a" and ".b" suffixes. They transmit traffic toward each other, simulating two ports that are interconnected.

Although, the most PXC sub-ports command options are configurable, specific command options are fixed and cannot be changed. For example, PXC sub-ports are created in a hybrid mode and this cannot be modified.

Each PXC sub-port is internally (within the system) represented by an internal four-byte VLAN tag which is not visible to the user. Therefore, traffic carried over the PXC contains four extra bytes, which must be accounted for in the QoS configured on PXC sub-ports.

Example: MD-CLI

```
[ex:/configure port-xc]
A:admin@node-2# info
  pxc 1 {
    admin-state enable
    port-id 1/1/c1/1
  }
  pxc 2 {
    admin-state enable
    port-id 1/1/c2/1
  }
```

Example: classic CLI

```
A:node-2>config>port-xc# info
-----
  pxc 1 create
    port 1/1/c1/1
    no shutdown
  exit
  pxc 2 create
    port 1/1/c2/1
    no shutdown
  exit
-----
```

The preceding configuration automatically creates the following PXC sub-ports. In the following example, the following ports are cross-connected:

- pxc-1.a is cross-connected with pxc-1.b
- pxc-1.b is cross-connected with pxc-1.a
- pxc-2.a is cross-connected with pxc-2.b
- pxc-2.b is cross-connected with pxc-2.a

Example: MD-CLI

```
[ex:/configure]
A:admin@node-2# info
...
port pxc-1.a {
}
port pxc-1.b {
}
port pxc-2.a {
}
port pxc-2.b {
}
```

Example: classic CLI

```
A:node-2# admin display-config
...
#-----
echo "Port Configuration"
#-----
port pxc-1.a
exit
exit
port pxc-1.b
exit
exit
port pxc-2.a
exit
exit
port pxc-2.b
exit
exit
```

5.5 Bandwidth considerations and QoS

Bandwidth consumed by PXCs based on faceplate ports correlates with the faceplate's port capacity. Because each PXC allocates a faceplate port for exclusive use, the PXC capacity cannot exceed the card capacity that is already allocated for the faceplate ports. In other words, a PXC based on a faceplate port does not add any additional bandwidth to the forwarding complex.

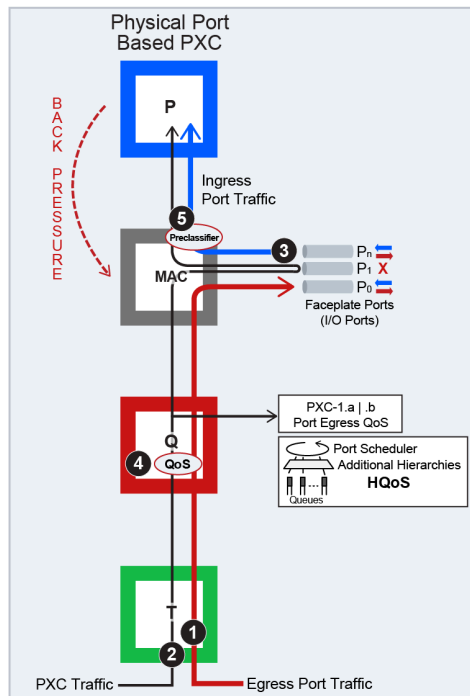
The bandwidth management in the PXC environment is performed through existing QoS mechanisms.

5.5.1 QoS

There are interactions between the PXC traffic and non-PXC traffic in the datapath. [Figure 19: Interaction between PXC and non-PXC traffic](#) shows this interaction as the traffic enters the egress forwarding path from the fabric tap (T). This traffic consists of non-PXC traffic (1) destined for the egress faceplate ports

and PXC traffic (2) that is sent (cross-connected) to the ingress forwarding path (P) within the same forwarding complex. Regular ingress traffic from the faceplate ports (3) is added to the stream and merged into the same ingress forwarding path as the PXC traffic.

Figure 19: Interaction between PXC and non-PXC traffic



sw4646

The physical port-based PXC configuration [Figure 19: Interaction between PXC and non-PXC traffic](#), shows interaction of the three traffic streams on the forwarding complex with a PXC based on the faceplate ports. To manage congestion, the user-configured input can be exerted in points 4 and 5.

Point 4 represents regular egress QoS in the traffic manager (Q) applied to an egress port. In this setup, the faceplate port P1 is reserved for PXC traffic which is represented by the two sub ports (PXC sub-ports **pxc-id.a** and **pxc-id.b**). Egress QoS is applied to each PXC subport.

Point 5 represents a pre-classifier in the MAC chip that manages ingress bandwidth if transient bursts occur in the ingress datapath (P), which then exerts back pressure toward the MAC. During congestion, the pre-classifier arbitrates between regular ingress traffic from the faceplate ports and the PXC traffic.

5.5.1.1 QoS on PXC sub-ports

The network user must understand the concept of the PXC sub-ports described in [Port-based PXC](#) for correct egress QoS configuration in the traffic manager (Q).

The following summarizes key points for the PXC sub ports:

- Each subport (**pxc-id.a** and **pxc-id.b**) in a PXC is, in the context of egress QoS, treated as a separate port with its own port scheduler policy.

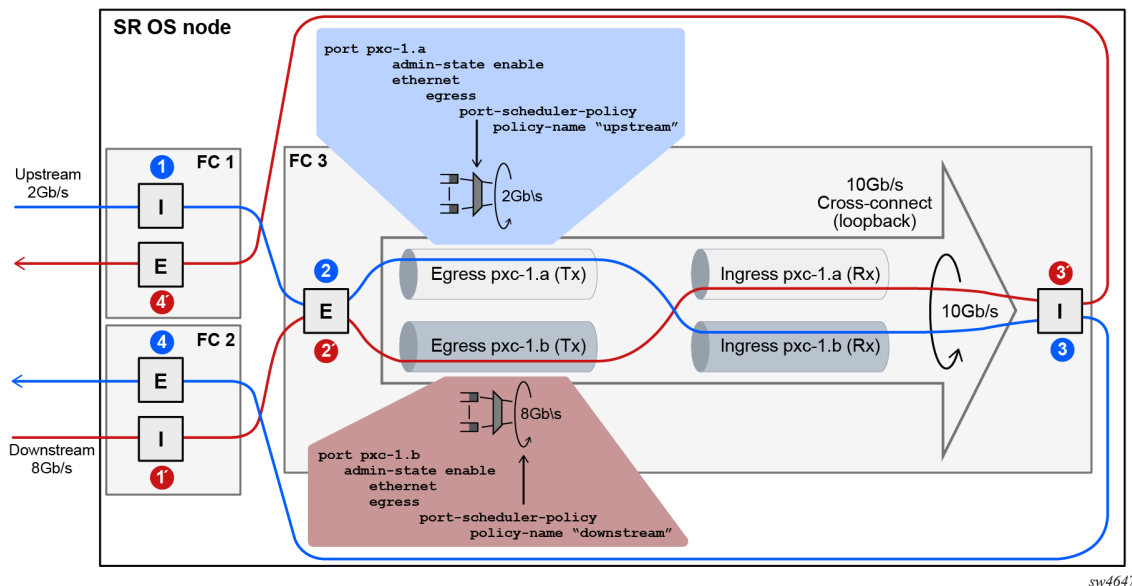
- Both sub-ports are created on top of the same loopback configuration (port-based or MAC-based). For faceplate ports, this bandwidth is determined by the port capabilities (for example, a 100 Gb/s port versus a 400 Gb/s port) and for the MAC loopback, this bandwidth is configurable.

Funneling traffic from two PXC sub-ports through the same loopback requires separate bandwidth management for each PXC sub-ports. The sum of the configured bandwidth caps for the Egress Port Scheduler (EPS) under the two PXC sub-ports should not exceed the bandwidth capacity of the underlying loopback. [Figure 20: Bandwidth management on PXC sub-ports](#) shows an example of this concept where each PXC sub-port is divided into two parts, the Tx or the egress part and the Rx or the ingress part. [Figure 20: Bandwidth management on PXC sub-ports](#) shows bidirectional traffic entering and exiting the SR node at forwarding complex 1 and 2, with PXC processing on forwarding complex 3. In the upstream direction, traffic enters SR node at the ingress forwarding complex 1 at point (1) and is redirected to the PXC for additional processing, points (2) and (3). From there, traffic is sent by the egress forwarding complex 2 out of the node, at point (4).

Similar logic can be followed in the downstream (opposite) direction where the traffic enters the ingress forwarding complex 2 at point (1'), it is redirected to the same PXC on forwarding complex 3 and exists the node on forwarding complex 1 at point (4').

In this example with the maximum loopback bandwidth of 10 Gb/s, port-schedulers under the PXC egress subports must be configured to support their respective anticipated bandwidth in each direction (2 Gb/s upstream and 8 Gb/s downstream), for the total bandwidth of 10 Gb/s supported on the cross-connect.

Figure 20: Bandwidth management on PXC sub-ports



Traffic traversing PXC contains an overhead of 4 bytes per packet that are attributed to the internal VLAN tag used for PXC sub-port identification within the SR node. However, these 4 bytes are not accounted for in the configured QoS rates. Therefore, the user should take this into consideration when configuring rates on QoS objects under PXC ports.

5.5.2 Queue allocation on PXC sub-ports

PXC sub-ports are auto-configured in hybrid mode and this cannot be changed by configuration. The PXC sub-ports each have a set of queues on the network egress side and a set of queues on the access egress and ingress (per SAP or ESM subscriber). Queues on network ingress are shared per FP or per MDA, as they are on non-PXC ports in hybrid mode.

Queue groups are allocated per PXC sub-ports.

5.5.3 Pool allocations on PXC ports

Queue buffers are created in buffer pools and are used for traffic buffering when queues are congested. Buffer pools are allocated per forwarding complex or per cross-connect.

Each cross-connect has three associated buffer pools:

- access ingress
- access egress
- network egress

The network ingress pool is shared between all faceplate ports on a forwarding complex. The size of the buffer pools is automatically determined by the system based on the forwarding complex type and cross-connect configuration.

5.6 Operational states

A port under a PXC (for example, port 1/1/c1/1), the PXC itself (PXC ID represented by the cross-connect port configuration port-xc pxc 1), and PXC sub-ports (for example, port pxc-1.a and pxc-1.b) all have administrative and operational states.

For a port-based PXC, when all layers of a PXC (PXC port, PXC ID, and PXC sub-ports) are operationally up, the faceplate port status LED on the faceplate blinks amber. The port activity LED lights green in the presence of traffic on PXC ports and turns off in the absence of traffic on PXC ports. The presence of the optical transceiver on the PXC has no effect on its operational state. Traffic cannot be sent out through the transceiver or be received through the transceiver from the outside. However, the existing traps related to insertion or removal of a transceiver (SFF Inserted/Removed) are supported. The "Signal-Fail" alarm on the PXC is suppressed.

The operational state of the PXC ID is derived from its administrative state and the operational state of the sub-ports.

The operational state of the PXC sub-ports is dependent on the operational state of the underlying port and the administrative state of the corresponding PXC ID.

5.7 PXC statistics

Two types of statistics can be collected on a regular, non-PXC Ethernet port:

- Low-level port statistics which provide information about conditions on the data-link layer and physical port, for example, the aggregate number of forwarded and dropped octets or bytes on the data-link layer (Layer 2 MAC), FCS errors, number of collisions, and so on. These statistics can be viewed with the **show port** command.
- Network-level statistics provide information about forwarded and dropped octets or packets on a per-queue level on network ports. These statistics can be viewed with the **show port detail** command.

5.7.1 Statistics on PXC ports

The statistics on the PXC ports are maintained only on the data-link layer (Layer 2 MAC). The internal Q-tag used for PXC sub-port identification within the router is included in the displayed octet count. The collected statistics represent the combined upstream and downstream traffic carried by the corresponding PXC sub-ports.

For example, in port level statistics output for a PXC port, the output count represents the upstream and downstream traffic flowing out of the faceplate port while the input count represents the same looped traffic returning into the same port.

```
show port 1/1/c1/1 detail
```

Output example

```
...
=====
Traffic Statistics
=====
```

	Input	Output
Octets	290164703	290164703
Packets	2712661	2712661
Errors	0	0

Statistics are cleared when a faceplate port is added or removed from the PXC.

Statistics collection to a local file is not supported for PXC ports.

Queues are not instantiated on the PXC ports, therefore, the network level (queue) statistics are not maintained in that location.

5.7.2 Statistics collection on PXC sub-ports and PXC LAG

PXC sub-ports (for example, pxc-1.a and pxc-1.b) provide aggregated network-level statistics (queue statistics). Physical-level statistics are not supported on PXC sub-ports because these ports do not relay on MAC statistics.

The statistics on a PXC sub-port are aggregated counts of all queues in each traffic direction for the following:

- forwarded packets
- forwarded octets
- dropped packets
- dropped octets

The statistics collection is triggered on demand at the time of executing either of the following commands.

```
show port pxc-1.a statistics queue-aggregate
monitor port pxc-1.a interval 30 0 aggregate-queue
```

The collected statistics are cached for 30 seconds. If multiple consecutive executions of these commands occur within the 30 second period, the statistics counters remains unchanged from the previous reads. Therefore, the minimum interval between two executions of the following command should be at least 30 seconds apart.

```
show port statistics queue-aggregate
```

Examples for PXC statistics on individual PXC sub-ports

Use the following command to display aggregate queue statistics.

```
show port pxc-1.a statistics aggregate-queue
```

Output example

```
=====
Port Statistics on Slot 1
=====
Port-id          Ingress Packets Fwd   Ingress Octets Fwd
                  Ingress Packets Drop Ingress Octets Drop
                  Egress Packets Fwd   Egress Octets Fwd
                  Egress Packets Drop   Egress Octets Drop
-----
pxc1.a           4654649               94523288
                  22544                 99852
                  98652214              65889554
                  55451                 22144
=====
```

Use the following command to display aggregate queue statistics with the **interval** and **repeat** option.

```
monitor port pxc-1.a interval 30 repeat 10 aggregate-queue
```

Output example

```
=====
Monitor statistics for port pxc-1.a
=====
Ingress Packets Fwd   Ingress Octets Fwd
Ingress Packets Drop   Ingress Octets Drop
Egress Packets Fwd     Egress Octets Fwd
Egress Packets Drop     Egress Octets Drop
-----
At time t = 0 sec (Base Statistics)
-----
                4654649       94523288
                22544         99852
                98652214      65889554
                55451         22144
-----
At time t = 30 sec (Mode: Delta)
-----
```

4654649	94523288
22544	99852
98652214	65889554
55451	22144

At time t = 60 sec (Mode: Delta)	

4654649	94523288
22544	99852
98652214	65889554
55451	22144

Use the following command to display aggregate queue statistics with the **interval**, **repeat**, and **rate** option.

```
monitor port pxc-1.a interval 30 repeat 10 rate aggregate-queue
```

Output example

=====		
Monitor statistics for port pxc-1.a		
=====		
	Input	Output

At time t = 0 sec (Base Statistics)		

Forwarded Packets	454649	94288
Forwarded Bytes	3343434	777998

At time t = 30 sec		

Rate [kbps]	4654649	94288
Utilization (% of port capacity)	22.54	9.98

At time t = 60 sec		

Rate [kbps]	4654649	94288
Utilization (% of port capacity)	22.54	9.98

Examples for PXC statistics on PXC LAG

Use the following command to display aggregate queue statistics on PXC LAG.

```
show lag 1 statistics aggregate-queue
```

Output example

=====			
LAG Statistics			
=====			
Description : N/A			

Port-id	Ingress Packets Fwd	Ingress Octets Fwd	
	Ingress Packets Drop	Ingress Octets Drop	
	Egress Packets Fwd	Egress Octets Fwd	

	Egress Packets Drop	Egress Octets Drop
pxc1.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
pxc2.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
Totals	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144

Use the following command to display aggregate queue statistics with the **interval** and **repeat** option.

```
monitor lag 1 interval 30 repeat 10 aggregate-queues
```

Output example

```
=====
Monitor statistics for LAG ID 1
=====
```

Port-id	Ingress Packets Fwd Ingress Packets Drop	Ingress Octets Fwd Ingress Octets Drop
	Egress Packets Fwd Egress Packets Drop	Egress Octets Fwd Egress Octets Drop

```
-----
```

At time t = 0 sec (Base Statistics)

```
-----
```

pxc1.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
pxc2.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
Totals	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144

```
-----
```

At time t = 30 sec (Mode: Delta)

```
-----
```

pxc1.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
pxc2.a	4654649	94523288
	22544	99852

	98652214	65889554
	55451	22144

Totals	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144

At time t = 60 sec (Mode: Delta)		

pxc1.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144
pxc2.a	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144

Totals	4654649	94523288
	22544	99852
	98652214	65889554
	55451	22144

Use the following command to display aggregate queue statistics with the **interval**, **repeat**, and **rate** option.

```
monitor lag 1 interval 30 repeat 10 rate aggregate-queues
```

Output example

=====		
Monitor statistics for LAG ID 1		
=====		
Port-id	Ingress Rate [kbps]	Egress Rate [kbps]
	Ingress Utilization	Egress Utilization
	% of port capacity)	(% of port capacity)

At time t = 0 sec (Base Statistics)		

pxc1.a	0	0
	0	0
pxc2.a	0	0
	0	0

Totals	4654649	94523288
	22.44	17.52

At time t = 30 sec (Mode: Delta)		

pxc1.a	4654649	94523288
	25.44	10.85

pxc2.a	4654649 22.44	94523288 11.52

Totals	4654649 22.44	94523288 17.52

At time t = 60 sec (Mode: Delta)		

pxc1.a	4654649 25.44	94523288 10.85

pxc2.a	4654649 22.44	94523288 11.52

Totals	4654649 22.44	94523288 17.52

5.7.2.1 MIBs

PXC sub-ports statistics are represented in a MIB table `tmnxPortAggQueueStatsTable` which is defined in `TIMETRA-PORT-MIB.mib` with the following entries.

```

TmnxPortAggQueueStatsEntry      ::= SEQUENCE
{
    tmnxPortAggQueueIngPktsFwd    Counter64,
    tmnxPortAggQueueIngOctsFwd    Counter64,
    tmnxPortAggQueueIngPktsDrop   Counter64,
    tmnxPortAggQueueIngOctsDrop   Counter64,
    tmnxPortAggQueueEgrPktsFwd    Counter64,
    tmnxPortAggQueueEgrOctsFwd    Counter64,
    tmnxPortAggQueueEgrPktsDrop   Counter64,
    tmnxPortAggQueueEgrOctsDrop   Counter64,
    tmnxPortLastClearedTime        TimeStamp,
    tmnxPortLastFetchedTime        TimeStamp
}

```

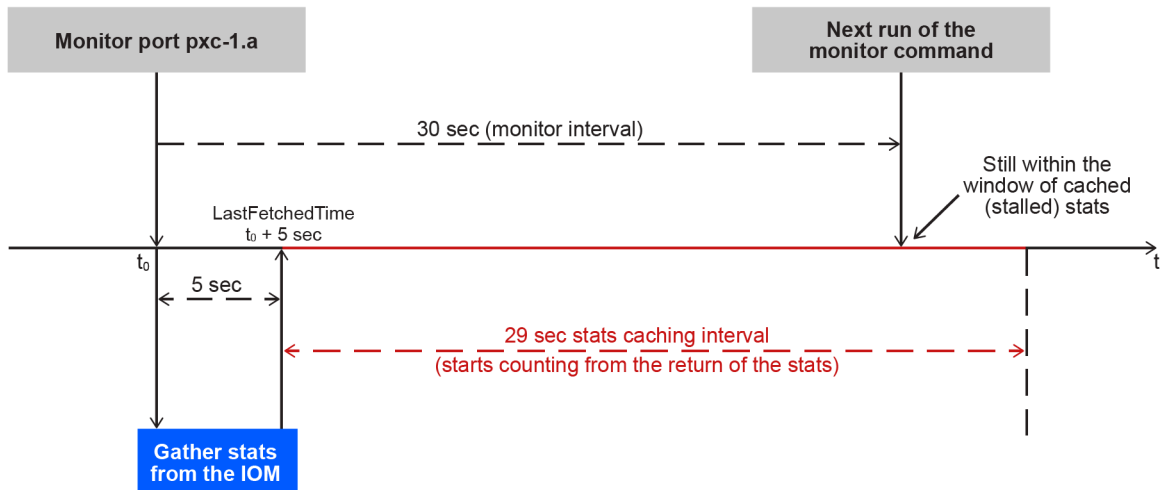
5.7.2.2 Restrictions

The following items describe monitor port restriction.

- Monitor port command allows monitoring of five simultaneous ports. Mixing of PXC and regular ports in the same monitor command is not supported.
- When monitoring ports with a large number of queues, it is possible that the longer time needed for statistics collection may lead to every other output of the monitor command displaying all zeros. This is particularly true at shorter monitoring intervals, such as the minimum of 30 seconds. To ensure consistent non-zero outputs, Nokia recommends gradually increasing the monitoring interval. The recommended monitoring interval with larger number of queues is 60 seconds.

The following diagram illustrates this issue.

Figure 21: Monitor port interval issue

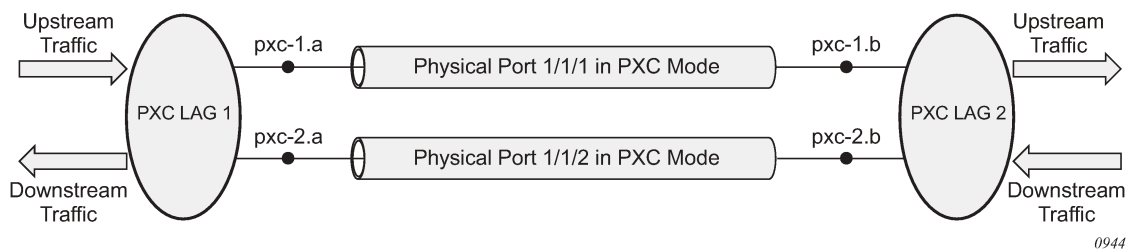


sw4219

5.8 PXC LAG

PXC sub-ports can be aggregated into a PXC LAG for increased capacity and card redundancy. A logical concept of a PXC LAG is shown in [Figure 22: Logical concept of a LAG on PXC ports](#).

Figure 22: Logical concept of a LAG on PXC ports



0944

Although the configuration allows for a mix of port-based PXCs and MAC-based PXCs in a LAG, the configuration should be used in a production network only during a short migration period when transitioning from one type of PXC to the other. Outside of the migration, the PXCs in a LAG should be of the same type, for example, a LAG should contain only port-based PXCs or only MAC-based PXCs but not both.

The LAGs on PXC ports must be configured in pairs as shown in the following example.

Example: MD-CLI

```
[ex:/configure]
A:admin@node-2# info
...
lag "lag-1" {
```

```

        description "lag in the up direction"
        port pxc-1.a {
        }
        port pxc-2.a {
        }
    }
    lag "lag-2" {
        description "lag in the down direction"
        port pxc-1.b {
        }
        port pxc-2.b {
        }
    }
}

```

Example: classic CLI

```

A:node-2# configure lag 1
A:node-2>config>lag$ info
-----
        description "lag in the up direction"
        port pxc-1.a
        port pxc-2.a
-----
A:node-2# configure lag 2
A:node-2>config>lag$ info
-----
        description "lag in the down direction"
        port pxc-1.b
        port pxc-2.b
        no shutdown
-----

```

Within the router, the two sides of the PXC LAG (LAG 1 and LAG 2 in the example configuration) are not aware of their interconnection. As a result, the operational state of one side of the PXC LAG is not influenced by the state of the PXC LAG on the other side.

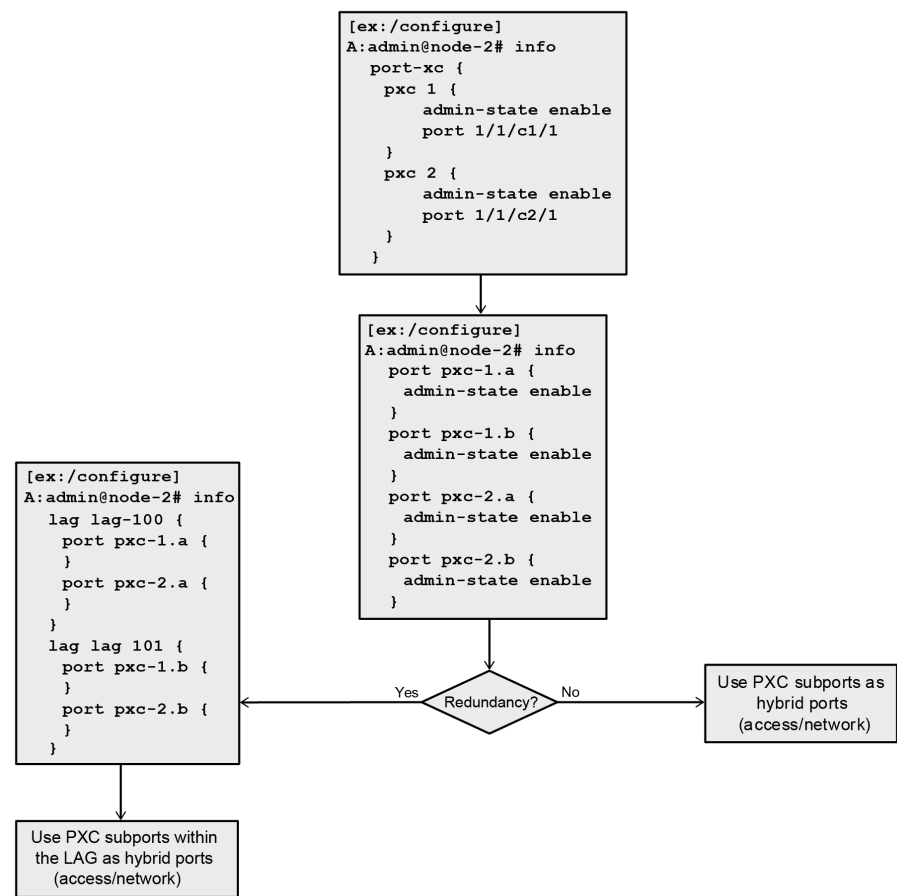
PXC sub-ports in a LAG must have the same properties (such as the same speed). Mixing PXC sub-ports and non-PXC ports is not allowed. The first port added to a LAG determines the type of LAG (PXC or non-PXC).

Statistics in the output of the **show lag statistics** command represent combined traffic carried over the referenced LAG and its pair (lag 1 and lag 2 in the above example).

5.9 Basic PXC provisioning

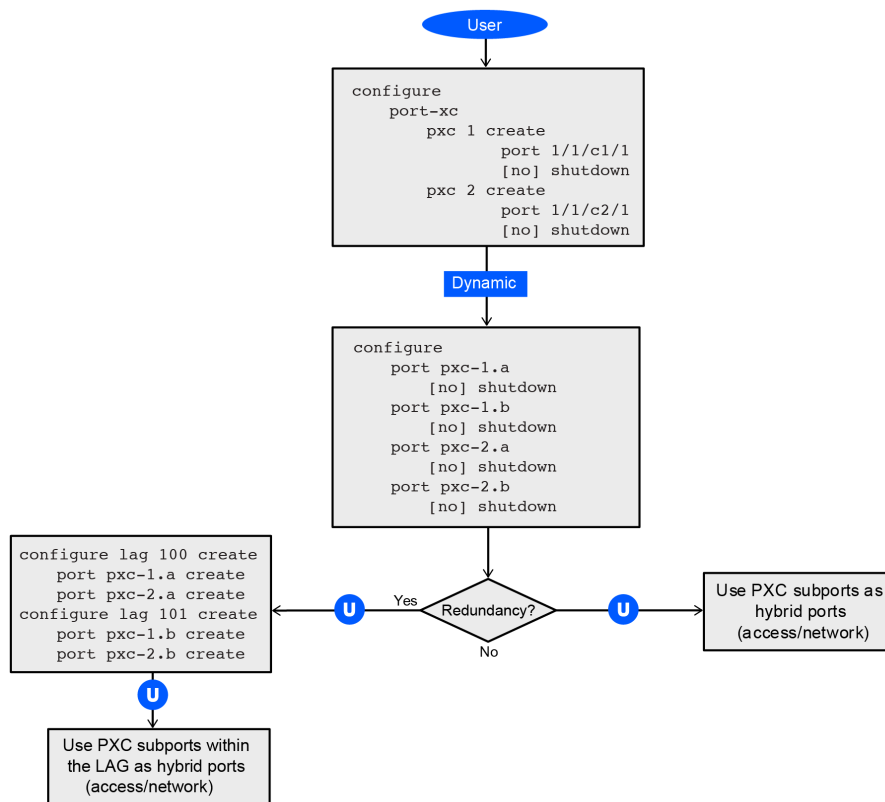
The CLI configuration flow example shown in the following figures represents a PXC configuration based on the faceplate port. The oval marked "User" represents a configuration step that the user must perform. The block marked "Dynamic" represents a step that the system performs automatically without a user's assistance.

Figure 23: MD-CLI flow



sw4648

Figure 24: Classic CLI flow



sw4649

5.10 PXC mirroring and LI

Traffic on a PXC sub-port can be mirrored or lawfully intercepted (LI). For example, subscriber "Annex1" traffic arriving on a PXC sub-port is mirrored if "Annex1" is configured as a mirror or LI source. A PXC sub-port can also be used to transmit mirror and LI traffic out from a mirror-destination service (such as a mirror-dest SAP or SDP can egress out a PXC sub-port, or a routable LI encapsulated packet can be forwarded and transmitted out a PXC sub-port).

A mirror destination can be configured to transmit mirrored and LI traffic out of a SAP on a PXC sub-port that is then cross connected into a VPLS service where a VXLAN encapsulation is added to the mirrored packets before transmission out of the node.

The internal Q-tag that represents the PXC sub-port within the system is included in the lawfully intercepted copy of the packet for traffic intercepted (mirrored) on the ingress side of a PXC sub-port, when the associate mirror-dest service is of type **ether** (the default) with routable lawful interception encapsulation in the following context.

Use the following command to configure a mirror destination to transmit mirrored and LI traffic from a SAP on a PXC sub-port.

```
configure mirror mirror-dest encap
```

See the *7705 SAR Gen 2 OAM and Diagnostics Guide* for information about LI.

5.11 Configuration example

The following example shows how a PXC port is used to transport an Epipe service over IPsec tunnels when the 7705 SAR Gen 2 is operating as an IPsec SecGW head-end router (for example, when aggregating Epipes over IPsec from a group of 7705 SAR-Hm series of routers). The Epipe service uses GRE transport, where this network egress traffic is sent over the PXC port and into a VPRN SAP configured on the other side of the PXC port. The VPRN SAP ingress traffic is routed over the relevant IPsec tunnel to reach the respective 7705 SAR-Hm series router that terminates the Epipe, using its own VPRN and PXC port to terminate the IPsec tunnel and GRE transport packets.

Example: MDA and port configuration (MD-CLI)

```
[ex:/configure]
A:admin@node-2# info
...
  card 1 {
    mda 2 {
      mda-type isa-tunnel-v
    }
    mda 3 {
      mda-type isa-bb-v
    }
  }
  port 1/1/c10 {
    admin-state enable
    connector {
      breakout c1-10g
    }
  }
  port 1/1/c10/1 {
    admin-state enable
    ethernet {
      mode hybrid
    }
  }
  port 1/1/c11 {
    admin-state enable
    connector {
      breakout c1-1g
    }
  }
  port 1/1/c11/1 {
    admin-state enable
    ethernet {
      mode hybrid
    }
  }
  port 1/1/c12 {
    admin-state enable
    connector {
      breakout c1-1g
    }
  }
  port 1/1/c12/1 {
    admin-state enable
    ethernet {
      mode access
    }
  }
}
```

```

        encap-type dot1q
    }
}
...

```

Example: Faceplate (physical) port configuration on cards 3 and 4 (classic CLI)

```

[ex:/configure]
A:admin@node-2# info
...
  card 1
    card-type iom-sar
    mda 1
      mda-type m10-sfp++6-sfp
      no shutdown
    exit
    mda 2
      mda-type isa-tunnel-v
      no shutdown
    exit
  exit
port 1/1/c10
  connector
  breakout c1-10g
  exit
  no shutdown
exit
port 1/1/c11
  connector
  breakout c1-1g
  exit
  no shutdown
exit
port 1/1/c12
  connector
  breakout c1-1g
  exit
  no shutdown
exit
port 1/1/c10/1
  ethernet
  mode hybrid
  exit
  no shutdown
exit
port 1/1/c11/1
  ethernet
  mode hybrid
  encap-type dot1q
  exit
  no shutdown
exit
port 1/1/c12/1
  ethernet
  mode access
  encap-type dot1q
  exit
  no shutdown
exit
...

```

Example

The user must manually configure the sub-port encapsulation (the default is dot1q). PXC sub-ports transparently pass traffic with preserved QinQ tags from the .b side of the PXC to the .a side of the PXC and the other way around.

Example: PXC and PXC sub-port configuration (MD-CLI)

```
[ex:/configure port-xc]
A:admin@node-2# info
  pxc 1 {
    admin-state enable
    port-id 1/1/c10/1
  }
  port pxc-1.a {
    admin-state enable
    description "VPRN for IPsec; SAP private interface configured here"
  }
  port pxc-1.b {
    admin-state enable
    description "GRE orig/term PXC side; NW interface will be configured here"
  }
```

Example: PXC and PXC sub-port configuration (classic CLI)

```
A:node-2>config>port-xc# info
-----
      pxc 1 create
        port 1/1/c10/1
        no shutdown
      exit
A:node-2>config# info
...
  port pxc-1.a
    description "VPRN for IPsec; SAP private interface configured here"
    ethernet
    exit
    no shutdown
  exit
  port pxc-1.b
    description "GRE orig/term PXC side; NW interface will be configured here"
    ethernet
    exit
    no shutdown
  exit
...
-----
```

Example: Configuration of the router interface on PXC for GRE transport (MD-CLI)

```
A:admin@node-2>config>router# info
...
  interface "toVPRN100" {
    port pxc-1.b:200
    ipv4 {
      primary {
        address 200.200.200.1
        prefix-length 30
        gre-termination true
      }
    }
  }
```

```

    }
  }
...

```

Example: Configuration of the router interface on PXC for GRE transport (classic CLI)

```

A:admin@node-2>config>router# info
...
    interface "toVPRN100"
      address 200.200.200.1/30 gre-termination
      port pxc-1.b:200
      no shutdown
    exit
...

```

Example

See the *7705 SAR Gen 2 Multiservice ISA and ESA Guide* for more information about IPsec configuration. See the *7705 SAR Gen 2 Layer 2 Services and EVPN Guide* for more information about BGP-VPWS Epipe service configuration.

Example: Configuration IPsec SecGW and the E-pipe and VPRN services (MD-CLI)

```

[ex:/configure service]
A:admin@node-2# info
...
    pw-template "3084" {
      allow-fragmentation true
      auto-gre-sdp true
    }
    epipe "VLL-Hmc03" {
      admin-state enable
      service-id 10573
      customer "1"
      bgp 1 {
        route-distinguisher "65001:10573"
        route-target {
          export "target:65001:10573"
          import "target:65001:10573"
        }
        pw-template-binding "3084" {
        }
      }
      bgp-vpws {
        admin-state enable
        local-ve {
          name "HEAD-END"
          id 8432
        }
        remote-ve "Hmc" {
          id 3284
        }
      }
      sap 1/1/c12/1:1101 {
      }
    }
    ies "10" {
      admin-state enable
      customer "1"
      interface "public-100" {
        sap tunnel-1.public:2 {
          ipsec-gateway "gw-vprn-100" {

```

```

        admin-state enable
        default-tunnel-template 2
        ike-policy 2
        pre-shared-key "HUWumavMAgAr6Q6r7PYdDE01aJC8 hash2"
        default-secure-service {
            service-name "VPRN_100"
            interface "Int-private-100"
        }
        local {
            gateway-address 10.107.2.1
        }
    }
}
ipv4 {
    primary {
        address 10.107.2.0
        prefix-length 31
    }
}
}
}
vprn "VPRN_100" {
    admin-state enable
    service-id 100
    customer "1"
    autonomous-system 65001
    ipsec {
        security-policy 1 {
            entry 10 {
                local-ip {
                    any true
                }
                remote-ip {
                    any true
                }
            }
        }
    }
    interface "Int-private-100" {
        tunnel true
        sap tunnel-1.private:2 {
        }
    }
    interface "toGRT" {
        ipv4 {
            primary {
                address 200.200.200.2
                prefix-length 30
            }
        }
        sap pxc-1.a:200 {
        }
    }
    static-routes {
        route 100.100.100.0/24 route-type unicast {
            description "to 7705 SAR-Hm series routers"
            interface "Int-private-100" {
                admin-state enable
            }
        }
    }
}
}
}
...

```

Example: Configuration IPsec SecGW and the E-pipe and VPRN services (classic CLI)

```

A:node-2>config>service$ info
...
pw-template 3084 name "3084" auto-gre-sdp create
  allow-fragmentation
exit
ies 10 name "10" customer 1 create
  interface "public-100" create
  exit
exit
vprn 100 name "VPRN_100" customer 1 create
  interface "toGRT" create
  exit
  interface "Int-private-100" tunnel create
  exit
exit
ies 10 name "10" customer 1 create
  interface "public-100" create
    address 10.107.2.0/31
    tos-marking-state untrusted
    sap tunnel-1.public:2 create
      ipsec-gw "gw-vprn-100"
        default-secure-service name "VPRN_100" interface "Int-private-100"
        default-tunnel-template 2
        ike-policy 2
        local-gateway-address 10.107.2.1
        pre-shared-key "HUWumavMAgAr6Q6r7PYdDE01aJC8" hash2
        no shutdown
      exit
    exit
  exit
  no shutdown
exit
vprn 100 name "VPRN_100" customer 1 create
  ipsec
    security-policy 1 create
      entry 10 create
        local-ip any
        remote-ip any
      exit
    exit
  exit
  autonomous-system 65001
  interface "toGRT" create
    address 200.200.200.2/30
    sap pxc-1.a:200 create
  exit
  exit
  interface "Int-private-100" tunnel create
    sap tunnel-1.private:2 create
  exit
  exit
  static-route-entry 100.100.100.0/24
    next-hop "Int-private-100"
    no shutdown
  exit
  exit
  no shutdown
exit
epipe 10573 name "VLL-Hmc03" customer 1 create
  service-mtu 1614
  bgp 1
    route-distinguisher 65001:10573

```

```
        route-target export target:65001:10573 import target:65001:10573
        pw-template-binding 3084
        exit
    exit
    bgp-vpws
        ve-name "HEAD-END"
        ve-id 8432
        exit
        remote-ve-name "Hmc"
        ve-id 3284
        exit
        no shutdown
    exit
    sap 1/1/c12/1:1101 create
        no shutdown
    exit
    no shutdown
exit
exit
-----
```

6 LAG

A Link Aggregation Group (LAG), based on the IEEE 802.1ax standard (formerly 802.3ad), increases the bandwidth available between two network devices by grouping multiple ports to form one logical interface.

Traffic forwarded to a LAG by the router is load balanced between all active ports in the LAG. The hashing algorithm deployed by Nokia routers ensures that packet sequencing is maintained for individual sessions. Load balancing for packets is performed by the hardware, which provides line rate forwarding for all port types.

LAGs can be either statically configured or formed dynamically with Link Aggregation Control Protocol (LACP). A LAG on the 7705 SAR Gen 2 can consist of same-speed ports only.

All ports within a LAG must be of the same Ethernet type (access, network, or hybrid) and have the same encapsulation type (dot1q, QinQ, or null).

The following is an example of static LAG configuration using dot1q access ports.

Example: MD-CLI

```
[ex:/configure lag "lag-1"]
A:admin@node-2# info
  admin-state enable
  encap-type dot1q
  mode access
  port 1/1/1 {
  }
  port 1/1/2 {
  }
```

Example: classic CLI

```
A:node-2>config>lag# info
-----
  mode access
  encap-type dot1q
  port 1/1/1
  port 1/1/2
  no shutdown
-----
```

6.1 LACP

The LACP control protocol, defined by the IEEE 802.3ad standard, specifies the method by which two devices establish and maintain LAGs. When LACP is enabled, SR OS automatically associates LACP-compatible ports into a LAG.

The following is an example of LACP LAG configuration using network ports and a default null encapsulation type.

Example: MD-CLI

```
[ex:/configure lag "lag-2"]
A:admin@node-2# info
  admin-state enable
  mode network
  lacp {
    mode active
    administrative-key 32768
  }
  port 1/1/3 {
  }
  port 1/1/4 {
  }
```

Example: classic CLI

```
A:node-2>config>lag# info
-----
  mode network
  port 1/1/3
  port 1/1/4
  lacp active administrative-key 32768
  no shutdown
-----
```

6.1.1 LACP multiplexing

The router supports two modes of multiplexing RX/TX control for LACP: coupled and independent.

In coupled mode (default), both RX and TX are enabled or disabled at the same time whenever a port is added or removed from a LAG group.

In independent mode, RX is first enabled when a link state is up. LACP sends an indication to the far-end system that it is ready to receive traffic. Upon receiving this indication, the far-end system can enable TX. Therefore, in independent RX/TX control, LACP adds a link into a LAG only when it detects that the other end is ready to receive traffic. This minimizes traffic loss that may occur in coupled mode if a port is added into a LAG before notifying the far-end system or before the far-end system is ready to receive traffic. Similarly, on link removals from LAG, LACP turns off the distributing and collecting bit and informs the far end about the state change. This allows the far-end side to stop sending traffic as soon as possible.

Independent control provides lossless operation for unicast traffic in most scenarios when adding new members to a LAG or when removing members from a LAG. It also reduces loss for multicast and broadcast traffic.



Note: Independent and coupled mode are interoperable (connected systems can have either mode set).

Independent and coupled modes are supported when using PXC ports, however, Nokia recommends independent mode, as it provides significant performance improvements.

6.1.2 LACP tunneling

LACP tunneling is supported on Epipe and VPLS services. In a VPLS service, the Layer 2 control frames are sent out of all the SAPs configured in the VPLS. This feature should only be used when a VPLS emulates an end-to-end Epipe service. That is, an Epipe is configured using a three-point VPLS, with one access SAP and two access-uplink SAPs or SDPs for redundant connectivity.



Note: Nokia does not recommend using LACP tunneling if the VPLS is used for multipoint connectivity.

When a Layer 2 control frame is forwarded out of a dot1q SAP or a QinQ SAP, the SAP tags of the egress SAP are added to the packet.

Depending on the port encapsulation, the following SAPs can be configured for tunneling the untagged LACP frames; the corresponding protocol tunneling must be enabled on the port:

- If the port encapsulation is null, a null SAP can be configured on a port to tunnel these packets.
- If the port encapsulation is dot1q, either a dot1q explicit null SAP (for example, 1/1/10:0) or a dot1q default SAP (for example, 1/1/11:*) can be used to tunnel these packets.
- If the port encapsulation is QinQ, a 0.* SAP (for example, 1/1/10:0.*) can be used to tunnel these packets.

LAG port states may be impacted if LACP frames are lost because of incorrect prioritization and congestion in the network carrying the tunnel.

6.2 LAG sub-group

LAG can provide active/standby redundancy by logically dividing LAG into sub-groups. The LAG is divided into sub-groups by either assigning each LAG's ports to an explicit sub-group (1 by default), or by automatically grouping all LAG's ports residing on the same line card into a unique sub-group (auto-iom) or by automatically grouping all LAG's ports residing on the same MDA into a unique sub-group (auto-mds).

When a LAG is divided into sub-groups, only a single sub-group is elected as active. Which sub-group is selected depends on the LAG selection criteria.

The standby state of a port in the LAG is communicated to the remote end using the LAG standby signaling, which can be either **lacp** for LACP LAG or **best-port** for static LAG. The following applies for standby state communication:

- **lacp**

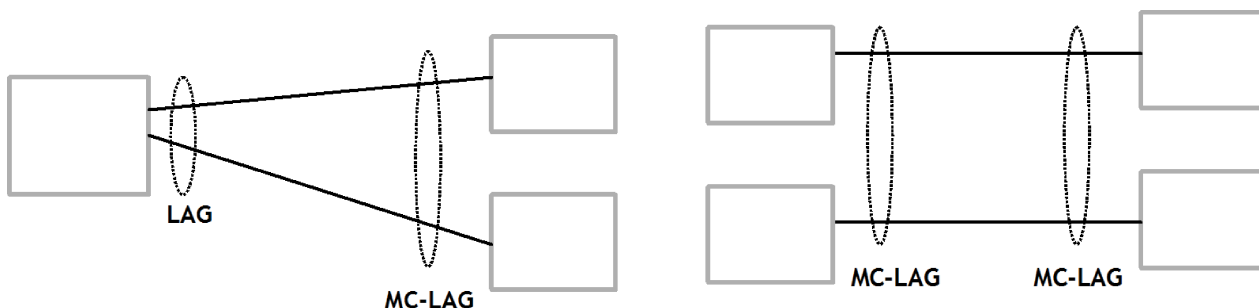
The standby state of a port is communicated to the remote system using the LACP protocol.

- **best-port**

The standby state of a port is communicated by switching the transmit laser off. This requires the LAG to be configured using **selection-criteria best-port** and **standby-signaling power-off**.

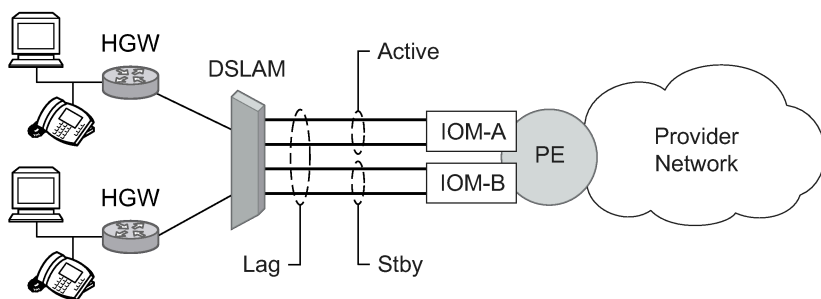
[Figure 25: Active/standby LAG operation deployment examples](#) shows how LAG in active/standby mode can be deployed toward a DSLAM access using sub-groups with auto-iom sub-group selection. LAG links are divided into two sub-groups (one per line card).

Figure 25: Active/standby LAG operation deployment examples



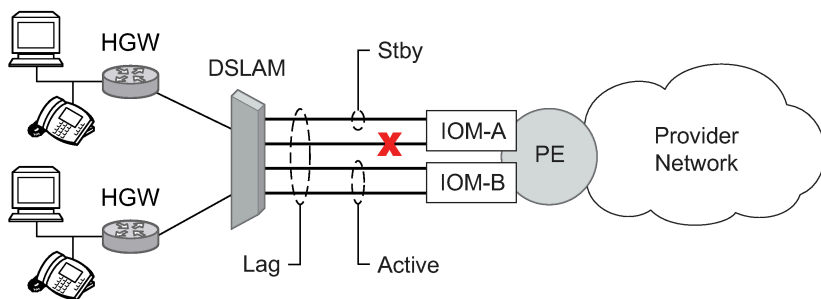
In case of a link failure, as shown in [Figure 26: LAG on access interconnection](#) and [Figure 27: LAG on access failure switchover](#), the switch over behavior ensures that all LAG-members connected to the same IOM as failing link become standby and LAG-members connected to other IOM become active. This way, QoS enforcement constraints are respected, while the maximum of available links is used.

Figure 26: LAG on access interconnection



OSSG095

Figure 27: LAG on access failure switchover



OSSG096

6.3 Traffic load balancing options

When a requirement exists to increase the available bandwidth for a logical link that exceeds the physical bandwidth or add redundancy for a physical link, typically one of two methods is applied: equal cost multi-

path (ECMP) or Link Aggregation (LAG). A system can deploy both at the same time using ECMP of two or more Link Aggregation Groups (LAG) or single links, or both.

Different types of hashing algorithms can be employed to achieve one of the following objectives:

- ECMP and LAG load balancing should be influenced solely by the offered flow packet. This is referred to as *per-flow* hashing.
- ECMP and LAG load balancing should maintain consistent forwarding within a specific service. This is achieved using *consistent per-service* hashing.
- LAG load balancing should maintain consistent forwarding on egress over a single LAG port for a specific network interface, SAP, and so on. This is referred as *per link* hashing (including explicit per-link hashing with LAG link map profiles). Note that if multiple ECMP paths use a LAG with per-link hashing, the ECMP load balancing is done using either *per flow* or *consistent per service* hashing.

These hashing methods are described in the following subsections. Although multiple hashing options may be configured for a specific flow at the same time, only one method is selected to hash the traffic based on the following decreasing priority order:

For ECMP load balancing:

1. Consistent per-service hashing
2. Per-flow hashing

For LAG load balancing:

1. LAG link map profile
2. Per-link hash
3. Consistent per-service hashing
4. Per-flow hashing

6.3.1 Per-flow hashing

Per-flow hashing uses information in a packet as an input to the hash function ensuring that any specific flow maps to the same egress LAG port/ECMP path. Note that because the hash uses information in the packet, traffic for the same SAP/interface may be sprayed across different ports of a LAG or different ECMP paths. If this is not wanted, other hashing methods described in this section can be used to change that behavior. Depending on the type of traffic that needs to be distributed into an ECMP or LAG, or both, different variables are used as input to the hashing algorithm that determines the next hop selection. The following describes default per-flow hashing behavior for those different types of traffic:

- VPLS known unicast traffic is hashed based on the IP source and destination addresses for IP traffic, or the MAC source and destination addresses for non-IP traffic. The MAC SA/DA are hashed and then, if the Ethertype is IPv4 or IPv6, the hash is replaced with one based on the IP source address/destination address.
- VPLS multicast, broadcast and unknown unicast traffic.
 - Traffic transmitted on SAPs is not sprayed on a per-frame basis, but instead, the service ID selects ECMP and LAG paths statically.
 - Traffic transmitted on SDPs is hashed on a per packet basis in the same way as VPLS unicast traffic. However, per packet hashing is applicable only to the distribution of traffic over LAG ports, as the ECMP path is still chosen statically based on the service ID.

Data is hashed twice to get the ECMP path. If LAG and ECMP are performed on the same frame, the data is hashed again to get the LAG port (three hashes for LAG). However, if only LAG is performed, then hashing is only performed twice to get the LAG port.

- Multicast traffic transmitted on SAPs with IGMP snooping enabled is load-balanced based on the internal multicast ID, which is unique for every (s,g) record. This way, multicast traffic pertaining to different streams is distributed across different LAG member ports.
- The hashing procedure that used to be applied for all VPLS BUM traffic would result in PBB BUM traffic being sent out on BVPLS SAP to follow only a single link when MMRP was not used. Therefore, traffic flooded out on egress BVPLS SAPs is now load spread using the algorithm described above for VPLS known unicast.
- Unicast IP traffic routed by a router is hashed using the IP SA/DA in the packet.
- MPLS packet hashing at an LSR is based on the whole label stack, along with the incoming port and system IP address. Note that the EXP/TTL information in each label is not included in the hash algorithm. This method is referred to as *Label-Only Hash* option and is enabled by default, or can be re-instated in CLI by entering the *lbl-only* option. A few options to further hash on the headers in the payload of the MPLS packet are also provided.
- VLL traffic from a service access point is not sprayed on a per-packet basis, but as for VPLS flooded traffic, the service ID selects one of the ECMP/LAG paths. The exception to this is when shared-queuing is configured on an Epipe SAP, or Lpipe SAP, or when H-POL is configured on an Epipe SAP. In those cases, traffic spraying is the same as for VPLS known unicast traffic. Packets of the above VLL services received on a spoke SDP are sprayed the same as for VPLS known unicast traffic.
- Note that Cpipe VLL packets are always sprayed based on the service-id in both directions.
- Multicast IP traffic is hashed based on an internal multicast ID, which is unique for every record similar to VPLS multicast traffic with IGMP snooping enabled.

If the ECMP index results in the selection of a LAG as the next hop, then the hash result is hashed again and the result of the second hash is input to the modulo like operation to determine the LAG port selection.

When the ECMP set includes an IP interface configured on a spoke SDP (IES/VP RN spoke interface), or a Routed VPLS spoke SDP interface, the unicast IP packets—which is sprayed over this interface—is not further sprayed over multiple RSVP LSPs/LDP FEC (part of the same SDP), or GRE SDP ECMP paths. In this case, a single RSVP LSP, LDP FEC next-hop or GRE SDP ECMP path is selected based on a modulo operation of the service ID. In case the ECMP path selected is a LAG, the second round of the hash, hashes traffic based on the system, port or interface load-balancing settings.

In addition to the above described per-flow hashing inputs, the system supports multiple options to modify default hash inputs.

6.3.1.1 LSR hashing

By default, the LSR hash routine operates on the label stack only. However, the system also offers the ability to hash on the IP header fields of the packet for the purpose of spraying labeled IP packets over ECMP paths in an LSP or over multiple links of a LAG group.

The LSR hashing options can be selected using the following system-wide command.

```
configure system load-balancing lsr-load-balancing
```

LSR label-only hash

The system hashes the packet using the labels in the MPLS stack and the incoming port (**port-id**). In the presence of entropy label, the system uses only the entropy label and does not use the incoming port (**port-id**) for the hash calculation.

The net result is used to select which LSP next hop to send the packet to using a modulo operation of the hash result with the number of next hops.

This same result feeds to a second round of hashing if there is LAG on the egress port where the selected LSP has its NHLFE programmed.

Use the following command to enable the label-only hash option.

```
configure system load-balancing lsr-load-balancing lbl-only
```

LSR label-IP hash

In the first hash round for ECMP, the algorithm parses down the label stack and after it reaches the bottom, it checks the next nibble. If the nibble value is 4, it assumes it is an IPv4 packet. If the nibble value is 6, it assumes it is an IPv6 packet. In both cases, the result of the label hash is fed into another hash along with source and destination address fields in the IP packet header. Otherwise, the algorithm uses the label stack hash already calculated for the ECMP path selection.



Note: Enable the control word for Layer 2 services to avoid hashing based on incorrect parameters in cases where the ETH header in the ETHoMPLS packets ingressing at the LSR has the first nibble set to 4 or 6.

The second round of hashing for LAG re-uses the net result of the first round of hashing.

Use the following command to enable the label-IP hash option.

```
configure system load-balancing lsr-load-balancing lbl-ip
```

LSR IP-only hash

This option behaves like the label-IP hash option, except that when the algorithm reaches the bottom of the label stack in the ECMP round and finds an IP packet, it throws the outcome of the label hash and only uses the source and destination address fields in the IP packet header.

Use the following command to enable the IP-only hash option.

```
configure system load-balancing lsr-load-balancing ip-only
```

LSR Ethernet encapsulated IP hash

This option behaves like LSR IP-only hash, except for how the IP SA/DA information is found.

After the bottom of the MPLS stack is reached, the hash algorithm verifies that what follows is an Ethernet II untagged or tagged frame. For untagged frames, the system determines the value of Ethertype at the expected packet location and checks whether it contains an Ethernet-encapsulated IPv4 (0x0800) or IPv6 (0x86DD) value. The system also supports Ethernet II tagged frames with up to two 802.1Q tags, provided that the Ethertype value for the tags is 0x8100.

When the Ethertype verification passes, the first nibble of the expected IP packet location is then verified to be 4 (IPv4) or 6 (IPv6).

Use the following command to enable the LSR Ethernet encapsulated IP hash option.

```
configure system load-balancing lsr-load-balancing eth-encap-ip
```

LSR hashing of MPLS-over-GRE encapsulated packet

When the router removes the GRE encapsulation, pops one or more labels and then swaps a label, it acts as an LSR. The LSR hashing for packets of a MPLS-over-GRE SDP or tunnel follows a different procedure, which is enabled automatically and overrides the LSR hashing option enabled on the incoming network IP interface.

On a packet-by-packet basis, the new hash routine parses through the label stack and the new hash routine hashes on the SA/DA fields and the Layer 4 SRC/DST Port fields of the inner IPv4/IPv6 header.



Note:

- If the GRE header and label stack sizes are such that the Layer4 SRC/DST Port fields are not read, it hashes on the SA/DA fields of the inner IPv4/IPv6 header.
- If the GRE header and label stack sizes are such that the SA/DA fields of the inner IPv4/IPv6 header are not read, it hashes on the SA/DA fields of the outer IPv4/IPv6 header.

LSR hashing when an Entropy Label (EL) is present in the packet's label stack

The LSR hashing procedures are modified as follows:

- If the **lbl-only** hashing command option is enabled, or if one of the other LSR hashing options are enabled but an IPv4 or IPv6 header is not detected below the bottom of the label stack, the LSR hashes on the EL only.
- If the **lbl-ip** hashing command option is enabled, the LSR hashes on the EL and the IP headers.
- If the **ip-only** or **eth-encap-ip** hashing command option is enabled, the LSR hashes on the IP headers only.

6.3.1.2 Layer 4 load balancing

Users can enable Layer 4 load balancing to include TCP/UDP source/destination port numbers in addition to source/destination IP addresses in per-flow hashing of IP packets. By including the Layer 4 information, a SA/DA default hash flow can be sub-divided into multiple finer-granularity flows if the ports used between a specific SA/DA vary.

Layer 4 load balancing can be enabled or disabled at the system or interface level to improve load balancing distribution by including the TCP or UDP source and destination port of the packet to the hash function.

Use the following command to enable layer 4 load balancing at the system level.

```
configure system load-balancing l4-load-balancing
```

This setting applies to unicast traffic.

6.3.1.3 System IP load balancing

This option, when enabled, enhances all per-flow load balancing by adding the system IP address to the hash calculation. This capability avoids polarization of flows when a packet is forwarded through multiple routers with a similar number of ECMP/LAG paths.



Note: The system IP address is not added to the hash calculation for packets load balanced based on service ID.

Use the following command to enable system IP address load balancing.

```
configure system load-balancing system-ip-load-balancing
```

6.3.1.4 Source-only/destination-only hash inputs

A user can include only the **source** command option or only the **destination** command option in the hash for inputs that have **source/destination** context (such as IP address and Layer 4 port). Command options that do not have source/destination context (such as TEID or System IP, for example) are also included in hash as per applicable hash configuration. The functionality ensures that both upstream and downstream traffic hash to the same ECMP path/LAG port on system egress when traffic is sent to a hair-pinned appliance (by configuring source-only hash for incoming traffic on upstream interfaces and destination-only hash for incoming traffic on downstream interfaces).



Note: The **source** or **destination** options do not affect LSR load balancing.

Use the **source** and **destination** command options in the following commands to enable source-only or destination-only hash inputs in load balancing at the Layer 3 interface (**service** or **router**) level:

- **MD-CLI**

```
configure router interface load-balancing ip-load-balancing
configure service vprn interface load-balancing ip-load-balancing
configure service ies interface load-balancing ip-load-balancing
```

- **classic CLI**

```
configure router interface load-balancing egr-ip-load-balancing
configure service vprn interface load-balancing egr-ip-load-balancing
configure service ies interface load-balancing egr-ip-load-balancing
```

6.3.2 LAG port hash weight

The LAG port **hash-weight** command customizes the flow hashing distribution between LAG ports by adjusting the weight of each port independently for same-speed LAGs.

The following are common rules for using the LAG port **hash-weight** command.

- The configured **hash-weight** value per port is ignored until the **hash-weight** command is configured for all the ports in the LAG.
- The **hash-weight** value can be set to **port-speed** or an integer value from 1 to 100000:

- **port-speed**

This assigns an implicit **hash-weight** value based on the physical port speed.

- **1 to 100000**

This value range allows for control of flow hashing distribution between LAG ports.

- The LAG port **hash-weight** value is normalized internally to distribute flows between LAG ports. The minimum value returned by this normalization is 1.
- When the LAG port **hash-weight** command is not configured, the value defaults to the **port-speed** value.

The following table lists the **hash-weight** values using **port-speed** per physical port types.

Table 18: Port types and speeds

Port type	Port speed
FE port	port-speed value 1
1GE port	port-speed value 1
10GE port	port-speed value 10
25GE port	port-speed value 25
40GE port	port-speed value 40
50GE port	port-speed value 50
100GE port	port-speed value 100
400GE port	port-speed value 400
800GE port	port-speed value 800
Other ports	port-speed value 1

The LAG port **hash-weight** capability is supported for same-speed LAGs only.

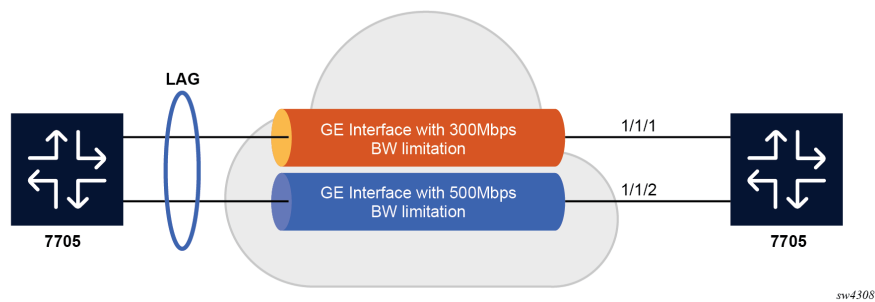
6.3.2.1 Configurable hash weight to control flow distribution

The user can use the LAG port **hash-weight** to control traffic distribution between LAG ports by adjusting the weight of each port independently.

This capability is especially useful when LAG links on Nokia routers are rate limited by a third-party transport operator providing the connectivity between two sites, as shown in the following figure, where:

- LAG links 1/1/1 and 1/1/2 are GE
- LAG link 1/1/1 is rate limited to 300 Mb/s by a third-party transport user
- LAG Link 1/1/2 is rate limited to 500 Mb/s by a third-party transport user

Figure 28: Same-speed LAG with ports of different hash weight



In this context, configure the LAG to adapt the flow distribution between LAG ports according to the bandwidth restrictions on each port that uses customized **hash-weight** values.

Example: MD-CLI

```
[ex:/configure lag "lag-5"]
A:admin@node-2# info
  admin-state enable
  port 1/1/1 {
    hash-weight 300
  }
  port 1/1/2 {
    hash-weight 500
  }
```

Example: classic CLI

```
A:node-2>config>lag# info
-----
  port 1/1/1 hash-weight 300
  port 1/1/2 hash-weight 500
  no shutdown
-----
```

Use the following command to display the resulting flow-distribution between active LAG ports.

```
show lag 3 flow-distribution
```

Output example

```
=====
Distribution of allocated flows
=====
```

Port	Bandwidth (Gbps)	Hash-weight	Flow-share (%)
1/1/1	10.000	300	37.50
1/1/2	10.000	500	62.50

```
-----
Total operational bandwidth: 20.000
=====
```



Note: The following applies for same-speed LAGs that use the **hash-weight** capability:

- If all ports have a **hash-weight** configured, other than **port-speed**, the configured value is used and normalized to modify the hashing between LAG ports.
- If the LAG ports are all configured to **port-speed**, or if only some of the ports have a customized **hash-weight** value, the system uses a hash weight of 1 for every port.

6.3.3 Consistent per-service hashing

The hashing feature described in this section applies to traffic going over LAG, Ethernet tunnels (**eth-tunnel**) in load-sharing mode, or CCAG load balancing for VSM redundancy. The feature does not apply to ECMP.

Per-service-hashing was introduced to ensure consistent forwarding of packets belonging to one service. The feature can be enabled using the **per-service-hashing** command under the following contexts and is valid for Epipe, VPLS, PBB Epipe, IVPLS, BVPLS, EVPN-VPWS and EVPN-VPLS.

```
configure service epipe load-balancing
configure service vpls load-balancing
```

The following behavior applies to the usage of the **per-service-hashing** option.

- The setting of the PBB Epipe or I-VPLS children dictates the hashing behavior of the traffic destined for or sourced from an Epipe or I-VPLS endpoint (PW/SAP).
- The setting of the B-VPLS parent dictates the hashing behavior only for transit traffic through the B-VPLS instance (not destined for or sourced from a local I-VPLS or Epipe children).

The following algorithm describes the hash-key used for hashing when the **per-service-hashing** option is enabled:

- If the packet is PBB encapsulated (contains an I-TAG Ethertype) at the ingress side and enters a B-VPLS service, use the ISID value from the I-TAG. For PBB encapsulated traffic entering other service types, use the related service ID.
- If the packet is not PBB encapsulated at the ingress side:
 - For regular (non-PBB) VPLS and Epipe services, use the related service ID.
 - If the packet is originated from an ingress IVPLS or PBB Epipe SAP:
 - If there is an ISID configured, use the related ISID value.
 - If there is no ISID configured, use the related service ID.
 - For BVPLS transit traffic use the related flood list ID.
 - Transit traffic is the traffic going between BVPLS endpoints.
 - An example of non-PBB transit traffic in BVPLS is the OAM traffic.
- The above rules apply to Unicast, BUM flooded without MMRP or with MMRP, IGMP snooped regardless of traffic type.

Users may sometimes require the capability to query the system for the link in a LAG or Ethernet tunnel that is currently assigned to a specific service-id or ISID.

Use the following command to query the system for the link in a LAG or Ethernet tunnel that is currently assigned to a specific service-id or ISID.

```
tools dump map-to-phy-port lag 11 service 1
```

Output example

```

ServiceId  ServiceName  ServiceType  Hashing  Physical Link
-----
1          i-vpls    per-service(if enabled)  3/2/8

A:Dut-B# tools dump map-to-phy-port lag 11 isid 1

ISID      Hashing  Physical Link
-----
1         per-service(if enabled)  3/2/8

A:Dut-B# tools dump map-to-phy-port lag 11 isid 1 end-isid 4
ISID      Hashing  Physical Link
-----
1         per-service(if enabled)  3/2/8
2         per-service(if enabled)  3/2/7
3         per-service(if enabled)  1/2/2
4         per-service(if enabled)  1/2/3

```

6.3.4 ESM

In ESM, egress traffic can be load balanced over LAG member ports based on the following entities:

- per subscriber, in weighted and non-weighted mode
- per Vport, on non HSQ cards in weighted and non-weighted
- per secondary shaper on HSQ cards
- per destination MAC address when ESM is configured in a VPLS (Bridged CO)

ESM over LAGs with configured PW ports require additional considerations:

- PW SAPs are not supported in VPLS services or on HSQ cards. This means that load balancing per secondary shaper or destination MAC are not supported on PW ports with a LAG configured under them.
- Load balancing on a PW port associated with a LAG with faceplate member ports (fixed PW ports) can be performed per subscriber or Vport.
- Load balancing on a FPE (or PXC)-based PW port is performed on two separate LAGs which can be thought of as two stages:
 - Load balancing on a PXC LAG where the subscribers are instantiated. In this first stage, the load balancing can be performed per subscriber or per Vport.
 - The second stage is the LAG over the network faceplate ports over which traffic exits the node. Load balancing is independent of ESM and must be examined in the context of Epipe or EVPN VPWS that is stitched to the PW port.

6.3.4.1 Load balancing per subscriber

Load balancing per subscriber has two modes of operation.

The first mode is native non-weighted per-subscriber load balancing in which traffic is directly hashed per subscriber. Use this mode in SAP and subscriber (1:1) deployments and in SAP and service (N:1) deployments. Examples of services in SAP and services deployments are VoIP, video, or data.

In this mode of operation, the following configuration requirements must be met.

- Any form of the **per-link-hash** command in a LAG under the **configure lag** context must be disabled. This is the default setting.
- If QoS schedulers or Vports are used on the LAG, their bandwidth must be distributed over LAG member ports in a port-fair operation.

```
configure lag access adapt-qos port-fair
```

In this scenario, setting this command option to in **adapt-qos** to mode **link** disables per-subscriber load balancing and enables per-Vport load balancing.

The second mode, the weighted per subscriber load balancing is supported only in SAP and subscriber (1:1) deployments, and it requires the following configurations.

```
configure lag per-link-hash weighted subscriber-hash-mode sap
```

In this scenario where hashing is performed per SAP, as reflected in the CLI above, in terms of load balancing, per-SAP hashing produces the same results as per-subscriber hashing because SAPs and subscribers are in a 1:1 relationship. The end result is that the traffic is load balanced per-subscribers, regardless of this indirection between hashing and load-balancing.

With the **per-link-hash** option enabled, the SAPs (and with this, the subscribers) are dynamically distributed over the LAG member links. This dynamic behavior can be overridden by configuring the **lag-link-map-profiles** command under the static SAPs or under the **msap-policy**. This way, each static SAP, or a group of MSAPs sharing the same **msap-policy** are statically and deterministically assigned to a preordained member port in a LAG.

This mode allows classes and weights to be configured for a group of subscribers with a shared subscriber profile under the following hierarchy.

- MD-CLI**

```
configure subscriber-mgmt sub-profile egress lag-per-link-hash class
configure subscriber-mgmt sub-profile egress lag-per-link-hash weight
```

- classic CLI**

```
configure subscriber-mgmt sub-profile egress lag-per-link-hash class weight
```

Default values for **class** and **weight** are 1. If all subscribers on a LAG are configured with the same values for class and weight, load balancing effectively becomes non-weighted.



Note: The second mode of operation, weighted per-subscriber load balancing, is not supported on 7705 SAR Gen 2 platforms.

If QoS schedulers and Vports are used on the LAG, their bandwidth should be distributed over LAG member ports in a port-fair operation.

- MD-CLI**

```
configure lag "lag-100" access adapt-qos mode port-fair
```

- **classic CLI**

```
configure lag access adapt-qos port-fair
```

6.3.4.2 Load balancing per Vport

Load balancing per Vport applies to user bearing traffic, and not to the control traffic originated or terminated on the BNG, required to setup and maintain sessions, such as PPPoE and DHCP setup and control messages.

Per Vport load balancing has two modes of operation.

In the first mode, non-weighted load balancing based on Vport hashing, the following LAG-related configuration is required.

The **per-link-hash** command must be disabled.

- **MD-CLI**

```
configure lag access adapt-qos mode link
```

- **classic CLI**

```
configure lag access adapt-qos link
```

If LAG member ports are distributed over multiple forwarding complexes, the following configuration is required.

```
configure subscriber-mgmt sub-profile vport-hashing
```

The second mode, weighted load balancing based on Vport hashing, supports **class** and **weight** command options per Vport. To enable weighted traffic load balancing per Vport, the following configuration must be enabled.

```
configure lag per-link-hash weighted subscriber-hash-mode vport
```

The class and weight can be optionally configured under the Vport definition.

- **MD-CLI**

```
configure port ethernet access egress virtual-port lag-per-link-hash class  
configure port ethernet access egress virtual-port lag-per-link-hash weight
```

- **classic CLI**

```
configure port ethernet access egress vport lag-per-link-hash class weight
```



Note: The second load-balancing mode is not supported on 7705 SAR Gen 2 platforms.

6.3.4.3 Load balancing per secondary shaper

Load balancing based on a secondary shaper is supported only on HSQ cards and only in non-weighted mode. The following LAG-related configuration is required. The **per-link-hash** command first must be disabled.

- **MD-CLI**

```
configure lag "lag-100" access adapt-qos mode link
```

- **classic CLI**

```
configure lag access adapt-qos link
```

Use the following command to disable **per-link-hash**.

- **MD-CLI**

```
configure lag delete per-link-hash
```

- **classic CLI**

```
configure lag no per-link-hash
```



Note: Per-link hashing is not supported on 7705 SAR Gen 2 platforms.

6.3.4.4 Load balancing per destination MAC

This load balancing mode is supported only when ESM is enabled in VPLS in Bridged Central Office (CO) deployments. In this mode of operation, the following configuration is required. The **per-link-hash** command first must be disabled.

```
configure subscriber-mgmt msap-policy vpls-only-sap-parameters mac-da-hashing  
configure service vpls sap sub-sla-mgmt mac-da-hashing
```

6.4 QoS consideration for access LAG

The following section describes various QoS related features applicable to LAG on access.

6.4.1 Adapt QoS modes

Link Aggregation is supported on the access side with access or hybrid ports. Similarly to LAG on the network side, LAG on access aggregates Ethernet ports into all active or active/standby LAG. The difference with LAG on networks lies in how the QoS or H-QoS is handled. Based on hashing configured, a SAP's traffic can be sprayed on egress over multiple LAG ports or can always use a single port of a LAG. There are three user-selectable modes that allow the user to best adapt QoS configured to a LAG the SAPs are using:

- **distribute (default)**

Use the following command to configure the distributed mode:

- **MD-CLI**

```
configure lag access adapt-qos mode distribute
```

- **classic CLI**

```
configure lag access adapt-qos distribute
```

In the distribute mode, the SLA is divided among all line cards proportionate to the number of ports that exist on that line card for a specific LAG. For example, a 100 Mb/s PIR with 2 LAG links on IOM A and 3 LAG links on IOM B would result in IOM A getting 40 Mb/s PIR and IOM B getting 60 Mb/s PIR. Because of this distribution, SLA can be enforced. The disadvantage is that a single flow is limited to IOM's share of the SLA. This mode of operation may also result in underrun because of hashing imbalance (traffic not sprayed equally over each link). This mode is best suited for services that spray traffic over all links of a LAG.

- **link**

Use the following command to configure the link mode:

- **MD-CLI**

```
configure lag access adapt-qos mode link
```

- **classic CLI**

```
configure lag access adapt-qos link
```

In a link mode the SLA is provided to each port of a LAG. With the example above, each port would get 100 Mb/s PIR. The advantage of this method is that a single flow can now achieve the full SLA. The disadvantage is that the overall SLA can be exceeded, if the flows span multiple ports. This mode is best suited for services that are guaranteed to hash to a single egress port.

- **port-fair**

Use the following command to configure the port-fair mode:

- **MD-CLI**

```
configure lag access adapt-qos mode port-fair
```

- **classic CLI**

```
configure lag access adapt-qos port-fair
```

Port-fair distributes the SLA across multiple line cards relative to the number of active LAG ports per card (in a similar way to distribute mode) with all LAG QoS objects parented to scheduler instances at the physical port level (in a similar way to link mode). This provides a fair distribution of bandwidth between cards and ports whilst ensuring that the port bandwidth is not exceeded. Optimal LAG utilization relies on an even hash spraying of traffic to maximize the use of the schedulers' and ports' bandwidth. With the example above, enabling port-fair would result in all five ports getting 20 Mb/s.

When port-fair mode is enabled, per-Vport hashing is automatically disabled for subscriber traffic such that traffic sent to the Vport no longer uses the Vport as part of the hashing algorithm. Any QoS object

for subscribers, and any QoS object for SAPs with explicitly configured hashing to a single egress LAG port, are given the full bandwidth configured for each object (in a similar way to link mode). A Vport used together with an egress port scheduler is supported with a LAG in port-fair mode, whereas it is not supported with a distribute mode LAG.

- **distribute include-egr-hash-cfg**

Use the following commands to configure the distributed include-egr-hash-cfg mode:

- **MD-CLI**

```
configure lag access adapt-qos mode distribute
configure lag access adapt-qos include-egr-hash-cfg
```

- **classic CLI**

```
configure lag access adapt-qos distribute include-egr-hash-cfg
```

This mode can be considered a mix of link and distributed mode. The mode uses the configured hashing for LAG/SAP/service to choose either link or distributed adapt-qos modes. The mode allows:

- SLA enforcement for SAPs that through configuration are guaranteed to hash to a single egress link using full QoS per port (as per link mode)
- SLA enforcement for SAPs that hash to all LAG links proportional distribution of QoS SLA amongst the line cards (as per distributed mode)
- SLA enforcement for multi service sites (MSS) that contain any SAPs regardless of their hash configuration using proportional distribution of QoS SLA amongst the line cards (as per distributed mode)

The following restrictions apply to adapt-qos distributed include-egr-hash-cfg:

- LAG mode must be access or hybrid.
- When link-map-profiles or per-link-hash is configured, the user cannot change from **include-egr-hash-cfg** mode to **distribute** mode.
- The user cannot change from **link** to **include-egr-hash-cfg** on a LAG with any configuration.

[Table 19: Adapt QoS bandwidth/rate distribution](#) shows examples of rate/BW distributions based on the **adapt-qos** mode used.

Table 19: Adapt QoS bandwidth/rate distribution

	distribute	link	port-fair	distribute include-egr-hash-cfg
SAP Queues	% # local links ¹²	100% rate	100% rate (SAP hash to one link) or %# all links ¹³ (SAP hash to all links)	100% rate (SAP hash to one link) or % # local linksa (SAP hash to all links)

¹² * % # local links = X * (number of local LAG members on a line card/ total number of LAG members)

¹³ %# all links = X* (link speed)/(total LAG speed)

	distribute	link	port-fair	distribute include-egr-hash-cfg
SAP Scheduler	% # local linksa	100% bandwidth	100% rate (SAP hash to one link) or %# all linksb (SAP hash to all links)	100% bandwidth (SAP hash to a one link) or % # local linksa (SAP hash to all links)
SAP MSS Scheduler	% # local linksa	100% bandwidth	% # local linksa	% # local linksa

6.4.2 Per-fp-ing-queuing

Per-fp-ing-queuing optimization for LAG ports provides the ability to reduce the number of hardware queues assigned on each LAG SAP on ingress when the flag at LAG level is set for per-fp-ing-queuing.

When the feature is enabled in the **configure lag access** context, the queue allocation for SAPs on a LAG are optimized and only one queuing set per ingress forwarding path (FP) is allocated instead of one per port.

The following rules apply for configuring the per-fp-ing-queuing at LAG level:

- To enable per-fp-ing-queuing, the LAG must be in access mode.
- The LAG mode cannot be set to network mode when the feature is enabled.
- Per-fp-ing-queuing can only be set if no port members exists in the LAG.

6.4.3 Per-fp-egr-queuing

Per-fp-egr-queuing optimization for LAG ports provides the ability to reduce the number of egress resources consumed by each SAP on a LAG, and by any encap groups that exist on those SAPs.

When the feature is enabled in the **configure lag access** context, the queue and virtual scheduler allocation are optimized. Only one queuing set and one H-QoS virtual scheduler tree per SAP/encap group is allocated per egress forwarding path (FP) instead of one set per each port of the LAG. In case of a link failure/recovery, egress traffic uses failover queues while the queues are moved over to a newly active link.

Per-fp-egr-queuing can be enabled on existing LAG with services as long as the following conditions are met.

- The mode of the LAG must be **access** or **hybrid**.
- The port-type of the LAGs must be **standard**.
- The LAG must have either **per-link-hash** enabled or all SAPs on the LAG must use **per-service-hashing** only and be of a type: VPLS SAP, i-VPLS SAP, or e-Pipe VLL or PBB SAP.

To disable per-fp-egr-queuing, all ports must first be removed from a specific LAG.

6.4.4 Per-fp-sap-instance

Per-fp-sap-instance optimization for LAG ports provides the ability to reduce the number of SAP instance resources consumed by each SAP on a lag.

When the feature is enabled, in the `config>lag>access` context, a single SAP instance is allocated on ingress and on egress per each forwarding path instead of one per port. Thanks to an optimized resource allocation, the SAP scale on a line card increases, if a LAG has more than one port on that line card. Because SAP instances are only allocated per forwarding path complex, hardware reprogramming must take place when as result of LAG links going down or up, a SAP is moved from one LAG port on a specific line card to another port on a specific line card within the same forwarding complex. This results in an increased data outage when compared to per-fp-sap-instance feature being disabled. During the reprogramming, failover queues are used when SAP queues are reprogrammed to a new port. Any traffic using failover queues is not accounted for in SAPs statistics and is processed at best-effort priority.

The following rules apply when configuring a per-fp-sap-instance on a LAG:

- Per-fp-ing-queuing and per-fp-egr-queuing must be enabled.
- The functionality can be enabled/disabled on LAG with no member ports only. Services can be configured.

Other restrictions:

- SAP instance optimization applies to LAG-level. Whether a LAG is sub-divided into sub-groups or not, the resources are allocated per forwarding path for all complexes LAG's links are configured on (that is irrespective of whether a sub-group a SAP is configured on uses that complex or not).
- Egress statistics continue to be returned per port when SAP instance optimization is enabled. If a LAG links are on a single forwarding complex, all ports but one have no change in statistics for the last interval – unless a SAP moved between ports during the interval.
- Rollback that changes per-fp-sap-instance configuration is service impacting.

6.5 LAG hold-down timers

Users can configure multiple hold-down timers that allow control how quickly LAG responds to operational port state changes. The following timers are supported:

- **port-level hold-time up/down timer**

This optional timer allows user to control delay for adding/removing a port from LAG when the port comes UP/goes DOWN. Each LAG port runs the same value of the timer, configured on the primary LAG link. See the Port Link Dampening description in [Port features](#) for more details on this timer.

- **sub-group-level hold-time timer**

This optional timer allows user to control delay for a switch to a new candidate sub-group selected by LAG sub-group selection algorithm from the current, operationally UP sub-group. The timer can also be configured to never expire, which prevents a switch from operationally up sub-group to a new candidate sub-group (manual switchover is possible using tools perform force lag command). Note that, if the port link dampening is deployed, the port level timer must expire before the sub-group-selection takes place and this timer is started. Sub-group-level hold-down timer is supported with LAGs running LACP only.

- **LAG-level hold-time down timer**

This optional timer allows user to control delay for declaring a LAG operationally down when the available links fall below the required port/BW minimum. The timer is recommended for LAG connecting to MC-LAG systems. The timer prevents a LAG going down when MC-LAG switchover executes break-before-make switch. Note that, if the port link dampening is deployed, the port level timer must expire before the LAG operational status is processed and this timer is started.

6.6 Multi-Chassis LAG

Multi-Chassis LAG (MC-LAG) is an extension of the LAG concept. MC-LAG provides node-level redundancy, in addition to the link-level redundancy provided by LAG.

Typically, MC-LAG is deployed in a network-wide scenario providing redundant connection between different end points. The whole scenario is then built by combination of different mechanisms (for example, MC-LAG and redundant pseudowire to provide e2e redundant p2p connection or dual homing of DSLAMs in Layer 2/3 TPSDA).

6.6.1 Overview

Multichassis LAG is a method of providing redundant Layer 2/3 access connectivity that extends beyond link level protection by allowing two systems to share a common LAG end point.

The multiservice access node (MSAN) node is connected with multiple links toward a redundant pair of Layer 2/3 aggregation nodes such that both link and node level redundancy, are provided. By using a multichassis LAG protocol, the paired Layer 2/3 aggregation nodes (referred to as redundant-pair) appears to be a single node utilizing LACP toward the access node. The multichassis LAG protocol between a redundant-pair ensures a synchronized forwarding plane to and from the access node and synchronizes the link state information between the redundant-pair nodes such that correct LACP messaging is provided to the access node from both redundant-pair nodes.

To ensure SLAs and deterministic forwarding characteristics between the access and the redundant-pair node, MC-LAG provides an active/standby operation to and from the access node. LACP is used to manage the available LAG links into active and standby states, which ensures that links from only one aggregation node are active at a time to and from the access node.

Alternatively, when access nodes do not support LACP, the following command can be used to enforce the active/standby operation.

```
configure lag standby-signaling power-off
```

In this case, the standby ports are **trx_disabled** (power off transmitter) to prevent usage of the LAG member by the access-node. Characteristics related to MC-LAG are:

- The selection of the common system ID, system-priority, and administrative-key are used in LACP messages so that partner systems consider all links as part of the same LAG.
- The selection algorithm is extended to allow the selection of the active sub-group.
 - A sub-group definition in the LAG context is local to the single box, which means that if sub-groups configured on two different systems have the same sub-group-id, they are still considered two separate sub-groups within the specified LAG.
 - Multiple sub-groups per PE in an MC-LAG are supported.

- In the case where there is a tie in the selection algorithm, (for example, two sub-groups with identical aggregate weight (or number of active links), the group that is local to the system with the lower system LACP priority and LAG system ID is used.
- An inter-chassis communication channel allows LACP support on both systems. The inter-chassis communication channel supports the following:
 - Connections at the IP level that do not require a direct link between two nodes. The IP address configured at the neighbor system is one of the addresses of the system (interface or loop-back IP address).
 - A communication protocol that provides a heartbeat mechanism to enhance the robustness of the MC-LAG operation and to detect node failures.
 - User actions on any node that force an operational change.
 - LAG group-ids that do not have to match between neighbor systems. At the same time, there can be multiple LAG groups between the same pair of neighbors.
 - Verifying the configuration of physical characteristics, such as speed and auto-negotiation, and initiating user notifications (traps) if errors exist. Consistency of MC-LAG configuration (system-id, administrative-key, and system-priority) is provided. Similarly, the load-balancing mode of operation must be consistently configured on both nodes.
 - Traffic over the signaling link encryption using a user-configurable message digest key.
- MC-LAG provides active/standby status to other software applications to build a reliable solution.

Figure 29: MC-LAG Layer 2 dual-homing to remote PE pairs and Figure 30: MC-LAG Layer 2 dual homing to local PE pairs show the different combinations of MC-LAG attachments that are supported. The supported configurations can be sub-divided into following sub-groups:

- Dual-homing to remote PE pairs
 - both end-points attached with MC-LAG
 - one end-point attached
- Dual-homing to local PE pair
 - both end-points attached with MC-LAG
 - one end-point attached with MC-LAG
 - both end-points attached with MC-LAG to two overlapping pairs

Figure 29: MC-LAG Layer 2 dual-homing to remote PE pairs

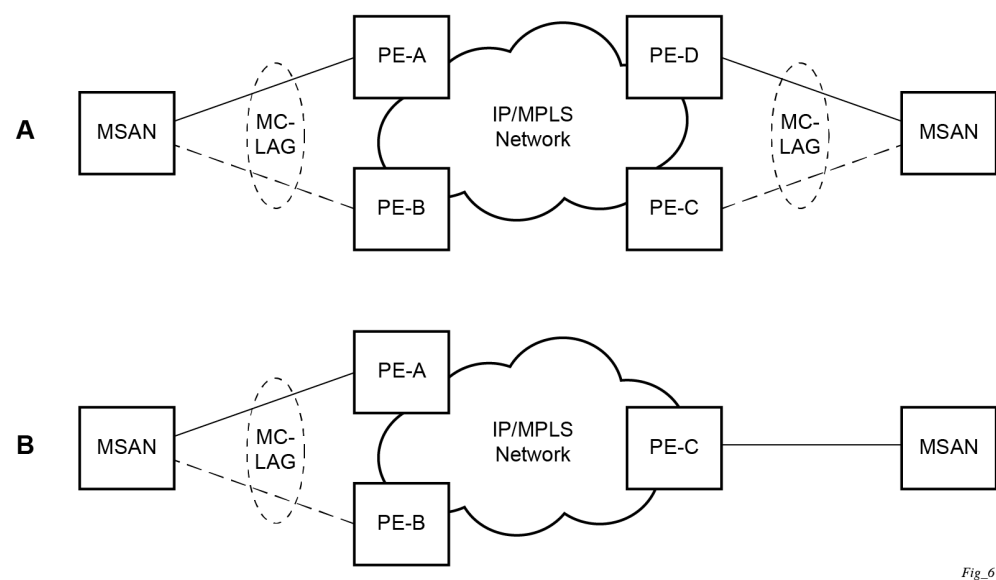


Figure 30: MC-LAG Layer 2 dual homing to local PE pairs

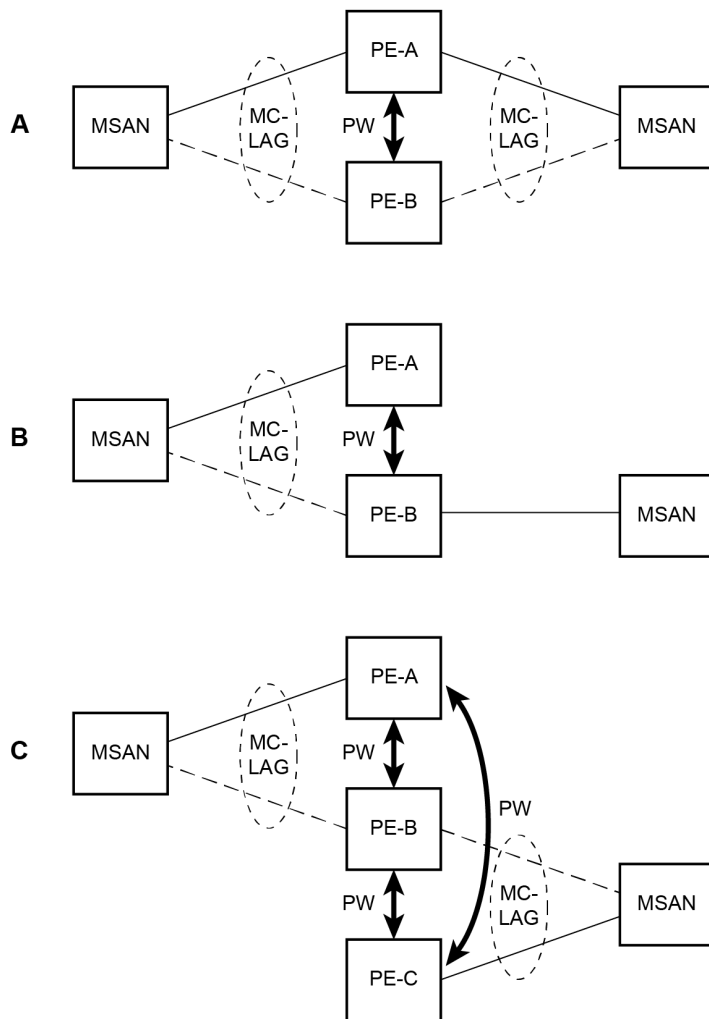


Fig_7

The forwarding behavior of the nodes abide by the following principles. Note that logical destination (actual forwarding decision) is primarily determined by the service (VPLS or VLL) and the principle below applies only if destination or source is based on MC-LAG:

- Packets received from the network are forwarded to all local active links of the specific destination-sap based on conversation hashing. In case there are no local active links, the packets are cross-connected to inter-chassis pseudowire.
- Packets received from the MC-LAG sap are forwarded to active destination pseudowire or active local links of destination-sap. In case there are no such objects available at the local node, the packets are cross-connected to inter-chassis pseudowire.

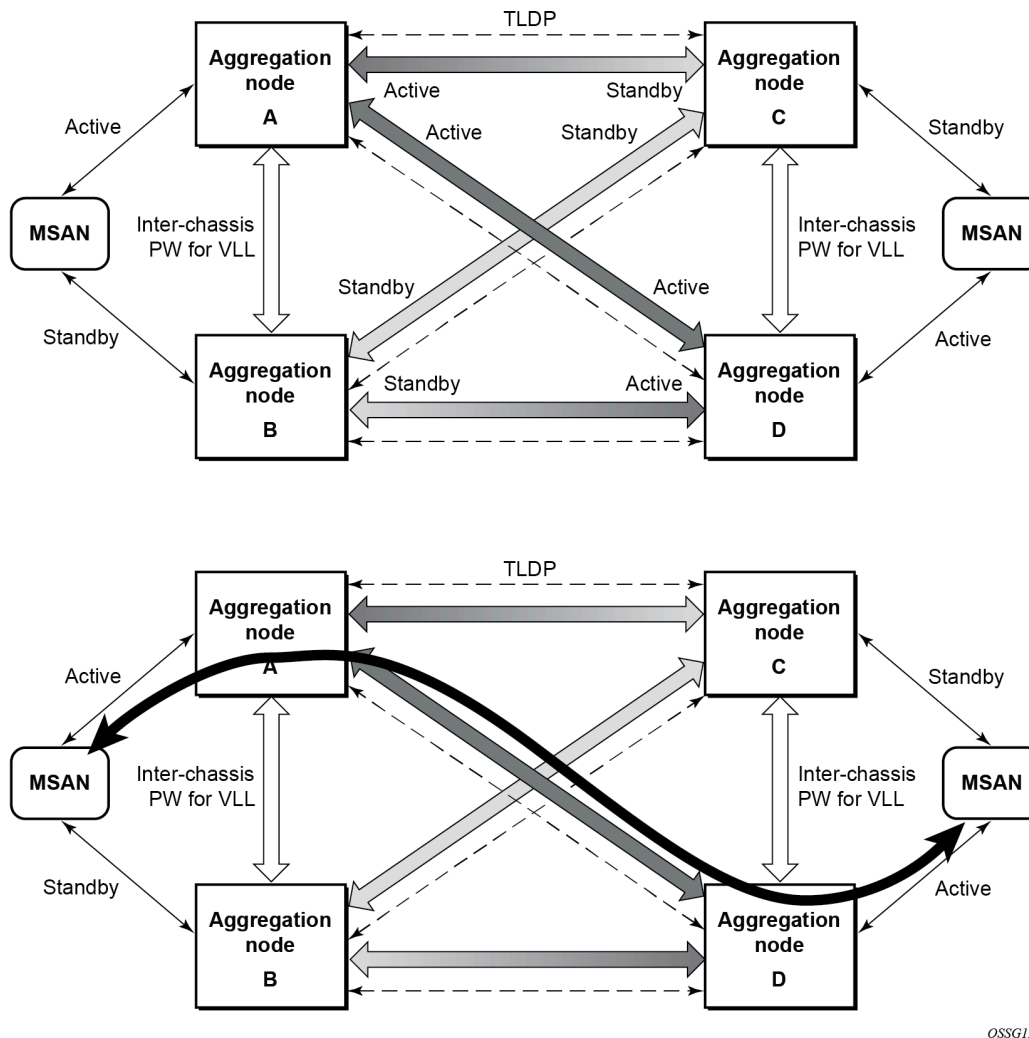
6.6.2 MC-LAG and SRRP

MC-LAG and Subscriber Routed Redundancy Protocol (SRRP) enable dual-homed links from any IEEE 802.1ax (formerly 802.3ad) standards-based access device (for example, a IP DSLAM, Ethernet switch or a Video on Demand server) to multiple Layer 2/3 or Layer 3 aggregation nodes. In contrast with slow recovery mechanisms such as Spanning Tree, multichassis LAG provides synchronized and stateful redundancy for VPN services or triple play subscribers in the event of the access link or aggregation node failing, with zero impact to end users and their services.

6.6.3 P2P redundant connection across Layer 2/3 VPN network

[Figure 31: Point-to-Point \(P2P\) redundant connection through a Layer 2 VPN network](#) shows the connection between two multiservice access nodes (MSANs) across a network based on Layer 2/3 VPN pseudowires. The connection between MSAN and a pair of PE routers is realized by MC-LAG. From an MSAN perspective, a redundant pair of PE routers acts as a single partner in LACP negotiation. At any time, only one of the routers has an active link in a specified LAG. The status of LAG links is reflected in status signaling of pseudowires set between all participating PEs. The combination of active and stand-by states across LAG links as well as pseudowires gives only one unique path between a pair of MSANs.

Figure 31: Point-to-Point (P2P) redundant connection through a Layer 2 VPN network



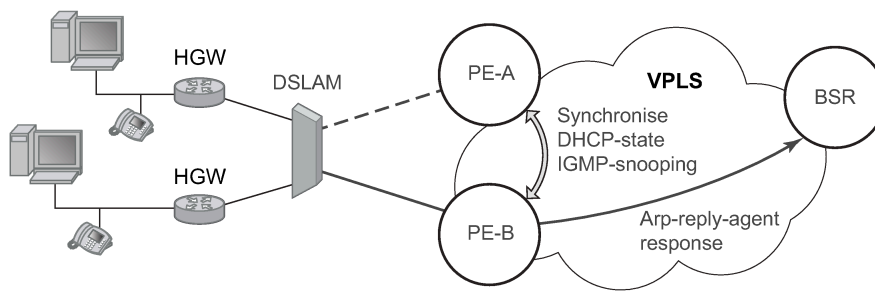
Note that the configuration in [Figure 31: Point-to-Point \(P2P\) redundant connection through a Layer 2 VPN network](#) shows one particular configuration of VLL connections based on MC-LAG, particularly the VLL connection where two ends (SAPs) are on two different redundant-pairs. In addition to this, other configurations are possible, such as:

- Both ends of the same VLL connections are local to the same redundant-pair.
- One end VLL endpoint is on a redundant-pair the other on single (local or remote) node.

6.6.4 DSLAM dual-homing in a Layer 2/3 TPSDA model

The following figure shows a network configuration where DSLAM is dual-homed to a pair of redundant PEs by using MC-LAG. In the aggregation network, a redundant pair of PEs is connecting to a VPLS service, which provides a reliable connection to a single or pair of Broadband Service Routers (BSRs).

Figure 32: DSLAM dual-homing using MC-LAG



OSSG110

MC-LAG and pseudowire connectivity, PE-A and PE-B implement enhanced subscriber management features based on DHCP-snooping and creating dynamic states for every subscriber-host. As in any point of time there is only one PE active, it is necessary to provide the mechanism for synchronizing subscriber-host state-information between active PE (where the state is learned) and stand-by PE. In addition, VPLS core must be aware of active PE to forward all subscriber traffic to a PE with an active LAG link. The mechanism for this synchronization is outside of the scope of this document.

6.7 LAG thresholds

The following sections provide information on LAG thresholds that are used by the system to adapt the IGP cost associated with the LAG or change its operational status. The thresholds can be configured to be either port-thresholds, hash-weight thresholds, or bandwidth-thresholds as a percentage relative to the overall configured LAG capacity.

6.7.1 LAG IGP cost

When using a LAG, it is possible to take an operational link degradation into consideration by setting a configurable degradation threshold. The following alternative settings are available through configuration:

```
configure lag port-threshold
configure lag hash-weight-threshold
configure lag bandwidth-threshold
```

When the LAG operates under normal circumstances and is included in an IS-IS or OSPF routing instance, the LAG must be associated with an IGP link cost. This LAG cost can either be statically configured in the IGP context or set dynamically by the LAG based upon the combination of the interface speed and reference bandwidth.

Under operational LAG degradation however, it is possible for the LAG to set a new updated dynamic or static threshold cost taking the gravity of the degradation into consideration.

As a consequence, there are some IGP link cost alternatives available, for which the most appropriate must be selected. The IGP uses the following priority rules to select the most appropriate IGP link cost:

1. Static LAG cost (from the LAG threshold action during degradation)
2. Explicit configured IGP cost (from the configuration under the IGP routing protocol context)

3. Dynamic link cost (from the LAG threshold action during degradation)
4. Default metric (no cost is set anywhere)

For example:

- Static LAG cost overrules the configured metric.
- Dynamic cost does not overrule configured metric or static LAG cost.

6.7.2 Adjusting the operational state of the LAG

Instead of changing the IGP cost, when using a LAG, a user can also configure to take the operational state of the links or link degradation into consideration to adjust the operational state of the LAG. Use the **action** command option of the following command to control the operational state of the LAG:

- **MD-CLI**

```
configure lag string port-threshold
```

- **classic CLI**

```
configure lag lag-id port-threshold
```

When the total number of operational links for the LAG is at or below the configured threshold value, the LAG operational state goes down. If the number of operational links for the LAG exceeds the threshold value, the operational state of LAG comes up.

For LAGs with PXC sub-ports also the operational state can be controlled through the **port-threshold action down** configuration described in the preceding information.

Similar to port threshold, use the hash-weight threshold to control the operational state of the LAG. Use the **action** option in the following the command to control the operational state of the LAG:

- **MD-CLI**

```
configure lag string hash-weight-threshold
```

- **classic CLI**

```
configure lag lag-id hash-weight-threshold
```

When the sum of hash weights of all the operational links of LAG is at or below the configured threshold value (weight), the LAG operational state goes down. If the sum of hash weights of all operational LAG links exceeds the hash-weight threshold value, the operational state of LAG comes up.

Also similar to port-threshold, use the bandwidth-threshold to control the operational state of the LAG. Use the **action** option in the following command to control the operational state of the LAG:

- **MD-CLI**

```
configure lag string bandwidth-threshold
```

- **classic CLI**

```
configure lag lag-id bandwidth-threshold
```

When the relative ratio of the sum of the bandwidth capacities of all operational LAG links to the total configured LAG bandwidth capacity, expressed as a percentage, is at or below the configured bandwidth-threshold value, the operational state of the LAG goes down. If the ratio exceeds the bandwidth-threshold value, the operational state of the LAG comes up.

7 Ethernet port monitoring

Ethernet ports can record and recognize various medium statistics and errors. There are two main types of errors:

- **frame based**

Frame based errors are counted when the arriving frame has an error that means the frame is invalid. These types of errors are only detectable when frames are presents on the wire.

- **symbol based**

Symbol errors are invalidly encoded symbols on the physical medium. Symbols are always present on an active Ethernet port regardless of the presence of frames.

CRC-Monitor and Symbol-Monitor allows the user to monitor ingress error conditions on the Ethernet medium and compare these error counts to the thresholds. CRC-Monitor monitors CRC errors. Symbol-Monitor monitors symbol errors. Symbol Error is not supported on all Ethernet ports. Crossing a signal degrade (SD) threshold causes a log event to be raised. Crossing the configured signal failure (SF) threshold causes the port to enter an operation state of down. The user may consider the configuration of other protocols to convey the failure, through timeout conditions.

The error rates are in the form of $M \cdot 10^E - N$. The user has the ability to configure both the threshold (N) and a multiplier (M). By default if the multiplier is not configured the multiplier is 1. As an example, sd-threshold 3 would result in a signal degrade error rate of $1 \cdot 10^E - 3$ (one error per 1000). Changing the configuration to would sd-threshold 3 multiplier 5 result in a signal degrade rate of $5 \cdot 10^E - 3$ (5 errors per 1000). The signal degrade value must be a lower error rate than the signal failure threshold. This threshold can be used to provide notification that the port is operating in a degraded but not failed condition. These do not equate to a bit error rate (BER). CRC-Monitor provides a CRC error rate. Symbol-Monitor provides a symbol error rate.

The configured error thresholds are compared to the user specified sliding window to determine if one or both of the thresholds have been crossed. Statistics are gathered every second. This means that every second the oldest statistics are dropped from the calculation. The default 10 second sliding window means that at the 11th second, the oldest 1-second statistical data is dropped and the 11th second is included.

Symbol error crossing differs slightly from CRC-based error crossing. The error threshold crossing is calculated based on the window size and the fixed number of symbols that arrive (ingress) on that port during that window.

The following configuration demonstrates this concept..

Example: MD-CLI

```
[ex:/configure port 2/1/2 ethernet]
A:admin@node-2# info
  symbol-monitor {
    admin-state enable
    signal-degrade {
      threshold 5
      multiplier 5
    }
    signal-failure {
      threshold 3
      multiplier 5
    }
  }
```

```
}
```

Example: classic CLI

```
A:node-2>config>port>ethernet# info detail
-----
symbol-monitor
sd-threshold 5 multiplier 5
sf-threshold 3 multiplier 5
no shutdown
exit
```

Use the following command to display Ethernet port statistics.

```
show port 2/1/2 ethernet
```

Output example

```
=====
Ethernet Interface
=====
Description      : 1-Gig/10-Gig Ethernet
Interface        : 2/1/2
Link-level       : Ethernet
Admin State      : down
Oper State       : down
Config Duplex    : N/A
Physical Link    : No
Single Fiber Mode : No
IfIndex          : 35684352
Last State Change : 11/29/2022 18:37:14
Hold Time Down Rmng: 0 cs
Last Cleared Time : N/A
Phys State Chng Cnt: 0
RS-FEC Config Mode : None
RS-FEC Oper Mode  : None

Oper Speed       : 10 Gbps
Config Speed    : 10 Gbps
Oper Duplex      : full

MTU              : 8704
Min Frame Length : 64 Bytes
Hold time up     : 0 seconds
Hold time down   : 0 seconds
Hold Time Up Rmng: 0 cs
DDM Events       : Enabled

Configured Mode  : network
Dot1Q Ethertype  : 0x8100
PBB Ethertype    : 0x88e7
Ing. Pool % Rate : 100
Net. Egr. Queue Pol: default
Egr. Sched. Pol  : n/a
DCPU Prot Policy : _default-port-policy
Oper DCPU Prot Plcy: _default-port-policy
Monitor Port Sched : Disabled
Monitor Agg Q Stats: Disabled
Monitor Oper Group : none
Auto-negotiate    : N/A
Oper Phy-tx-clock : not-applicable
Accounting Policy : None
Acct Plcy Eth Phys : None
Egress Rate       : Default
Oper Egress Rate  : Unrestricted
Load-balance-algo : Default
Access Bandwidth  : Not-Applicable
Access Available BW: 0
Access Booked BW  : 0
Discard Rx Pause  : Disabled

Encap Type       : null
QinQ Ethertype   : 0x8100
Egr. Pool % Rate : 100

MDI/MDX          : N/A
Collect-stats    : Disabled
Collect Eth Phys : Disabled
Ingress Rate     : Default

LACP Tunnel      : Disabled
Booking Factor   : 100

Suppress Threshold : 2000
Max Penalties      : 16000

Reuse Threshold   : 1000
Max Suppress Time : 20 seconds
=====
```

```

Half Life           : 5 seconds

Down-when-looped    : Disabled
Loop Detected       : False
Use Broadcast Addr   : False

Sync. Status Msg.   : Disabled
Tx DUS/DNU          : Disabled
SSM Code Type       : sdh

Down On Int. Error   : Disabled

CRC Mon SD Thresh    : Disabled
CRC Mon SF Thresh    : Disabled

Sym Mon SD Thresh    : 5*10E-5
Sym Mon SF Thresh    : 5*10E-3

EFM OAM             : Disabled
Ignr EFM OAM State   : False

Configured Address   : b6:1b:01:01:00:01
Hardware Address     : b6:1b:01:01:00:01
Cfg Alarm            : remote local

Transceiver Data

Transceiver Status   : operational
Transceiver Type     : SFP
Model Number         : 3HE04823AAAA01 ALA IPU3ANKEAA
TX Laser Wavelength : 1310 nm
Connector Code       : LC
Manufacture date     : 2009/12/17
Serial Number        : UGR04DK
Part Number          : FTLX1471D3BCL-A5
Optical Compliance   : 10GBASE-LR
Link Length support  : 10km for SMF

=====
Transceiver Digital Diagnostic Monitoring (DDM), Internally Calibrated
=====

```

	Value	High Alarm	High Warn	Low Warn	Low Alarm
Temperature (C)	+25.4	+78.0	+73.0	-8.0	-13.0
Supply Voltage (V)	3.31	3.70	3.60	3.00	2.90
Tx Bias Current (mA)	35.6	85.0	80.0	20.0	15.0
Tx Output Power (dBm)	-1.46	2.00	1.00	-7.00	-8.00
Rx Optical Power (avg dBm)	-2.18	2.50	2.00	-18.01	-20.00

```

=====

Traffic Statistics
=====

```

	Input	Output
Octets	0	0
Packets	0	0
Errors	0	0
Utilization (300 seconds)	0.00%	0.00%

```

=====

Port Statistics
=====

```

	Input	Output
-----	-----	-----
Unicast Packets	0	0
Multicast Packets	0	0
Broadcast Packets	0	0
Discards	0	0
Unknown Proto Discards	0	
=====		
Ethernet-like Medium Statistics		
=====		
Alignment Errors :	0	Sngl Collisions : 0
FCS Errors :	0	Mult Collisions : 0
SQE Test Errors :	0	Late Collisions : 0
CSE :	0	Excess Collisns : 0
Too long Frames :	0	Int MAC Tx Errs : 0
Symbol Errors :	0	Int MAC Rx Errs : 0
In Pause Frames :	0	Out Pause Frames : 0
=====		

The above configuration results in an SD threshold of 5×10^{-5} (0.00005) and an SF threshold of 5×10^{-3} (0.005) over the default 10-second window. If this port is a 1GbE port supporting symbol monitoring then the error rate is compared against 1,250,000,000 symbols (10 seconds worth of symbols on a 1GbE port 125,000,000). If the error count in the current 10 second sliding window is less than 62,500 then the error rate is below the signal degrade threshold and no action is taken. If the error count is between 62,501 and 6,250,000 then the error rate is above signal degrade but has not breached the signal failure signal threshold and a log event is raised. If the error count is above 6,250,000 the signal failure threshold is crossed and the port enters an operation state of down. Consider that this is a very simple example meant to demonstrate the function and not meant to be used as a guide for configuring the various thresholds and window times.

A port is not returned to service automatically when a port enters the failed condition as a result of crossing a signal failure threshold for both CRC-Monitor and Symbol-Monitor. Because the port is operationally down without a physical link error monitoring stops. In MD-CLI, the user may enable the port using the **admin-state enable** and **admin-state disable** commands. In classic CLI, the user may enable the port using the **shutdown** and **no shutdown** commands. Other port transition functions like clearing the MDA or slot, removing the cable, and other physical link transition functions.

8 IEEE 802.3ah OAM

IEEE 802.3ah Clause 57 (EFM OAM) defines the OAM sublayer, which provides useful mechanisms for monitoring link operation, such as remote fault indication and remote loopback control. In general, OAM provides network operators the ability to monitor the network health and determine the location of failing links or fault conditions. EFM OAM described in this clause provides data link layer mechanisms that complement applications that may reside in higher layers.

OAM information is conveyed in slow protocol frames called OAM PDUs. OAM PDUs contain the control and status information used to monitor, test, and troubleshoot OAM-enabled links. OAM PDUs traverse a single link, passed between peer OAM entities. As a result, they are not forwarded by MAC clients (like bridges or switches).

The following EFM OAM functions are supported:

- EFM OAM capability discovery – allows the port to discover and advertise their OAM capabilities to the peer
- operational mode – supports both active and passive modes
- Remote Failure Indication (RFI) - enables the port to signal to its peer that a critical fault (for example, link fault, dying gasp) has occurred on its local receive path
- loopback – allows for a data link layer frame-level loopback mode. Both remote and local loopback modes are supported.
- EFM OAM PDU tunneling – allows PDUs to pass transparently through service provider network
- high-resolution timer for EFM OAM in 100 ms interval (minimum) – allows a more precise detection of connectivity failures
- non-zero Vendor Specific Information Field – adds extra device-specific detail to the standard Ethernet OAM protocol. The 32-bit field is encoded using the format 00:PP:CC:CC and references TIMETRA-CHASSIS-MIB:
 - 00 – must be zeros
 - PP – represents the platform type based on the installed IOM from `tmnxHwEquippedPlatform`.
 - CC:CC – represents the chassis type index value from `tmnxChassisType` which is indexed in `tmnxChassisTypeTable`. The table identifies the specific chassis backplane.

The 00:00:00:00 value is sent for all software releases that do not support the non-zero value or are unable to identify the required elements. Peer or local vendor information fields are not decoded on the network element. The hexadecimal value is included in the output of the following command.

```
show port ethernet efm-oam
```

When the EFM-OAM protocol fails to negotiate a peer session or encounters a protocol failure following an established session, the *Port State* enters the *Link Up* condition. This port state is used by many protocols to indicate the port is administratively UP, has physical connectivity, but the ports operational state is in a DOWN state due to a protocol, such as EFM-OAM. A reason code is added to help discern if the EFM-OAM protocol is the underlying reason for the *Link Up* condition.

Use the following command to display port information.

```
show port
```

Output example

```
=====
Ports on Slot 1
=====
Port      Admin Link Port   Cfg  Oper LAG/  Port Port Port   C/QS/S/XFP/
Id        State      State MTU  MTU  Bndl Mode Encp Type MDIMDX
-----
1/1/1     Down  No   Down   1578 1578  - netw null xcme
1/1/2     Down  No   Down   1578 1578  - netw null xcme
1/1/3     Up    Yes  Link Up 1522 1522  - accs qinq xcme
1/1/4     Down  No   Down   1578 1578  - netw null xcme
1/1/5     Down  No   Down   1578 1578  - netw null xcme
1/1/6     Down  No   Down   1578 1578  - netw null xcme
```

Use the following command to display information about a specific port.

```
show port 1/1/3
```

Output example

```
=====
Ethernet Interface
=====
Description      : 10/100/Gig Ethernet SFP
Interface        : 1/1/3
Link-level       : Ethernet
Admin State      : up
Oper State       : down
Reason Down      : efm0amDown
Physical Link     : Yes
Single Fiber Mode : No
IfIndex          : 35749888
Last State Change : 12/18/2012 15:58:29
Last Cleared Time : N/A
Phys State Chng Cnt: 1

Configured Mode   : access
Dot1Q Ethertype   : 0x8100
PBB Ethertype     : 0x88e7
Ing. Pool % Rate  : 100
Ing. Pool Policy  : n/a
Egr. Pool Policy  : n/a
Net. Egr. Queue Pol: default
Egr. Sched. Pol   : n/a
Auto-negotiate    : true
Oper Phy-tx-clock : not-applicable
Accounting Policy  : None
Acct Plcy Eth Phys : None
Egress Rate       : Default
Load-balance-algo : Default

Encap Type        : QinQ
QinQ Ethertype    : 0x8100
Egr. Pool % Rate  : 100

MTU               : 1522
Min Frame Length  : 64 Bytes
Hold time up      : 0 seconds
Hold time down    : 0 seconds
DDM Events        : Enabled

MDI/MDX           : unknown
Collect-stats     : Disabled
Collect Eth Phys  : Disabled
Ingress Rate      : Default
LACP Tunnel       : Disabled

Down-when-looped  : Disabled
Loop Detected     : False
Use Broadcast Addr : False

Sync. Status Msg. : Disabled
Rx Quality Level  : N/A
```

```

Tx DUS/DNU      : Disabled      Tx Quality Level : N/A
SSM Code Type   : sdh           ESMC Tunnel     : Disabled

Down On Int. Error : Disabled

CRC Mon SD Thresh : Disabled      CRC Mon Window  : 10 seconds
CRC Mon SF Thresh : Disabled

Configured Address : d8:ef:01:01:00:03
Hardware Address   : d8:ef:01:01:00:03

```

The user can also decouple the EFM-OAM protocol from the port state and operational state. To remove the protocol, monitor the protocol only, migrate, or make changes, the user should configure the following command.

```
configure port ethernet efm-oam ignore-efm-state
```

When the **ignore-efm-state** command is configured on a port, the protocol continues as normal. However, any failure in the protocol state machine (discovery, configuration, time-out, loops, and so on) does not impact the port on which the protocol is active and the optional **ignore-efm-state** command is configured. A protocol warning message is generated only if there are protocol issues. When this optional command is not configured, the default behavior is that the port state is affected by any EFM-OAM protocol fault or clear conditions. Adding and removing this optional **ignore-efm-state** command immediately represents the *Port State* and *Oper State* based on the active configuration. For example, if the **ignore-efm-state** command is configured on a port that is exhibiting a protocol error, this error does not affect the port state or operational state and there is no *Reason Down* code. If the **ignore-efm-state** is removed from a port with an existing EFM-OAM protocol error, the port transitions to *Link UP*, *Oper Down* with the reason code *efmOamDown*.

8.1 OAM events

The Information OAMPDU is transmitted by each peer at the configured intervals. This OAMPDU performs keepalive and critical notification functions. Various local conditions are conveyed through the setting of the Flags field. The following Critical Link Events, defined in IEEE 802.3 Section 57.2.10.1, are supported:

- link fault – the PHY has determined a fault has occurred in the receive direction of the local DTE
- dying gasp – an unrecoverable local failure condition has occurred
- critical event – an unspecified critical event has occurred

The local node can set and unset the various Flag fields based on the operational state of the port, shutdown or activation of the EFM-OAM protocol, or locally raised events. These Flag fields maintain the setting for a specific event. Changing port conditions, protocol state, or user intervention may impact the setting of these fields in the Information OAMPDU.

A peer processing the Information OAMPDU can take a configured action when one or more of these Flag fields are set. By default, receiving a set value for any of the Flag fields causes the local port to enter the Link Up port state and an event is logged. To bypass the default behavior and log the event without affecting the local port, use options under the following context, configurable per Flag field.

```
configure port ethernet efm-oam peer-rdi-rx
```

8.2 Remote loopback

EFM OAM provides a link-layer frame loopback mode that can be remotely controlled.

To initiate remote loopback, the local EFM OAM client sends a loopback control OAM PDU by enabling the OAM **remote-loopback** command. After receiving the loopback control OAM PDU, the remote OAM client places the remote port in local loopback mode.

To exit remote loopback, the local EFM OAM client sends a loopback control OAM PDU by disabling the OAM **remote-loopback** command. After receiving the loopback control OAM PDU, the remote OAM client places the port back in normal forwarding mode.

During a remote loopback test operation, all frames except EFM OAM PDUs are dropped at the local port for the receive direction, where remote loopback is enabled. To forward the received looped back non-EFM frames must be forwarded, use the following command to enable forwarding:

- **MD-CLI**

```
configure port ethernet efm-oam remote-loopback-forward-non-efm-frames true
```

- **classic CLI**

```
configure port ethernet efm-oam remote-lb-fwd-non-efm-frames true
```

If local loopback is enabled, all frames received on the port are looped back, and any frames generated or forwarded by the node are dropped in the transmit direction. This behavior may result in some protocols (for example, STP or LAG) resetting the state machines.

When a port is in loopback mode, service mirroring does not work if the port is a mirror-source or a mirror-destination.

8.3 802.3ah OAM PDU tunneling for Epipe service

Nokia routers support 802.3ah. Customers who subscribe to Epipe service treat the Epipe as a wire, and expect the ability to run 802.3ah between their devices located at either end of the Epipe.

This feature applies only to port-based Epipe SAPs because 802.3ah runs at the port level, not at the VLAN level. These ports must be configured as null encapsulated SAPs.

When OAM PDU tunneling is enabled, 802.3ah OAM PDUs received at one end of an Epipe are forwarded through the Epipe. 802.3ah can run between devices that are located at either end of the Epipe. When OAM PDU tunneling is disabled (the default setting), OAM PDUs are either dropped or processed locally, according to the EFM-OAM configuration state.

Enabling 802.3ah and OAM PDU tunneling on the same port are mutually exclusive operations.

9 Serial transport over raw sockets

Serial transport over raw sockets provides the capability of transporting serial data, in the form of characters, over an IP transport service within a Layer 3 IP/MPLS service (IES or VPRN). A raw socket allows direct sending and receiving of IP packets without any protocol-specific transport layer formatting. For information about raw socket IP transport services, see the *7705 SAR Gen 2 Services Overview Guide*.

This feature provides the functionality for a local host to listen to and open raw socket sessions from remote hosts, and for a remote host to initiate and open raw socket sessions to local hosts. The local and remote host functions support TCP or UDP sessions (but not both concurrently) over the IP transport service.

Raw sockets are supported on RS-232 ports on the 4-port RS-232 and 4-port C37.94 interface card.

**Note:**

- RS-232 serial data can be carried over Cpipes or over raw sockets using IP transport. To use Cpipes, the RS-232 port must be configured with a channel ID. To use raw sockets, the RS-232 port must be configured with a socket ID.
- The node supports a mix of Cpipes and raw socket serial links on the same card.

Figure 33: Serial transport over raw socket application on the 7705 SAR-Hx shows an example of a raw socket application, where serial data is transferred between RTUs and the utility SCADA management system using an IP transport service across a Layer 3 service (IES or VPRN), that includes 7705 SAR-Hx.

A raw socket local host (acting as a server) at the 7705 SAR-Hx substation listens to TCP sessions that originate at the 7705 SAR-8 Shelf V2/7705 SAR-18 central location network operations center (NOC). The 7705 SAR-8 Shelf V2/7705 SAR-18 at the NOC is connected to two front-end processors (FEPs), one via a serial port and another via an Ethernet port. The serial port on the 7705 SAR-8 Shelf V2/7705 SAR-18 is configured as a remote host (acting as a client) that initiates TCP/UDP sessions toward the RTU at the 7705 SAR-Hx substation when traffic is received from the FEP over the serial port. These TCP/UDP sessions are transported over the IP/MPLS network using IP transport service over an IES or VPRN service. The serial data that is transported over the TCP/UDP session and received at the 7705 SAR-Hx is sent over the serial link toward the RTU. TCP/UDP sessions received from the FEP over the Ethernet port are transported over an IES or VPRN service (that is, there is no need for serial remote host configuration in this case).

Multiple FEPs can poll a single RTU. If multiple sessions attempt to transmit serial data on the serial port simultaneously, the 7705 SAR Gen 2 queues packets per session and ensures that all data for one session is sent out before processing another session's data, ensuring that sessions do not overlap one another.

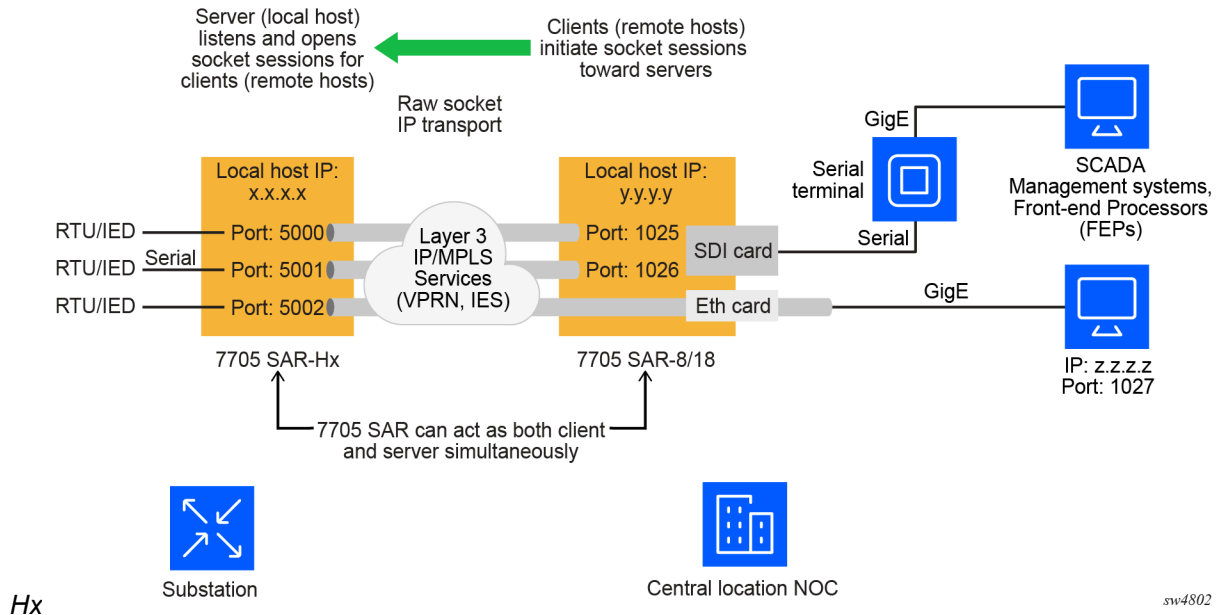


Note: A serial port can be concurrently configured as both a server (local host) and a client (remote host) using the following commands.

```
configure service ies ip-transport local-host
configure service vprn ip-transport local-host
configure service ies ip-transport remote-host
configure service vprn ip-transport remote-host
```

This is accomplished with the **local-host** configuration to support the server function and the **remote-host** configuration to set up client sessions to far-end remote hosts.

Figure 33: Serial transport over raw socket application on the 7705 SAR-



9.1 Raw socket configuration

A raw socket IP transport interface can be configured for each RS-232 serial port on a node. This allows the serial port to receive TCP connections or UDP session packets from multiple remote hosts, or to create new sessions to remote hosts to send and receive serial data.

There are port-level and service-level configuration requirements for a raw socket serial port to send and receive serial data in either server mode, client mode, or both.

A raw socket port-level configuration includes defining the EOP checking parameters (idle time, length, special character) and the inter-session delay for transmitting session data over the serial link.

At the service level, an IP transport subservice is created within an IES or VPRN service to associate the serial port with the respective IES or VPRN service. TCP/UDP encapsulated serial data is routed within the corresponding Layer 3 IES or VPRN service. The required configuration includes IP transport subservice local-host and remote-host configuration, TCP timers, and session control. See the *7705 SAR Gen 2 Services Overview Guide* for the required information.

9.2 Raw socket packet processing

Figure 34: [Raw socket packet processing](#) displays how raw socket packets are processed over a serial link.

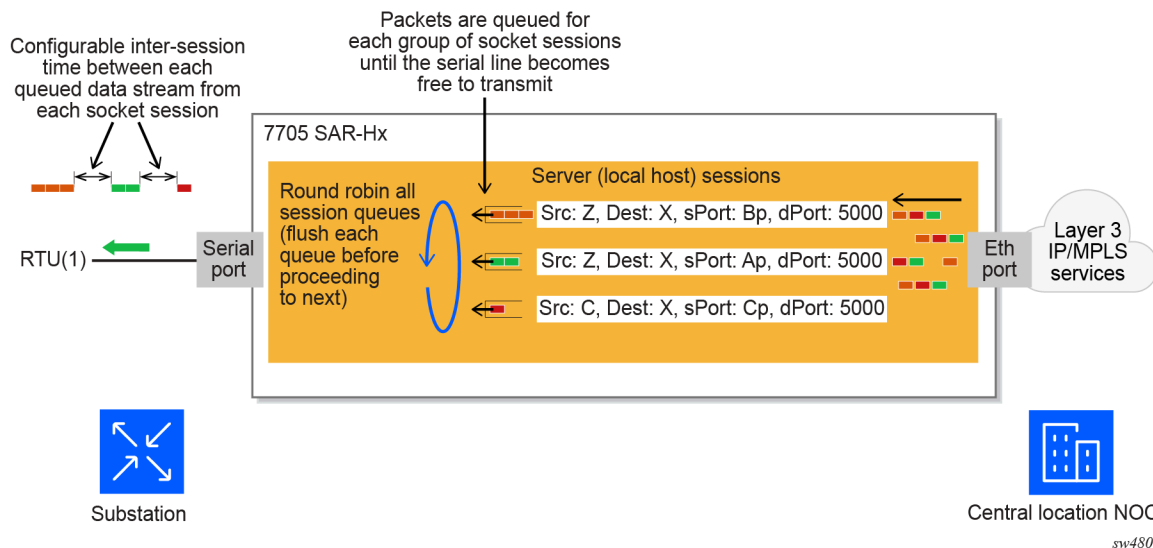
Session data attempting to access the serial port is queued. One queue is maintained per session. The purpose of the session queue is to prevent two different flows of packets from interleaving out of the serial

port and creating unreadable messages. When data is transmitted over the serial link for a session, any other session's data is queued until the first session has emptied its queue. The next session's data is transmitted over the serial link only after the following command timer expires.

```
configure port serial rs232 socket tx inter-session-delay
```

Each session's data is sent out in round-robin fashion.

Figure 34: Raw socket packet processing



9.2.1 Raw socket processing for UDP sessions

When the local host receives a UDP packet from a remote host, it queues the packet and sends it over the serial link. The local host remembers the UDP session while there is still data to send from the serial link. If further packets are received for the same session, they are queued behind the already-queued packet. After all the queued data has been sent over the serial link, the session is removed from the system. An associated UDP remote host for the serial link must be configured to have serial data sent back to the remote host from the serial port.

When a packet is received from the serial link based on EOP requirements, the data is copied and sent in a UDP packet to each configured remote host.

9.2.2 Raw socket processing for TCP sessions

An open TCP session from a remote host to a raw socket's local host is kept open until either the remote host terminates the session or the TCP inactivity timer expires. When a TCP session is open, all packets received from the remote host are queued for the raw socket serial link and sent over the serial link until no packets remain in the queue.

If multiple sessions are open toward the local host and each is receiving data, each session's data is queued and sent over the serial link in round-robin fashion until no packets remain. When a packet is received over the serial link, it is copied to each open TCP session and transmitted to the remote host.

9.3 Raw socket squelch functionality

A condition may occur where the end device connected to the serial port continues to send out a continuous stream of data after the normal response period has expired. This can prevent the far-end remote host or master equipment from receiving data from other end devices in the network. To resolve this condition, the **squelch** command can be used on the raw socket at the port level (it is disabled by default). This stops the socket from receiving any more data from the problematic device.

If the command is enabled, the 7705 SAR Gen 2 monitors the serial port for a constant character stream. A configurable squelch delay period, using the following command, determines how long to measure the constant character stream before initiating the squelch function.

```
configure port serial rs232 socket rx squelch-delay
```

If the squelch function is initiated, the port is considered locked up, and an alarm is raised indicating the lock-up and that the squelching function has been triggered.

The serial port can be forced out of squelch and put back to normal, either manually using the **squelch-reset** command or automatically using the following command.

```
configure port serial rs232 socket rx unsquelch-delay
```

The **unsquelch-delay** command defines the time to wait after squelch is initiated before it is removed.

10 MTU configuration guidelines

The following MTU configuration guidelines apply:

- The router provides the option to configure MTU limitations at many service points. The physical (access and network) port, service, and SDP MTU values must be individually defined.
- Identify ports designated as network ports and intended to carry service traffic.
- MTU values should not be modified frequently.
- The service MTU values must conform to the following conditions:
 - must be less than or equal to the SDP path MTU
 - must be less than or equal to the access port (SAP) MTU
- When the network group encryption (NGE) feature is enabled, additional bytes because of NGE packet overhead must be considered. See the "NGE Packet Overhead and MTU Considerations" section in the *7705 SAR Gen 2 Services Overview Guide* for more information.

10.1 Default MTU values

The following table lists the default MTU values that are dependent upon the (sub-) port type, mode, and encapsulation.

Table 20: MTU default values

Port type	Mode	Encap type	Default (bytes)
Ethernet	access	null	1514
Ethernet	access	dot1q	1518
Fast Ethernet ¹⁴	network	—	1514
Other Ethernet	network	—	9212 ¹⁵

10.2 Modifying MTU defaults

MTU command options must be modified at the service level as well as the port level.

- The service-level MTU command options configure the service payload (Maximum Transmission Unit – MTU) in bytes for the service ID overriding the service-type default MTU.

¹⁴ Physical/native Fast Ethernet only.

¹⁵ The default MTU for Ethernet ports other than Fast Ethernet is actually the lesser of 9212 and any MTU limitations imposed by hardware, which is typically 16K.

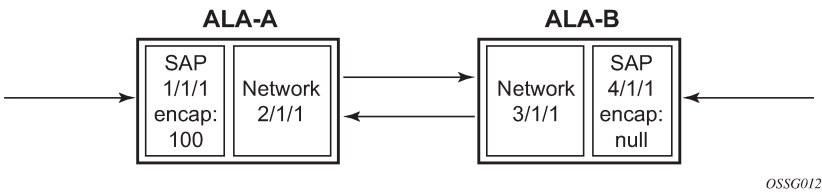
- The port-level MTU command options configure the maximum payload MTU size for an Ethernet port, LAG, or SONET/SDH SONET path (sub-port) or TDM port/channel.

The default MTU values must be modified to ensure that packets are not dropped because of frame size limitations. The service MTU must be less than or equal to both the SAP port MTU and the SDP path MTU values. When an SDP is configured on a network port using default port MTU values, the operational path MTU can be less than the service MTU. In this case, enter the `show service sdp` command to check the operational state. If the operational state is down, then modify the MTU value accordingly.

10.3 Configuration example

In order for the maximum length service frame to successfully travel from a local ingress SAP to a remote egress SAP, the MTU values configured on the local ingress SAP, the SDP (GRE or MPLS), and the egress SAP must be coordinated to accept the maximum frame size the service can forward. For example, the targeted MTU values to configure for a distributed Epipe service (ALA-A and ALA-B) are shown in [Figure 35: MTU configuration example](#).

Figure 35: MTU configuration example



Because ALA-A uses Dot1q encapsulation, the SAP MTU must be set to 1518 to be able to accept a 1514 byte service frame (see [Default MTU values](#) for MTU default values). Each SDP MTU must be set to at least 1514 as well. If ALA-A's network port (2/1/1) is configured as an Ethernet port with a GRE SDP encapsulation type, then the MTU value of network ports 2/1/1 and 3/1/1 must each be at least 1556 bytes (1514 MTU + 28 GRE/Martini + 14 Ethernet). Finally, the MTU of ALA-B's SAP (access port 4/1/1) must be at least 1514, as it uses null encapsulation.

[Table 21: MTU configuration example values](#) shows example MTU configuration values.

Table 21: MTU configuration example values

	ALA-A		ALA-B	
	Access (SAP)	Network	Network	Access (SAP)
Port (slot/MDA/port)	1/1/1	2/1/12	3/1/1	4/1/1
Mode or ECAP-type	dot1q	network	network	null
MTU	1518	1556	1556	1514

11 Deploying preprovisioned components

When a card, or MDA is installed in a preprovisioned slot, the device detects discrepancies between the preprovisioned card type configurations and the types actually installed. Error messages display if there are inconsistencies and the card does not initialize.

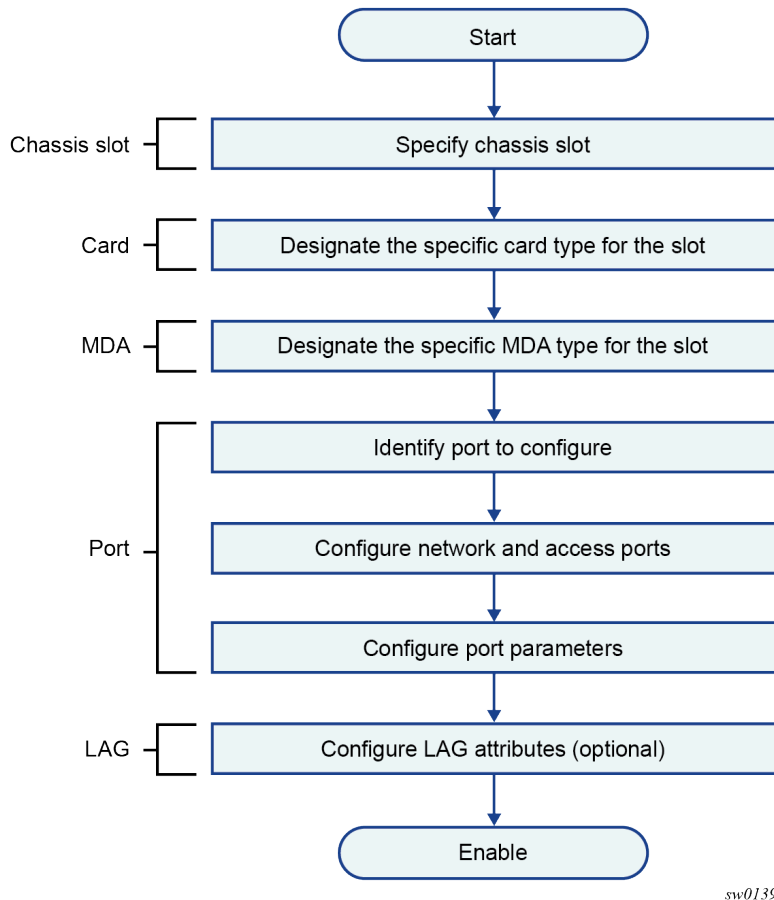
When the correct preprovisioned cards are installed into the appropriate chassis slot, alarm, status, and performance details are displayed.

On bootup, preprovisioned (and administratively enabled) slots are checked after an initial 15-minute period to ensure that they are present in the slot. If the card is not detected the system raises an alarm.

12 Configuration process overview

The following figure shows the process to provision chassis slots, cards, MDAs, and ports.

Figure 36: Slot, card, MDA, port configuration, and implementation flow



sw0139

12.1 Configuration notes

The following information describes provisioning restrictions.

- If a card or MDA type is installed in a slot provisioned for a different type, the card does not initialize.
- A card or MDA installed in an unprovisioned slot remains administratively and operationally down until the card type and MDA are specified.
- Ports cannot be provisioned until the slot, card and MDA type are specified.

13 Configuring physical ports with CLI

This section provides information to configure cards, MDAs, and ports.

13.1 Preprovisioning guidelines

7705 SAR Gen 2 platforms have a console port, either located on the CPM or integrated into the chassis, to connect terminals to the router.

Configure command options from a system console connected to a router console port, using Telnet to access a router remotely or SSH to open a secure shell connection.

13.1.1 Predefining entities

To initialize a card, the chassis slot, line card type, and MDA type must match the preprovisioned command options. In this context, preprovisioning means to configure the entity type (such as the card type, MDA type, port, and interface) that is planned for a chassis slot, card, or MDA. Preprovisioned entities can be installed but not enabled or the slots can be configured but remain empty until populated. Provisioning means that the preprovisioned entity is installed and enabled.

You can:

- Preprovision ports and interfaces after the line card and MDA types are specified.
- Install line cards in slots with no pre-configuration command options specified. After the card is installed, the card and MDA types must be specified.
- Install a line card in a slot provisioned for a different card type (the card does not initialize). The existing card and MDA configuration must be deleted and replaced with the current information.

13.1.2 Preprovisioning a port

Before a port can be configured, the slot must be preprovisioned with an allowed card type and the MDA must be preprovisioned with an allowed MDA type. Some recommendations to configure a port include:

- **Ethernet**
 - Configure an access port for customer facing traffic on which services are configured.
 - An encapsulation type may be specified to distinguish services on the port or channel. Encapsulation types are not required for network ports.
 - To configure an Ethernet access port, see [Configuring Ethernet access ports](#).

13.1.3 Maximizing bandwidth use

After ports are preprovisioned, Link Aggregation Groups (LAGs) can be configured to increase the bandwidth available between two nodes.

All physical links or channels in a LAG/bundle combine to form one logical connection. A LAG/bundle also provides redundancy in case one or more links that participate in the LAG/bundle fail. For command syntax for LAG, see [Configuring LAG](#).

13.2 Basic configuration

The most basic configuration must specify the following:

- line card type (must be an allowed card type)
- MDA slot
- MDA (must be an allowed MDA type)
- specific port to configure

The following is an example of card configuration for the 7705 SAR Gen 2.

Example: MD-CLI

```
[ex:]
A:admin@node-2# admin show configuration
configure {
  card 6 {
    card-type iom4-e
    mda 1 {
      mda-type me1-100gb-cfp2
    }
    fp 1 {
    }
  }
  card 7 {
    card-type iom4-e
    mda 1 {
      mda-type me10-10gb-sfp+
    }
    mda 2 {
      mda-type me1-100gb-cfp2
    }
    fp 1 {
    }
  }
  card 8 {
    card-type iom4-e
    mda 1 {
      mda-type me10-10gb-sfp+
    }
  }
}
```

Example: classic CLI

```
A:node-2> admin display-config
echo "Card Configuration"
#-----
card 6
  card-type iom4-e
  no shutdown
exit
card 7
  card-type iom4-e
```

```

mda 1
  mda-type me10-10gb-sfp+
  no shutdown
exit
mda 2
  mda-type me1-100gb-cfp2
  no shutdown
exit
no shutdown
exit
card 8
  card-type iom4-e
  no shutdown
exit
#-----

```

13.3 Common configuration tasks

The following sections are basic system tasks that must be performed.

13.3.1 Configuring cards and MDAs

Card configurations include a chassis slot designation. A slot must be preconfigured with the type of cards and MDAs which are allowed to be provisioned.

The following example shows card and MDA configurations for the 7705 SAR Gen 2.



Note: On 7705 SAR Gen 2 platforms, the card is permanently equipped and provisioned in slot 1 cannot be removed.

Example: MD-CLI

```

[ex: /configure card 1]
A:admin@node-2# info
mda 1 {
  admin-state enable
  mda-type m2-1g-sfp+2-10g-sfp+
}
mda 2 {
  admin-state disable
}
mda 3 {
}
mda 4 {
}

```

Example: classic CLI

```

A:node-2>config>card# info
#-----
card-type iom-sar-1x
mda 1
  no shutdown
  mda-type m2-1g-sfp+2-10g-sfp+
exit
mda 2

```

```

        shutdown
        mda-type m4-1g-rj+6-10g-sfp++2-25g-sfp28
    exit
    no shutdown
#-----

```

13.3.2 Configuring ports

This section provides the CLI and examples to configure port command options.

13.3.2.1 Configuring port pools

The buffer space is portioned out on a per port basis. Each port gets an amount of buffering which is its fair-share based on the port's bandwidth compared to the overall active bandwidth.

This mechanism takes the buffer space available and divides it into a portion for each port based on the port's active bandwidth relative to the amount of active bandwidth for all ports associated with the buffer space. The number of ports sharing the same buffer space depends on the type of MDAs populated on the IOM. An active port is considered to be any port that has an active queue associated. After a queue is created for the port, the system allocates the appropriate amount of buffer space to the port. This process is independently performed for both ingress and egress.

Normally, the amount of active bandwidth is considered as opposed to total potential bandwidth for the port when determining the port's fair share. If a port is channelized and not all bandwidth is allocated, only the bandwidth represented by the configured channels with queues configured is counted toward the bandwidth represented by the port. Also, if a port may operate at variable speeds (as in some Ethernet ports), only the current speed is considered. Based on the above, the number of buffers managed by a port may change because of queue creation and deletion, channel creation and deletion and port speed variance on the local port or other ports sharing the same buffer space.

After the active bandwidth is calculated for the port, the result may be modified through the use of the following commands.

- **MD-CLI**

```

configure port modify-buffer-allocation percentage-of-rate egress
configure port modify-buffer-allocation percentage-of-rate ingress

```

- **classic CLI**

```

configure port modify-buffer-allocation egr-percentage-of-rate
configure port modify-buffer-allocation ing-percentage-of-rate

```

The default value of each is 100% which allows the system to use all of the ports active bandwidth when deciding the relative amount of buffer space to allocate to the port. When the value is explicitly modified, the active bandwidth on the port is changed according to the specified percentage. If a value of 50% is given, the ports active bandwidth is multiplied by 5, if a value of 150% is given, the active bandwidth is multiplied by 1.5. The ports rate percentage command options may be modified at any time.

To modify (in this example, to double) the size of buffer allocated on ingress for a port.

Example: MD-CLI

```

configure port 1/2/1 modify-buffer-allocation-rate percentage-of-rate ingress 200

```

Example: classic CLI

```
configure port 1/2/1 modify-buffer-allocation-rate ing-percentage-of-rate 200
```

To modify (in this example, to double) the size of buffer allocated on egress for a port.

Example: MD-CLI

```
configure port 1/2/1 modify-buffer-allocation-rate percentage-of-rate egress 200
```

Example: classic CLI

```
configure port 1/2/1 modify-buffer-allocation-rate egr-percentage-of-rate 200
```

The default buffer allocation has the following characteristics:

- Each port manages a buffer according to its active bandwidth (ports with equal active bandwidth get the same buffer size).
- An access port has 2 default pools created: access-ingress and access-egress.
- A network port has 2 default pools created: ingress-FP (common pool for all ingress network ports) and network-egress.
- All queues defined for a port receive buffers from the same buffer pool.

The following example shows port pool configurations.

Example: MD-CLI

```
[ex:/configure port 1/1/1]
A:admin@node-2# info
  admin-state enable
  access {
    egress {
      pool "default" {
        slope-policy "slopePolicy1"
      }
    }
  }
  network {
    egress {
      pool "default" {
        slope-policy "slopePolicy2"
      }
    }
  }
}
```

Example: classic CLI

```
A:node-2>config>port# info
-----
  access
    egress
      pool
        slope-policy "slopePolicy1"
      exit
    exit
  exit
  network
    egress
```

```

        pool
            slope-policy "slopePolicy2"
        exit
    exit
exit
no shutdown
-----

```

The following shows a CBS configuration over subscription example.

Example: MD-CLI

```

[ex:/configure port 1/1/1]
A:admin@node-2# info
  admin-state enable
  access {
    ingress {
      pool "default" {
        amber-alarm-threshold 10
        resv-cbs {
          cbs 10
          amber-alarm-action {
            step 1
            max 30
          }
        }
      }
    }
  }
}
ethernet {
  mode access
  encap-type dot1q
}

```

Example: classic CLI

```

A:node-2>config>port# info
-----
  access
    ingress
      pool
        amber-alarm-threshold 10
        resv-cbs 10 amber-alarm-action step 1 max 30
      exit
    exit
  exit
ethernet
  mode access
  encap-type dot1q
exit
no shutdown

```

13.3.2.2 Changing hybrid-buffer-allocation

The following example shows a hybrid-buffer-allocation value change (from default) for ingress. In this example, the network-egress buffer pool is two times the size of the access-egress.

Example: MD-CLI

```
[ex:/configure port 1/1/2 hybrid-buffer-allocation]
A:admin@node-2# info
  egress-weight {
    access 20
    network 40
  }
```

Example: classic CLI

```
A:node-2config>port>hybrid-buffer-allocation# info
-----
egr-weight access 20 network 40
```

13.3.2.3 Configuring Ethernet ports

13.3.2.3.1 Configuring Ethernet network ports

A network port is network-facing and participates in the service provider transport or infrastructure network processes.

The following example shows a network port configuration.

Example: MD-CLI

```
[ex:/configure port A/3]
A:admin@node-2# info
  admin-state enable
  description "Ethernet network port"
```

Example: classic CLI

```
A:node-2config>port# info
-----
  description "Ethernet network port"
  ethernet
  exit
  no shutdown
-----
```

13.3.2.3.2 Configuring Ethernet access ports

Services are configured on access ports that are used for customer-facing traffic. If a SAP is to be configured on a port, it must be configured as access mode. When a port is configured for access mode, the appropriate encapsulation type can be specified to distinguish the services on the port. After a port has been configured for access mode, multiple services can be configured on the port.

The following example shows an Ethernet access port configuration.

Example: MD-CLI

```
[ex:/configure port 1/1/cl/1]
A:admin@node-2# info
  admin-state enable
  description "Ethernet access port"
  ethernet {
    mode access
    encap-type dot1q
  }
```

Example: classic CLI

```
A:node-2>config>port# info
-----
description "Ethernet access port"
ethernet
  mode access
  encap-type dot1q
exit
no shutdown
-----
```

13.3.2.3.3 Configuring an 802.1x authentication port

The following example shows an 802.1x port configuration.

Example: MD-CLI

```
[ex:/configure port 1/2/4 ethernet dot1x]
A:admin@node-2# info detail
...
  admin-state enable
  max-authentication-requests 2
  port-control auto
  quiet-period 60
  radius-policy dot1xpolicy
  server-timeout 30
  supplicant-timeout 30
  transmit-period 30
  tunneling false
  tunnel-dot1q true
  tunnel-qinq true
  re-authentication {
    period 3600
  }
...
```

Example: classic CLI

```
A:node-2>config>port>ethernet>dot1x# info detail
-----
port-control auto
radius-plcy dot1xpolicy
re-authentication
re-auth-period 3600
max-auth-req 2
transmit-period 30
```

```

quiet-period 60
supplicant-timeout 30
server-timeout 30
no tunneling
no shutdown
-----

```

13.3.2.4 Configuring LAG

LAG configurations should include at least two ports. Other considerations include the following.

- A maximum of 64 ports (depending on the lag-id) can be included in a LAG. All ports in the LAG must share the port characteristics inherited from the primary port.
- Auto-negotiation must be disabled or set to limited mode for ports that are part of a LAG, to guarantee a specific port speed.
- Ports in a LAG must be configured as full duplex.

The following example shows the LAG configuration output.

Example: MD-CLI

```

[ex:/configure lag "lag-2"]
A:admin@node-2# info
  description "LAG2"
  mac-address 04:68:ff:00:00:01
  dynamic-cost true
  port-threshold {
    value 4
    action down
  }
  port 1/1/1 {
  }
  port 1/3/1 {
  }
  port 1/5/1 {
  }
  port 1/7/1 {
  }
  port 1/9/1 {
  }

```

Example: classic CLI

```

A:node-2>config>lag# info detail
-----
  description "LAG2"
  mac 04:68:ff:00:00:01
  port 1/1/1
  port 1/3/1
  port 1/5/1
  port 1/7/1
  port 1/9/1
  dynamic-cost
  port-threshold 4 action down
-----

```

13.3.2.5 Configuring GNSS ports

The 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx are equipped with an integrated GNSS receiver and GNSS RF port for retrieval and recovery of GPS and Galileo signals.



Note: GPS signal recovery must always be enabled in the system when using the GNSS receiver.

Use the commands in the following context to configure integrated GNSS RF ports, including antenna-cable delay, signal recovery, and elevation-mask angle.

```
configure port gnss
```

GNSS port settings allow you to configure specific constellations and antenna cable delay. You can also set the minimum carrier-to-noise ratio and elevation masks to ignore satellites that do not meet the specified thresholds.

13.3.2.6 Configuring TDM ports

13.3.2.6.1 Configuring T1/E1 ports

Use the commands in the following context to configure TDM for T1 and E1 on the 7705 SAR-10x, 7705 SAR-Hx, and 7705 SAR-Mx.

```
configure port tdm
```

The following example shows a TDM configuration for E1.

Example

```
A:node-2>config>port>tdm# info detail
-----
no length
e1
  shutdown
  framing g704
  loopback line
  clock-source node-timed
  signal-mode cas
  report-alarm ais los
  no report-alarm oof rai
  no hold-time
  ssm
    shutdown
    ssm-bit 8
    no tx-dus
  exit
  channel-group 1
    description "DS0GRP"
    mode access
    encap-type cem
    timeslots 2-16
    loopback line
```

```

        idle-payload-fill all-ones
        idle-signal-fill all-ones
        no shutdown
    exit
exit
line-impedance 120
-----

```

13.3.2.6.2 Configuring E1 SSM

Use the commands in the following context to configure Synchronization Status Messaging (SSM) for E1 TDM ports.

```
configure port tdm e1 ssm
```



Note:

- Use only the **g704** framing mode with E1 SSM. The **no-crc-g704** and **e1-unframed** framing modes are not compatible with E1 SSM.
- The **ssm-bit** applies only to E1 SSM.

The following example shows an SSM configuration on an E1 port.

Example

```

A:node-2>config>port>tdm# info
-----
    e1
        ssm
            ssm-bit 4
            tx-dus
            no shutdown
        exit
    no shutdown
exit
-----

```

13.3.2.7 Configuring teleprotection ports

Use the following command to configure an IEEE C37.94 TPIF port on an 8-port RS-232 and C37.94 Combination MDA or 8-port C37.94 Teleprotection card.

```
configure port tdm tpif
```

Output example: TPIF port configuration

```

A:node-2>config>port>tdm>tpif$ info detail
-----
        shutdown
        no loopback
        report-alarm los rai
        channel-group 1
            shutdown
            no description

```

```
timeslots 1
exit
-----
```

13.3.2.8 Configuring serial ports for raw socket transport

Use the following CLI syntax to configure raw socket parameters on an RS-232 serial port.



Note: Raw sockets are only supported on RS-232 serial ports.

Example: Raw socket configuration on an RS-232 serial port

```
A:node-2>config>port>serial>rs232$ info detail
-----
shutdown
no loopback
character-length 8
no parity
stop-bits 1
control-lead
  input
    dtr-dsr high
    rts-dcd high
  exit
  output
    dcd-rts high
    cts-alb high
    ri-rdl high
  exit
  monitor
    dtr-dsr off
    rts-dcd off
  exit
exit
s-bit-signaling off
no data-position
no mon-data-inactivity
no hold-time
report-alarm ctrlLeadMon monDataInac
no report-alarm hcm0of hcmRai
speed 9600
socket 1
  shutdown
  description "SOCKET"
  rx
    eop
      length 1500
      idle-timeout 50
      no special-char
    exit
    no squelch-delay
    no unsquelch-delay
  exit
  tx
    inter-session-delay 10
  exit
exit
-----
```

14 Service management tasks

This section discusses basic procedures to complete service management tasks.

14.1 Modifying or deleting an MDA

To change an MDA type already provisioned for a specific slot or card, first you must shut down the slot/MDA/port configuration and then delete the MDA from the configuration.

You must reset the IOM after changing the MDA type from MS-ISA to any other MDA type.

The following example shows how to modify the configuration of an MDA on the 7705 SAR Gen 2.

Example: MD-CLI

```
[ex:/configure port 1/1/c1]
A:admin@node-2# admin-state disable

*[ex:/configure port 1/1/c1]
A:admin@node-2# back

*[ex:/configure]
A:admin@node-2# card 1 mda 1

*[ex:/configure card 1 mda 1]
A:admin@node-2# admin-state disable

*[ex:/configure card 1 mda 1]
A:admin@node-2# mda-type m2-1g-sfp+2-10g-sfp+
```

Example: classic CLI

```
A:node-2>config# port 1/1/c1
*A:node-2>config>port# shutdown
*A:node-2>config>port# back
*A:node-2>config# card 1 mda 1
*A:node-2>config>card>mda# shutdown
*A:node-2>config>card>mda# no mda-type
*A:node-2>config>card>mda# mda-type "m2-1g-sfp+2-10g-sfp+"
*A:node-2>config>card>mda#
```

14.2 Modifying a card type

To modify the card type already provisioned for a specific slot, you must shutdown existing port configurations and shutdown and remove all MDA configurations.

You must reset the IOM after changing the MDA type from MS-ISA to any other MDA type.

The following example shows how to administratively disable a port and card before you modify a card type already provisioned for a specific slot.

Example: MD-CLI

```
*[ex:/ configure]
A:admin@node-2# port 1/1/c1

*[ex:/ configure port]
A:admin@node-2# admin-state disable

*[ex:/ configure card]
A:admin@node-2# mda 1

*[ex:/ configure card mda]
A:admin@node-2# admin-state disable
```

Example: classic CLI

```
*A:node-2>config# port 1/1/c1
*A:node-2>config>port# shutdown
*A:node-2>config>card> mda 1
*A:node-2>config>card>mda# shutdown
*A:node-2>config>card>mda# no mda-type
```

14.3 Deleting a card

On 7705 SAR Gen 2 platforms, cards are permanently built into the chassis and have preassigned parameters. The cards cannot be modified or removed.

14.4 Deleting port command options

The following example shows the deletion of a port provisioned for a specific card:

Example: MD-CLI

```
*[ex:/ configure]
A:admin@node-2# port 1/2/12

*[ex:/ configure port]
A:admin@node-2# admin-state disable
```

Example: classic CLI

```
*A:node-2>config# port 1/2/12
*A:node-2>config>port# shutdown
*A:node-2>config>port# exit
*A:node-2>config# no port 1/2/12
```

14.5 Soft IOM reset

This section provides basic procedures for soft IOM reset service management tasks.

14.5.1 Soft reset

Soft reset is an advanced high availability feature that greatly reduces the impact of IOM/IMM resets either during a software upgrade or during other maintenance or debug operations. The combination of In Service Software Upgrade (ISSU) and Soft reset maximizes service availability in an operational network.

A soft reset re-initializes the control plane while the data plane continues operation with only very minimal impact to data forwarding. During the soft reset some processes that rely on the IOM control plane do not run for a duration that is similar to the duration of an IOM Hard reset. These processes include the updating of the IP forwarding table on the IOM (IP FIB downloads from the CPM), Layer 2 learning of new MAC addresses on the IOM, updating of the MAC forwarding table (for MAC addresses learned from other IOMs), ARP, Ethernet OAM 802.3ah, LLDP and handling for specific ICMP functions such as Can't Fragment, Redirect, Host Unreachable, Network Unreachable and TTL Expired. Note that protocols and processes on the CPM continue to operate during a Soft Reset (BGP continues to learn new routes from peers, and the new routes are downloaded to the IOM after the Soft Reset has completed).

The combination of the very small data plane impact and special soft reset enhancements for protocols ensures that most protocols do not go down and no visible impacts to most protocols are detected externally to the 7705 SAR Gen 2 platforms. BFD timers are temporarily increased for the duration of a soft reset to keep BFD sessions up. Protocols such as BGP, OSPF, IS-IS, PIM, and so on with default timers remain up. A protocol using aggressive timers may go down momentarily during a soft reset.

Although the majority of protocols stay up during a Soft Reset, there are some limitations for a few protocols. See *Known Limitations* in the *Release Notes* for the relevant release for details.

Configuration changes are not allowed while any card is in the process of a soft reset.

The soft IOM reset procedure is applicable during the ISSU process and for a manual soft reset procedure.

To manually perform a soft IOM reset, enter the following command.

```
clear card soft
```

Soft Reset is supported on Ethernet IMMs and on IOMs that have Ethernet MDAs provisioned. The user can optionally force a Soft Reset on an IOM that contains at least one MDA that supports Soft Reset but also has an MDA that does not support Soft Reset or is operationally down. To force Soft Reset in this case the following command is used and the supported MDAs and the card itself are soft reset while the MDAs that do not support soft reset (or are operationally down) are hard reset.

```
clear card soft hard-reset-unsupported-mdas
```

The **show card** and **show mda** commands indicate that a soft IOM reset is occurring during the soft reset process.

14.5.2 Deferred MDA reset

As part of an ISSU, soft reset is supported even if the (old) firmware version on the MDAs is not the same as the (new) firmware version in the software load to which the user is upgrading. The soft reset is allowed to proceed by leaving the previous version of the firmware running while upgrading the rest of the MDA/IOM/IMM. The user can then issue a hard reset of the MDA/IMM at some time in the future to upgrade the firmware.

The soft reset is only allowed to proceed if the older firmware is compatible with the new IOM/IMM software load. Otherwise the soft reset is blocked and a hard reset must be used instead.

After a soft reset has been completed, a log event is raised to warn the user that the MDA (or IMM) is running older firmware and that they can perform a hard reset of the MDA (or IMM) at some point if required.

If the MDA/IMM is not hard reset by the user, and then a software upgrade is performed, and the older firmware is no longer compatible with the newest load being upgraded to, then the soft reset is blocked (or an automatic hard reset occurs for ISSU).

The user can see whether they are running with older MDA/IMM firmware at any time by using the following command.

```
show mda detail
```

15 Standards and protocol support

**Note:**

The information provided in this chapter is subject to change without notice and may not apply to all platforms.

Nokia assumes no responsibility for inaccuracies.

15.1 Bidirectional Forwarding Detection (BFD)

RFC 5880, *Bidirectional Forwarding Detection (BFD)*

RFC 5881, *Bidirectional Forwarding Detection (BFD) IPv4 and IPv6 (Single Hop)*

RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*

15.2 Border Gateway Protocol (BGP)

draft-hares-idr-update-attr-low-bits-fix-01, *Update Attribute Flag Low Bits Clarification*

draft-ietf-idr-add-paths-guidelines-08, *Best Practices for Advertisement of Multiple Paths in IBGP*

draft-ietf-idr-best-external-03, *Advertisement of the best external route in BGP*

draft-ietf-idr-bgp-gr-notification-01, *Notification Message support for BGP Graceful Restart*

draft-ietf-idr-bgp-optimal-route-reflection-10, *BGP Optimal Route Reflection (BGP-ORR)*

draft-ietf-idr-error-handling-03, *Revised Error Handling for BGP UPDATE Messages*

draft-ietf-idr-link-bandwidth-03, *BGP Link Bandwidth Extended Community*

RFC 1772, *Application of the Border Gateway Protocol in the Internet*

RFC 1997, *BGP Communities Attribute*

RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*

RFC 2439, *BGP Route Flap Damping*

RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*

RFC 2858, *Multiprotocol Extensions for BGP-4*

RFC 2918, *Route Refresh Capability for BGP-4*

RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*

RFC 4360, *BGP Extended Communities Attribute*

RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4456, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)*

RFC 4486, *Subcodes for BGP Cease Notification Message*

RFC 4659, *BGP/MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*

RFC 4684, *Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)*

RFC 4724, *Graceful Restart Mechanism for BGP – helper mode*

RFC 4760, *Multiprotocol Extensions for BGP-4*

RFC 4798, *Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)*

RFC 5004, *Avoid BGP Best Path Transitions from One External to Another*

RFC 5065, *Autonomous System Confederations for BGP*

RFC 5291, *Outbound Route Filtering Capability for BGP-4*

RFC 5396, *Textual Representation of Autonomous System (AS) Numbers – asplain*

RFC 5492, *Capabilities Advertisement with BGP-4*

RFC 5668, *4-Octet AS Specific BGP Extended Community*

RFC 6286, *Autonomous-System-Wide Unique BGP Identifier for BGP-4*

RFC 6368, *Internal BGP as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 6793, *BGP Support for Four-Octet Autonomous System (AS) Number Space*

RFC 6810, *The Resource Public Key Infrastructure (RPKI) to Router Protocol*

RFC 6811, *Prefix Origin Validation*

RFC 6996, *Autonomous System (AS) Reservation for Private Use*

RFC 7311, *The Accumulated IGP Metric Attribute for BGP*

RFC 7606, *Revised Error Handling for BGP UPDATE Messages*

RFC 7607, *Codification of AS 0 Processing*

RFC 7854, *BGP Monitoring Protocol (BMP)*

RFC 7911, *Advertisement of Multiple Paths in BGP*

RFC 7999, *BLACKHOLE Community*

RFC 8092, *BGP Large Communities Attribute*

RFC 8097, *BGP Prefix Origin Validation State Extended Community*

RFC 8212, *Default External BGP (EBGP) Route Propagation Behavior without Policies*

RFC 8277, *Using BGP to Bind MPLS Labels to Address Prefixes*

RFC 9294, *Application-Specific Link Attributes Advertisement Using the Border Gateway Protocol - Link State (BGP LS)*

RFC 9494, *Long-Lived Graceful Restart for BGP*

RFC 9552, *Distribution of Link-State and Traffic Engineering Information Using BGP*

15.3 Bridging and management

IEEE 802.1AB, *Station and Media Access Control Connectivity Discovery*

IEEE 802.1ad, *Provider Bridges*

IEEE 802.1AX, *Link Aggregation*
IEEE 802.1D, *MAC Bridges*
IEEE 802.1p, *Traffic Class Expediting*
IEEE 802.1Q, *Virtual LANs*
IEEE 802.1s, *Multiple Spanning Trees*
IEEE 802.1w, *Rapid Reconfiguration of Spanning Tree*

15.4 Certificate management

RFC 4210, *Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*
RFC 4211, *Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)*
RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*
RFC 6712, *Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)*
RFC 7030, *Enrollment over Secure Transport*
RFC 7468, *Textual Encodings of PKIX, PKCS, and CMS Structures*

15.5 Ethernet

IEEE 802.3x, *Ethernet Flow Control*

15.6 Ethernet VPN (EVPN)

draft-ietf-bess-evpn-ip-mac-proxy-adv-00, *Proxy MAC-IP Advertisement in EVPNs*
draft-ietf-bess-evpn-l3mh-proto-00, *EVPN Multi-Homing support for L3 services*
RFC 7432, *BGP MPLS-Based Ethernet VPN*
RFC 8214, *Virtual Private Wire Service Support in Ethernet VPN*
RFC 8365, *A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)*
RFC 8560, *Seamless Integration of Ethernet VPN (EVPN) with Virtual Private LAN Service (VPLS) and Their Provider Backbone Bridge (PBB) Equivalents*
RFC 8584, *Framework for Ethernet VPN Designated Forwarder Election Extensibility*
RFC 9047, *Propagation of ARP/ND Flags in an Ethernet Virtual Private Network (EVPN)*
RFC 9135, *Integrated Routing and Bridging in Ethernet VPN (EVPN)*
RFC 9136, *IP Prefix Advertisement in Ethernet VPN (EVPN)*
RFC 9161, *Operational Aspects of Proxy ARP/ND in Ethernet Virtual Private Networks*
RFC 9251, *Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Proxies for Ethernet VPN (EVPN)*

RFC 9785, *Preference-Based EVPN Designated Forwarder (DF) Election*
RFC 9786, *EVPN Port-Active Redundancy Mode*
RFC 10039, *Interconnecting EVPN and IPVPN Domains*

15.7 gRPC Remote Procedure Calls (gRPC)

cert.proto version 0.1.0, *gNOI Certificate Management Service*
file.proto version 0.1.0, *gNOI File Service*
gnmi.proto version 0.8.0, *gNMI Service Specification*
gnmi_ext.proto, *gNMI Commit Confirmed Extension*
gnmi_ext.proto, *gNMI Config Subscription Extension*
gnmi_ext.proto, *gNMI Depth Extension*
system.proto version 1.0.0, *gNOI System Service*
tunnel.proto version 0.2, *gRPC Tunnel Service*
PROTOCOL-HTTP2, *gRPC over HTTP2*

15.8 Intermediate System to Intermediate System (IS-IS)

draft-ietf-isis-mi-02, *IS-IS Multi-Instance*
draft-ietf-lsr-igp-ureach-prefix-announce-01, *IGP Unreachable Prefix Announcement* – without U-Flag and UP-Flag
draft-kaplan-isis-ext-eth-02, *Extended Ethernet Frame Size Support*
ISO/IEC 10589:2002 Second Edition, *Intermediate system to Intermediate system intra-domain routing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)*
RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*
RFC 2973, *IS-IS Mesh Groups*
RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*
RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*
RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*
RFC 5120, *M-ISIS: Multi Topology (MT) Routing in IS-IS*
RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*
RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*
RFC 5302, *Domain-wide Prefix Distribution with Two-Level IS-IS*
RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*

RFC 5304, *IS-IS Cryptographic Authentication*
RFC 5305, *IS-IS Extensions for Traffic Engineering TE*
RFC 5306, *Restart Signaling for IS-IS – helper mode*
RFC 5308, *Routing IPv6 with IS-IS*
RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*
RFC 5310, *IS-IS Generic Cryptographic Authentication*
RFC 5311, *Simplified Extension of Link State PDU (LSP) Space for IS-IS*
RFC 6213, *IS-IS BFD-Enabled TLV*
RFC 6232, *Purge Originator Identification TLV for IS-IS*
RFC 6233, *IS-IS Registry Extension for Purges*
RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*
RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability – sections 2.1 and 2.3*
RFC 7981, *IS-IS Extensions for Advertising Router Information*
RFC 7987, *IS-IS Minimum Remaining Lifetime*
RFC 8202, *IS-IS Multi-Instance – single topology*
RFC 8570, *IS-IS Traffic Engineering (TE) Metric Extensions – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE*
RFC 8919, *IS-IS Application-Specific Link Attributes*
RFC 9681, *IS-IS Fast Flooding*
RFC 9885, *Multi-Part TLVs in IS-IS*

15.9 Internet Protocol (IP) general

RFC 768, *User Datagram Protocol*
RFC 793, *Transmission Control Protocol*
RFC 854, *Telnet Protocol Specifications*
RFC 1350, *The TFTP Protocol (revision 2)*
RFC 2784, *Generic Routing Encapsulation (GRE)*
RFC 3164, *The BSD syslog Protocol*
RFC 4250, *The Secure Shell (SSH) Protocol Assigned Numbers*
RFC 4251, *The Secure Shell (SSH) Protocol Architecture*
RFC 4252, *The Secure Shell (SSH) Authentication Protocol – publickey, password*
RFC 4253, *The Secure Shell (SSH) Transport Layer Protocol*
RFC 4254, *The Secure Shell (SSH) Connection Protocol*
RFC 4511, *Lightweight Directory Access Protocol (LDAP): The Protocol*
RFC 4513, *Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms – TLS*

RFC 4632, *Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan*

RFC 5082, *The Generalized TTL Security Mechanism (GTSM)*

RFC 5246, *The Transport Layer Security (TLS) Protocol Version 1.2* – TLS client, RSA public key

RFC 5289, *TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)*

RFC 5424, *The Syslog Protocol* – excluding structured data

RFC 5425, *Transport Layer Security (TLS) Transport Mapping for Syslog* – RFC 3164 with TLS

RFC 5656, *Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer* – ECDSA

RFC 5925, *The TCP Authentication Option*

RFC 5926, *Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)*

RFC 6398, *IP Router Alert Considerations and Usage* – MLD

RFC 6528, *Defending against Sequence Number Attacks*

RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*

RFC 8907, *The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol*

RFC 9887, *Terminal Access Controller Access-Control System Plus (TACACS+) over TLS 1.3*

15.10 Internet Protocol (IP) multicast

RFC 1112, *Host Extensions for IP Multicasting*

RFC 2236, *Internet Group Management Protocol, Version 2*

RFC 2365, *Administratively Scoped IP Multicast*

RFC 2375, *IPv6 Multicast Address Assignments*

RFC 2710, *Multicast Listener Discovery (MLD) for IPv6*

RFC 3376, *Internet Group Management Protocol, Version 3*

RFC 3446, *Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)*

RFC 3590, *Source Address Selection for the Multicast Listener Discovery (MLD) Protocol*

RFC 3618, *Multicast Source Discovery Protocol (MSDP)*

RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*

RFC 4541, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*

RFC 4604, *Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast*

RFC 4610, *Anycast-RP Using Protocol Independent Multicast (PIM)*

RFC 4611, *Multicast Source Discovery Protocol (MSDP) Deployment Scenarios*

RFC 5059, *Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)*

RFC 5186, *Internet Group Management Protocol Version 3 (IGMPv3) / Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction*

RFC 5384, *The Protocol Independent Multicast (PIM) Join Attribute Format*
RFC 7761, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*
RFC 8487, *Mtrace Version 2: Traceroute Facility for IP Multicast*

15.11 Internet Protocol (IP) version 4

RFC 791, *Internet Protocol*
RFC 792, *Internet Control Message Protocol*
RFC 826, *An Ethernet Address Resolution Protocol*
RFC 1034, *Domain Names - Concepts and Facilities*
RFC 1035, *Domain Names - Implementation and Specification*
RFC 1191, *Path MTU Discovery* – router specification
RFC 1519, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*
RFC 1812, *Requirements for IPv4 Routers*
RFC 1918, *Address Allocation for Private Internets*
RFC 2131, *Dynamic Host Configuration Protocol*; Relay only
RFC 2132, *DHCP Options and BOOTP Vendor Extensions* – DHCP
RFC 2401, *Security Architecture for Internet Protocol*
RFC 3021, *Using 31-Bit Prefixes on IPv4 Point-to-Point Links*
RFC 3046, *DHCP Relay Agent Information Option (Option 82)*
RFC 3768, *Virtual Router Redundancy Protocol (VRRP)*
RFC 4884, *Extended ICMP to Support Multi-Part Messages* – ICMPv4 and ICMPv6 Time Exceeded

15.12 Internet Protocol (IP) version 6

RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*
RFC 2529, *Transmission of IPv6 over IPv4 Domains without Explicit Tunnels*
RFC 3122, *Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification*
RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*
RFC 3587, *IPv6 Global Unicast Address Format*
RFC 3596, *DNS Extensions to Support IP version 6*
RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*
RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*
RFC 3971, *SEcure Neighbor Discovery (SEND)*
RFC 4007, *IPv6 Scoped Address Architecture*
RFC 4191, *Default Router Preferences and More-Specific Routes* – Default Router Preference

RFC 4193, *Unique Local IPv6 Unicast Addresses*
RFC 4291, *Internet Protocol Version 6 (IPv6) Addressing Architecture*
RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*
RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*
RFC 5095, *Deprecation of Type 0 Routing Headers in IPv6*
RFC 5722, *Handling of Overlapping IPv6 Fragments*
RFC 5798, *Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6 – IPv6*
RFC 5952, *A Recommendation for IPv6 Address Text Representation*
RFC 6164, *Using 127-Bit IPv6 Prefixes on Inter-Router Links*
RFC 8021, *Generation of IPv6 Atomic Fragments Considered Harmful*
RFC 8200, *Internet Protocol, Version 6 (IPv6) Specification*

15.13 Internet Protocol Security (IPsec)

draft-ietf-ipsec-isakmp-mode-cfg-05, *The ISAKMP Configuration Method*
draft-ietf-ipsec-isakmp-xauth-06, *Extended Authentication within ISAKMP/Oakley (XAUTH)*
RFC 2401, *Security Architecture for the Internet Protocol*
RFC 2403, *The Use of HMAC-MD5-96 within ESP and AH*
RFC 2404, *The Use of HMAC-SHA-1-96 within ESP and AH*
RFC 2405, *The ESP DES-CBC Cipher Algorithm With Explicit IV*
RFC 2406, *IP Encapsulating Security Payload (ESP)*
RFC 2407, *IPsec Domain of Interpretation for ISAKMP (IPsec DoI)*
RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*
RFC 2409, *The Internet Key Exchange (IKE)*
RFC 2410, *The NULL Encryption Algorithm and Its Use With IPsec*
RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*
RFC 3526, *More Modular Exponential (MODP) Diffie-Hellman group for Internet Key Exchange (IKE)*
RFC 3566, *The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec*
RFC 3602, *The AES-CBC Cipher Algorithm and Its Use with IPsec*
RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*
RFC 3947, *Negotiation of NAT-Traversal in the IKE*
RFC 3948, *UDP Encapsulation of IPsec ESP Packets*
RFC 4106, *The Use of Galois/Counter Mode (GCM) in IPsec ESP*
RFC 4109, *Algorithms for Internet Key Exchange version 1 (IKEv1)*
RFC 4301, *Security Architecture for the Internet Protocol*
RFC 4303, *IP Encapsulating Security Payload*

RFC 4307, *Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)*
RFC 4308, *Cryptographic Suites for IPsec*
RFC 4434, *The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol (IKE)*
RFC 4543, *The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH*
RFC 4754, *IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA)*
RFC 4835, *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)*
RFC 4868, *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec*
RFC 4945, *The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2 and PKIX*
RFC 5019, *The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments*
RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the IKEv2 Protocol*
RFC 5903, *ECP Groups for IKE and IKEv2*
RFC 5996, *Internet Key Exchange Protocol Version 2 (IKEv2)*
RFC 5998, *An Extension for EAP-Only Authentication in IKEv2*
RFC 6379, *Suite B Cryptographic Suites for IPsec*
RFC 6380, *Suite B Profile for Internet Protocol Security (IPsec)*
RFC 6960, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*
RFC 7296, *Internet Key Exchange Protocol Version 2 (IKEv2)*
RFC 7321, *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)*
RFC 7383, *Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation*
RFC 7427, *Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)*
RFC 8784, *Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security*

15.14 Label Distribution Protocol (LDP)

draft-pdutta-mpls-ldp-adj-capability-00, *LDP Adjacency Capabilities*
draft-pdutta-mpls-ldp-v2-00, *LDP Version 2*
RFC 3037, *LDP Applicability*
RFC 3478, *Graceful Restart Mechanism for Label Distribution Protocol – helper mode*
RFC 5036, *LDP Specification*
RFC 5283, *LDP Extension for Inter-Area Label Switched Paths (LSPs)*
RFC 5443, *LDP IGP Synchronization*
RFC 5561, *LDP Capabilities*
RFC 5919, *Signaling LDP Label Advertisement Completion*

15.15 Multiprotocol Label Switching (MPLS)

RFC 3031, *Multiprotocol Label Switching Architecture*

RFC 3032, *MPLS Label Stack Encoding*

RFC 3270, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services – E-LSP*

RFC 3443, *Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks*

RFC 5332, *MPLS Multicast Encapsulations*

RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*

RFC 6424, *Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels*

RFC 7308, *Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)*

RFC 7746, *Label Switched Path (LSP) Self-Ping*

15.16 Network Address Translation (NAT)

RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*

RFC 5382, *NAT Behavioral Requirements for TCP*

RFC 5508, *NAT Behavioral Requirements for ICMP*

15.17 Network Configuration Protocol (NETCONF)

RFC 5277, *NETCONF Event Notifications*

RFC 6020, *YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)*

RFC 6022, *YANG Module for NETCONF Monitoring*

RFC 6241, *Network Configuration Protocol (NETCONF)*

RFC 6242, *Using the NETCONF Protocol over Secure Shell (SSH)*

RFC 6243, *With-defaults Capability for NETCONF*

RFC 8071, *NETCONF Call Home and RESTCONF Call Home – NETCONF*

RFC 8342, *Network Management Datastore Architecture (NMDA) – Startup, Candidate, Running and Intended datastores*

RFC 8525, *YANG Library*

RFC 8526, *NETCONF Extensions to Support the Network Management Datastore Architecture – <get-data> operation*

15.18 Open Shortest Path First (OSPF)

RFC 1765, *OSPF Database Overflow*

RFC 2328, *OSPF Version 2*
RFC 3101, *The OSPF Not-So-Stubby Area (NSSA) Option*
RFC 3509, *Alternative Implementations of OSPF Area Border Routers*
RFC 3623, *Graceful OSPF Restart Graceful OSPF Restart – helper mode*
RFC 3630, *Traffic Engineering (TE) Extensions to OSPF Version 2*
RFC 4811, *OSPF Out-of-Band Link State Database (LSDB) Resynchronization – OSPFv2*
RFC 4812, *OSPF Restart Signaling – OSPFv2*
RFC 4222, *Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance*
RFC 4552, *Authentication/Confidentiality for OSPFv3*
RFC 4576, *Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS IP Virtual Private Networks (VPNs)*
RFC 4577, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*
RFC 5185, *OSPF Multi-Area Adjacency*
RFC 5243, *OSPF Database Exchange Summary List Optimization*
RFC 5250, *The OSPF Opaque LSA Option*
RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*
RFC 5340, *OSPF for IPv6*
RFC 5642, *Dynamic Hostname Exchange Mechanism for OSPF*
RFC 5709, *OSPFv2 HMAC-SHA Cryptographic Authentication*
RFC 5838, *Support of Address Families in OSPFv3*
RFC 6549, *OSPFv2 Multi-Instance Extensions*
RFC 6987, *OSPF Stub Router Advertisement*
RFC 7471, *OSPF Traffic Engineering (TE) Metric Extensions – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE*
RFC 7684, *OSPFv2 Prefix/Link Attribute Advertisement*
RFC 7770, *Extensions to OSPF for Advertising Optional Router Capabilities*
RFC 8362, *OSPFv3 Link State Advertisement (LSA) Extensibility*
RFC 8920, *OSPF Application-Specific Link Attributes*

15.19 Path Computation Element Protocol (PCEP)

draft-alvarez-pce-path-profiles-04, *PCE Path Profiles*
draft-ietf-pce-binding-label-sid-15, *Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks. – MPLS binding SIDs*
RFC 5440, *Path Computation Element (PCE) Communication Protocol (PCEP)*
RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*

RFC 8253, *PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)*

RFC 8281, *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model*

RFC 8408, *Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages*

RFC 8664, *Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing*

15.20 Pseudowire (PW)

draft-ietf-l2vpn-vpws-iw-oam-04, *OAM Procedures for VPWS Interworking*

MFA Forum 12.0.0, *Multiservice Interworking - Ethernet over MPLS*

MFA Forum 13.0.0, *Fault Management for Multiservice Interworking v1.0*

MFA Forum 16.0.0, *Multiservice Interworking - IP over MPLS*

RFC 3916, *Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)*

RFC 3985, *Pseudo Wire Emulation Edge-to-Edge (PWE3)*

RFC 4385, *Pseudo Wire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN*

RFC 4446, *IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)*

RFC 4447, *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*

RFC 4448, *Encapsulation Methods for Transport of Ethernet over MPLS Networks*

RFC 5085, *Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires*

RFC 5659, *An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge*

RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*

RFC 6073, *Segmented Pseudowire*

RFC 6310, *Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping*

RFC 6391, *Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network*

RFC 6575, *Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs*

RFC 6718, *Pseudowire Redundancy*

RFC 6829, *Label Switched Path (LSP) Ping for Pseudowire Forwarding Equivalence Classes (FECs) Advertised over IPv6*

RFC 6870, *Pseudowire Preferential Forwarding Status bit*

RFC 7023, *MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking*

RFC 7267, *Dynamic Placement of Multi-Segment Pseudowires*

RFC 7392, *Explicit Path Routing for Dynamic Multi-Segment Pseudowires – ER-TLV and ER-HOP IPv4 Prefix*

RFC 8395, *Extensions to BGP-Signaled Pseudowires to Support Flow-Aware Transport Labels*

15.21 Quality of Service (QoS)

RFC 2430, *A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)*
RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*
RFC 2597, *Assured Forwarding PHB Group*
RFC 3140, *Per Hop Behavior Identification Codes*
RFC 3246, *An Expedited Forwarding PHB (Per-Hop Behavior)*

15.22 Remote Authentication Dial In User Service (RADIUS)

draft-oscca-cfrg-sm3-02, *The SM3 Cryptographic Hash Function*
RFC 2865, *Remote Authentication Dial In User Service (RADIUS)*
RFC 2866, *RADIUS Accounting*
RFC 3162, *RADIUS and IPv6*
RFC 6613, *RADIUS over TCP – with TLS*
RFC 6614, *Transport Layer Security (TLS) Encryption for RADIUS*
RFC 6929, *Remote Authentication Dial-In User Service (RADIUS) Protocol Extensions*

15.23 Resource Reservation Protocol - Traffic Engineering (RSVP-TE)

RFC 2702, *Requirements for Traffic Engineering over MPLS*
RFC 2747, *RSVP Cryptographic Authentication*
RFC 2961, *RSVP Refresh Overhead Reduction Extensions*
RFC 3097, *RSVP Cryptographic Authentication -- Updated Message Type Value*
RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*
RFC 4124, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*
RFC 4125, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*
RFC 4127, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*
RFC 4561, *Definition of a Record Route Object (RRO) Node-Id Sub-Object*
RFC 5817, *Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks*

15.24 Routing Information Protocol (RIP)

RFC 1058, *Routing Information Protocol*
RFC 2080, *RIPng for IPv6*
RFC 2082, *RIP-2 MD5 Authentication*

RFC 2453, *RIP Version 2*

15.25 Segment Routing (SR)

RFC 8287, *Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes*

RFC 8426, *Recommendations for RSVP-TE and Segment Routing (SR) Label Switched Path (LSP) Coexistence*

RFC 8476, *Signaling Maximum SID Depth (MSD) Using OSPF – node MSD*

RFC 8491, *Signaling Maximum SID Depth (MSD) Using IS-IS – node MSD*

RFC 8660, *Segment Routing with the MPLS Data Plane*

RFC 8661, *Segment Routing MPLS Interworking with LDP*

RFC 8665, *OSPF Extensions for Segment Routing*

RFC 8667, *IS-IS Extensions for Segment Routing*

RFC 8669, *Segment Routing Prefix Segment Identifier Extensions for BGP*

RFC 9256, *Segment Routing Policy Architecture*

RFC 9350, *IGP Flexible Algorithm*

15.26 Simple Network Management Protocol (SNMP)

draft-blumenthal-aes-usm-04, *The AES Cipher Algorithm in the SNMP's User-based Security Model – CFB128-AES-192 and CFB128-AES-256*

draft-ietf-isis-wg-mib-06, *Management Information Base for Intermediate System to Intermediate System (IS-IS)*

draft-ietf-mpls-ldp-mib-07, *Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)*

draft-ietf-mpls-lsr-mib-06, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base Using SMIv2*

draft-ietf-mpls-te-mib-04, *Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base*

draft-ietf-ospf-mib-update-08, *OSPF Version 2 Management Information Base*

draft-ietf-vrrp-unified-mib-06, *Definitions of Managed Objects for the VRRP over IPv4 and IPv6 – IPv6*

ESO-CONSORTIUM-MIB revision 200406230000Z, *esoConsortiumMIB*

IANA-ADDRESS-FAMILY-NUMBERS-MIB revision 200203140000Z, *ianaAddressFamilyNumbers*

IANAifType-MIB revision 200505270000Z, *ianaifType*

IANA-RTPROTO-MIB revision 200009260000Z, *ianaRtProtoMIB*

IEEE8021-CFM-MIB revision 200706100000Z, *ieee8021CfmMib*

IEEE8021-PAE-MIB revision 200101160000Z, *ieee8021paeMIB*

IEEE8023-LAG-MIB revision 200006270000Z, *lagMIB*
LLDP-MIB revision 200505060000Z, *lldpMIB*
RFC 1157, *A Simple Network Management Protocol (SNMP)*
RFC 1212, *Concise MIB Definitions*
RFC 1215, *A Convention for Defining Traps for use with the SNMP*
RFC 1724, *RIP Version 2 MIB Extension*
RFC 1901, *Introduction to Community-based SNMPv2*
RFC 2021, *Remote Network Monitoring Management Information Base Version 2 using SMIv2*
RFC 2206, *RSVP Management Information Base using SMIv2*
RFC 2578, *Structure of Management Information Version 2 (SMIv2)*
RFC 2579, *Textual Conventions for SMIv2*
RFC 2580, *Conformance Statements for SMIv2*
RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*
RFC 2819, *Remote Network Monitoring Management Information Base*
RFC 2856, *Textual Conventions for Additional High Capacity Data Types*
RFC 2863, *The Interfaces Group MIB*
RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*
RFC 2933, *Internet Group Management Protocol MIB*
RFC 3014, *Notification Log MIB*
RFC 3273, *Remote Network Monitoring Management Information Base for High Capacity Networks*
RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*
RFC 3430, *Simple Network Management Protocol (SNMP) over Transmission Control Protocol (TCP) Transport Mapping*
RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*
RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*
RFC 3413, *Simple Network Management Protocol (SNMP) Applications*
RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*
RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*
RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP) – SNMP over UDP over IPv4*
RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*
RFC 3419, *Textual Conventions for Transport Addresses*
RFC 3434, *Remote Monitoring MIB Extensions for High Capacity Alarms*

RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

RFC 3593, *Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals*

RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*

RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*

RFC 3877, *Alarm Management Information Base (MIB)*

RFC 4001, *Textual Conventions for Internet Network Addresses*

RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*

RFC 4113, *Management Information Base for the User Datagram Protocol (UDP)*

RFC 4273, *Definitions of Managed Objects for BGP-4*

RFC 4292, *IP Forwarding Table MIB*

RFC 4293, *Management Information Base for the Internet Protocol (IP)*

RFC 4878, *Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) Functions on Ethernet-Like Interfaces*

RFC 7420, *Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module*

RFC 7630, *HMAC-SHA-2 Authentication Protocols in the User-based Security Model (USM) for SNMPv3*

15.27 Storage sanitization

IEEE 2883-2022, *IEEE Standard for Sanitizing Storage* – clear sanitization method for CF, MMC, SSD, SD, USB

15.28 Timing

RFC 3339, *Date and Time on the Internet: Timestamps*

RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*

RFC 8573, *Message Authentication Code for the Network Time Protocol*

15.29 Two-Way Active Measurement Protocol (TWAMP)

RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP) – server, unauthenticated mode*

RFC 5938, *Individual Session Control Feature for the Two-Way Active Measurement Protocol (TWAMP)*

RFC 6038, *Two-Way Active Measurement Protocol (TWAMP) Reflect Octets and Symmetrical Size Features*

RFC 8545, *Well-Known Port Assignments for the One-Way Active Measurement Protocol (OWAMP) and the Two-Way Active Measurement Protocol (TWAMP) – TWAMP*

RFC 8762, *Simple Two-Way Active Measurement Protocol* – unauthenticated

RFC 8972, *Simple Two-Way Active Measurement Protocol Optional Extensions* – unauthenticated

RFC 9503, *Simple Two-Way Active Measurement Protocol (STAMP) Extensions for Segment Routing Networks* – excluding Sections 3, 4.1.2 and 4.1.3

15.30 Virtual Private LAN Service (VPLS)

RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*

RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*

RFC 5501, *Requirements for Multicast Support in Virtual Private LAN Services*

RFC 6074, *Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)*

RFC 7117, *Multicast in Virtual Private LAN Service (VPLS)*

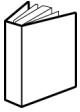
15.31 Yet Another Next Generation (YANG)

RFC 6991, *Common YANG Data Types*

RFC 7950, *The YANG 1.1 Data Modeling Language*

RFC 7951, *JSON Encoding of Data Modeled with YANG*

Customer document and product support



Customer documentation

[Customer documentation welcome page](#)



Technical support

[Product support portal](#)



Documentation feedback

[Customer documentation feedback](#)