



7450 Ethernet Service Switch
7750 Service Router
7950 Extensible Routing System
Virtualized Service Router
Release 26.10.R1

Basic System Configuration Guide

3HE 22299 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

1	Getting started.....	13
1.1	About this guide.....	13
1.2	Conventions.....	13
1.2.1	Precautionary and information messages.....	14
1.2.2	Options or substeps in procedures and sequential workflows.....	14
2	File management.....	16
2.1	SR OS file system.....	16
2.1.1	Storage devices.....	16
2.1.2	URLs.....	16
2.1.3	HTTP digest authentication.....	19
2.1.4	Wildcards and special characters.....	19
2.2	Text editor.....	20
2.2.1	Summary of text editor commands.....	20
2.2.2	Using text editor commands.....	20
2.3	File management tasks in the classic CLI.....	27
2.3.1	Managing storage devices.....	28
2.3.2	Displaying directory and file information.....	29
2.3.3	Modifying file attributes.....	30
2.3.4	Creating directories.....	30
2.3.5	Copying files.....	31
2.3.6	Moving files.....	32
2.3.7	Deleting files and removing directories.....	35
2.3.8	Unzipping files.....	35
2.3.9	Displaying file checksums.....	36
2.4	File management tasks in the MD-CLI.....	36
2.4.1	Managing storage devices.....	37
2.4.2	Displaying directory and file information.....	38
2.4.3	Modifying file attributes.....	39
2.4.4	Creating and navigating directories.....	40
2.4.5	Copying files.....	41
2.4.6	Moving files.....	43
2.4.7	Deleting files and removing directories.....	46

2.4.8	Unzipping files.....	47
2.4.9	Displaying file checksums.....	47
3	System initialization and boot options.....	49
3.1	Boot process.....	49
3.2	Boot Loader.....	51
3.3	Boot Options File.....	51
3.3.1	BOF manual mode.....	51
3.4	Software and configuration.....	52
3.4.1	Management interface configuration modes.....	53
3.5	Initial installation and software update.....	54
3.6	Storage card content.....	54
3.6.1	7750 SR-1x-48D, SR-1x-92S, SR-1-24D, SR-1-48D, SR-1-46S, SR-1-92S, SR-1se, and SR-2se.....	55
3.6.2	7450 ESS, 7950 XRS, 7750 SR 7/7s/12/12e/14s, SR-a, SR-e, and SR-2s.....	56
3.6.3	7750 SR-1 and SR-1s.....	58
3.6.4	ISA and ESA applications.....	59
3.7	Persistent indexes in classic and mixed configuration mode.....	59
3.8	BOF and configuration file encryption.....	59
3.9	System profiles.....	60
3.10	FIPS-140 mode.....	62
3.10.1	FIPS operating environment.....	62
3.10.2	Displaying the OK and the CMV.....	62
3.10.3	Booting up the system in FIPS mode.....	63
3.10.4	Supported algorithms in FIPS mode.....	64
3.10.5	FIPS image upgrade validation using a DS.....	64
3.10.5.1	Digital signature.....	64
3.10.5.2	DS verification.....	65
3.10.5.3	DS support for files.....	66
3.10.5.4	DS file directories.....	66
3.10.5.5	DS verification when configuring the fips-140 flag in the BOF.....	67
3.10.5.6	DS verification for ISA-AA.....	67
3.10.5.7	Checking a file DS.....	67
3.10.5.8	Upgrade procedure to a FIPS image.....	67
3.11	Lawful Intercept.....	68
3.12	Configuring the Boot Options File with CLI.....	68

3.12.1	Basic BOF configuration.....	68
3.12.2	Common configuration tasks.....	70
3.12.2.1	Searching for the BOF.....	70
3.12.2.2	Accessing the CLI.....	70
3.12.2.3	Console connection.....	70
3.12.2.4	Configuring BOF encryption.....	71
3.12.2.5	Configuring the BOF interactive menu password.....	71
3.12.2.6	Configuring configuration file encryption.....	71
3.12.3	Autoconfigure.....	72
3.12.3.1	Autoconfigure restrictions.....	72
3.12.3.2	DHCP discovery of MAC addresses.....	72
3.12.3.3	IPv6 DUID.....	73
3.12.3.4	IPv6 DHCP RAs.....	73
3.12.4	System management tasks.....	74
3.12.4.1	System management commands in the classic CLI.....	74
3.12.4.2	System management commands in the MD-CLI.....	77
3.13	Anti-theft.....	81
3.13.1	Node behavior when the anti-theft password is set.....	82
3.13.2	BOF and configuration file behavior.....	83
3.13.3	Management interface interaction in anti-theft mode.....	84
4	Debug configuration.....	86
4.1	Debug configuration in the classic CLI.....	86
4.1.1	Logging debug events in the classic CLI.....	86
4.2	Debug configuration in the MD-CLI.....	87
4.2.1	Logging debug events in the MD-CLI.....	88
4.3	Debug configuration in mixed and model-driven mode.....	89
5	Secure Boot.....	91
5.1	Secure Boot chain.....	91
5.2	Activate Secure Boot.....	92
5.2.1	Satellite.....	93
5.3	Operational commands and logs.....	94
5.3.1	Secure Boot state.....	94
5.3.2	Software update.....	95
5.3.3	Update Secure Boot variables.....	95

6	System management.....	96
6.1	System management commands.....	96
6.1.1	System information.....	96
6.1.1.1	Name.....	96
6.1.1.2	Contact.....	96
6.1.1.3	Location.....	96
6.1.1.4	Coordinates.....	97
6.1.1.5	Naming objects.....	97
6.1.1.6	Common language location identifier.....	97
6.1.1.7	DNS security extensions.....	97
6.1.2	System time.....	97
6.1.2.1	Time zones.....	97
6.1.2.2	NTP.....	100
6.1.2.3	Synchronization.....	101
6.1.2.4	GNSS.....	101
6.1.2.5	CRON.....	103
6.2	High Availability.....	103
6.2.1	HA features.....	104
6.2.1.1	Redundancy.....	104
6.2.1.2	Nonstop forwarding.....	106
6.2.1.3	Nonstop Routing.....	106
6.2.1.4	CPM switchover.....	106
6.2.1.5	Synchronization.....	107
6.3	Network synchronization.....	107
6.3.1	Central synchronization subsystem.....	109
6.3.2	7950 XRS-40 extension chassis central clocks.....	112
6.3.3	Synchronization Status Messages.....	112
6.3.3.1	DS1 signals.....	113
6.3.3.2	E1 signals.....	113
6.3.3.3	DS3/E3.....	113
6.3.4	Synchronous Ethernet.....	113
6.3.4.1	Timing reference selection based on quality level.....	114
6.3.4.2	Extended QL TLV for ESMC messaging.....	116
6.3.5	Clock source quality level definitions.....	116
6.3.6	Advanced G.781 features.....	119

6.3.7	IEEE 1588v2 PTP.....	119
6.3.7.1	PTP clock synchronization.....	126
6.3.7.2	Performance considerations.....	129
6.3.7.3	PTP capabilities.....	131
6.3.7.4	PTP ordinary timeReceiver clock for frequency.....	131
6.3.7.5	PTP ordinary timeTransmitter clock for frequency.....	132
6.3.7.6	PTP boundary clock for frequency and time.....	133
6.3.7.7	PTP timeTransmitter clock for frequency and time distribution.....	134
6.3.7.8	ITU-T G.8275.2 profile and APTS.....	135
6.3.7.9	PTP clock redundancy.....	137
6.3.7.10	PTP message encapsulations.....	137
6.3.7.11	PTP time for system time and OAM time.....	137
6.3.7.12	PTP within routing instances.....	138
6.3.7.13	PTSF conditions for G.8275.1.....	138
6.3.7.14	Profile interworking.....	139
6.3.7.15	Annex J performance monitoring statistics.....	140
6.3.7.16	PTP path trace.....	141
6.3.8	Synchronization with Ethernet satellites.....	142
6.4	QinQ network interface support.....	143
6.5	LLDP.....	144
6.6	IP hashing as an LSR.....	147
6.7	Satellites.....	148
6.7.1	Ethernet satellites.....	149
6.7.2	Software repositories for satellites.....	150
6.7.3	Upgrading satellite software.....	151
6.7.4	Provisioning a 7250 IXR satellite.....	152
6.7.5	Synchronization features with satellites.....	153
6.7.6	Satellite configuration.....	153
6.7.6.1	Satellite client and uplink port ID formats.....	154
6.7.6.2	Port naming conventions.....	154
6.7.6.3	Local forwarding.....	156
6.7.6.4	Port template.....	157
6.7.6.5	7210 SAS 10GE client ports.....	157
6.7.6.6	7210 SAS 100GE client ports.....	157
6.7.6.7	10GE uplinks on the 64x10GE+4xQSFP28 satellite.....	157
6.7.6.8	Satellite uplink resiliency.....	159

6.7.6.9	Dynamic uplink resiliency.....	160
6.7.6.10	Ethernet LAGs with satellite member links.....	161
6.7.6.11	CRC monitoring.....	162
6.8	Auto-provisioning.....	164
6.8.1	Auto-provisioning limits.....	165
6.8.2	Auto-provisioning process.....	166
6.8.3	Auto-provisioning DHCP rules.....	166
6.8.4	Auto-provisioning failure.....	167
6.9	Administrative tasks.....	167
6.9.1	Saving configurations.....	167
6.9.2	Specifying post-boot configuration files.....	167
6.9.3	Network timing.....	168
6.9.4	Power supplies.....	168
6.9.5	CPM file synchronization.....	169
6.9.6	Forcing a switchover.....	169
6.10	System router instances.....	169
6.11	System configuration process overview.....	170
6.12	Saving configurations manually.....	171
6.13	Common configuration tasks.....	171
6.13.1	System information.....	172
6.13.1.1	System name.....	172
6.13.1.2	Contact.....	172
6.13.1.3	Location.....	172
6.13.1.4	CLLI code.....	172
6.13.1.5	GPS coordinates.....	173
6.13.2	System time elements.....	173
6.13.2.1	Zone.....	173
6.13.2.2	Summer (daylight saving) time.....	173
6.13.2.3	NTP.....	173
6.13.2.4	SNTP.....	180
6.13.2.5	CRON.....	181
6.13.3	ANCP enhancements.....	182
6.13.4	Configuration backups.....	182
6.14	System timing.....	184
6.14.1	Entering edit mode.....	184
6.14.2	Configuring timing references.....	184

6.14.3	Using the revert command.....	185
6.14.4	Committing and discarding changes.....	185
6.14.5	Forcing a specific reference.....	186
6.14.6	Configuring system timing to use a GNSS RF port.....	186
6.15	Configuring synchronization and redundancy.....	187
6.15.1	Configuring subscriber management persistence.....	187
6.15.2	Configuring a CLI script file for synchronization.....	187
6.15.3	Configuring synchronization options.....	187
6.15.4	Displaying synchronization options.....	188
6.16	Configuring multichassis redundancy for LAG.....	188
6.17	Post-boot configuration extension files.....	189
6.17.1	Show command output and console messages.....	189
6.18	Configuring system monitoring thresholds.....	190
6.18.1	Creating events.....	191
6.19	Configuring LLDP.....	192
6.20	Configuring low-power mode features.....	193
6.20.1	7750 SR-s low-power switch fabric mode.....	193
6.20.2	7750 SR low-power optic mode.....	198
6.20.3	Low-power card mode.....	199
7	CPM redundancy.....	201
7.1	File synchronization.....	201
7.2	Active and standby designations.....	203
7.3	When the active CPM goes offline.....	204
7.4	OOB management Ethernet port redundancy.....	204
7.5	DHCP persistence.....	205
7.5.1	DDP access optimization for DHCP leases.....	206
8	Zero touch provisioning.....	208
8.1	ZTP overview.....	208
8.1.1	Network requirements.....	208
8.1.2	Network support.....	209
8.2	ZTP process overview.....	211
8.2.1	Auto-boot process.....	211
8.2.2	Auto-provisioning process.....	211
8.3	DHCP support for ZTP.....	211

8.3.1	DHCP server offer options.....	212
8.3.1.1	Nokia-specific TLV.....	212
8.3.2	Supported DHCP client options for ZTP.....	212
8.3.3	Supported DHCP server options for ZTP.....	213
8.3.4	DHCP discovery and solicitation.....	214
8.3.4.1	DHCP discovery (IPv4 and IPv6).....	214
8.3.4.2	DHCP solicitation (IPv6).....	215
8.3.5	IPv4 and IPv6 DHCP support.....	215
8.3.5.1	IPv4 route installation details.....	215
8.3.5.2	IPv6 DHCP/RA details.....	215
8.3.5.3	ZTP and DHCP timeouts.....	216
8.4	ZTP procedure details.....	216
8.4.1	Node bootup.....	216
8.4.1.1	Reinitiating ZTP during normal node bootup.....	216
8.4.2	BOF.....	216
8.4.2.1	SD card and compact flash support.....	217
8.4.3	Auto-boot process details.....	217
8.4.3.1	Options and option modification.....	217
8.4.3.2	CLI access.....	218
8.4.3.3	Interrupting auto-boot.....	218
8.4.4	Auto-provisioning process.....	218
8.4.4.1	VLAN discovery.....	219
8.4.4.2	Auto-provisioning procedure.....	219
8.4.4.3	Out-of-band management versus in-band management.....	220
8.4.5	Provisioning files.....	221
8.4.5.1	Provisioning file download.....	222
8.4.5.2	Provisioning file resolution using DNS.....	222
8.4.5.3	File download and redundancy.....	222
8.4.5.4	Configuring the ZTP timeout in the provisioning file.....	222
8.4.5.5	Downloading the image file.....	223
8.4.5.6	Example provisioning file.....	224
8.4.5.7	Proxy support.....	225
8.4.6	Day 0 configuration.....	226
8.4.6.1	Day 0 configuration for multi-slot routers.....	226
8.4.6.2	Day 0 symbols.....	227
8.4.6.3	Sample day 0 configuration template.....	230

8.4.7	Logs and events.....	232
8.4.7.1	Syslog.....	232
8.5	Legacy SZTP (non RFC 8572-compliant).....	232
8.5.1	Staging the secure environment.....	234
8.5.2	Bootstrapping methods.....	234
8.5.3	Installation site process.....	235
8.5.3.1	Initial conveyed information file.....	236
8.5.3.2	Onboarding information.....	238
8.5.3.3	Conveyed information.....	242
8.6	RFC 8572-compliant SZTP.....	243
8.6.1	TLS details.....	245
8.6.2	Bootstrap methods.....	246
8.6.3	Secure ZTP BOF flag.....	247
8.6.4	Conveyed, redirect, and onboarding information support.....	247
8.6.5	Image download rules.....	248
8.6.6	Configuration script.....	249
8.6.6.1	Provisioning file format.....	249
8.6.6.2	Provisioning script format.....	250
8.6.6.3	Examples of the pre-configuration-script.....	251
8.6.7	Installation site.....	253
8.6.8	Non-TPM 2.0 platforms.....	254
8.6.9	Image download directory.....	254
8.6.10	Logs.....	254
8.6.11	NTP and OV certificate date and time validation.....	254
9	Standards and protocol support.....	255
9.1	Access Node Control Protocol (ANCP).....	255
9.2	Bidirectional Forwarding Detection (BFD).....	255
9.3	Border Gateway Protocol (BGP).....	255
9.4	Bridging and management.....	257
9.5	Broadband Network Gateway (BNG) Control and User Plane Separation (CUPS).....	258
9.6	Certificate management.....	258
9.7	Circuit emulation.....	259
9.8	Ethernet.....	259
9.9	Ethernet VPN (EVPN).....	259
9.10	gRPC Remote Procedure Calls (gRPC).....	260

9.11	Intermediate System to Intermediate System (IS-IS).....	260
9.12	Internet Protocol (IP) Fast Reroute (FRR).....	262
9.13	Internet Protocol (IP) general.....	262
9.14	Internet Protocol (IP) multicast.....	263
9.15	Internet Protocol (IP) version 4.....	265
9.16	Internet Protocol (IP) version 6.....	265
9.17	Internet Protocol Security (IPsec).....	266
9.18	Label Distribution Protocol (LDP).....	268
9.19	Layer Two Tunneling Protocol (L2TP) Network Server (LNS).....	268
9.20	Multiprotocol Label Switching (MPLS).....	269
9.21	Multiprotocol Label Switching - Transport Profile (MPLS-TP).....	269
9.22	Network Address Translation (NAT).....	270
9.23	Network Configuration Protocol (NETCONF).....	270
9.24	Open Shortest Path First (OSPF).....	271
9.25	Path Computation Element Protocol (PCEP).....	271
9.26	Point-to-Point Protocol (PPP).....	272
9.27	Policy management and credit control.....	272
9.28	Pseudowire (PW).....	273
9.29	Quality of Service (QoS).....	273
9.30	Remote Authentication Dial In User Service (RADIUS).....	274
9.31	Resource Reservation Protocol - Traffic Engineering (RSVP-TE).....	274
9.32	Routing Information Protocol (RIP).....	275
9.33	Segment Routing (SR).....	275
9.34	Simple Network Management Protocol (SNMP).....	276
9.35	Storage sanitization.....	278
9.36	Timing.....	279
9.37	Two-Way Active Measurement Protocol (TWAMP).....	279
9.38	Virtual Private LAN Service (VPLS).....	280
9.39	Voice and video.....	280
9.40	Yet Another Next Generation (YANG).....	280
9.41	Yet Another Next Generation (YANG) OpenConfig Models.....	280

1 Getting started

1.1 About this guide

This guide describes system concepts and provides configuration explanations and examples to configure SR OS boot option file (BOF), file system, and system management functions.



Note: See the *7450 ESS, 7750 SR, 7950 XRS, and VSR Basic System Configuration Advanced Configuration Guide for Classic CLI* for information about advanced configurations.

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.

The topics and commands described in this guide apply to the:

- 7450 ESS
- 7750 SR
- 7950 XRS
- Virtualized Service Router (VSR)

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 MD-CLI and the classic CLI.

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

- *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Clear, Monitor, Show, and Tools CLI Command Reference Guide* (for both the MD-CLI and the classic CLI)
- *7450 ESS, 7750 SR, 7950 XRS, and VSR 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. For information about features supported in each load of the Release 26.x.Rx software or for a list of unsupported features by platform and chassis, see the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000x TQZZA.

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 File management

This chapter provides information about file system management.

2.1 SR OS file system

The SR OS file system is used to store files used and generated by the system; for example, image files, configuration files, logging files, and accounting files.

The file commands allow you to copy, create, move, and delete files and directories, navigate to a different directory, and display file or directory contents and the image version.

Although some of the storage devices on routers are not actually compact flash devices (for example, cf1: on the 7950 XRS is an internal SSD), all storage devices are referred to as compact flash.

2.1.1 Storage devices

In the 7750 SR and 7450 ESS, each control processor can have up to three storage devices numbered 1 through 3. In the 7950 XRS, each CCM has an SSD and up to two compact flash devices. The names for these devices are:

- cf1:
- cf2:
- cf3:

The preceding device names are relative names: they refer to devices that are local to the control processor of the current console session. The colon (":") at the end of the name indicates this is a device.

The three compact flash devices on the 7450 ESS and 7750 SR OS are removable and have an administrative state.

The cf2: and cf3: compact flash devices on the 7950 XRS routers are removable and have an administrative state. cf1: is an internal SSD.

Devices vary by platform as compact flash, SD card, USB, or embedded SSD.



Note: To prevent corrupting open files in the file system, only remove a compact flash that is administratively disabled. SR OS gracefully closes any open files on the device, allowing the storage device to be safely removed.

2.1.2 URLs

The arguments for the SR OS file commands are modeled after standard universal resource locator (URL). A URL refers to a file (a *file-url*) or a directory (a *directory-url*).

SR OS supports operations on both the local file system and on remote files. To categorize the applicability of commands to local and remote file operations, URLs are divided into five types of URLs: **local**, **ftp**,

sftp, **tftp**, **http**, **https**, and **scp**. The syntax for each of the URL types are listed in [Table 1: URL types and syntax](#).

Table 1: URL types and syntax

URL type	Syntax	Notes
<i>local-url</i>	<code>[cflash-id:\]path</code>	<i>cflash-id</i> is the compact flash device name. Values: cf1: , cf2: , cf3:
<i>ftp-url</i>	<code>ftp://[username[:password]@]host/path</code>	An absolute FTP path from the root of the remote file system. <i>username</i> is the FTP username <i>password</i> is the FTP user password <i>host</i> is the FTP server <i>path</i> is the path to the directory or file
	<code>ftp://[username[:password]@]host/.path</code>	A relative FTP path from the user's home directory. Note the period and slash (".") in this syntax compared to the absolute path.
<i>sftp-url</i>	<code>sftp://[username[:password]@]host/path</code>	An absolute SFTP path from the root of the remote file system. <i>username</i> is the SFTP username <i>password</i> is the SFTP user password <i>host</i> is the SFTP server <i>path</i> is the path to the directory or file
	<code>sftp://[username[:password]@]host/.path</code>	A relative SFTP path from the user's home directory. Note the period and slash (".") in this syntax compared to the absolute path.
<i>tftp-url</i>	<code>tftp://host[/path]/filename</code>	TFTP is only supported for operations on a <i>file-url</i> .
<i>http-url</i>	<code>http://[username[:password]@]host[:port]/path</code>	<i>host</i> is the HTTP server <i>port</i> defaults to 80
<i>https-url</i>	<code>https://[username[:password]@]host[:port]/path</code>	<i>host</i> is the HTTPS server <i>port</i> defaults to 443
<i>scp-url</i>	<code>scp://username @host:path</code>	<i>username</i> is the SSH username <i>host</i> is the SSH server <i>path</i> is the path to the directory or file

If the host portion of a URL is an IPv6 address, enclose the URL in quotes and the address in square brackets, as shown in the following examples.

Example

- "ftp://username:password@[2001:db8:3333:4444:5555:6666:7777:8888]/testfile.txt"
- "scp://username@[2001:db8:3333:4444:5555:6666:7777:8888]/testfile.txt"

The system accepts forward slash (/) or backslash (\) characters to delimit directory and/or filenames in URLs. Similarly, the SR OS SCP client application can use either slash or backslash characters, but not all SCP clients treat backslash characters as equivalent to slash characters. In particular, UNIX systems often interpret the backslash character as an escape character. This can cause problems when using an external SCP client application to send files to the SCP server. If the external system treats the backslash like an escape character, the backslash delimiter gets stripped by the parser and is not transmitted to the SCP server.

For example, a destination directory specified as "cf1:\dir1\file1" is transmitted to the SCP server as "cf1:dir1file1" where the backslash escape characters are stripped by the SCP client system before transmission. On systems where the client treats the backslash like an escape character, a double backslash (\\) or the forward slash (/) can typically be used to properly delimit directories and the filename.

When a special character is used in a password, it can cause issues when that password is encoded as part of a URL. To prevent this issue, percent encoding can be used. Percent encoding is a mechanism to encode 8-bit characters that have specific meaning in the context of URLs. The encoding consists of substitution of a percent character (%) followed by the hexadecimal representation of the ASCII value of the replaced character.

Some file manipulation commands such as copying, removing, or moving files, may request access to an HTTP or HTTPS server. If an HTTP or HTTPS server redirects the system to a different URL (from an "HTTP 301" error or similar response), the system prompts the user "y/n" to either repeat the operation with the new URL or terminate it. These file commands can be configured to force the HTTP redirects without prompting or they can be configured to refuse HTTP redirects. If a file command is redirected more than eight times, or if it queries an HTTPS URL and gets redirected to an HTTP URL, the command automatically terminates as a security measure.

Use the following command to refuse HTTP redirects:

- **MD-CLI**

```
copy source-url destination-url direct-http
```

- **classic CLI**

```
copy source-url dest-url no-redirect
```

Use the following command to force the HTTP redirects without prompting:

- **MD-CLI**

```
copy source-url destination-url force
```

- **classic CLI**

```
copy source-url dest-url force
```

When connecting to an HTTPS server, the system verifies the server's TLS certificate. For the certificate to pass verification, the system must have a CA profile already configured for the server's Certificate Authority (CA), which specifies up-to-date certificate and CRL files. HTTPS file commands do not use the Online

Certificate Status Protocol (OCSP). If the certificate was issued by an intermediate CA, the system must have a CA profile for every CA tracing back to the root CA. If the server's certificate fails verification for any reason, the file command terminates. See the *7450 ESS, 7750 SR, and VSR Multiservice Integrated Service Adapter and Extended Services Appliance Guide* for more information about CA profiles.

Use the following CLI command to configure the CA profile:

```
configure system security pki ca-profile
```

An HTTPS **file** command may also include a **client-tls-profile** configuration parameter, referring to a client TLS profile that provides the cipher list, client certificate, and trust anchor the system uses when communicating with the HTTPS server. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR System Management Guide* for more information about client TLS profiles.

A **file** command that connects to an HTTP or HTTPS server outside the local network may need to use an HTTP proxy. The user may specify a proxy server (which must be an HTTP URL).

2.1.3 HTTP digest authentication

For HTTP or HTTPS **file** commands only, the HTTP digest authentication scheme can be used with the HTTP authentication mechanism as described in RFC 7616 and RFC 2617. The following hash algorithms are supported:

- MD5
- SHA-256
- SHA-512/256

2.1.4 Wildcards and special characters

SR OS supports the standard wildcard characters. The asterisk (*) can represent zero or more characters in a string of characters, and the question mark (?) can represent any one character and must be enclosed in quotation marks (" ").

Example: MD-CLI

```
[file "cf3:\"]  
A:admin@node-2# copy bof.* testdir  
11 file(s) copied.
```

```
[file "cf3:\"]  
A:admin@node-2#
```

Example: classic CLI

```
A:node-2>file cf3:\ # copy bof.* testdir  
Copying file cf3:\bof.cfg-1 ... OK  
Copying file cf3:\bof.cfg-2 ... OK  
Copying file cf3:\bof.cfg-3 ... OK  
Copying file cf3:\bof.cfg-4 ... OK  
Copying file cf3:\bof.cfg-5 ... OK  
Copying file cf3:\bof.cfg-6 ... OK  
Copying file cf3:\bof.cfg-7 ... OK  
Copying file cf3:\bof.cfg-8 ... OK  
Copying file cf3:\bof.cfg-9 ... OK
```

```
Copying file cf3:\bof.cfg.1 ... OK
Copying file cf3:\bof.cfg ... OK
11 file(s) copied.
A:node-2>file cf3:\ #
```

2.2 Text editor

The text editor allows the user to edit an ASCII text file. When the user modifies the configuration using a configuration mode, the system validates whether the user is allowed to perform configuration changes. When the text editor is used to edit a configuration file, these validation checks do not occur. For this reason, administrator privileges are required to access the text editor, and the user profile must be modified to allow this access.

Access permission for the directory where the file resides must be granted before a user can open, read, or write a file. If the user does not have permission to access the directory, the operation is denied.

Use the following command to start the text editor:

- **MD-CLI**

```
file edit url
```

- **classic CLI**

```
file vi local-url
```

2.2.1 Summary of text editor commands

The text editor operates in the following modes:

- **command mode**

In this mode, every character entered is a command that causes an action to be taken on the text file. For example, the character “O” typed in command mode causes the text editor to enter insert mode.

- **insert mode**

In this mode, every character typed is added to the text in the file. Pressing Esc turns off insert mode.

2.2.2 Using text editor commands

Use the commands described in the tables in this section to do the following:

- start and end text editor sessions
- move within a file
- enter new text
- modify, move, and delete existing text
- read from and write to other files

The following table describes the commands to cut, paste, and delete text.

Table 2: Cutting and pasting or deleting text

Text editor command	Description
"	Specify a buffer to be used with any of the commands using buffers. Follow the quotation mark (") character with a letter or a number that corresponds to a buffer.
d	Delete text. The "dd" command deletes the current line. A count specifies the number of lines to delete. Deleted text is placed in the buffer specified with the " command. If no buffer is specified, the general buffer is used.
D	Delete to the end of the line from the current cursor position.
p	Paste the specified buffer after the current cursor position or line. If no buffer is specified using the " command, the general buffer is used.
P	Paste the specified buffer before the current cursor position or line. If no buffer is specified using the " command, the general buffer is used.
x	Delete the character under the cursor. A count specifies the number of characters to delete. The characters after the cursor are deleted.
X	Delete the character before the cursor.
y	Yank text and place the result into a buffer. The "yy" command yanks the current line. Entering a number yanks the specified number of lines. The buffer can be specified with the " command. If no buffer is specified, the general buffer is used.
Y	Yank the current line into the specified buffer. If no buffer is specified, the general buffer is used.

The following table describes the commands to insert new text.

Table 3: Inserting new text

Text editor command	Description
A	Append text at the end of the current line.
I	Insert text from the beginning of a line.
O	Enter insert mode in a new line above the current cursor position.

Text editor command	Description
a	Enter insert mode and append text after the current cursor position. A preceding count inserts all the text that was inserted the specified number of times.
i	Enter insert mode and insert typed text before the current cursor position. A preceding count inserts all the text that was inserted the specified number of times.
o	Enter insert mode in a new line below the current cursor position.

The following table describes the commands used to move the cursor within the file.

Table 4: Moving the cursor within the file

Text editor command	Description
^B	Scroll backward one page. A count scrolls that many pages.
^D	Scroll forward half a window. A count scrolls that many lines.
^F	Scroll forward one page. A count scrolls that many pages.
^H	Move the cursor one space to the left. A count moves that many spaces.
^J	Move the cursor down one line in the same column. A count moves that many lines down.
^M	Move to the first character on the next line.
^N	Move the cursor down one line in the same column. A count moves that many lines down.
^P	Move the cursor up one line in the same column. A count moves that many lines up.
^U	Scroll backward half a window. A count scrolls that many lines.
\$	Move the cursor to the end of the current line. A count moves to the end of the following lines.
%	Move the cursor to the matching parenthesis or brace.
^	Move the cursor to the first non-whitespace character.
(	Move the cursor to the beginning of a sentence.

Text editor command	Description
)	Move the cursor to the beginning of the next sentence.
{	Move the cursor to the preceding paragraph.
}	Move the cursor to the next paragraph.
	Move the cursor to the column specified by the count.
+	Move the cursor to the first non-whitespace character in the next line.
-	Move the cursor to the first non-whitespace character in the previous line.
—	Move the cursor to the first non-whitespace character in the current line.
0	Move the cursor to the first column of the current line.
B	Move the cursor back one word, skipping over punctuation.
E	Move the cursor forward to the end of a word, skipping over punctuation.
G	Go to the line number specified as the count. If no count is specified, go to the end of the file.
H	Move the cursor to the first non-whitespace character at the top of the screen.
L	Move the cursor to the first non-whitespace character at the bottom of the screen.
M	Move the cursor to the first non-whitespace character in the middle of the screen.
W	Move forward to the beginning of a word, skipping over punctuation.
b	Move the cursor back one word. If the cursor is in the middle of a word, move the cursor to the first character of that word.
e	Move the cursor forward one word. If the cursor is in the middle of a word, move the cursor to the last character of that word.
h	Move the cursor one character position to the left.
j	Move the cursor down one line.
k	Move the cursor up one line.

Text editor command	Description
l	Move the cursor one character position to the right.
w	Move the cursor forward one word. If the cursor is in the middle of a word, move the cursor to the first character of the next word.

The following table describes the commands to move the cursor around the screen.

Table 5: Moving the cursor around the screen

Text editor command	Description
^E	Scroll forward one line. A count scrolls that many lines.
^Y	Scroll backward one line. A count scrolls that many lines.
z	<p>Redraw the screen with the following options:</p> <ul style="list-style-type: none"> • z<return> puts the current line on the top of the screen. • z. puts the current line on the center of the screen. • z- puts the current line on the bottom of the screen. <p>If you specify a count before the z command, it changes the current line to the line specified. For example, 16z. puts line 16 on the center of the screen.</p>

The following table describes the commands to replace text.

Table 6: Replacing text

Text editor command	Description
C	Change to the end of the line from the current cursor position.
R	Replace characters on the screen with a set of characters entered, ending with Esc.
S	Change an entire line.
c	The cc command changes the current line. A count changes that many lines.
r	Replace one character under the cursor. Specify a count to replace a number of characters.

Text editor command	Description
s	Substitute one character under the cursor, and enter insert mode. Specify a count to substitute a number of characters. A dollar sign (\$) is placed at the last character to be substituted.

The following table describes the commands to search for text or characters in the file.

Table 7: Searching for text or characters

Text editor command	Description
,	Repeat the last f , F , t or T command in the reverse direction.
/	Search the file forward for the string specified after the forward slash (/).
;	Repeat the last f , F , t or T command.
?	Search the file backward for the string specified after the ?.
F	Search the current line backward for the character specified after the F command. If found, move the cursor to the position.
N	Repeat the last search done by forward slash (/) or question mark (?) in the backward direction.
T	Search the current line backward for the character specified after the T command, and move to the column after the character, if it is found.
f	Search the current line for the character specified after the f command. If found, move the cursor to the position.
n	Repeat the last search done by forward slash (/) or question mark (?) in the forward direction
t	Search the current line for the character specified after the t command, and move to the column before the character, if it is found.

The following table describes the commands to manipulate character and line formatting.

Table 8: Manipulating character and line formatting

Text editor command	Description
~	Switch the case of the character under the cursor.

Text editor command	Description
<	Shift the lines up to the left by one shiftwidth. The << command shifts the current line to the left and can be specified with a count.
>	Shift the lines up to the right by one shiftwidth. The >> command shifts the current line to the right and can be specified with a count.
J	Join the current line with the next one. A count joins that many lines.

The following table describes miscellaneous commands.

Table 9: Miscellaneous commands

Text editor command	Description
^G	Show the current file name and the status.
^L	Clear and redraw the screen.
^R	Redraw the screen removing false lines.
^[	Cancel a partially formed command (Esc).
^^	Go back to the last file edited.
&	Repeat the previous :s command.
.	Repeat the last command that modified the file.
:	Begin typing a line editing command. The command is executed when the user presses Enter.
@	Type the command stored in the specified buffer.
U	Restore the current line to the previous state before the cursor entered the line.
m	Mark the current position with the character specified after the m command.
u	Undo the last change to the file. Redo the change by typing u again.
ZZ	Exit the editor, saving if any changes were made.

From the text editor, use the : command to enter a line editing command. To modify more than one line using specific commands (such as :s and :w), the range must be specified before the command. For example, to substitute lines 3 through 15, the command is :3,15s/from/this/g.

The following table describes the commonly used line editing commands.

Table 10: Line editing commands

Text editor command	Description
:ab string strings	Abbreviation. If a word is typed in the text editor corresponding to string1, the editor automatically inserts the corresponding words. For example, the abbreviation ":ab vprn Virtual Private Routed Network" inserts the words "Virtual Private Routed Network" whenever the word "vprn" is typed.
:map keys new_seq	Map a key or a sequence of keys to another key or sequence of keys.
:q	Quit the text editor. If changes have been made, the editor issues a warning message.
:q!	Quit the text editor without saving changes.
:s/pattern/to_pattern/options	Substitute the specified pattern with the string in the to_pattern . Without options, only the first occurrence of the pattern is substituted. If a g is specified, all occurrences are substituted.
:set [all]	Sets some customizing options. The :set all command displays all options.
:una string	Removes the abbreviation previously defined by :ab .
:unm keys	Removes the mapping defined by :map .
:vi filename	Starts editing a new file. If changes have not been saved, the editor displays a warning message.
:w	Write contents of the current file.
:w filename	Write the contents of the buffer to the file name specified.
:w >> filename	Append the contents of the buffer to the file name.
:wq	Write the contents of the buffer and quit.

2.3 File management tasks in the classic CLI

The following sections describe file management tasks that can be performed in the classic CLI.

For more information about the supported classic CLI commands, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*.

When a file system operation is performed that can potentially remove or overwrite a file system entry, a prompt appears to confirm the action. The **force** keyword performs the operation without displaying the confirmation prompt.

All the commands can operate on the local file system with a default of cf3:. The following table indicates the commands that also support remote file operations.

Table 11: File command local and remote file system support in the classic CLI

Command	local-url	ftp-url	sftp-url	tftp-url	http-url	https-url
attrib	✓					
cd	✓	✓	✓			
copy	✓	✓	✓	source only	✓	✓
copy (recursive)	✓	✓	✓			
checksum	✓	✓	✓	✓		
delete	✓	✓	✓		✓	✓
dir	✓	✓	✓			
md	✓	✓	✓			
move	✓	✓	✓		✓	✓
move (recursive)	✓	✓	✓			
rd	✓	✓	✓			
scp	source only					
type	✓	✓	✓	✓	✓	✓
unzip	✓	✓	✓	source only		
version	✓	✓	✓	✓		
vi	✓					

2.3.1 Managing storage devices

Use the **repair** command to check a storage device for errors and repair any errors found. The device does not need to be administratively disabled.

Example: Repair command syntax

```
A:node-2# file repair cf3:
Checking drive cf3: on slot A for errors...
Drive cf31: on slot A is OK.
```

Use the **format** command to format a storage device with a new file system without erasing the data. The device must be administratively disabled first.

Example: Format command syntax

```
A:node-2# file shutdown cf1:
A:node-2# file format cf1:
Formatting Drive cf1: on Slot A ...
Drive cf1: on Slot A is formatted
A:node-2# file no shutdown cf1:
```

Use the **secure-erase** command to secure erase and format a storage device with a new file system using the clear sanitization method to sanitize media as defined in IEEE 2883-2022, *IEEE Standard for Sanitizing Storage*. This action may take several minutes to complete depending on the storage device size, because the data is overwritten in one or more passes. The device must be administratively disabled first.

Example: Secure erase command syntax

```
A:node-2# file shutdown cf1:
A:node-2# file secure-erase cf1:
Are you sure you want to secure erase and format drive cf1: (y/n)?y
Secure erasing drive cf2: on slot A - 100% complete ...
Formatting drive cf1: on slot A ...
Drive cf1: is formatted
Drive cf1: is secure erased
```



Note: New system data may be written to a storage devices after it is administratively enabled. Do not administratively enable a storage device unless it is to be reused.

2.3.2 Displaying directory and file information

Use the **dir** command to display a list of files on a file system. The **type** command displays the contents of a file. The **version** command displays the version of an SR OS .tim image file.

Example: Display directory and file information

```
A:node-2# file dir
Volume in drive cf1 on slot A has no label.
Directory of cf1:\
01/01/1980 12:00a          7597 test.cfg
01/01/1980 12:00a           957 b.
08/19/2001 02:14p       230110 BOOTROM.SYS
01/01/1980 12:00a         133 NVRAM.DAT
04/03/2003 05:32a         1709 103.ndx
01/28/2003 05:06a         1341 103.cftg.ndx
01/28/2003 05:06a       20754 103.cftg
04/05/2003 02:20a    <DIR>          test
                15 File(s)          338240 bytes.
                3 Dir(s)           1097728 bytes free.
```

```
A:node-2# file type example.cfg
File: example.cfg
```

```
-----
exit all
config
#-----
# Chassis Commands
#-----
card 2 card-type faste-tx-32
exit
#-----
```

```
# Interface Commands
#-----
# Physical port configuration
interface faste 2/1
    shutdown
    mode network
exit
interface faste 2/2
    shutdown
exit
interface faste 2/3
    shutdown
exit
Press any key to continue (Q to quit)

A:node-2# file version boot.tim
TiMOS-L-24.10.R1
Thu Oct 31 23:49:01 UTC 2024 by builder in /builds/2410B/R1/panos/main/sr
```

2.3.3 Modifying file attributes

The system administrator can change the attribute of a local file or files in a directory. Enter the **attrib** command with no options to display the contents of the directory and the file attributes.



Note: A file with an "R" preceding the filename indicates that the file is read-only.

Example: File configuration output

```
A:node-2>file cf3:\ # attrib
cf3:\bootlog.txt
cf3:\bof.cfg
cf3:\boot.ldr
cf3:\bootlog_prev.txt
cf3:\B0F.SAV
A:node-2>file cf3:\ # attrib +r B0F.SAV
A:node-2>file cf3:\ # attrib
cf3:\bootlog.txt
cf3:\bof.cfg
cf3:\boot.ldr
cf3:\bootlog_prev.txt
R   cf3:\B0F.SAV
```

2.3.4 Creating directories

Use the **md** command to create a new directory in the local file system, one level at a time.

Enter the **cd** command to navigate to different directories.

Example: Creating three levels of directories

```
A:node-2>file cfl:\ # md test1file cfl:\ # cd test1
A:node-2>file cfl:\test1\ # md test2
A:node-2>file cfl:\test1\ # cd test2
A:node-2>file cfl:\test1\test2\ # md test3
A:node-2>file cfl:\test1\test2\ # cd test3
```

```
A:node-2>file cf1:\test1\test2\test3 #
```

2.3.5 Copying files

Use the **copy** command to copy files to or from a flash device or an FTP/TFTP server.

The **copy** command supports wildcards.

The **scp** command copies files between hosts on a network. It uses SSH for data transfer, uses the same authentication, and provides the same security as SSH.

The source file for the **scp** command must be local. The file must reside on the router. The destination file does not need to be local, but it must be in the `user@host:file-name` format.

Example: Image file and network host copy

```
A:node-2>file cf1:\ # copy 104.cfg cf1:\test1\test2\test3\test.cfg
A:node-2>file cf1:\ # scp file1 admin@192.168.x.x:cf1:\file1
A:node-2>file cf1:\ # scp file2 user2@192.168.x.x:/user2/file2
A:node-2>file cf1:\ # scp cf2:/file3 admin@192.168.x.x:cf1:\file3
```

Use the **recursive** keyword to recursively copy files and directories. If files or directories already exist, the user is prompted to overwrite them. When the **force** keyword is enabled, a positive response to the overwrite prompts is assumed. The user is not prompted for confirmation and the existing files or directories are overwritten.

Example: Recursive directory copy

```
A:node-2# file dir
Volume in drive cf3 on slot A is SR0S VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 06:11p 101 NVRAM.DAT
09/19/2021 07:22p SUPPORT/
09/19/2021 06:11p SYSLINUX/
09/19/2021 06:11p TIMOS/
09/23/2021 09:03a 27459 bootlog.txt
09/20/2021 09:56a 27326 bootlog_prev.txt
09/23/2021 01:21p 319 nvsys.info
09/21/2021 07:19p recursive3/
09/23/2021 08:22a ssh/
7 File(s) 55402 bytes.
6 Dir(s) 612319232 bytes free.

A:node-2# file dir recursive3

Volume in drive cf3 on slot A is SR0S VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive3

09/21/2021 07:18p ./
09/21/2021 07:18p ../
09/21/2021 07:19p 7 file1.txt
```

```

09/21/2021 07:19p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612319232 bytes free.

A:node-2# file copy recursive3 recursive1 recursive
Copying directory cf3:\recursive3
Copying directory cf3:\recursive3\recursive2
Copying file cf3:\recursive3\recursive2\file2.txt ... OK
Copying file cf3:\recursive3\recursive2\file3.txt ... OK
Copying file cf3:\recursive3\file1.txt ... OK
2 dir(s) and 3 file(s) copied.
A:sros# file dir

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 06:11p 101 NVRAM.DAT
09/19/2021 07:22p SUPPORT/
09/19/2021 06:11p SYSLINUX/
09/19/2021 06:11p TIMOS/
09/23/2021 09:03a 27459 bootlog.txt
09/20/2021 09:56a 27326 bootlog_prev.txt
09/23/2021 01:21p 319 nvsys.info
09/23/2021 02:42p recursive1/
09/21/2021 07:19p recursive3/
09/23/2021 08:22a ssh/
7 File(s) 55402 bytes.
7 Dir(s) 612298752 bytes free.

A:node-2# file dir recursive1

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive1

09/23/2021 02:42p ./
09/23/2021 02:42p ../
09/23/2021 02:42p 7 file1.txt
09/23/2021 02:42p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612298752 bytes free.

```

2.3.6 Moving files

Use the **move** command to move a file or directory from one location to another.

The **move** command supports wildcards, recursively moves files and directories, and overwrites existing content without prompting for confirmation.

Example: Moving files and directories

```

A:node-2>file cf3:\ # dir

Volume in drive cf3 on slot A is TIMOS_VM_CF.

```

```

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/28/2021  11:42a                933 B0F.CFG
09/28/2021  11:42a                166 NVRAM.DAT
09/29/2021  03:51p            12831 bootlog.txt
09/29/2021  03:46p            12828 bootlog_prev.txt
04/16/2014  10:15a                 0 config.cfg
09/29/2021  03:51p             318 nvsys.info
04/16/2014  10:15a      <DIR>      syslinux/
09/29/2021  03:55p            12831 test.txt
09/29/2021  03:54p      <DIR>      test_dir1/
04/16/2014  10:15a      <DIR>      timos/
                                7 File(s)          39907 bytes.
                                3 Dir(s)          14452736 bytes free.

A:node-2>file cf3:\ # move test.txt /test_dir1/test_dir2/test_dir3
Moving file cf3:\test.txt ... OK
cf3:\test.txt

A:node-2>file cf3:\ # dir

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/28/2021  11:42a                933 B0F.CFG
09/28/2021  11:42a                166 NVRAM.DAT
09/29/2021  03:51p            12831 bootlog.txt
09/29/2021  03:46p            12828 bootlog_prev.txt
04/16/2014  10:15a                 0 config.cfg
09/29/2021  03:51p             318 nvsys.info
04/16/2014  10:15a      <DIR>      syslinux/
09/29/2021  03:54p      <DIR>      test_dir1/
04/16/2014  10:15a      <DIR>      timos/
                                6 File(s)          27076 bytes.
                                3 Dir(s)          14452736 bytes free.

A:node-2>file cf3:\ # dir test_dir1/test_dir2/test_dir3

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test_dir1\test_dir2\test_dir3

09/29/2021  03:54p      <DIR>      ./
09/29/2021  03:54p      <DIR>      ../
09/29/2021  03:55p            12831 test.txt
                                1 File(s)          12831 bytes.
                                2 Dir(s)          14452736 bytes free.

```

Example: Recursive directory move

```

A:node-2>file cf3:\ # dir

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

```

```
Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 06:11p 101 NVRAM.DAT
09/19/2021 07:22p  SUPPORT/
09/19/2021 06:11p  SYSLINUX/
09/19/2021 06:11p  TIMOS/
09/23/2021 09:03a 27459 bootlog.txt
09/20/2021 09:56a 27326 bootlog_prev.txt
09/23/2021 01:21p 319 nvsys.info
09/23/2021 02:42p recursive1/
09/23/2021 08:22a  ssh/
7 File(s) 55402 bytes.
7 Dir(s) 612311040 bytes free.

A:node-2>file cf3:\ # dir recursive1

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive1

09/23/2021 02:42p ./
09/23/2021 02:42p ../
09/23/2021 02:42p 7 file1.txt
09/23/2021 02:42p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612311040 bytes free.

A:node-2>file cf3:\ # move recursive1 recursive4
Moving file cf3:\recursive1 ... OK
cf3:\recursive1
A:sros>file cf3:\ # dir

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 06:11p 101 NVRAM.DAT
09/19/2021 07:22p  SUPPORT/
09/19/2021 06:11p  SYSLINUX/
09/19/2021 06:11p  TIMOS/
09/23/2021 09:03a 27459 bootlog.txt
09/20/2021 09:56a 27326 bootlog_prev.txt
09/23/2021 01:21p 319 nvsys.info
09/23/2021 02:42p recursive4/
09/23/2021 08:22a  ssh/
7 File(s) 55402 bytes.
7 Dir(s) 612311040 bytes free.

A:node-2>file cf3:\ # dir recursive4

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive4
```

```
09/23/2021 02:42p ./
09/23/2021 02:42p ../
09/23/2021 02:42p 7 file1.txt
09/23/2021 02:42p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612311040 bytes free.
```

2.3.7 Deleting files and removing directories

Use the **delete** and **rd** commands to delete files and remove directories. Directories can be removed even if they contain files or subdirectories. To remove a directory that contains files and subdirectories, use the **rd /s** command. When files or directories are deleted, they cannot be recovered. The **force** option deletes the file or directory without prompting the user to confirm.

Example: Delete files and remove directories

```
A:node2>file cfl:\test1\ # delete test.cfg
A:node-2>file cfl:\test1\ # delete abc.cfg
A:node-2>file cfl:\test1\test2\ # cd test3
A:node-2>file cfl:\test1\test2\test3\ # cd ..
A:node-2>file cfl:\test1\test2\ # rd test3
A:node-2>file cfl:\test1\test2\ # cd ..
A:node-2>file cfl:\test1\ # rd test2
A:node-2>file cfl:\test1\ # cd ..
A:node-2>file cfl:\ # rd test1
A:node-2>file cfl:\ #
```

2.3.8 Unzipping files

Use the **unzip** command to expand the contents of a ZIP file to the local file system. Any file that is zipped using the store, deflate, or zip64 compression methods can be unzipped. An example is the SR OS software image available from the Nokia customer support portal.

The source ZIP file can be located locally on the installed solid-state storage device, or remotely on an FTP or TFTP server.

The **create-destination** keyword ensures that any non-existent directory structure that is explicitly entered as the destination file URL is created as part of the unzip operation.



Note:

- The destination for the unzipped files and directories must be a locally installed solid-state storage device in the active CPM.
- ZIP filenames, or the filenames of any contained files, must not include special characters.

Example: Unzip command

```
A:node-2# file unzip demo.zip cf3:/mynewfolder/mynewsfolder create-destination force
Verifying cf3:\demo.zip .. ... OK
Unzipping cf3:\demo.zip to cf3:\mynewfolder\mynewsfolder\ .. .Processing demodir/
Processing demodir/myfile1.txt
Processing demodir/myfile2.txt
Processing demodir/demosubdir/
Processing demodir/demosubdir/myfile3.txt
```

```
Writing...OK
```

2.3.9 Displaying file checksums

Use the **checksum** command to display file checksums.

Use the **version** command to check the version of an SR OS .tim image file.

Example: Checking the version of an SR OS .tim image file

```
A:node-2# file version cpm.tim
TiMOS-C-20.10.R1
Wed Nov 4 09:18:17 PST 2020 by builder in /builds/c/2010B/R1/panos/main/sros

A:node-2># file version cpm.tim check
TiMOS-C-20.10.R1
Wed Nov 4 09:18:17 PST 2020 by builder in /builds/c/2010B/R1/panos/main/sros
Checking file ... OK
```

Use the **checksum** command to display checksums.

Example: Output of the checksum operation

The following example shows the output of the checksum operation to compute and display a checksum based on the MD5 and SHA256 algorithms for the cpm.tim file on cf3.

```
A:node-2# file checksum md5 cpm.tim
Checking file cf3:cpm.tim ...
c65699dc05e6e35a2172eaac80485aa2

A:node-2# file checksum sha256 cpm.tim
Checking file cf3:cpm.tim ...
a1a813a696be04906f9faf1df9db0f90a990ff51cb3383099ade21241203bc1c
```

2.4 File management tasks in the MD-CLI

The following sections describe file management tasks that can be performed in the MD-CLI.

For more information about the supported MD-CLI commands, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI Command Reference Guide*.

When a file system operation is performed that can potentially remove or overwrite a file system entry, a prompt appears to confirm the action. The **force** keyword performs the operation without displaying the confirmation prompt.

All the commands can operate on the local file system with a default of cf3:. The following table lists which commands also support remote file operations.

Table 12: File command local and remote file system support in the MD-CLI

Command	local-url	ftp-url	sftp-url	tftp-url	http-url	https-url
change-directory	✓	✓	✓			

Command	local-url	ftp-url	sftp-url	tftp-url	http-url	https-url
checksum	✓	✓	✓	✓		
copy	✓	✓	✓	source only	✓	✓
copy (recursive)	✓	✓	✓			
list	✓	✓	✓			
make-directory	✓	✓	✓			
move	✓	✓	✓		✓	✓
move (recursive)	✓	✓	✓			
permission	✓					
remove	✓	✓	✓		✓	✓
remove-directory	✓	✓	✓			
show	✓	✓	✓	✓	✓	✓
unzip	✓	✓	✓	source only		
version	✓	✓	✓	✓		

2.4.1 Managing storage devices

Use the **repair** command to check a storage device for errors and repair any errors found. The device does not need to be administratively disabled.

Example: Repair command syntax

```
[/]
A:admin@node-2# file repair cf3:
Checking drive cf3: on slot A for errors...
Drive cf3: on slot A is OK.
```

Use the **format** command to format a storage device with a new file system without erasing the data. The device must be administratively disabled first.

Example: Format command syntax

```
[/]
A:admin@node-2# file disable cflash-id cf1:

[/]
A:admin@node-2# file format cf1:
Formatting Drive cf1: on Slot A ...
Drive cf1: on Slot A is formatted

[/]
```

```
A:admin@node-2# file enable cflash-id cf1:
```

Use the **secure-erase** command to secure erase and format a storage device with a new file system using the clear sanitization method to sanitize media as defined in IEEE 2883-2022, *IEEE Standard for Sanitizing Storage*. This action may take several minutes to complete depending on the storage device size, because the data is overwritten in one or more passes. The device must be administratively disabled first.

Example: Secure erase command syntax

```
[/]
A:admin@node-2# file disable cflash-id cf1:

[/]
A:admin@node-2# file secure-erase cf1:
Are you sure you want to secure erase and format drive cf1: (y/n)?y
Secure erasing drive cf2: on slot A - 100% complete ...
Formatting drive cf1: on slot A ...
Drive cf1: is formatted
Drive cf1: is secure erased
```



Note: New system data may be written to a storage devices after it is administratively enabled. Do not administratively enable a storage device unless it is to be reused.

2.4.2 Displaying directory and file information

Use the **list** command to list the files on a file system, with an option to indicate the list order based on the date, name, or size of the files. The **show** command displays the contents of a specified file or multiple files. The **version** command displays the version of an SR OS .tim image file.

Example: Display directory and file information

```
[/]
A:admin@node-2# file list

Volume in drive cf3 on slot A is .

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/01/2020  11:27p      <DIR>          .ssh/
01/01/1980  12:00a          170 NVRAM.DAT
01/01/1980  12:00a          679 bof.cfg
09/01/2020  11:27p          319 nvsys.info
09/01/2020  11:27p           1 restcntr.txt
09/02/2020  04:32p      <DIR>          tstidir/
               4 File(s)              1169 bytes.
               2 Dir(s)                 0 bytes free.

[/]
A:admin@node-2# file list size

Volume in drive cf3 on slot A is .

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\
```

```

09/01/2020 11:27p <DIR> .ssh/
09/02/2020 04:32p <DIR> tstdir/
09/01/2020 11:27p 1 restcntr.txt
01/01/1980 12:00a 170 NVRAM.DAT
09/01/2020 11:27p 319 nvsys.info
01/01/1980 12:00a 679 bof.cfg
                4 File(s)          1169 bytes.
                2 Dir(s)           0 bytes free.

[/]
A:admin@node-2# file show example.cfg
File: example.cfg
-----
configure {
  card 1 {
    mda 1 {
    }
  }
  log {
    filter 1001 {
      entry 10 {
        description "Collect only events of major severity or higher"
        action forward
        match {
          severity {
            gte major
          }
        }
      }
    }
  }
  log-id 99 {
    description "Default System Log"
    source {
      main true
    }
  }
}
--(more)--(5%)--(lines 1-29/464)--

[/]
A:admin@node-2# file version boot.ldr
TiMOS-L-24.10.R1
Thu Oct 31 23:49:01 UTC 2024 by builder in /builds/2410B/R1/panos/main/sros

```

2.4.3 Modifying file attributes

The system administrator can change the attribute of a local file or files in a directory.

Enter the **permission** command with no options to display the contents of the directory and the file attributes.

A single local file can be specified or the wildcard character (*) can be used to indicate multiple files. If no URL is specified, the command applies to all files in the directory.

A file with an "R" preceding the filename indicates the file is read-only; otherwise, the file is read-write.

Example: Modify file attributes

```

[/]
A:admin@node-2# file permission
cf3:\NVRAM.DAT
cf3:\bof.cfg
cf3:\nvsys.info

```

```

        cf3:\restcntr.txt
        cf3:\.ssh
        cf3:\my.txt

[/]
A:admin@node-2# file permission read-only my.txt

[/]
A:admin@node-2# file permission
        cf3:\NVRAM.DAT
        cf3:\bof.cfg
        cf3:\nvsys.info
        cf3:\restcntr.txt
        cf3:\.ssh
R       cf3:\my.txt

[/]
A:admin@node-2# file permission read-only

[/]
A:admin@node-2# file permission
R       cf3:\NVRAM.DAT
R       cf3:\bof.cfg
R       cf3:\nvsys.info
R       cf3:\restcntr.txt
R       cf3:\.ssh
R       cf3:\my.txt

```

2.4.4 Creating and navigating directories

New directories can be created in the local file system, one level at a time.

Use the **make-directory** command to create a new directory.

The **change-directory** command navigates to different directories.

Example: Create and navigate directories

```

[/]
A:admin@node-2# file

[file "cf3:\"]
A:admin@node-2# make-directory test1

[file "cf3:\"]
A:admin@node-2# change-directory test1

[file "cf3:\test1"]
A:admin@node-2# make-directory test2

[file "cf3:\test1"]
A:admin@node-2# change-directory test2

[file "cf3:\test1\test2"]
A:admin@node-2# make-directory test3

[file "cf3:\test1\test2"]
A:admin@node-2# change-directory test3

[file "cf3:\test1\test2\test3"]
A:admin@node-2# change-directory ..

```

```
[file "cf3:\test1\test2"]
A:admin@node-2#
```

2.4.5 Copying files

Use the **copy** command to copy files to or from a flash device, or an FTP, TFTP, or SSH server. The **copy** command supports wildcards.

Use the **recursive** option to recursively copy files and directories. If files or directories already exist, the user is prompted to overwrite them. Use the **force** option to automatically overwrite the existing files or directories, without being prompted for confirmation.

The following example shows how to copy the `config.cfg` file to the `test_dir1` directory.

Example: Local file copy

```
[/]
A:admin@node-2# file list test_dir1

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test_dir1

09/29/2021  05:07p      <DIR>          ./
09/29/2021  05:07p      <DIR>          ../
              0 File(s)                  0 bytes.
              2 Dir(s)                14458880 bytes free.

[/]
A:admin@node-2# file list

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/28/2021  11:43a                931 B0F.CFG
09/28/2021  11:43a                165 NVRAM.DAT
09/28/2021  04:34p            11319 bootlog.txt
09/28/2021  03:50p            11259 bootlog_prev.txt
09/29/2021  05:08p            11319 config.cfg
09/28/2021  04:34p             319 nvsys.info
04/16/2014  10:15a      <DIR>          syslinux/
09/29/2021  05:08p      <DIR>          test_dir1/
04/16/2014  10:15a      <DIR>          timos/
              7 File(s)                35312 bytes.
              3 Dir(s)                14458880 bytes free.

[/]
A:admin@node-2# file copy config.cfg test_dir1
Copying file cf3:\config.cfg ... OK
1 file copied.

[/]
A:admin@node-2# file list test_dir1
```

```

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test_dir1

09/29/2021  05:07p      <DIR>          ./
09/29/2021  05:07p      <DIR>          ../
09/29/2021  05:09p                11319 config.cfg
                        1 File(s)                11319 bytes.
                        2 Dir(s)                14447104 bytes free.

```

The following example shows how to copy the `config.cfg` file to a file on an SSH server.

Example: File copy to an SSH server

```

[/]
A:admin@node-2# file copy cf3:config.cfg scp://user@10.231.216.68:~/
user@10.231.216.68's password:
config.cfg                                100% 625      0.6KB/s   00:00

```

The following example shows a recursive move of the `recursive4` directory to `recursive5`.

Example: Recursive directory move

```

[/]
A:admin@node-2# file list

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021  06:11p  196 B0F.CFG
09/19/2021  06:11p   0 CONFIG.CFG
09/19/2021  07:22p  SUPPORT/
09/23/2021  09:03a 27459 bootlog.txt
09/20/2021  09:56a 27326 bootlog_prev.txt
09/23/2021  01:21p  319 nvsys.info
09/23/2021  02:42p recursive4/
09/23/2021  08:22a ssh/
7 File(s) 55402 bytes.
7 Dir(s) 612311040 bytes free.

[/]
A:admin@node-2# file copy recursive4 recursive5 recursive
Copying directory cf3:\recursive4
Copying directory cf3:\recursive4\recursive2
Copying file cf3:\recursive4\recursive2\file2.txt ... OK
Copying file cf3:\recursive4\recursive2\file3.txt ... OK
Copying file cf3:\recursive4\file1.txt ... OK
2 dir(s) and 3 file(s) copied.

```

2.4.6 Moving files

Files or directories can be moved from one location to another. The **move** command recursively moves files and directories, and overwrites existing content without prompting for confirmation. The **move** command supports wildcards.

The following example moves the `md-config.cfg` file to the `test_dir1` directory.

Example: Moving files and directories

```
[/file "cf3:\"]
A:admin@node-2# list test_dir1

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test_dir1

09/29/2021  05:07p      <DIR>          ./
09/29/2021  05:07p      <DIR>          ../
              0 File(s)                  0 bytes.
              2 Dir(s)                14458880 bytes free.

[/file "cf3:\"]
A:admin@node-2# list

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/28/2021  11:43a              931 B0F.CFG
09/28/2021  11:43a              165 NVRAM.DAT
09/28/2021  04:34p            11319 bootlog.txt
09/28/2021  03:50p            11259 bootlog_prev.txt
09/29/2021  05:08p            11319 md-config.cfg
09/28/2021  04:34p              319 nvsys.info
04/16/2014  10:15a      <DIR>          syslinux/
09/29/2021  05:09p      <DIR>          test_dir1/
04/16/2014  10:15a      <DIR>          timos/
              7 File(s)                35312 bytes.
              3 Dir(s)                14458880 bytes free.

[/file "cf3:\"]
A:admin@node-2# move md-config.cfg test_dir1
Moving file cf3:\md-config.cfg ... OK
cf3:\md-config.cfg

[/file "cf3:\"]
A:admin@node-2# list test_dir1

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test_dir1

09/29/2021  05:07p      <DIR>          ./
```

```

09/29/2021 05:07p <DIR> ../
09/29/2021 05:08p 11319 md-config.cfg
                  1 File(s) 11319 bytes.
                  2 Dir(s) 14458880 bytes free.

[/file "cf3:\"]
A:admin@node-2# list

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/28/2021 11:43a 931 B0F.CFG
09/28/2021 11:43a 165 NVRAM.DAT
09/28/2021 04:34p 11319 bootlog.txt
09/28/2021 03:50p 11259 bootlog_prev.txt
09/28/2021 04:34p 319 nvsys.info
04/16/2014 10:15a <DIR> syslinux/
10/01/2021 04:17p <DIR> test_dir1/
04/16/2014 10:15a <DIR> timos/
                  6 File(s) 23993 bytes.
                  3 Dir(s) 14458880 bytes free.

```

The following example shows a recursive move of the recursive1 directory to recursive3.

Example: Recursive directory move

```

[/file "cf3:\"]
A:admin@node-2# list

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 07:22p SUPPORT/
09/20/2021 09:56a 27326 bootlog.txt
09/20/2021 02:55p 319 nvsys.info
09/21/2021 07:19p recursive1/
09/20/2021 09:55a ssh/
6 File(s) 27943 bytes.
6 Dir(s) 612347904 bytes free.

[/file "cf3:\"]
A:admin@node-2# list recursive1

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive1
09/21/2021 07:18p ./
09/21/2021 07:18p ../
09/21/2021 07:19p 7 file1.txt
09/21/2021 07:19p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612347904 bytes free.

```

```
[/file "cf3:\"]
A:admin@node-2# list recursive1/recursive2

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive1\recursive2

09/21/2021 07:19p ./
09/21/2021 07:19p ../
09/21/2021 07:19p 7 file2.txt
09/21/2021 07:19p 7 file3.txt
2 File(s) 14 bytes.
2 Dir(s) 612347904 bytes free.

[/file "cf3:\"]
A:admin@sros# move recursive1 recursive3
Moving file cf3:\recursive1 ... OK
cf3:\recursive1

[/file "cf3:\"]
A:admin@node-2# list

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\

09/19/2021 06:11p 196 B0F.CFG
09/19/2021 06:11p 0 CONFIG.CFG
09/19/2021 06:11p 101 NVRAM.DAT
09/19/2021 07:22p SUPPORT/
09/19/2021 06:11p SYSLINUX/
09/19/2021 06:11p TIMOS/
09/20/2021 09:56a 27326 bootlog.txt
09/20/2021 02:55p 319 nvsys.info
09/21/2021 07:19p recursive3/
09/20/2021 09:55a ssh/
6 File(s) 27943 bytes.
6 Dir(s) 612347904 bytes free.

[/file "cf3:\"]
A:admin@node-2# list recursive3

Volume in drive cf3 on slot A is SROS VM.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\recursive3

09/21/2021 07:18p ./
09/21/2021 07:18p ../
09/21/2021 07:19p 7 file1.txt
09/21/2021 07:19p recursive2/
1 File(s) 7 bytes.
3 Dir(s) 612347904 bytes free.
```

2.4.7 Deleting files and removing directories

Use the **remove** and **remove-directory** commands to delete files and remove directories. Directories can be removed even if they contain files or subdirectories.

Example: Removing directories

```
[file "cf3:\test1\test2"]
A:admin@node-2# list

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test1\test2

09/01/2020  08:13p      <DIR>      ./
09/01/2020  08:13p      <DIR>      ../
09/04/2020  06:36p                874 bof.cfg
04/28/2020  03:15p            11401 md-config.cfg
09/04/2020  06:43p      <DIR>      test3/
                2 File(s)                12275 bytes.
                3 Dir(s)                10641920 bytes free.

[file "cf3:\test1\test2"]
A:admin@node-2# list test3

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test1\test2\test3

09/01/2020  08:13p      <DIR>      ./
09/01/2020  08:13p      <DIR>      ../
09/04/2020  06:43p            11788 conf3.cfg
09/04/2020  04:24p            6645 mybof.cfg
                2 File(s)                18433 bytes.
                2 Dir(s)                10641920 bytes free.

[file "cf3:\test1\test2"]
A:admin@node-2# remove-directory test3 ?

remove-directory
force                - Force removal without prompting
recursive            - Remove directory and its content recursively

[file "cf3:\test1\test2"]
A:admin@node-2# remove-directory test3 recursive
Deleting all subdirectories and files in specified directory. y/n ?y
Deleting file cf3:\test1\test2\test3\mybof.cfg ... OK
Deleting file cf3:\test1\test2\test3\conf3.cfg ... OK
Deleting directory cf3:\test1\test2\test3 ... OK

[file "cf3:\test1\test2"]
A:admin@node-2# list

Volume in drive cf3 on slot A is TIMOS_VM_CF.

Volume in drive cf3 on slot A is formatted as FAT32

Directory of cf3:\test1\test2
```

```

09/01/2020 08:13p <DIR> ./
09/01/2020 08:13p <DIR> ../
09/04/2020 06:36p      874 bof.cfg
04/28/2020 03:15p    11401 md-config.cfg
                2 File(s)          12275 bytes.
                2 Dir(s)          10661376 bytes free.

```

2.4.8 Unzipping files

Use the **unzip** command to expand the contents of a ZIP file to the local file system. Any file zipped using the store, deflate, or zip64 compression methods can be unzipped. An example is the SR OS software image available from the Nokia customer support portal.

The source ZIP file location can be a locally installed solid-state storage device or a remote FTP or TFTP server.

The **create-destination** keyword ensures that any non-existent directory structure that is explicitly entered as the destination file URL is created as part of the unzip operation. This parameter is required to create new directories.



Note:

- The destination for the unzipped files and directories must be a locally installed solid-state storage device in the active CPM.
- ZIP filenames, or the filenames of any contained files, must not include special characters.

Example: Unzipping a file

```

[/]
A:admin@node-2# file unzip demo.zip cf3:/mynewfolder/mynewsfolder create-
destination force
Verifying cf3:\demo.zip .. ... OK
Unzipping cf3:\demo.zip to cf3:\mynewfolder\mynewsfolder\ .. .Processing demodir/
Processing demodir/myfile1.txt
Processing demodir/myfile2.txt
Processing demodir/demosubdir/
Processing demodir/demosubdir/myfile3.txt
Writing...OK

```

2.4.9 Displaying file checksums

Use the **checksum** command to display file checksums.

Example: Check a .tim image file checksum

The following example shows the output of the operation to check an SR OS .tim image file checksum.

```

[/]
A:admin@node-2# file checksum image cpm.tim
TiMOS-C-20.2.R1
Sat Feb 29 10:39:32 PST 2020 by builder in /builds/c/202B/R1/panos/main/sros
Checking file ... OK

```

Example: Output of the checksum operation

The following example shows the output of the checksum operation to compute and display a checksum based on the MD5 and SHA256 algorithms for the `cpm.tim` file on `cf3`.

```
[/]  
A:admin@node-2# file checksum md5 cpm.tim  
Checking file cf3:cpm.tim  
c65699dc05e6e35a2172eaac80485aa2  
  
[/]  
A:admin@node-2# file checksum sha256 cpm.tim  
Checking file cf3:cpm.tim  
a1a813a696be04906f9faf1df9db0f90a990ff51cb3383099ade21241203bc1c
```

3 System initialization and boot options

This section describes the system initialization and boot option process.

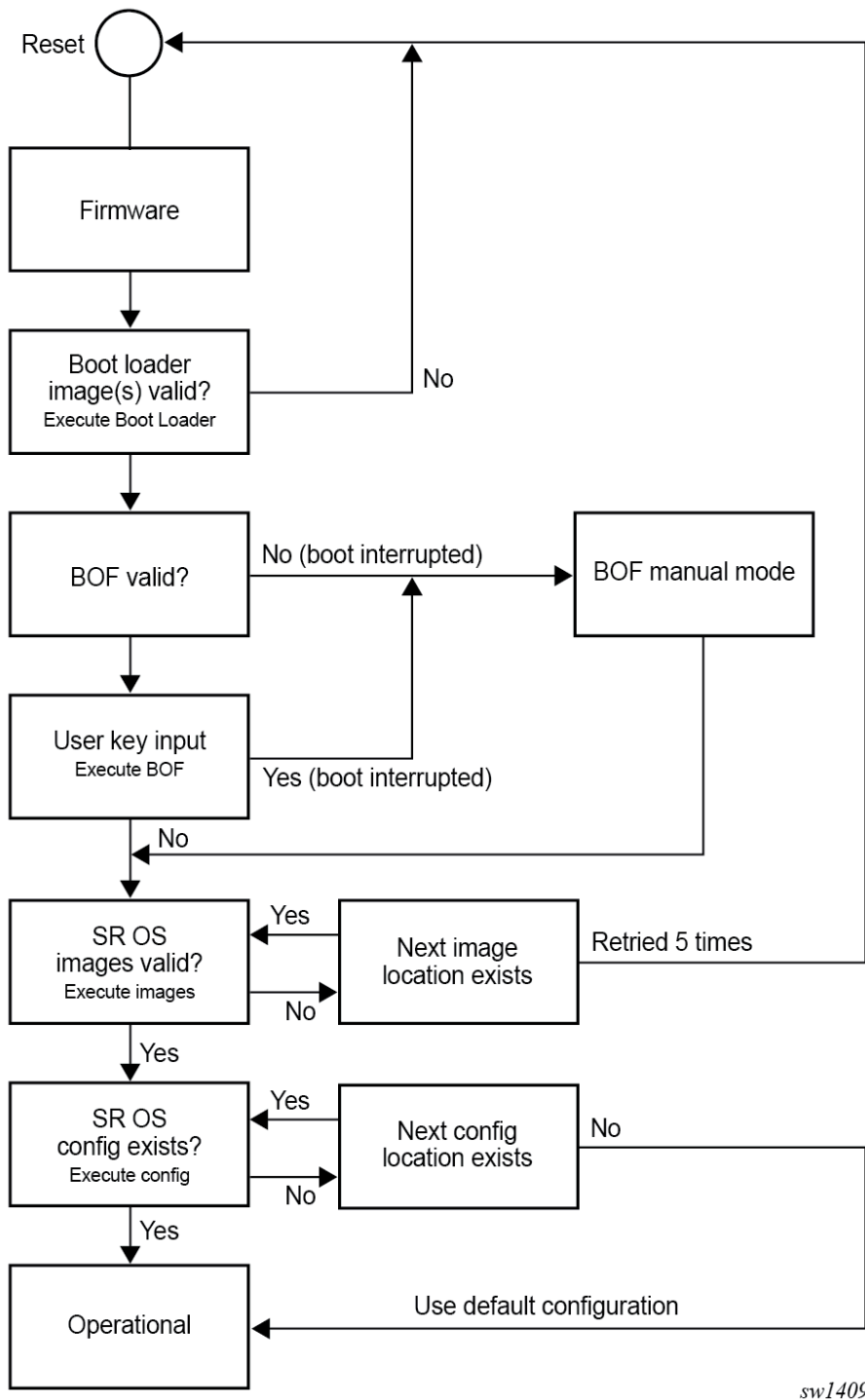
3.1 Boot process

The router startup process begins after a reset or power cycle with the firmware initializing the hardware before executing the Boot Loader images. The Boot Loader then executes the Boot Options File (BOF) to load the SR OS software image and configuration. The BOF file contains system initialization commands including the software image and configuration locations.

SR OS Boot Loader, software images, and configuration files are stored in storage media cards referred to as CF in the system. See [Storage devices](#) for more information about the type of storage supported for each platform.

The following diagram shows the system boot process from the firmware up to the SR OS image and configuration file.

Figure 1: Boot process



sw1409

3.2 Boot Loader

The Boot Loader executes the initialization parameters from the BOF to load the software images and configuration file.

The Boot Loader phase can be manually interrupted even if the BOF is present by pressing any key on the console connected to the console port. This is done by typing **sros** and pressing the **Enter** key within 30 seconds to enter the BOF Manual Mode. This mode allows the configuration of the BOF system initialization commands manually and overwrites the existing BOF file if present.

Different Boot Loader images are used depending on the CPM control module:

- 7450 ESS, 7750 SR, 7750 SR-7s, 7750 SR-14s, 7950 XRS
 - `boot.ldr` is the Boot Loader image that executes the BOF file before loading SR OS TiMOS software images and configuration
 - `boot.ldr` must be located at the root directory of the CF3 card (for systems using `boot.ldr`)
- 7750 SR-1x-48D, SR-1x-92S, SR-1-24D, SR-1-48D, SR-1-46S, SR-1-92S, SR-1se, and SR-2se
 - `bootx64.efi` is the original Boot Loader image located in `/EFI/B00T`
 - `bootx64.efi` executes the other images in `/EFI/B00T/x86_64` before executing the BOF file and loading SR OS TiMOS software images and configuration

3.3 Boot Options File

The BOF file (`bof.cfg`) must be located at the root of the CF3 card directory and contains various system initialization commands including:

- management Ethernet port (speed, duplex, IP address, static routes)
- console port speed
- software image locations
- configuration file locations
- BOF and configuration file encryption settings
- BOF password
- system profile
- wait time
- Zero Touch Provisioning
- licenses
- persistency

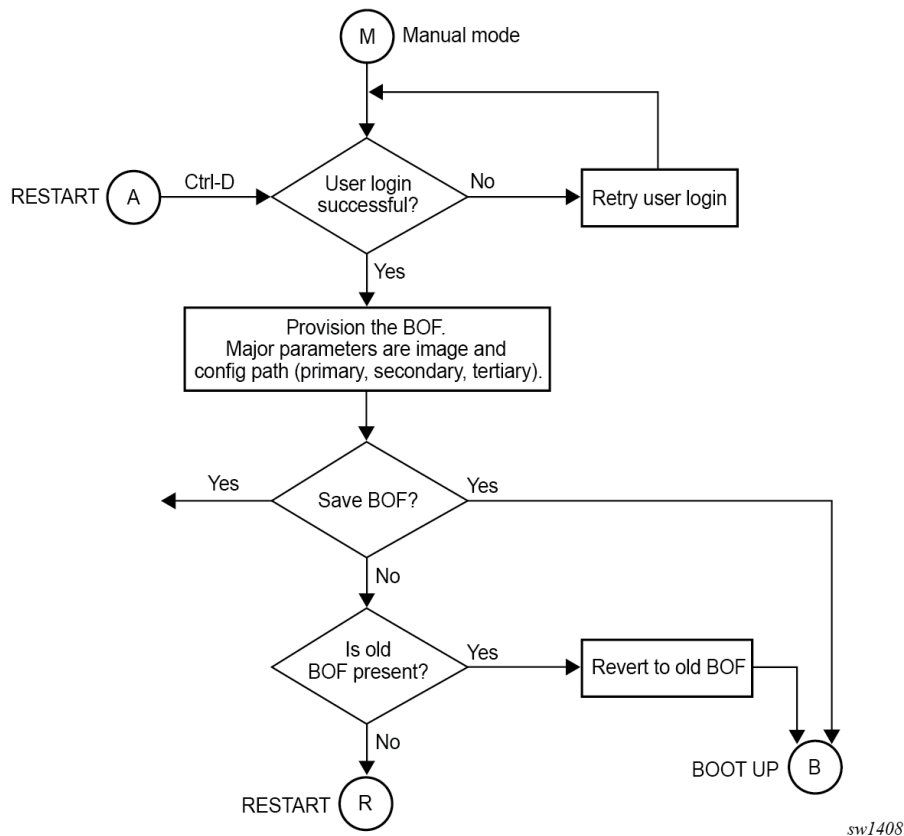
3.3.1 BOF manual mode

The system enters the BOF manual mode if the BOF is not present in the CF3 card or if the user interrupts the Boot Loader phase and requires access to the console port for configuration.

After the manual BOF configuration is completed and saved, a `bof.cfg` file with the new configured command options is created in the CF3 card and used for subsequent reboots. The Boot Loader image then processes the new BOF command options to boot the system.

This process is described in the following diagram.

Figure 2: BOF manual mode



3.4 Software and configuration

The software image and configuration file location are configured in the BOF.

Up to three locations, local or remote, can be configured for the software image and configuration file defined as primary image, secondary image, tertiary image, primary config, secondary config and tertiary config in the BOF.

Before loading the configuration, the software first attempts to read the license file if one has been included in the BOF. If a license file is found, it is activated. If there are any issues with the activation, a log event is raised, and the startup processing continues with the reading of the configuration file.

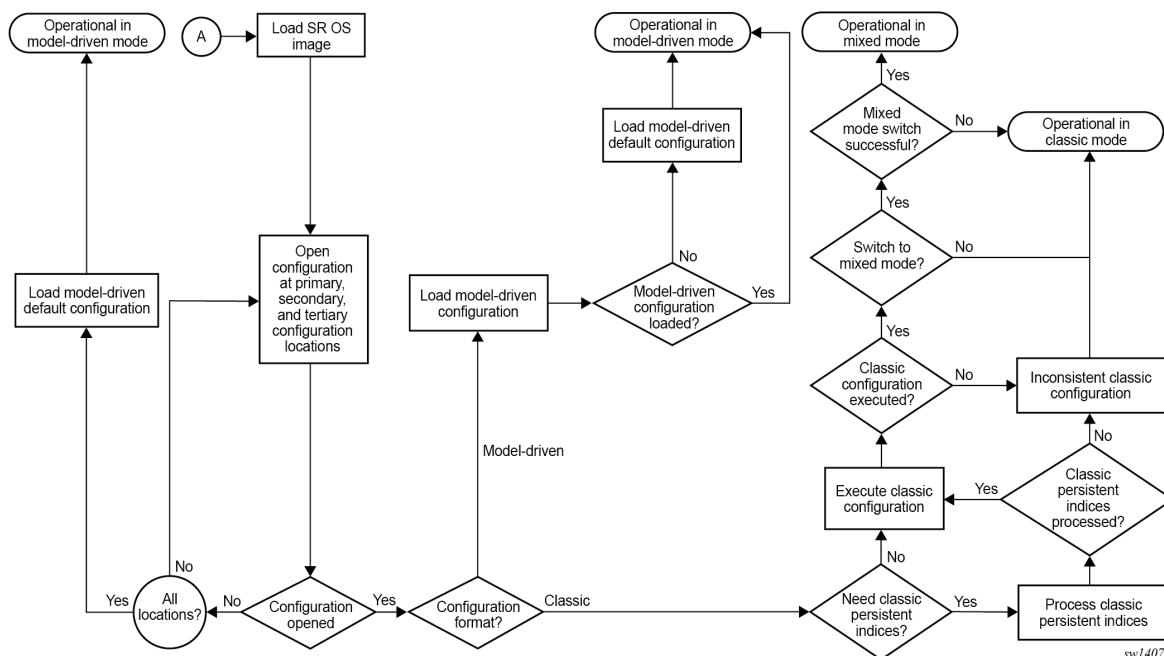
The following usage guidelines apply:

- The primary, secondary, and tertiary image locations must have the same version of software. If the secondary or tertiary image is configured with an older software image, this may result in a failure to load the configuration file as the file may contain commands only applicable to the more recent release.

- Similarly, the secondary and tertiary configuration files, if used, should be saved with the same version of software as the software executed on boot as it can result in a failure to load the configuration file otherwise.
- In the model-driven configuration mode, with incremental saved configuration files enabled, the primary configuration location supports complete and incremental saved configuration files. The secondary and tertiary configuration locations support complete saved configuration files. The user must ensure that complete saved configuration files are stored at both locations.

The following diagram provides additional details on the boot process differences between classic and MD-CLI configuration file processing.

Figure 3: Load SR OS configuration



3.4.1 Management interface configuration modes

The system can operate in different management interface configuration modes, which affect the CLI and network management protocols that can be used to configure the system. When the system boots and loads the configuration file, the configuration mode is set as follows:

- The default configuration mode is **model-driven** and a model-driven configuration file format is loaded. The value of **configure system management-interface configuration-mode** in the configuration file must not be **classic** or **mixed**.
- If the configuration file has **exit all** as the first executable line, the configuration mode is set to **classic** and a classic configuration file format is loaded. Lines beginning with a number sign character (#) are ignored.
 - The configuration mode may be changed to **mixed** if the value of **configure system management-interface configuration-mode** is **mixed** in the configuration file.

- The value of **configure system management-interface configuration-mode** in the configuration file must not be **model-driven**.

See "Management interface configuration modes" in the *7450 ESS, 7750 SR, 7950 XRS, and VSR System Management Guide* for more information.

3.5 Initial installation and software update

The following requirements apply to all systems that use the Extensible Firmware Interface (EFI):

- When installing a new CF3 SD card for the first time, all files including the installer files listed in the storage card content section of the respective platforms must be present in the SD card, and the SD card must be in FAT32 format. See [Storage card content](#) for more information. The first boot of the system on a new CF3 SD card initializes the software and partitions on the SD card. The SD card contains critical OS installer files that are distributed over three partitions. Two partitions are available for internal use, and the third partition, visible from TiMOS as CF3:, is for use by the user.



Note: An SD card containing necessary files and partitions can be inserted into an operational router; replacing the CF3 SD card with a new SD card, however, requires a reboot for the system to initialize the new card.

- The `bof.cfg` file must be at the root of the CF3:\ directory.
- When upgrading or changing the software on a system that has previously been booted correctly, the OS installer files in the \EFI directory must be manually deleted.
- The `boot.ldr` file is not present in the CF3:\ directory.

The following requirements apply to all systems that use the `boot.ldr`:

- The `boot.ldr` and `bof.cfg` files must be at the root of the CF3:\ directory.
- When upgrading or changing the software, the corresponding `boot.ldr` image must also be updated in the CF3:\ directory.
- The \EFI directory is not present in the CF3:\ directory.

3.6 Storage card content

SR OS software downloaded from the Nokia support website includes boot and operating system images for all platforms. This section describes the required storage media card directory and filenames on a per-platform basis to clarify which files and directory apply to which platform.

The primary copy of the SR OS software is located on a Compact Flash or SD card which is shipped with each software license agreement. The content of this storage media card can differ depending on the platform.

Configurations and executable images can be stored in any storage media supported by the platform while the boot loader images, and boot option file must be installed in the storage media card slot 3 (cf3:).

See the [Storage devices](#) section for the list of storage media names, locations, and support for each SR OS platform.

3.6.1 7750 SR-1x-48D, SR-1x-92S, SR-1-24D, SR-1-48D, SR-1-46S, SR-1-92S, SR-1se, and SR-2se

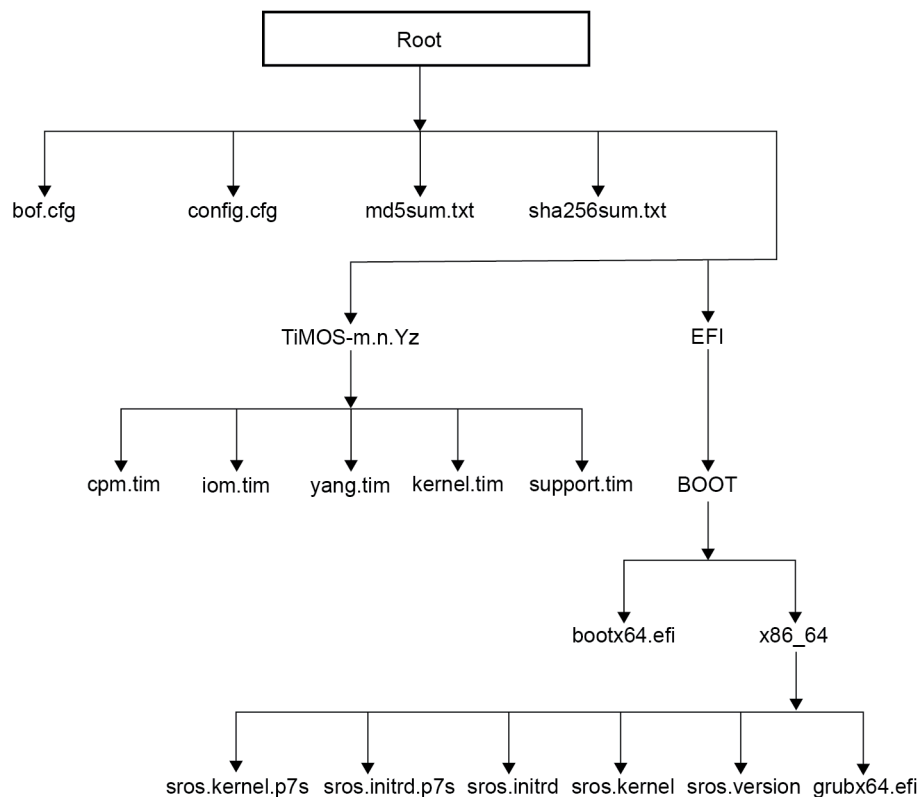
When installing a new storage media card into the system for the first time, ensure that the media card contains only the software files shipped by Nokia.

The following figure shows the required storage media card directory structure and filenames.



Note: Although cards in CF1/CF2 and CF3 are hot-pluggable, cards are not interchangeable between CF3 and CF1/CF2. Swapping SD cards between CF3 and CF1/CF2 can result in file system corruption.

Figure 4: Files on storage card — 7750 SR-1x-48D, SR-1x-92S, SR-1-24D, SR-1-48D, SR-1-46S, SR-1-92S, SR-1se, or SR-2se



sw4132

The following files are present on the storage media card:

- `bof.cfg` – boot option file
- `config.cfg` – default configuration file
- `md5sum.txt` – MD5 checksum file
- `sha256sum.txt` – SHA256 checksum file
- `TiMOS-m.n.Yz`:

- m – signifies a major release number
- n – signifies a minor release number
- Y: A – signifies an alpha release
- B – signifies a beta release
- M – signifies a maintenance release
- R – signifies a released software
- z – signifies a version number
- `cpm.tim` – CPM image file
- `iom.tim` – IOM image file
- `kernel.tim` – host operating system
- `support.tim` – required data for SR OS .tim files
- `yang.tim` – YANG model library

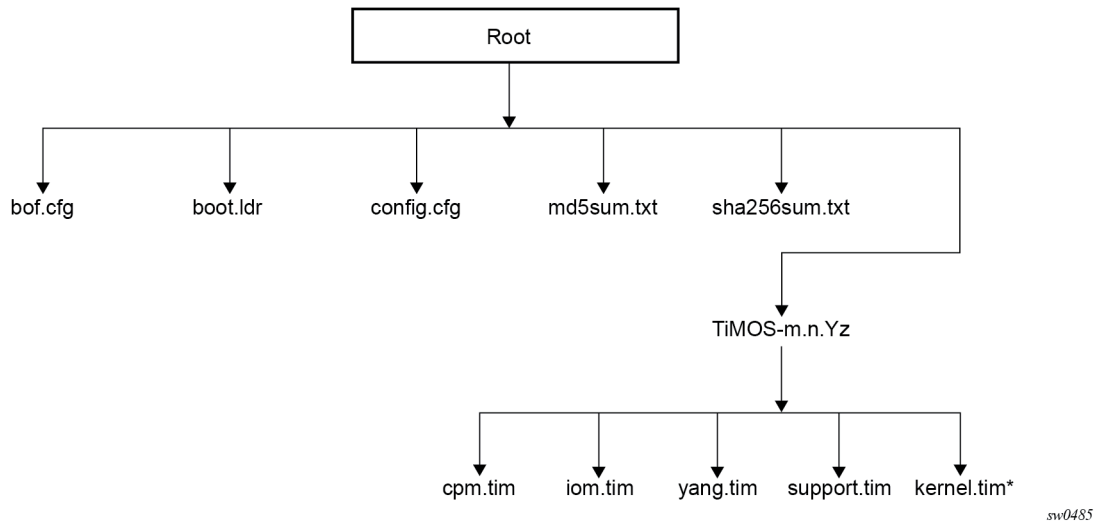
The following files and folder structure under /EFI should only be included if the system is booted with a new storage media card installed for the first time:

- EFI:
 - BOOT:
 - `bootx64.efi` – EFI file; Boot Loader
 - x86_64:
 - `sros.initrd` – OS installer file; installer rootfs
 - `sros.kernel` – OS installer file; installer kernel
 - `sros.version` – OS installer file; installer version
 - `grubx64.efi` – EFI file; GRUB boot loader

3.6.2 7450 ESS, 7950 XRS, 7750 SR 7/7s/12/12e/14s, SR-a, SR-e, and SR-2s

The following figure shows the required storage media card directory structure and filenames.

Figure 5: Files on storage card — 7450 ESS, 7950 XRS, 7750 SR 7/7s/12/12e/14s, SR-a, SR-e, and SR-2s



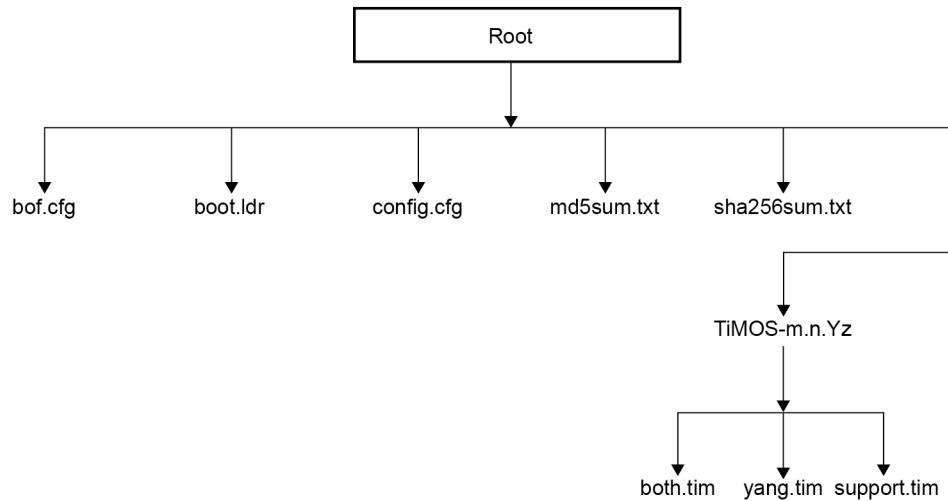
The following files are on the storage media card:

- `bof.cfg` – boot option file
- `boot.ldr` – bootstrap image file
- `config.cfg` – default configuration file
- `md5sum.txt` – MD5 checksum file
- `sha256sum.txt` – SHA256 checksum file
- `TiMOS-m.n.Yz`:
 - `m` – signifies a major release number
 - `n` – signifies a minor release number
 - `Y`: `A` – signifies an alpha release
 - `B` – signifies a beta release
 - `M` – signifies a maintenance release
 - `R` – signifies a released software
 - `z` – signifies a version number
 - `cpm.tim` – CPM image file
 - `iom.tim` – IOM image file
 - `support.tim` – required data for SR OS .tim files
 - `kernel.tim` – host operating system, required for 7750 SR-7s and 7750 SR-14s only
 - `yang.tim` – YANG model library

3.6.3 7750 SR-1 and SR-1s

The following figure shows the required storage media card directory structure and filenames.

Figure 6: Files on storage card — 7750 SR-1 and SR-1s



sw0486

The following files are present on the storage media card:

- bof.cfg – boot option file
- boot.ldr – bootstrap image file
- config.cfg – default configuration file
- md5sum.txt – MD5 checksum file
- sha256sum.txt – SHA256 checksum file
- TIMOS-m.n.Yz:
 - m – signifies a major release number
 - n – signifies a minor release number
 - Y: A – signifies an alpha release
 - B – signifies a beta release
 - M – signifies a maintenance release
 - R – signifies a released software
 - z – signifies a version number
 - both.tim – CPM and IOM image file
 - support.tim – required data for SR OS .tim files
 - yang.tim – YANG model library

3.6.4 ISA and ESA applications

The following images must also be present in the SR OS software TiMOS-m-n-Yz directory depending if ESA or Application Assurance are used:

- `hypervisors.tim`: hypervisor image for the ESA cards
- `isa-aa.tim`: Application Assurance software image

3.7 Persistent indexes in classic and mixed configuration mode

The BOF **persist** command option specifies whether the system should preserve system indexes when the configuration is saved. During a subsequent boot, the index file is read along with the configuration file. As a result, a number of system indexes are preserved between reboots, including the SNMP interface index, LSP IDs, path IDs, and so on. If persistence is not required and the configuration file is successfully processed, the system becomes operational. If **persist** is required, a matching `config.ndx` file must be successfully processed before the system can become operational. Configuration and index files must have the same filename prefix such as `config.cfg` and `config.ndx` and are created at the same time when an **admin save** command is executed. Note that the persistence option must be enabled to deploy a Network Management System (NMS) using SNMP. The default is off.



Note: System indexes in model-driven configuration mode are always persistent.

3.8 BOF and configuration file encryption

In cases where the platform is not installed in a physically secure location, the user can encrypt the BOF and the configuration file to halt or hinder interpretation of the file content.

By default, the BOF and configuration files are not encrypted. When encryption is enabled for either file and a change is saved, the original file is moved to `filename.1` and the encrypted file becomes the new `filename.cfg`.



Caution: The first time a file is encrypted and the original file is moved to `filename.1`, the `filename.1` file is unencrypted. Delete the unencrypted file to maintain node security.

When the BOF is encrypted on the Compact Flash, the BOF interactive menu can be used during node startup to access the file and modify BOF fields. To prevent unauthorized modification of the BOF using the BOF interactive menu, configure a password using the following command:

- **MD-CLI**

```
bof configuration password
```

- **classic CLI**

```
bof password
```

The BOF interactive menu is accessible only when the configured password is entered. If the correct password is not entered in 30 seconds, the node reboots.

See [Configuring BOF encryption](#) for information about configuring BOF encryption. See [Configuring the BOF interactive menu password](#) for information about configuring the BOF interactive menu password. See [Configuring configuration file encryption](#) for information about configuring configuration file encryption.

3.9 System profiles

System profiles provide flexibility when using line cards based on FP4 and later generations by supporting different system capabilities. The system profile is defined in the BOF and is used by the system when it is next rebooted. Contact your Nokia representative for system profile information.

The following system profiles are supported:

- **profile none**

This profile represents the existing system capabilities and allows hardware based on FP3 and later generations to coexist within a system. This profile is indicated by the omission of the profile parameter in the BOF.

- **profile A**

This profile is primarily targeted at subscriber services and Layer 2 and 3 VPN business services. Use the following command to configure the profile:

- **MD-CLI**

```
bof system profile profile-a
```

- **classic CLI**

```
bof system-profile profile-a
```

- **profile B**

This profile is primarily targeted at infrastructure routing, core, peering, and DC-GW applications.

System profiles **profile-a** and **profile-b** support only line cards based on FP4 and later generations. Provisioning FP2- or FP3-based line cards is prohibited when the system profile is set to **profile-a** or **profile-b**. If FP2- or FP3-based card types are present in the boot configuration when using these profiles, the boot sequence aborts the loading of the configuration file when it encounters their configuration.

When changing between system profiles, it is mandatory to remove all configuration commands for features that are not supported in the target system profile before rebooting the system, otherwise the reboot fails at the unsupported configuration command on startup.

On 7750 SR-1 and 7750 SR-s systems, the following conditions apply about the profile parameter:

- The parameter should be configured to either **profile-a** or **profile-b**.
- If the parameter is omitted, profile **profile-a** is used by the system.
- If the parameter is configured to an invalid value, it is ignored and profile **profile-a** is used by the system.

On 7750 SR-7-B/12-B/12e and 7950 XRS-20/20e systems, the following conditions apply about the profile parameter:

- The default system profile is **none** when the parameter is omitted.

- The parameter can be configured to either **profile-a** or **profile-b**, in which case only FP4-based line cards are supported.
- If the parameter is configured to an invalid value, it is ignored and profile **none** is used by the system.

On all other systems, the following conditions apply about the profile parameter:

- These systems must use profile **none** (the existing system capabilities). As a result, the parameter must not be configured.
- If the parameter is configured to **profile-a** or **profile-b**, the system boots, allowing access using the console and CPM management interface, but FP2-based and FP3-based line cards cannot be provisioned; if these card types are present in the boot configuration, the boot sequence aborts loading the configuration file when it encounters their configuration. This issue can be corrected by removing the parameter and rebooting the system.
- If the parameter is configured to an invalid value, it is ignored and profile **none** is used by the system.

If a system has two CPMs, and the standby CPM boots with a different profile parameter than is used on the active CPM, the active CPM reboots the standby CPM and keep it in a down state. To correct the situation, the BOF can be reconfigured on the standby CPM to match the one configured on the active CPM, and then reboot the system. Alternatively, use the following command to enable automatic BOF synchronization to keep both CPMs in sync.

```
configure redundancy synchronize boot-env
```

When performing a minor or major ISSU software upgrade on dual CPM systems, it is important that the system profile in the BOF on both the active and standby CPM is the same and has a value supported on the pre-upgrade software release. If the standby CPM happened to have a system profile which is only supported in the post-upgrade release, the active CPM reboots the standby and keeps it down because of a system profile mis-match.

Use the following command to display the BOF system profile:

- **MD-CLI**

```
admin show configuration bof | match profile
```

- **classic CLI**

```
show bof | match system-profile
```

The BOF system profile used by the system when it booted can be seen in the boot messages (using the **show boot-messages** command), which display the BOF read when rebooting.

Use the following command to display the system profile that is in use on the system.

```
show chassis | match "System Profile"
```

Use the following command to configure the system profile:

- **MD-CLI**

```
bof system profile
```

- **classic CLI**

```
bof system-profile
```

3.10 FIPS-140 mode

Starting in Release 24.3.R3, SR OS is in the process of obtaining the FIPS-140-3 certification with the Crypto Module Version (CMV) **SRCM 5.0**. After the certificate is obtained, all SR OS releases with CMV **SRCM 5.0** can be considered to be FIPS-approved. You can use the **show system information** command to display the CMV.

The system can be configured in FIPS mode using the **fips-140** flag in the BOF. See [Booting up the system in FIPS mode](#).

Not all SR OS releases are FIPS certified. The FIPS-certified SR OS CMV is listed on the [National Institute of Standards and Technology \(NIST\) website](#). The uncertified releases can operate in FIPS-140 mode.

3.10.1 FIPS operating environment

The SR OS FIPS certification extends over multiple minor or major releases. The system software is FIPS certified against an operating kernel (OK) “TiMOS SMP Kernel Version 2.5”, which does not change. The OK contains a Cryptographic Module (also known as Crypto Module) version (CMV).

The CMV contains the FIPS Provider version (FPV) and the Entropy Source version (ESV). The CMV changes when any of its components change. For example, if the FPV changes to 3.1, the CMV changes to 5.1. If the ESV changes, the CMV changes also.

The FIPS certification is valid for a specified OK version and CMV. Software images that do not contain the certified OK and CMV are not officially FIPS certified. The user must check the [NIST website](#) to identify the FIPS-certified OK and CMV.

3.10.2 Displaying the OK and the CMV

The OK and the CMV are displayed only on systems that support the FIPS mode and are configured in FIPS mode. The CMV, FPV, and ESV are displayed on systems supporting FIPS when the systems are in FIPS mode. The systems that do not support the FIPS mode do not display all the information.

When the system is in FIPS mode, the **show system information** command displays the CMV, FPV and ESV.

The following example shows the FIPS operating environment information on a system in FIPS mode.

Example: System in FIPS mode

```
=====
System Information
=====
System Name       : system-1
System Type       : <system-type>
Chassis Topology  : Standalone
System Version    : <system-version>
Kernel Version    : TiMOS SMP version 2.5
Crypto Module Version : SRCM 5.0
  FIPS-Provider Version: 3.1.5-nokia.1.0
  Entropy Source Version: 3.4.1-nokia.1.0
...
```

The following is the corresponding MD-CLI state information.

Example: MD-CLI state information about the FIPS operating environment on a system in FIPS mode

```
[/state system]
A:admin@node-2# info
...
  crypto-module {
    crypto-module-version "SRCM 5.0"
    fips-provider-version "3.1.5-nokia.1.0"
    entropy-source-version "3.4.1-nokia.1.0"
  }
```

The following is an example of operating environment information on a system that is not in FIPS mode.

Example: Operating environment information on a system that is not in FIPS mode

```
=====
System Information
=====
System Name       : system-2
System Type       : <system-type>
Chassis Topology  : Standalone
System Version    : <system-version>
Kernel Version    : TiMOS SMP version 2.5
Crypto Module Version :
FIPS-Provider Version:
Entropy Source Version: 3.4.1-nokia.1.0
```

The following is the corresponding MD-CLI state information of the preceding system example.

Example: MD-CLI state information about the operating environment on a system that is not in FIPS mode

```
[/state system]
A:admin@node-2# info
...
  crypto-module {
    entropy-source-version "3.4.1-nokia.1.0"
  }
```

3.10.3 Booting up the system in FIPS mode

To start using the system in FIPS mode, perform the following steps:

1. Use the following command to configure the **fips-140** flag in the BOF:

- **MD-CLI**

```
bof system fips-140
```

- **classic CLI**

```
bof fips-140
```

2. Save the BOF.
3. Reboot the system.

4. Starting in Release 24.10.R1, a digital signature (DS) validation using SHA256 is performed for upgrades on the image and the .tim files. See [FIPS image upgrade validation using a DS](#).

3.10.4 Supported algorithms in FIPS mode

When the system is configured in FIPS mode, the ciphers and hash algorithms that are not FIPS approved are removed from the SSH and TLS protocols. The FIPS certificate security target on the [NIST website](#) provides all the supported ciphers, hash, HMAC, and signature algorithms.

3.10.5 FIPS image upgrade validation using a DS

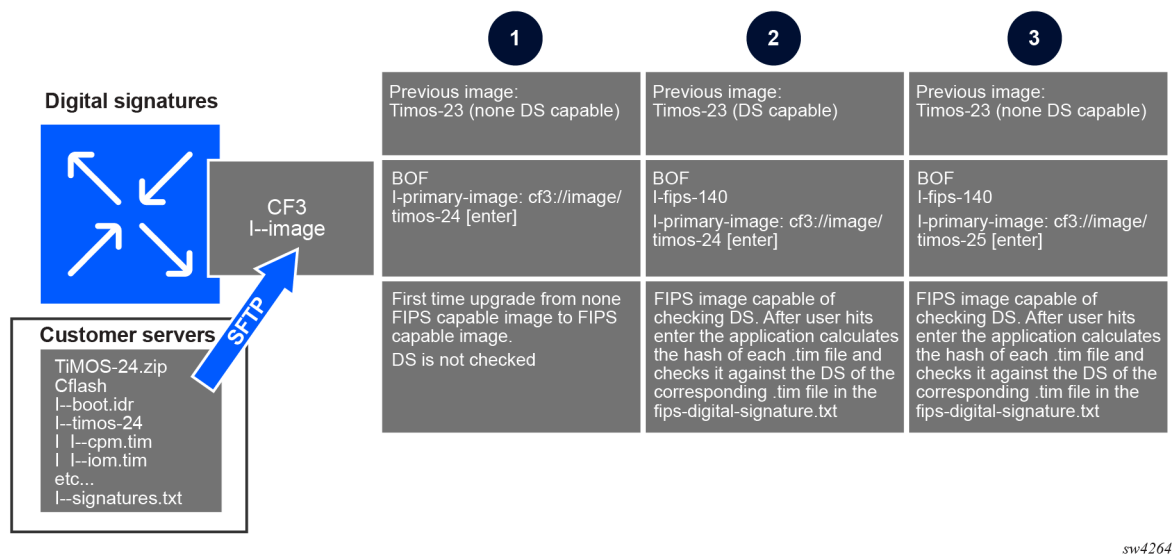
When upgrading to a new image, the National Information Assurance Partnership's (NIAP) collaborative Network Device Protection Profile (NDCPP) requires the new image be verified using a digital signature (DS). For more information about the SR OS DS environment, see [Digital signature](#).

The NIAP functionality is available when the SR OS is configured in FIPS-140 mode. NIAP testing was performed with the SR OS in FIPS-140 mode. When a node is in the FIPS-140 mode of operation in accordance with the FIPS-140-3 certification requirements, it validates the image using a DS before it uses the image for the upgrade. For more information, see [DS verification](#). If the DS of the image does not validate, the node does not upgrade to the new primary image.

3.10.5.1 Digital signature

The following figure shows the components of the SR OS digital signature (DS) environment.

Figure 7: SR OS DS environment



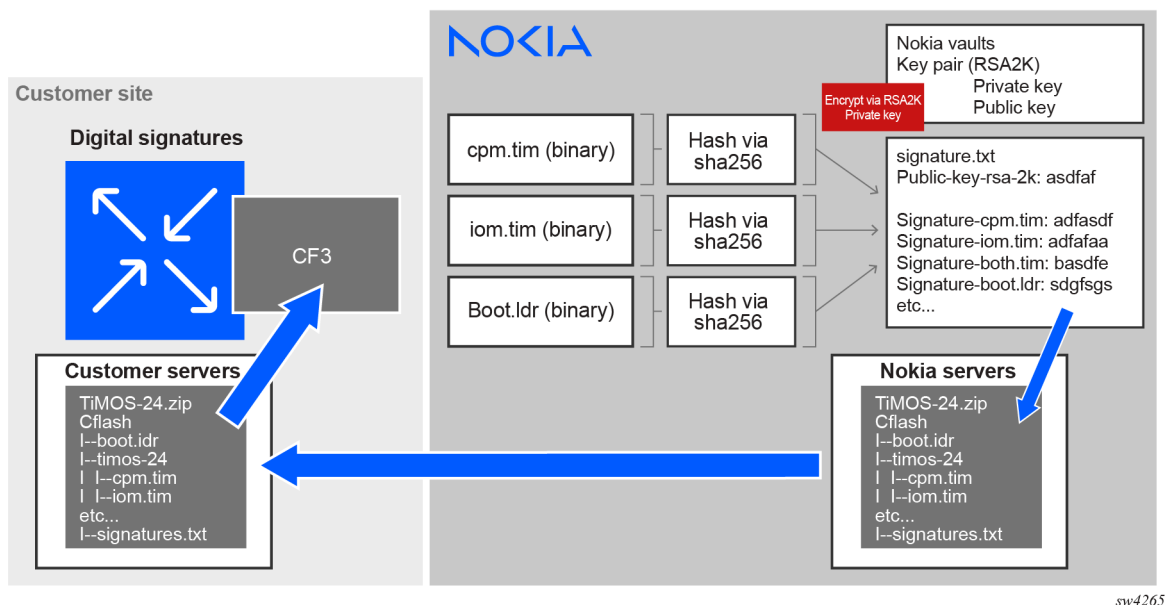
For the FIPS and NIAP validation of new SR OS images, the hash over different .tim files is calculated using SHA256 at Nokia premises. This hash is then encrypted using an RSA key pair (private key) with a key size of 2K. The encrypted hash is known as DS.

The compact flash (CF) image downloaded from the Nokia servers contains a file named `signatures.txt`, which contains the DSs of selected `.tim` files and the public key to validate these DSs.

3.10.5.2 DS verification

The following figure shows the DS verification components.

Figure 8: DS verification components



After the TiMOS `.zip` file with DS capabilities is downloaded from the SR OS servers, the `cflash` directory must be transferred to the system's CF. The user should ensure that the transfer is secure.



Note:

Nokia does not recommend, nor support upgrading to a new image using FTP. The image must be located on the CF.

After transferring the `cflash` to the CF, the system must be upgraded to the FIPS-140 image before configuring the **fips-140** flag in the BOF. After the upgrade, to use the system in FIPS-140 mode, the user must first run the **admin** command to validate the image DS. Then, after the DS is validated, the user must enable the FIPS mode by configuring the **fips-140** flag in the BOF. If the DS check fails, the system is not allowed to enter the FIPS-140 mode.

After upgrading to the FIPS image and entering the FIPS mode, any time the user wants to upgrade the system again, the DS of the `.tim` files must be validated first using the corresponding **admin** command. Use the following command to validate the DS before proceeding with the upgrade:

- **MD-CLI**

```
admin system security image-digital-signature validate software-image url
```

- **classic CLI**

```
admin system security image-digital-signature-validate url
```

The parameter of the preceding command specifies the URL of the software image location. If the DS validation passes, the URL is accepted and the image can be upgraded. If the validation does not pass, the primary image is rejected and the system is not upgraded. The system verifies the DS using SHA256 on the .tim files and compares this hash with the corresponding hash that is stored in the signatures.txt file. The hash in the signatures.txt file is decrypted using the public key. If the two hashes match, the .tim file is validated. This process is repeated for all selected .tim files that must be updated on the system.

3.10.5.3 DS support for files

The following files support the DS:

- cpm.tim
- both.tim
- iom.tim
- boot.ldr
- support.tim
- kernel.tim
- hypervisor.tim
- isa-aa.tim

3.10.5.4 DS file directories

The following directories and files must be in the same directory as the .tim image files.

- signatures.txt
- – cflash directory:
 - cflash/TiMOS-SR-version/signatures.txt
 - cflash/TiMOS-SR-version/signatures-isa-aa.txt

FIPS signatures files are not included for the following components:

- cflash-nl (No-LI version)
- SAR-Hm
- VSR-VMware, VSR-KVM, VSR-Container, VSIM-VMware, VSIM-KVM
- vm-nl (No-LI VM)
- pysros, PROTOBUF, MIBs, YANG

3.10.5.5 DS verification when configuring the fips-140 flag in the BOF

When the attempt is made to configure the **fips-140** flag in the BOF, the system verifies the DS of the selected .tim files. If the DS verification fails, the flag cannot be configured in the BOF, the system does not enter FIPS-140 mode, and a security log is generated.

To check the DS of the image, use the following command:

- **MD-CLI**

```
admin system security image-digital-signature validate software-image
```

- **classic CLI**

```
admin system security image-digital-signature-validate
```

The command caches the result of the DS verification. While the DS verification is in progress, the **fips-140** flag cannot be configured in the BOF. You must wait for the DS verification to be completed on the image. After the verification passes, the **fips-140** flag can be configured in the BOF and the system can be rebooted into FIPS mode.

3.10.5.6 DS verification for ISA-AA

When the system is in FIPS mode, ISA-AA is part of the DS verification. See the mechanism described in [DS verification when configuring the fips-140 flag in the BOF](#).

3.10.5.7 Checking a file DS

In FIPS mode, you can use the **file version url check** command to check the DS of the .tim file located at the specified URL. If the DS check fails, a message is displayed.

3.10.5.8 Upgrade procedure to a FIPS image

Prerequisites

Ensure that the `signatures.txt` file is synchronized between the CPMs. The `signatures.txt` file and the DS verification are only required in FIPS mode; they are not required in non-FIPS mode.

About this task

This procedure describes how to upgrade from a non-FIPS image to a FIPS image with DS for the first time.

Procedure

- Step 1.** SFTP the context of the software bundle to the active CF3. The `signatures.txt` file is copied to the active CF3 as part of this transfer.
- Step 2.** Perform an admin redundancy synchronization to ensure all files are synchronized between the active and standby CF3. An image that does not support the `signatures.txt` file and the DS verification does not synchronize the `signatures.txt` file to the standby CF3. (Such an

image has no understanding that the `signatures.txt` file is a critical component that must be synchronized between the two CPMs.)

- Step 3.** After upgrading to the FIPS release, the `signatures.txt` file will not be present on the standby CF3. Before enabling the FIPS-140 mode, perform an admin redundancy synchronization to copy the `signatures.txt` file to the standby CF3. After this step, both CPMs have the `signatures.txt` file and you can enable the FIPS-140 mode in the BOF.

3.11 Lawful Intercept

Lawful Intercept (LI) describes a process to intercept telecommunications by which law enforcement authorities can unobtrusively monitor voice and data communications to combat crime and terrorism with higher security standards of lawful intercept capabilities in accordance with local law and after following due process and receiving correct authorization from competent authorities. The interception capabilities are sought by various telecommunications providers.

As lawful interception is subject to national regulation, requirements vary from one country to another. -This implementation satisfies most national standard's requirements. LI is configurable for all service types.

3.12 Configuring the Boot Options File with CLI

This section provides information about configuring BOF parameters with CLI.

3.12.1 Basic BOF configuration

The parameters which specify the location of the image filename that the router tries to boot from and the configuration file are in the BOF.

The most basic BOF configuration should include the following:

- primary address
- primary image location
- primary configuration location

The following is an example of a basic BOF configuration.

Example: MD-CLI

```
[ ]
A:admin@node-2# admin show configuration bof
# TiMOS-B-22.2.R1 both/x86_64 Nokia 7750 SR Copyright (c) 2000-2022 Nokia.
# All rights reserved. All use subject to applicable license agreements.
# Built on Sat Feb 26 15:31:00 PST 2022 by builder in /builds/c/222B/R1/panos/main/sros
# Configuration format version 22.2 revision 0

# Generated 2022-03-07T17:08:41.4+00:00 by admin from Console

bof {
  configuration {
    primary-location "cf3:\config.cfg"
  }
  console {
```

```

        speed 115200
    }
    dns {
        domain "example.com"
        primary-server 10.251.72.68
        secondary-server 10.251.10.29
    }
    image {
        primary-location "cf3:\timos\"
    }
    li {
        local-save false
        separate false
    }
    license {
        primary-location "cf3:\\license.txt"
    }
    port "management" {
        autonegotiate true
    }
    router "management" {
        interface "management" {
            cpm active {
                ipv4 {
                    ip-address 192.168.189.52
                    prefix-length 24
                }
            }
            cpm standby {
            }
        }
        static-routes {
            route 192.168.0.0/16 {
                next-hop 192.168.189.1
            }
            route 172.16.0.0/16 {
                next-hop 192.168.189.1
            }
        }
    }
    system {
        fips-140 false
        persistent-indices true
    }
}

# Finished 2022-03-07T17:09:40.4+00:00

```

Example: classic CLI

```

A:node-2# show bof
=====
BOF (Memory)
=====
primary-image      ftp://*:~@192.168.15.1/./images/
primary-config     ftp://*:~@192.168.15.1/./images/dut-a.cfg
address            192.168.189.53/16 active
address            192.168.189.54/16 standby
static-route       192.0.2.0/24 next-hop 192.0.2.254
static-route       192.168.0.0/16 next-hop 192.0.2.254
static-route       192.168.10.10/16 next-hop 192.0.2.254
autonegotiate

```

```

duplex      full
speed       100
wait        3
persist     off
console-speed 115200
=====

```

3.12.2 Common configuration tasks

This sections describes basic system tasks that must be performed to configure BOF.

For more information about hardware installation and initial router connections, see the specific hardware installation guide.

3.12.2.1 Searching for the BOF

The BOF should be on the same drive (cf3:) as the bootstrap image file. If the system cannot load or cannot find the BOF, the system checks whether the boot sequence was manually interrupted.

3.12.2.2 Accessing the CLI

To access the CLI to configure the software for the first time, perform the following steps:

- When the power to the chassis is turned on, the SR OS software automatically begins the boot sequence.
- When the boot loader and BOF image and configuration files are successfully located, establish a router connection (console session).

3.12.2.3 Console connection

To establish a console connection, you need the following:

- a ASCII terminal or a PC running terminal emulation software set to the parameters shown in the following table
- a standard serial cable connector for connecting to an RS232 port (provides an RJ-45 connector)

Table 13: Console configuration parameter values

Parameter	Value
Baud rate	115200
Data bits	8
Parity	None
Stop bits	1
Flow control	None

3.12.2.4 Configuring BOF encryption

The BOF contents can be encrypted using AES-256.

Use the following command to enable BOF encryption:

- **MD-CLI**

```
bof configuration encrypt
```

- **classic CLI**

```
bof encrypt
```

3.12.2.5 Configuring the BOF interactive menu password

Access to the BOF interactive menu can be controlled using a password.

Use the following command to configure a BOF interactive menu password:

- **MD-CLI**

```
bof configuration password
```

- **classic CLI**

```
bof password
```

3.12.2.6 Configuring configuration file encryption

The configuration file contents can be encrypted using AES-256.

Use the following command to configure a configuration file encryption key:

- **MD-CLI**

```
bof configuration encryption-key
```

- **classic CLI**

```
bof encryption-key
```



Caution: The **admin save** command must be executed after changing configuration file encryption keys to ensure that a complete saved configuration file is saved with the new encryption key. After changing the encryption key, previously saved configuration files are no longer readable or loadable with the following command:

- **MD-CLI**

```
rollback
```

- **classic CLI**

```
admin rollback
```



Caution: Previously saved unencrypted configuration files, including MD-CLI incremental saved configuration files, are not automatically removed and must be removed manually.

3.12.3 Autoconfigure

When autoconfigure is enabled, the router performs a DHCP discovery or solicit (IPv6) to get the IP address of the out-of-band (OOB) management port.

The OOB management port can support a DHCP client for IPv4, IPv6, or dual stack. For dual stack, both IPv4 and IPv6 DHCP are configured. When the offer for either of the address families arrives, the management port is configured with the IP address in the offer. Eventually, both offers arrive and the management port is configured with both address families.

When a DHCP client is configured using autoconfigure, all image and license files should be placed and loaded from the CF. The configuration file could be loaded from the network, but Nokia recommends that the config file be on the CF as well. The configuration file is not loaded until the DHCP client offer is received and programmed successfully for the management port IP address, or the DHCP client timeout is expired.

3.12.3.1 Autoconfigure restrictions

When autoconfigure is enabled, a static IP address or static route cannot be configured in the BOF.

Similarly, a DNS server cannot be configured in the BOF, and only the DNS server provided by the DHCP offer can be used to resolve URLs.

The option 15 DNS domain name is not supported. The user can configure the DNS domain in the BOF so that the domain is not blocked when autoconfigure is used. Otherwise, the user must use the absolute URL with the hostname and domain included.

3.12.3.2 DHCP discovery of MAC addresses

When autoconfigure is used on redundant CPM chassis, the DHCP discovery uses the chassis MAC address. Only the active CPM performs a DHCP discovery and not the inactive CPM. When the offer arrives, the node uses that IP and the chassis MAC as addresses for management. Consequently, the inactive CPM is not reachable by the network, because it has no separate IP address. On activity switch, the inactive CPM inherits the active IP and chassis MAC.

For non-redundant CPMs, the management port MAC is used.



Note: The router must be rebooted when enabling autoconfigure for the first time to ensure that the CPM card uses the chassis MAC address.

3.12.3.3 IPv6 DUID

The SR OS supports type 2 DUID (link local), which is set to the chassis serial number. Type 3 (enterprise) is set to the chassis MAC address. Type 1 is not supported.

For type 2 DUID, the SR OS sends the Nokia Enterprise ID as the second byte of the DUID, followed by the chassis serial number. The first byte is the DUID type code. The chassis serial number starts with capital ASCII letters, which ensures that the serial number is unique as an application ID in the SR OS IPv6 DHCP application domain.

DUID type codes are as follows:

- DHCP6C_DUID_ENT_ID__IPSEC_IPV4ADDR - 1
- DHCP6C_DUID_ENT_ID__IPSEC_ASN1DN - 2
- DHCP6C_DUID_ENT_ID__IPSEC_FQDN - 3
- DHCP6C_DUID_ENT_ID__IPSEC_USER_FQDN - 4
- DHCP6C_DUID_ENT_ID__IPSEC_IPV6ADDR - 5
- DHCP6C_DUID_ENT_ID__IPSEC_ASN1GN - 6
- DHCP6C_DUID_ENT_ID__IPSEC_KEYID - 7
- DHCP6C_DUID_ENT_ID__WLAN_GW - 8
- DHCP6C_DUID_ENT_ID__AUTOBOOT - 9
- DHCP6C_DUID_ENT_ID__ZTP_BOF_AUTOP - Capital letters in ASCII

3.12.3.4 IPv6 DHCP RAs

An IPv6 DHCP offer does not have an IP prefix within the offer, unlike an IPv4 DHCP offer. The IPv6 prefix is usually obtained from the IPv6 Route Advertisement (RA) arriving from the upstream router. For ZTP, the SR OS is a host and assigns a /128 prefix to the IPv6 address obtained from the DHCP offer. In addition, the SR OS supports the installation of IPv6 default and static routes from upstream routers using the IPv6 RA. Multiple upstream routers can respond to a route solicitation with their own RA. The SR OS installs all the routes advertised by the RA. If the same route is advertised by multiple upstream routers (next hops), the SR OS installs the route with the highest preference. The SR OS does not support ECMP when the same route is advertised from multiple next hops by multiple RAs.

To ensure that all the RAs are obtained before the auto-provisioning process is started for IPv6, the SR OS follows the RFC 4861 recommendation that the host (in this case, the SR OS) send a minimum of three route solicitations. This is to ensure that if a route solicitation is lost, at least one of the three would reach the upstream routers. Each route solicitation is followed by a 4 s timeout. If the first route solicitation is sent at T_0 , the second is sent at T_0+4 s and the third is sent at T_0+8 s. The upstream routers must respond to the route solicitation within 0.5 s. This means that the SR OS has all of the RAs and the routes within 8.5 s of the first route solicitation. Therefore, the SR OS waits for a maximum of 9 s to receive all RAs.

If the DHCPv6 timeout is less than 9 s, the DHCPv6 timeout is honored even for the RA wait time. If the node has received a single RA and DHCP offer, the process is considered a success. However, it is possible that not all the RAs have arrived on the node because the node has waited less than 9 s.

3.12.4 System management tasks

This section describes system management tasks and commands.

3.12.4.1 System management commands in the classic CLI

For more information about the supported classic CLI commands, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*.

Use the following administrative commands to perform management tasks.

```
admin display-config
admin reboot
admin save
```

3.12.4.1.1 Viewing the current configuration

Use the following command to display the current configuration. The **detail** option displays all default values. The **index** option applies to the classic CLI and displays only the persistent indexes.

```
admin display-config
```

Use the following command to display context-level information.

```
info detail
```

The following example shows a configuration file for the 7750 SR.

Example

```
A:7750-3>admin# display-config
# TiMOS-B-22.2.R1 both/x86_64 Nokia 7750 SR Copyright (c) 2000-2022 Nokia.
# All rights reserved. All use subject to applicable license agreements.
# Built on Sat Feb 26 15:31:00 PST 2022 by builder in /builds/c/222B/R1/panos/main/sros
# Configuration format version 22.2 revision 0

# Generated MON MAR 07 16:50:13 2022 UTC

exit all
configure
#-----
echo "System Configuration"
#-----
system
  name "7750-3"
  contact "Fred Information Technology"
  location "Bldg.1-floor 2-Room 201"
  clli-code "abcdefg1234"
  coordinates "N 45 58 23, W 34 56 12"
  ccm 1
  exit
  snmp
  exit
  login-control
    idle-timeout 1440
```

```

        motd text "7750-3"
    exit
    time
        sntp
            shutdown
        exit
        zone UTC
    exit
    thresholds
        rmon
        exit
    exit
exit...
...
-----
echo "Redundancy Configuration"
#-----
    redundancy
        synchronize boot-env
    exit
...exit all

# Finished MON MAR 07 16:50:58 2022 UTC
A:7750#

```

3.12.4.1.2 Modifying and saving a configuration

If you modify a configuration file, the changes remain in effect only during the current power cycle unless a **save** command is executed. Changes are lost if the system is powered down or the router is rebooted without saving:

- Specify the file URL location to save the running configuration. If a destination is not specified, the files are saved to the location where the files were found for that boot sequence. The same configuration can be saved with different filenames to the same location or to different locations.
- The **detail** option adds the default parameters to the saved configuration.
- The **index** option forces a save of the index file.
- Changing the active and standby addresses without reboot standby CPM may cause a boot-env sync to fail.

Example: Saving the BOF configuration

```

A:node-2# bof save
Writing configuration to cf3:/bof.cfg ... OK
Completed.

```

Example: Saving the system configuration

```

A:node-2# admin save
Writing configuration to cf3:/config.cfg
Saving configuration ... OK
Completed.

```



Note: If the **persist** option is enabled and the **admin save** command is executed with an FTP path used as the file URL, two FTP sessions simultaneously open to the FTP server. The

FTP server must be configured to allow multiple sessions from the same login; otherwise, the configuration and index files will not be saved correctly.

3.12.4.1.3 Deleting BOF parameters

You can delete specific BOF parameters. The changes remain in effect only during the current power cycle unless a **save** command is executed. Changes are lost if the system is powered down or the router is rebooted without saving.

Deleting a BOF address entry is not allowed from a remote session.

Use the **no** form of following commands to remove and save BOF configuration parameters.

```
bof address <ip-prefix/ip-prefix-length> [<cpm>]
bof autonegotiate
bof console-speed <baud-rate>
bof dns-domain <dns-name>
bof duplex <duplex>
bof ess-system-type
bof ip-mtu <octets>
bof li-local-save
bof li-separate
bof license-file <file-url>
bof persist {on|off}
bof primary-config <file-url>
bof primary-dns <ip-address>
bof primary-image <file-url>
bof save [<cflash-id>]
bof secondary-config <file-url>
bof secondary-dns <ip-address>
bof secondary-image <file-url>
bof speed <speed>
bof static-route <ip-prefix/ip-prefix-length> next-hop <ip-address>
bof system-base-mac <mac-address>
bof system-profile <profile>
bof tertiary-config <file-url>
bof tertiary-dns <ip-address>
bof tertiary-image <file-url>
bof wait <seconds>
```

Example: Saving BOF configuration parameters

```
A:node-2# bof save
Writing configuration to cf3:/bof.cfg ... OK
Completed.
```

3.12.4.1.4 Saving a configuration to a different filename

Save the current configuration with a unique filename to have additional backup copies and to edit parameters with a text editor. You can save your current configuration to an ASCII file.

The following example shows saving a configuration to a different location.

Example: Using the admin save command

```
A:node-2>admin save cf3:\testABC.cfg
Writing configuration to cf3:\testABC.cfg
```

```
Saving configuration ... OK  
Completed.
```

3.12.4.1.5 Rebooting

When an **admin>reboot** command is issued, routers with redundant CPM are rebooted as well as the XMAs, XCMs, and IOMs. Changes are lost unless the configuration is saved. Use the **admin>save file-url** command to save the current configuration. If no command line options are specified, the user is prompted to confirm the reboot operation.

The following example shows a reboot.

Example

```
A:node-2>admin# reboot  
Are you sure you want to reboot (y/n)? y
```

3.12.4.1.6 Setting the MTU value for the management port

You can configure the MTU for IP packets transmitted out the interface of the management router instance associated with the management port. The command applies to the SR OS, however, it does not necessarily apply during the boot loader processing.

The operational MTU for the port is set to the lesser of the values configured with the **ip-mtu** command and the management port MTU. For example, with the port MTU fixed at 1514 bytes and an Ethernet header size of 14 bytes, the MTU of the management port is 1500 bytes (the default operational IP MTU).

If the interface supports IPv6 packets, the command value must be set to 1280 or higher, in accordance with RFC 2460, *Internet Protocol, Version 6 (IPv6) Specification*. Use the following command to configure the MTU for IP packets transmitted out the interface of the management router instance.

```
bof ip-mtu
```

3.12.4.2 System management commands in the MD-CLI

For more information about the supported MD-CLI commands, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI Command Reference Guide*.

Use the following administrative commands to perform management tasks.

```
admin show configuration  
admin reboot  
admin save
```

3.12.4.2.1 Viewing the current configuration

The **admin show configuration** command displays the current configuration for a specified configuration region (the default region is **configure**). The **booted** and **cflash-id** options are valid only for the **bof** configuration region.

Example: Detailed show output of BOF configuration file

The following example shows a BOF configuration file with the **detail** option to display all default and unconfigured values and the **units** option to show units where applicable.

```
A:admin@node-2# admin show configuration bof units detail
# TiMOS-B-22.10.R1 both/x86_64 Nokia 7750 SR Copyright (c) 2000-2022 Nokia.
# All rights reserved. All use subject to applicable license agreements.
# Built on Sun Oct 30 14:49:55 PDT 2022 by builder in /builds/c/2210B/R1/panos/main/sros
# Configuration format version 22.10 revision 0
# Generated 2023-01-12T16:57:57.6-05:00 by admin from Console

bof {
  configuration {
    primary-location "cf3:\config.cfg"
    ## secondary-location
    ## tertiary-location
  }
  console {
    speed 115200 bps
    wait-time 3 seconds
  }
  dns {
    ## domain
    ## primary-server
    ## secondary-server
    ## tertiary-server
  }
  image {
    primary-location "cf3:\timos\"
    ## secondary-location
    ## tertiary-location
  }
  li {
    local-save false
    separate false
  }
  license {
    primary-location "cf3:\license.txt"
  }
  port "management" {
    autonegotiate true
    duplex full
    speed 100 megabps
  }
  router "management" {
    interface "management" {
      ## ip-mtu
      cpm active {
        ipv4 {
          ip-address 192.168.189.52
          prefix-length 24
        }
        ## ipv6
      }
      cpm standby {
        ## ipv4
        ## ipv6
      }
    }
    static-routes {
      route 192.168.0.0/16 {
        next-hop 192.168.189.1
      }
    }
  }
}
```

```

    }
    route 172.16.0.0/16 {
        next-hop 192.168.189.1
    }
}
}
system {
    ## base-mac-address
    fips-140 false
    ## gateway-role
    persistent-indices true
    ## profile
}
}
# Finished 2023-01-12T16:57:57.6-05:00

```

3.12.4.2.2 Modifying BOF parameters

BOF parameters can be modified via a BOF session in exclusive, private, or read-only configuration mode in the MD-CLI. The same configuration management commands that are available in the configure region are available in the bof region.



Note: Changing the active and standby addresses without rebooting the standby CPM may cause synchronization with the **boot-env** option to fail.
Deleting a BOF address entry is not allowed from a remote session.

Example

```

[/]
A:admin@node-2# bof exclusive
INFO: CLI #2060: Entering exclusive configuration mode
INFO: CLI #2061: Uncommitted changes are discarded on configuration mode exit

[ex:/bof]
A:admin@node-2# ?

configuration      + Enter the configuration context
console            + Enter the console context
dns                + Enter the dns context
image              + Enter the image context
li                 + Enter the li context
license            + Enter the license context
port               + Enter the port list instance
router             + Enter the router list instance
system             + Enter the system context

```

See the *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI Command Reference Guide* and the *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI User Guide* for more information.

3.12.4.2.3 Saving a configuration

Configuration changes are lost if the system is powered down or the router is rebooted before the changes are saved. If the URL location to save the running configuration is not specified, the files are saved to the location where the files were found for the boot sequence. The same configuration can be saved with different filenames to the same location or to different locations.

Changing the active and standby addresses without rebooting the standby CPM may cause synchronization with the **boot-env** option to fail.

The following command saves the running configuration for the configure region. If no URL is specified, the configuration is saved to the `config.cfg` file.

```
admin save
```

Example: Configuration save output

```
[admin]
A:admin@node-2# save
Writing configuration to cf3:\config.cfg
Saving configuration OK
Completed.
```

The BOF configuration is saved to `cf3:\bof.cfg` with every **commit** command.

Example

The BOF configuration can be manually saved to a backup file on a server or to a different location, as shown in the following example.

```
[]
A:admin@node-2# admin save bof ftp://10.9.236.68/backup/node-2/bof.cfg
Writing configuration to ftp://10.9.236.68/backup/node-2/bof.cfg OK
Completed.
```

Example

The following example saves the BOF configuration to a file, named `testbof.cfg` on `cf3:`.

```
[]
A:admin@node-2# admin save bof testbof.cfg
Writing configuration to cf3:\testbof.cfg OK
Completed.
```



Note: The BOF configuration file is saved in classic CLI format.

3.12.4.2.4 Rebooting

When a **reboot** command is issued, routers with redundant CPM are rebooted as well as the XMA, XCMs, and IOMs. If the **now** option is not specified, the user is prompted to confirm the reboot operation.

3.12.4.2.5 Setting the MTU value for the management port

The following command configures the MTU for IP packets transmitted out the interface of the management router instance associated with the management port.

```
bof router "management" interface "management" ip-mtu
```

The command applies to the SR OS but does not necessarily apply during the boot loader processing.

The operational MTU for the port is set to the lesser of the values configured with the **ip-mtu** command and the management port MTU. For example, with the port MTU fixed at 1514 bytes and an Ethernet header size of 14 bytes, the MTU of the management port is 1500 bytes (the default operational IP MTU).

If the interface supports IPv6 packets, the command value must be set to 1280 or higher, in accordance with RFC 2460 *Internet Protocol, Version 6 (IPv6) Specification*.

Example

```
[ex:bof]
A:admin@node-2# router "management" interface "management" ip-mtu ?

ip-mtu <number>
<number> - <512..9786> - bytes

Interface IP MTU

Note: The new value of this element takes effect when the candidate is
committed.
```

3.13 Anti-theft



Note: This feature is supported only on the 7750 SR-2s, 7750 SR-2se, 7750 SR-7s, and 7750 SR-14s.

To discourage theft, the SR OS includes an anti-theft feature that prevents the system from being redeployed in other networks when it is powered down and removed from the original network. This feature also ensures that the router cannot be deployed in any network other than the intended network.

Before activating the anti-theft feature, the user must configure the OS security password. The password length must be between 12 and 32 characters, and must not contain any non-printable characters, spaces, or double quotes (").

Use the following command to configure the OS security password.

```
admin system security os-security set-password
```

Users can reconfigure the OS security password using the **set-password** command. When prompted, the user must enter the **current-password** followed by the **new-password**. Use the **activate-password** command to activate the newly entered password. The system immediately reboots and encrypts the BOF and configuration files with the new password.



Note:

- Nokia recommends using a random password generator to ensure a strong and secure password.
- After the OS security password is configured, the router reboots to save the password.
- The anti-theft feature does not need to be deactivated to reconfigure the OS security password.

After the OS security password is set, the anti-theft feature can be enabled on a per-CPM basis. When anti-theft is enabled, after a router reboot, the router blocks the CLI configuration until the administrative user logs on to the router and enters the OS security password to unlock anti-theft. If the entered OS security password is incorrect, the CLI configuration remains blocked.

Use the following command to activate the anti-theft feature for a CPM.

```
admin system security os-security anti-theft activate
```

When prompted, the user must enter their OS security password to enable the anti-theft feature. For each incorrect password attempt, the initial unsuccessful authentication wait period is doubled. This wait period is common for all users. Entering the correct password resets the wait period to the initial value for all users.

Example: Enable anti-theft (MD-CLI)

```
[/admin system security os-security]
A:admin@node-2# set-password
New password:
Re-enter new password:

A:admin@node-2# admin system security os-security anti-theft activate card "A"
force password
```

Example: Enable anti-theft (classic CLI)

```
*A:node-2>admin>system>security>os-security#
set-password
New password:
Re-enter new password:

A:node-2>admin>system>security>os-security>anti-theft#
activate card "A"
Current password:
```

When the anti-theft feature is enabled and the router reboots, all CLI configuration is blocked for the CPM after booting up, until the user unlocks the OS. Use the following command to unlock the OS.

```
admin system security os-security anti-theft unlock
```

When prompted, the user must enter the OS security password to unlock the router. For each incorrect password attempt, the initial unsuccessful authentication wait period is doubled. This wait period is common for all users. When the correct password is entered, the router resumes normal operation and allows configuration.

Use the following command to disable the anti-theft feature. When prompted, the user must enter the OS security password to disable anti-theft.

```
admin system security os-security anti-theft deactivate
```

Use the following command to remove the OS security password.

```
admin system security os-security remove-password
```

3.13.1 Node behavior when the anti-theft password is set



Note: This feature is supported only on the 7750 SR-2s, 7750 SR-2se, 7750 SR-7s, and 7750 SR-14s.

If it reboots when in the anti-theft mode, the node performs the following actions in the listed order upon booting up:

1. The node can be accessed using the console connection at this point. Remote connections using SSH/Telnet, SFTP/SCP, SNMP, NETCONF, and so on, are not possible until after the BOF and configuration are successfully loaded.
2. The node loads the configuration only if the BOF and the configuration file is decrypted successfully.
The configuration file is loaded, and a remote connection is allowed for management and entering the OS security password to put the router into a normal mode of operation.
It is assumed that the current configuration file is not useful to the organization that stole the router.
3. After connecting to the node, configuration using any management interface is not allowed.

Example: Anti-theft node error

```
*A:node-2#ssh admin@192.168.239.179
KANVMPLM6 - Dut-B
admin@192.168.239.179's password:
*A:node-2# configure
ERROR: "MINOR: MGMT_CORE #2XXX: Operation not permitted - OS lockdown activated - enter OS
security password"
```

4. The user can enter the OS security password after the reboot by using interactive management sessions, the corresponding NETCONF YANG, and so on.
5. After the OS security password is set, the node starts normal operation and allows configuration.

3.13.2 BOF and configuration file behavior



Note: This feature is supported only on the 7750 SR-2s, 7750 SR-2se, 7750 SR-7s, and 7750 SR-14s.

When the anti-theft feature is enabled, the BOF and configure files are saved in an encrypted format, and the router reboots. The router loads the configuration only if the BOF and configuration files are successfully decrypted.

If there is a failure in decryption and execution of the BOF, the router reboots. If the configuration file is not decrypted correctly, the router does not load the configuration file. This ensures that the router does not accept any configuration files that are loaded from unknown sources.

BOF and configuration files can be decrypted and stored as a single backup file. They are not saved as the active configuration or BOF, and do not overwrite the existing encrypted files.

All encrypted configuration files can be saved as backup configuration in cleartext using the **cleartext** option in the following command:

```
admin save
```

All encrypted BOF files can be saved as cleartext using the **cleartext** option in the following command.

- **MD-CLI**

```
admin save bof
```

- **classic CLI**

```
bof save
```

When anti-theft is enabled on the system and the **cleartext** option is used to save in cleartext, the user must enter the anti-theft OS security password or the BOF encryption key to perform the save operation.



Note: The **bof save cleartext** command does not require a password when **bof encryption** is enabled.

If the BOF and configuration files are already in cleartext and are saved using the **cleartext** option, there is no effect on the **admin save** command.



Note: When anti-theft is enabled, Nokia recommends transferring the files and then deleting the cleartext copies as soon as possible to prevent insecure file leakage.

3.13.3 Management interface interaction in anti-theft mode



Note: This feature is supported only on the 7750 SR-2s, 7750 SR-2se, 7750 SR-7s, and 7750 SR-14s.

In the anti-theft mode, the **configure** and **bof** branches are blocked in both the classic CLI and MD-CLI after a reboot of the router. The branches become available again only if the OS security password is entered.

The following branches are allowed for debugging the router in the anti-theft mode:

- **show**
- **tools**
- **debug**
- **clear**

Commands in the **admin** branch, with the following exceptions, can be used in the anti-theft mode:

MD-CLI

```
admin clear security lockdown all
admin clear security lockdown user
global-commands rollback
admin show configuration
admin system security hash-control custom-hash key
admin system security system-password admin-password
admin system security pki import type
admin system security pki crl-update ca-profile
admin save clear-text
admin system security os-security
admin system management-interface commit confirmed
```

classic CLI

```
admin clear lockdown all
admin clear lockdown user
admin rollback
admin display-config
admin system security hash-control custom-hash key
```

```
admin system security system-password admin-password
admin system security system-password dynsvc-password
admin certificate import type
admin certificate crl-update ca
admin save clear-text
admin system security os-security
```

The SCP, SFTP, and FTP are accessible in anti-theft mode. Actions such as removing and copying files are allowed.



Note: Deleting the **configure** or **bof** branches will cause encryption or authentication failure, and the router will not boot up.

The **exec** command is only allowed for non-configuration command contexts (for example, **show**, **tools**, **debug**, **clear**). Any configuration command in the script is blocked. The router stops executing the script as soon as it encounters a configuration command in the script. The configuration commands are blocked until the following command is executed.

```
admin system security os-security anti-theft unlock
```

4 Debug configuration

The **debug** configuration commands enable detailed debugging information for various protocols.

4.1 Debug configuration in the classic CLI

The **debug** commands in the classic CLI are available by entering the **debug** configuration context.

Debugging configuration is not persistent across CPM switchovers or router reboots. The **show debug** command displays debugging information.

Example: Show debug information

```
A:node-2# show debug
debug
  system
  netconf info
  exit
exit
```

The **admin debug-save** command saves the debugging configuration to `config.dbg` at the BOF **primary-config** location if a URL is not specified. Debug configuration files are not synchronized to the standby CPM in a system with redundant CPMs.

Example: Save debug configuration

```
A:node-2# admin debug-save
Writing configuration to cf3:\config.dbg
Saving configuration OK
Completed.
```

For a description of individual **debug** commands, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*.

4.1.1 Logging debug events in the classic CLI

The following is an example configuration for debug events that are stored in destination CLI log identifier 7. The log entries wrap at 50 entries (the configured value of **cli**).

Example: Configuration for stored debug events

```
A:node-2>config>log# log-id 7
A:node-2>config>log>log-id$ from debug-trace
A:node-2>config>log>log-id$ to cli 50
A:node-2>config>log>log-id$ info
-----
      from debug-trace
      to cli 50
      no shutdown
-----
```

After the log is configured, execute the following **tools** command in the CLI session that is intended to display output of the debug events. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR Clear, Monitor, Show, and Tools CLI Command Reference Guide* for more information about the **tools** command.

Example: Subscribe to debug log output to the CLI session

```
A:node-2# tools perform log subscribe-to log-id 7
```

Debug events can be displayed using the **show log** command and cleared using the **clear log** command.

Example: Unsubscribe from debug log output to the CLI session

```
A:node-2# show log log-id 7
=====
Event Log 7 log-name 7
=====
Description : (Not Specified)
Log contents [size=50 next event=2 (not wrapped)]

---snip---

A:node-2# clear log log-id 7
```

The following is an example of terminating the output of the logs to the CLI session using the **unsubscribe-from** command.

Example: Unsubscribe from debug log output to the CLI session

```
A:node-2# tools perform log unsubscribe-from log-id 7
```

4.2 Debug configuration in the MD-CLI

The **debug** commands in the MD-CLI are available in an exclusive, private, or read-only session using the explicit or implicit configuration mode. The same configuration management commands that are available in the configure region are available in the debug region.

Debugging configuration is not persistent across router reboots. Use the following command to display debugging information.

```
admin show configuration debug
```

The command displays debugging information and supports all configuration display formats, datastores, and output formats that are supported for other regions.

Example: Display debug information

```
[/]
A:admin@node-2# admin show configuration debug
# TiMOS-B-22.2.R1 both/x86_64 Nokia 7750 SR Copyright (c) 2000-2022 Nokia.
# All rights reserved. All use subject to applicable license agreements.
# Built on Sat Feb 26 15:31:00 PST 2022 by builder in /builds/c/222B/R1/panos/main/sros
# Configuration format version 22.2 revision 0

# Generated 2022-03-07T16:51:54.1+00:00 by admin from Console
debug {
```

```

system {
    management-interface {
        netconf info
    }
}

# Finished 2022-03-07T16:51:54.1+00:00

```

The **admin save debug** command saves the debugging configuration to `debug.cfg` at the following location if a URL is not specified. Debug configuration files are not synchronized to the standby CPM in a system with redundant CPMs.

```
bof configuration primary-location
```

Example: Save debug configuration

```

[/]
A:admin@node-2# admin save debug
Writing configuration to cf3:\debug.cfg
Saving configuration OK
Completed.

```

For descriptions of individual **debug** commands, see the *7450 ESS*, *7750 SR*, *7950 XRS*, and *VSR MD-CLI Command Reference Guide*.

4.2.1 Logging debug events in the MD-CLI

Example: Configuring a CLI log for debug events

The following is an example of a configuration for debug events that are stored in destination CLI log identifier 7. The log entries wrap at 50 entries (the configured value of **max-entries**).

```

*(ex)[configure log]
A:admin@node-2# log-id 7

*(ex)[configure log log-id "7"]
A:admin@node-2# source debug

*(ex)[configure log log-id "7"]
A:admin@node-2# destination cli max-entries 50

*(ex)[configure log log-id "7"]
A:admin@node-2# info
    source {
        debug true
    }
    destination {
        cli {
            max-entries 50
        }
    }

```

Example: Subscribing to a CLI log

After the **commit** command is issued to include the log in the running configuration, the following **tools** command can be executed in the CLI session that is intended to display output of the debug

events. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR Clear, Monitor, Show, and Tools CLI Command Reference Guide* for more information about the **tools** command.

```
[/]
A:admin@node-2# tools perform log subscribe-to log-id 7
```

Example: Displaying and clearing debug events

Debug events can be displayed using the **show log** command and cleared using the **clear log** command.

```
[/]
A:admin@node-2# show log log-id 7
=====
Event Log 7 log-name 7
=====
Description : (Not Specified)
Log contents [size=50 next event=2 (not wrapped)]
...

[/]
A:admin@node-2# clear log log-id 7
```

4.3 Debug configuration in mixed and model-driven mode

When debugging is configured in mixed or model-driven management mode, the following usage guidelines apply.

If the commands are available in the MD-CLI, the MD-CLI commands must be used to configure debugging:

- The classic CLI cannot be used.
- Debug configuration commands entered in the MD-CLI are only displayed in the MD-CLI **info** and in the following command output.

```
admin show configuration debug
```

- Debug configuration entered in the MD-CLI can be saved to `debug.cfg` or a file URL with the **admin save debug** command.
- Debug configuration commands entered in the MD-CLI are not displayed in the classic CLI **show debug** output.



Note: References must be configured with the MD-CLI names, not the classic CLI IDs. For example, the MD-CLI VPRN service name, not the classic CLI VPRN service ID.

If the commands are not available in the MD-CLI, the classic CLI must be used to configure debugging:

- The MD-CLI cannot be used.
- Debug configuration commands entered in the classic CLI are only displayed in the classic CLI **show debug** output.

- Debug configuration entered in the classic CLI can be saved to `config.dbg` or a file URL with the **admin debug-save** command.
- Debug configuration commands entered in the classic CLI are not displayed in the MD-CLI **info** or in the following command output.

```
admin show configuration debug
```

The user must manually remove the classic and model-driven debug configuration before changing the management interface configuration mode from model-driven to mixed mode. The system automatically removes the classic and model-driven debug configuration during all other mode switches.

5 Secure Boot



Note: All platforms support Secure Boot except the 7950 XRS-40 and VSR.

SR OS Secure Boot ensures that the system executes only trusted software that originates from Nokia IP Routing.

At every boot of the control card, each step in the boot process verifies the digital signature of the next software element to boot for integrity and authenticity up to the SR OS operating system images. This boot sequence forms the Secure Boot chain of trust.

Software image signatures use RSA-4096 keys and SHA-384 hashes.

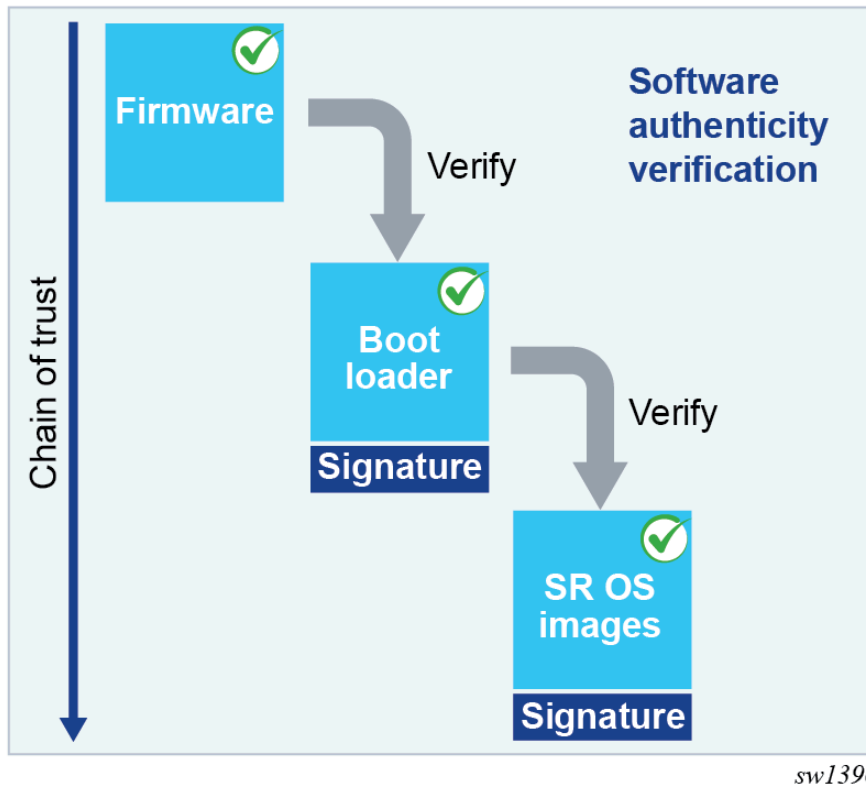
The Secure Boot chain is rooted in the platform CPM firmware, which is based on UEFI specifications. As such, the Nokia Platform Key, Key Exchange Key, and allowed and disallowed databases are provisioned when the user activates Secure Boot to perform the required signature verification.

Firmware updates are also digitally signed and verified using the same process. The existing firmware verifies the signature of a firmware update at boot time, before the firmware update can proceed.

5.1 Secure Boot chain

The following figure displays the Secure Boot chain of trust for SR OS platforms.

Figure 9: Secure boot chain of trust



The software images part of the Secure Boot chain varies among SR OS platforms. This list of software images per platform is described in [Storage card content](#) and includes the Boot Loader, `boot.ldr` or `/EFI/BOOT/` and installer images, and the SR OS `*.tim` software images.

5.2 Activate Secure Boot

For systems that require Secure Boot activation (systems for which Secure Boot is not enabled by default), Secure Boot can be enabled, per CPM card, by providing the card slot, card serial number, and confirmation code.

Use the following command to activate Secure Boot:

- **MD-CLI**

```
admin system security secure-boot activate card A
admin system security secure-boot activate confirmation-code secure-boot-permanent
admin system security secure-boot activate serial-number NS123456789
```

- **classic CLI**

```
admin system security secure-boot activate card "A" serial-number NS123456789 confirmation-
code secure-boot-permanent
```

The following example shows the warning messages and a prompt for proceeding with Secure Boot activation.

Example

```
WARNING: CLI This operation will permanently activate secure boot on card A and cannot be
reversed.
WARNING: CLI After activation, the system will only accept digitally signed software and
will not boot using un-signed software.
WARNING: CLI This operation will immediately reset card A.
WARNING: CLI Configuration and/or Boot options may have changed since the last save.
Are you sure you want to continue (y/n)?
```

The system requires the card serial number and Secure Boot confirmation code to prevent unintentionally activating Secure Boot in the network. The confirmation code is *secure-boot-permanent*.

The Secure Boot **activate** command verifies that the BOF primary image uses the same software release as the currently running software and automatically reboots the designated CPM card if the software release matches. Otherwise, an error is generated in the CLI.



Note: The system also verifies the boot.ldr version against the running software version on applicable platforms. These verifications ensure that the entire boot chain up to the primary image supports Secure Boot before activating Secure Boot and rebooting the CPM.



WARNING: After Secure Boot is activated on a CPM, the capability is permanently enabled and cannot be disabled. The CPM permanently refuses to execute unsigned software for security reasons. As a result, a user cannot downgrade to a software release published before the release that introduced Secure Boot for a specific platform. For example, 7750 SR-1s Secure Boot support was introduced in software Release 23.7.R1. After activating Secure Boot on this platform the system cannot be downgraded to software releases that predate Release 23.7.R1.

5.2.1 Satellite

The 7750 SR host can be used to control 7250 IXR satellite Secure Boot activation, update, and revocation.

To activate Secure Boot, use the command in the following example based on Satellite 1.

Example: Secure Boot configuration with satellite

```
admin system security secure-boot activate ethernet-satellite 1 serial-number NS252461306
confirmation-code secure-boot-permanent
```

The command requires the satellite number, system serial number, and Secure Boot confirmation code to prevent unintentionally activating Secure Boot in the network. The confirmation code is **secure-boot-permanent**.



WARNING: After Secure Boot is activated on a satellite, the capability is permanently enabled and cannot be disabled. The system permanently refuses to execute unsigned software for security reasons. As a result, a user cannot downgrade to a software release published before the release that introduced Secure Boot for a specific platform.

Secure Boot key updates and revocation use the commands in the following example by providing the satellite software repository name, when required.

Example: Secure Boot update and revoke commands

```
admin system security secure-boot update-key ethernet-satellite 1 serial-number
NS252461306 confirmation-code secure-boot-permanent software-repository sat_ixr_upg
admin system security secure-boot revoke-key ethernet-satellite 1 serial-number
NS252461306 confirmation-code secure-boot-permanent
```

State information can be obtained using the commands in the following example.

Example: State information

```
show system satellite eth-sat 1
state satellite ethernet-satellite 1
perform system security secure-boot show uefi-variables ethernet-satellite 1
tools dump system security secure-boot uefi-vars eth-sat 1
```

5.3 Operational commands and logs

This section describes the following:

- Secure Boot state
- software update process
- update Secure Boot variables

5.3.1 Secure Boot state

Use the following command to display Secure Boot state information, including the Secure Boot status and UEFI variables status per CPM.

```
show card A detail
```

Output example

```
Hardware Data
Secure boot status      : enabled
UEFI variables status   : ok
```

where

- Secure Boot status — indicates if Secure Boot is enabled or disabled
- UEFI variables status — indicates if Secure Boot variables need updating

At each boot, the system records in the security log whether Secure Boot is enabled or disabled for each CPM. The following is an example of such a log message.

```
24 2023/05/17 06:09:03.140 EDT MAJOR: SECURITY #2241 Base Card A
"CPM A has booted with a secure-boot status of enabled"
```

Use the following command to obtain Secure Boot UEFI variables for each CPM card:

- **MD-CLI**

```
perform system security secure-boot show uefi-variables card
```

- **classic CLI**

```
tools dump system security secure-boot uefi-var card
```

The command displays the following x509 certificates and SHA-256 hash UEFI variables:

- Platform Key (PK)
- Key Exchange Key (KEK)
- Allowed Database (DB)
- Disallowed Database (DBx)

5.3.2 Software update

After enabling Secure Boot, and before upgrading to a new software release, the user must verify that the new software image is properly signed. This additional verification is required on systems with Secure Boot enabled because the system only boots Nokia-signed software images and does not boot unsigned or improperly signed images.

Use the following command to validate the signature of the TiMOS *.tim images contained in the **software-image** *url* location referenced in the command. This verification includes `cpm.tim`, `iom.tim`, `support.tim`, `both.tim`, `kernel.tim`, as well as the `boot.ldr` if present in the CF3 directory.

```
admin system security secure-boot validate software-image url
```

5.3.3 Update Secure Boot variables

Use the following commands to configure Secure Boot UEFI key updates and revocation.

```
admin system security secure-boot update-key  
admin system security secure-boot revoke-key
```

6 System management

This chapter provides information about configuring basic system management parameters.

6.1 System management commands

System management commands allow you to configure basic system management functions, such as the system name, contact, router location and coordinates, as well as time zones, Network Time Protocol (NTP), Simple Network Time Protocol (SNTP) synchronization, Precision Time Protocol (PTP), and CRON.

6.1.1 System information

This section describes the system information components.

6.1.1.1 Name

You can configure a name for the system device. The name is used in the prompt string. Only one system name can be configured. If multiple system names are configured the last one encountered overwrites the previous entry. Use the following command to configure the system name.

```
configure system name
```

6.1.1.2 Contact

Use the **contact** command to specify the name of a system administrator, IT staff member, or other administrative entity. Use the following command to configure the system contact.

```
configure system contact
```

6.1.1.3 Location

Use the **location** command to specify the system location of the device. For example, enter the city, building address, floor, room number, and so on, where the router is located. Use the following command to configure the system location.

```
configure system location
```

6.1.1.4 Coordinates

You can optionally configure the GPS location of the device. If the string contains special characters (#, \$, spaces, and so on), the entire string must be enclosed within double quotes. Use the following command to configure the system coordinates.

```
configure system coordinates
```

6.1.1.5 Naming objects

Avoid configuring named objects with a name that starts with "_tmnx_" and with "_" in general.

6.1.1.6 Common language location identifier

A Common Language Location Identifier (CLLI) for the device is an 11-character standardized code string that uniquely identifies the geographic location of places and specific functional categories of equipment unique to the telecommunications industry. The CLLI code is stored in the Nokia Chassis MIB tmnxChassisCLLIcode object.

The CLLI code can be any ASCII printable text string of up to 11 characters.

6.1.1.7 DNS security extensions

DNS Security (DNSSEC) Extensions are now implemented in the SR OS, allowing users to configure DNS behavior of the router to evaluate whether the Authenticated Data bit was set in the response received from the recursive name server and to trust the response, or ignore it.

6.1.2 System time

Routers are equipped with a real-time system clock for timekeeping purposes. When set, the system clock always operates on Coordinated Universal Time (UTC), but the software has options for local time translation as well as system clock synchronization.

6.1.2.1 Time zones

Setting a time zone in SR OS allows for times to be displayed in the local time rather instead of UTC. SR OS has both user-defined and system-defined time zones.

A user-defined time zone has a user-assigned name of up to four printable ASCII characters in length and is unique from the system-defined time zones. For user-defined time zones, the offset from UTC is configured as well as any summer time adjustment for the time zone.

SR OS includes multiple commands to control the presentation of times in either UTC or local time zone format. For a CLI session, the environment variable time-display may be set to indicate UTC or local time zone. This setting only affects time strings shown during that specific CLI session. A global setting of the

following command can be used to control time strings for objects with larger scope than a single CLI session.

```
configure system time prefer-local-time
```

Time strings include the following:

- log filenames and log header information
- times in rollback information
- times in rollback and configuration files header information
- times related to CRON scripts
- times related to CRON scripts
- times in the event handler system
- times in NETCONF and gRPC date-and-time leafs

A separate control per log file controls the format of the time strings on the event recorded into the logs (separate from the log filename and header information). Use the following command to set these time strings.

```
configure log log-id time-format
```

The SR OS system-defined time zones are listed in the following table, which includes both time zones with and without daylight saving (summer) time adjustment.

Table 14: System-defined time zones and UTC offsets

Acronym	Time zone name	UTC offset
Europe:		
GMT	Greenwich Mean Time	UTC
BST	British Summer Time	UTC +1
IST	Irish Summer Time	UTC +1*
WET	Western Europe Time	UTC
WEST	Western Europe Summer Time	UTC +1
CET	Central Europe Time	UTC +1
CEST	Central Europe Summer Time	UTC +2
EET	Eastern Europe Time	UTC +2
EEST	Eastern Europe Summer Time	UTC +3
MSK	Moscow Time	UTC +3
MSD	Moscow Summer Time	UTC +4
US and Canada:		

Acronym	Time zone name	UTC offset
AST	Atlantic Standard Time	UTC -4
ADT	Atlantic Daylight Time	UTC -3
EST	Eastern Standard Time	UTC -5
EDT	Eastern Daylight Saving Time	UTC -4
ET	Eastern Time	Either as EST or EDT, depending on place and time of year
CST	Central Standard Time	UTC -6
CDT	Central Daylight Saving Time	UTC -5
CT	Central Time	Either as CST or CDT, depending on place and time of year
MST	Mountain Standard Time	UTC -7
MDT	Mountain Daylight Saving Time	UTC -6
MT	Mountain Time	Either as MST or MDT, depending on place and time of year
PST	Pacific Standard Time	UTC -8
PDT	Pacific Daylight Saving Time	UTC -7
PT	Pacific Time	Either as PST or PDT, depending on place and time of year
HST	Hawaiian Standard Time	UTC -10
AKST	Alaska Standard Time	UTC -9
AKDT	Alaska Standard Daylight Saving Time	UTC -8
Australia and New Zealand:		
AWST	Western Standard Time (for example, Perth)	UTC +8 hours
ACST	Central Standard Time (for example, Darwin)	UTC +9.5 hours
AEST	Eastern Standard/Summer Time (for example, Canberra)	UTC +10 hours
NZT	New Zealand Standard Time	UTC +12 hours
NZDT	New Zealand Daylight Saving Time	UTC +13 hours

6.1.2.2 NTP

NTP is the Network Time Protocol defined in RFC 1305, *Network Time Protocol (Version 3) Specification, Implementation and Analysis* and RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*. It allows for the participating network nodes to keep time more accurately and more importantly they can maintain time in a more synchronized fashion between all participating network nodes.

SR OS uses an NTP process based on a reference build provided by the Network Time Foundation. Nokia strongly recommends that the users review RFC 8633, *Network Time Protocol Best Current Practices*, when they plan to use NTP with the router. The RFC section "Using Enough Time Sources" indicates that using only two time sources (NTP servers) can introduce instability if they provide conflicting information. To maintain accurate time, Nokia recommends configuring three or more NTP servers.

NTP uses stratum levels to define the number of hops from a reference clock. The reference clock is considered to be a stratum-0 device that is assumed to be accurate with little or no delay. Stratum-0 servers cannot be used in a network. However, they can be directly connected to devices that operate as stratum-1 servers. A stratum-1 server is an NTP server with a directly-connected device that provides Coordinated Universal Time (UTC), such as a GPS or atomic clock.

The higher stratum levels are separated from the stratum-1 server over a network path, therefore, a stratum-2 server receives its time over a network link from a stratum-1 server. A stratum-3 server receives its time over a network link from a stratum-2 server.

SR OS routers normally operate as a stratum-2 or higher device. The router relies on an external stratum-1 server to source accurate time into the network. However, SR OS also allows for the use of the local PTP recovered time to be sourced into NTP. In this latter case, the local PTP source appears as a stratum-0 server and SR OS advertises itself as a stratum-1 server. Activation of the PTP source into NTP may impact the network NTP topology because the SR OS router is promoted to stratum-1.

SR OS router runs a single NTP clock which then operates NTP message exchanges with external NTP clocks. Exchanges can be made with external NTP clients, servers, and peers. These exchanges can be through the base, management, or VPRN routing instances.

NTP operates associations between clocks as either client or server, symmetric active and symmetric passive, or broadcast modes. These modes of operation are applied according to which elements are configured on the router. To run server mode, the user must enable NTP server mode for the base and each needed VPRN routing instance. To run client mode, the user must configure external servers. If both the local router and remote router are configured with each other as peers, then the router operates in symmetric active mode. If only one side of the association has peering configured, then the modes are symmetric passive. To operate using broadcast mode, interfaces must be configured to transmit as broadcast servers or receive as broadcast clients.

NTP server operation for both unicast and broadcast communication within a VPRN is configured within the VPRN (see the NTP Within a VPRN Service section in *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN*).



Note: NTP provides lightweight synchronization across a network for alignment of system time for logging purposes. NTP does not provide the high accuracy time needed for the on-air applications of the mobile base stations. The more recent PTP protocol has been developed for these applications (see [Network synchronization](#)).

The following NTP elements are supported:

- **server mode**

In this mode, the node advertises the ability to act as a clock source for other network elements. The node, by default, transmits NTP packets in NTP version 4 mode.

- **authentication keys**

Authentication keys implement increased security support in carrier and other networks. Both DES and MD5 authentication are supported, as well as multiple keys.

- **operation in symmetric active mode**

This capability requires that NTP be synchronized with a specific node that is considered more trustworthy or accurate than other nodes carrying NTP in the system. This mode requires that a specific peer is set.

- **server and peer addressing using IPv6**

Both external servers and external peers may be defined using IPv6 or IPv4 addresses. Other features (such as multicast, broadcast) use IPv4 addressing only.

- **broadcast or multicast modes**

When operating in these modes, the node receives or sends using either a multicast (default 224.0.1.1) or a broadcast address. Multicast is supported only on the CPM MGMT port.

- **alert when NTP server is not available**

When none of the configured servers are reachable on the node, the system reverts to manual timekeeping and issues a critical alarm. When a server becomes available, a trap is issued indicating that standard operation has resumed.

- **NTP and SNTP**

If both NTP and SNTP are enabled on the node, then SNTP transitions to an operationally down state. If NTP is removed from the configuration or shut down, then SNTP resumes an operationally up state.

- **gradual clock adjustment**

As several applications (such as Service Assurance Agent (SAA)) can use the clock, and if determined that a major (128 ms or more) adjustment needs to be performed, the adjustment is performed by programmatically stepping the clock. If a minor (less than 128 ms) adjustment must be performed, then the adjustment is performed by either speeding up or slowing down the clock.

- To avoid the generation of too many events/trap the NTP module rates limit the generation of events/traps to three per second. At that point a single trap is generated that indicates that event/trap squashing is taking place.

6.1.2.3 Synchronization

Synchronization between the CPMs includes the following:

- Configuration and boot-env synchronization ([File synchronization](#))
- [CPM synchronization](#)

6.1.2.4 GNSS

The 7750 SR supports frequency synchronization using a Layer 1 interface such as synchronous Ethernet, and ToD synchronization using a protocol such as NTP or PTP. In cases where these methods are not possible, or where accuracy cannot be ensured for the service, you can deploy a GNSS receiver as a

synchronous timing source. GNSS data is used to provide network-independent frequency and ToD synchronization.

GNSS receivers on the following platforms support GPS and Galileo reference using an integrated GNSS RF port:

- 7750 FP5 SR-1x-48D
- 7750 FP5 SR-1-24D
- 7750 FP5 SR-1-48D
- 7750 FP5 SR-1x-92S
- 7750 FP5 SR-1-46S
- 7750 FP5 SR-1-92S
- 7750 FP5 SR-1se
- 7750 FP5 SR-2se

A 7750 SR chassis equipped with a GNSS receiver and an attached GNSS antenna can be configured to receive frequency traceable to Stratum-1 (PRC/PRS). The GNSS receiver provides a synchronization clock to the SSU in the router with the corresponding QL for SSM. This frequency is distributed to the rest of the router from the SSU as configured with the following commands:

- **MD-CLI**

```
configure system central-frequency-clock ref-order
configure system central-frequency-clock ql-selection
```

- **classic CLI**

```
configure system sync-if-timing ref-order
configure system sync-if-timing ql-selection
```

The GNSS reference is qualified only if the GNSS receiver is in a position hold state and has a frequency successfully recovered. A PTP timeTransmitter or boundary clock can also use this frequency reference with PTP peers.

If GNSS signal loss or jamming result in the unavailability of timing information, the GNSS receiver automatically prevents output of clock or synchronization data to the system, and the system can revert to alternate timing sources. With Assisted Partial Timing Support (APTS), the system can perform a seamless switch when reverting to a backup PTP session; see [GNSS failure with APTS](#).

6.1.2.4.1 GNSS redundancy

The 7750 SR-2se chassis can be equipped with redundant CPMs. Each CPM includes an integrated GNSS receiver. Each integrated GNSS receiver can be connected to its own dedicated GNSS antenna, or both GNSS receivers can be connected to one shared GNSS antenna using a splitter.

For maximum resiliency, each CPM can use its own integrated GNSS receiver as well as the integrated GNSS receiver in the mate CPM installed in the same 7750 SR-2se chassis. The GNSS receivers in the redundant pair of CPMs actively synchronize with GNSS satellites so they are always in a hot standby state.

The active CPM has a startup preference for its own integrated GNSS receiver. If its own integrated GNSS receiver is down or the signal is degraded, the active CPM can automatically select and use the integrated GNSS receiver in the standby CPM, provided that receiver is up and the signal is not degraded.

After a CPM switchover, the integrated GNSS receiver in the newly standby CPM is reset.

6.1.2.4.2 GNSS failure with APTS

When the G.8275.2 profile is used for GNSS-enabled 7750 SR platforms, the APTS capability frequently measures and stores the delay offset between the GNSS time and a backup PTP session time. If a GNSS failure occurs, the backup PTP session automatically becomes the selected reference for time and frequency, and the stored delay offset value is added to or subtracted from the backup PTP session time to keep time and phase for the router as accurate as possible.

When GNSS has recovered and is stable, the system automatically switches back to GNSS for time and frequency reference, and backup PTP monitoring and delay measurement resumes.

6.1.2.5 CRON

The CRON feature supports periodic and date and time-based scheduling in SR OS. CRON can be used, for example, to schedule Service Assurance Agent (SAA) functions. CRON functionality includes the ability to specify scripts that need to be run, when they are scheduled, including one-time only functionality (one-shot), interval and calendar functions. Scheduled reboots, peer turn ups, service assurance agent tests and more can all be scheduled with CRON, as well as OAM events, such as connectivity checks, or troubleshooting runs.

CRON supports the schedule element. The schedule function configures the type of schedule to run, including one-time only (one-shot), periodic, or calendar-based runs. All runs are determined by month, day of month or weekday, hour, minute, and interval (seconds).

6.2 High Availability

This section describes the High Availability (HA) features that service providers can use to diminish vulnerability at the network or service provider edge and alleviate the effects of a lengthy outage on IP networks.

HA is an important feature in service provider routing systems because the demand from enterprise and residential communities has led to unprecedented growth of IP services and applications in service provider networks. Downtime can be very costly, and, in addition to lost revenue, customer information and business-critical communications can be lost. Availability is the combination of continuous uptime over long periods (Mean Time Between Failures (MTBF)) and the speed at which failover or recovery occurs (Mean Time To Repair (MTTR)).

The advantage of HA routing is evident at the network or service provider edge, where thousands of connections are hosted and rerouting options around a failed piece of equipment are often limited. As service providers converge business-critical services, such as real-time voice (VoIP), video, and VPN applications over their IP networks, the requirements for HA become much more stringent compared to the requirements for best-effort data.

Network and service availability become critical aspects when offering advanced IP services which dictates that IP routers that are used to construct the foundations of these networks be resilient to component and software outages.

For high availability configuration information, see [File synchronization](#).

6.2.1 HA features

As more and more critical commercial applications move onto the IP/MPLS networks, providing high availability services becomes increasingly important. This section describes high availability features for routers. Most of these features only apply to routers with two Control Processor Modules (CPM).

6.2.1.1 Redundancy

The redundancy features enable the duplication of data elements to maintain service continuation in case of outages or component failure.

See the *7450 ESS, 7750 SR, and VSR Multiservice ISA and ESA Guide* for information about redundancy for the Integrated Service Adapter (ISA).

6.2.1.1.1 Software redundancy

Software outages are challenging even when baseline hardware redundancy is in place. A balance should be maintained when providing HA routing, otherwise router problems typically propagate not only throughout the service provider network, but also externally to other connected networks possibly belonging to other service providers. This could affect customers on a broad scale. Currently supports several software availability features are supported that contribute to the percentage of time that a router is available to process and forward traffic.

All routing protocols specify minimum time intervals in which the peer device must receive an acknowledgment before it disconnects the session:

- OSPF default session timeout is approximately 40 seconds. The timeout intervals are configurable.
- BGP default session timeout is approximately 120 seconds. The timeout intervals are configurable for the 7750 SR and 7950 XRS only.

Therefore, the router software must recover faster than the specified time interval to maintain up time.

6.2.1.1.2 Configuration redundancy

Features configured on the active device CPM are configured on the standby CPM as well. When the active CPM fails, these features are brought up on the standby CPM when it becomes active.

6.2.1.1.3 Component redundancy

Component redundancy is critical to reduce MTTR for the system and primarily consists of the following router features:

- **dual route processor modules**

For a highly available architecture, redundant Control Processor Modules (CPM) are essential. The route processing functions of the CPM calculate the most efficient route to an Internet destination and communicate the best path information to peer routers. Rapid information synchronization between the primary and secondary CPMs is crucial to minimize recovery time.

- **switch fabric (SFM) redundancy**

Failure of a single switch fabric card can occur with little to no loss of traffic.

- **redundant line cards**

LAG, ECMP and other techniques are employed to spread traffic over multiple line cards so that a failure of one line card does not impact the services being delivered.

- **redundant power supply**

A power module can be removed without impact on traffic.

- **redundant fan**

Failure of a fan module can occur without impacting traffic.

- **hot swap**

Components in a live system can be replaced or become active without taking the system down or affecting traffic flow to/from other modules.

Router hardware architecture plays a key role in the availability of the system. The principle router architecture styles are centralized and distributed. In these architectures, both active and standby route processors, I/O modules (IOMs) (also called line cards), fans, and power supplies maintain a low MTTR for the routing system.

However, in a centralized architecture, packet processing and forwarding is performed in a central shared route processor and the individual line cards are relatively simple. The cards rely solely on the route processor for routing and forwarding intelligence and, should the centralized route processor fail, there is greater impact to the system overall, as all routing and packet forwarding stops.

In a distributed system, the packet forwarding functionality is situated on each line card. Distributing the forwarding engines off the central route processor and positioning one on each line card lowers the impact of route processor failure as the line cards can continue to forward traffic during an outage.

The distributed system is better suited to enable the convergence of business critical services such as real-time voice (VoIP), Video, and VPN applications over IP networks with superior performance and scalability. The centralized architecture can be prone to performance bottleneck issues and limits service offerings through poor scalability which may lead to customer and service SLA violations.

6.2.1.1.4 Service redundancy

All service-related statistics are kept during a switchover. Services, SDPs, and SAPs remain up with a minimum loss of forwarded traffic during a CPM switchover.

6.2.1.1.5 Accounting configuration redundancy

When there is a switchover and the standby CPM becomes active, the accounting servers are checked and if they are administratively up and capable of coming online (media present, and so on), the standby is brought online and new accounting files are created. Users must manually copy the accounting records from the failed CPM.

6.2.1.2 Nonstop forwarding

In a control plane failure or a forced switchover event, the router continues to forward packets using the existing stale forwarding information. Nonstop forwarding requires clean control plane and data plane separation.

Usually, the forwarding information is distributed to the IOMs, XCMs and XMAAs.

Nonstop forwarding is used to notify peer routers to continue forwarding and receiving packets, even if the route processor (control plane) is not working or is in a switch-over state. Nonstop forwarding requires clean control plane and data plane separation and usually the forwarding information is distributed to the line cards. This method of availability has both advantages and disadvantages. Nonstop forwarding continues to forward packets using the existing stale forwarding information during a failure. This may cause routing loops and black holes, and also requires that surrounding routers adhere to separate extension standards for each protocol. Every router vendor must support protocol extensions for interoperability.

6.2.1.3 Nonstop Routing

With nonstop routing (NSR) on the 7450 ESS, 7750 SR, 7950 XRS, and VSR router devices, routing neighbors are unaware of a routing process fault. If a fault occurs, a reliable and deterministic activity switch to the inactive control complex occurs such that routing topology and reachability are not affected, even in the presence of routing updates. NSR achieves high availability through parallelization by maintaining up to date routing state information, at all times, on the standby route processor. This capability is achieved independently of protocols or protocol extensions, providing a more robust solution than graceful restart protocols between network routers.

The NSR implementation on the 7450 ESS, 7750 SR, 7950 XRS, and VSR routers supports all routing protocols. NSR makes it possible to keep the existing sessions (BGP, LDP, OSPF, and so on) during a CPM switchover, including support for MPLS signaling protocols. Peers do not see any change.

Protocol extensions are not required. There are no interoperability issues and there is no need to define protocol extensions for every protocol. Unlike nonstop forwarding and graceful restart, the forwarding information in NSR is always up to date, which eliminates possible black holes or forwarding loops.

Traditionally, addressing HA issues have been patched through nonstop forwarding solutions. With the implementation of NSR, these limitations are overcome by delivering an intelligent hitless failover solution. This enables a carrier-class foundation for transparent networks, required to support business IP services backed by stringent SLAs. This level of HA poses a major issue for conventional routers whose architectural design limits or prevents them from implementing NSR.

6.2.1.4 CPM switchover

During a switchover, system control and routing protocol execution are transferred from the active to the standby CPM.

An automatic switchover may occur under the following conditions:

- A fault condition that causes the active CPM to crash or reboot.
- The active CPM is declared down (not responding).
- Online removal of the active CPM.

A manual switchover can occur if a switchover is forced from an active CPM to the standby, using the **admin redundancy force-switchover** command.

After a CPM switchover, SFMs, IOMs, MDAs and other cards in the system briefly appear as "not equipped" or "booting" while the newly active CPM rediscovers them, even though the control plane and data plane forwarding continues uninterrupted.

Use the following command to configure a batch file that executes automatically after a CPM switchover:

- **MD-CLI**

```
configure redundancy switchover-exec
```

- **classic CLI**

```
configure system switchover-exec
```

6.2.1.5 Synchronization

Synchronization between the CPMs includes the following:

- Configuration and boot-env synchronization ([File synchronization](#))
- [CPM synchronization](#)

6.2.1.5.1 CPM synchronization

When a new standby CPM is inserted into the system, it synchronizes with the active CPM upon a successful boot process.

When the standby CPM is rebooted, it synchronizes with the active CPM upon a successful boot process.

When configuration or state changes occur, an incremental synchronization is conducted from the active CPM to the standby CPM.

If the synchronization fails, the standby does not reboot automatically. Use the following command to display synchronization output information.

```
show redundancy synchronization
```

If the active and standby are not synchronized, use the following command on the active or standby CPM to manually reboot and synchronize the standby CPM.

```
admin reboot standby
```

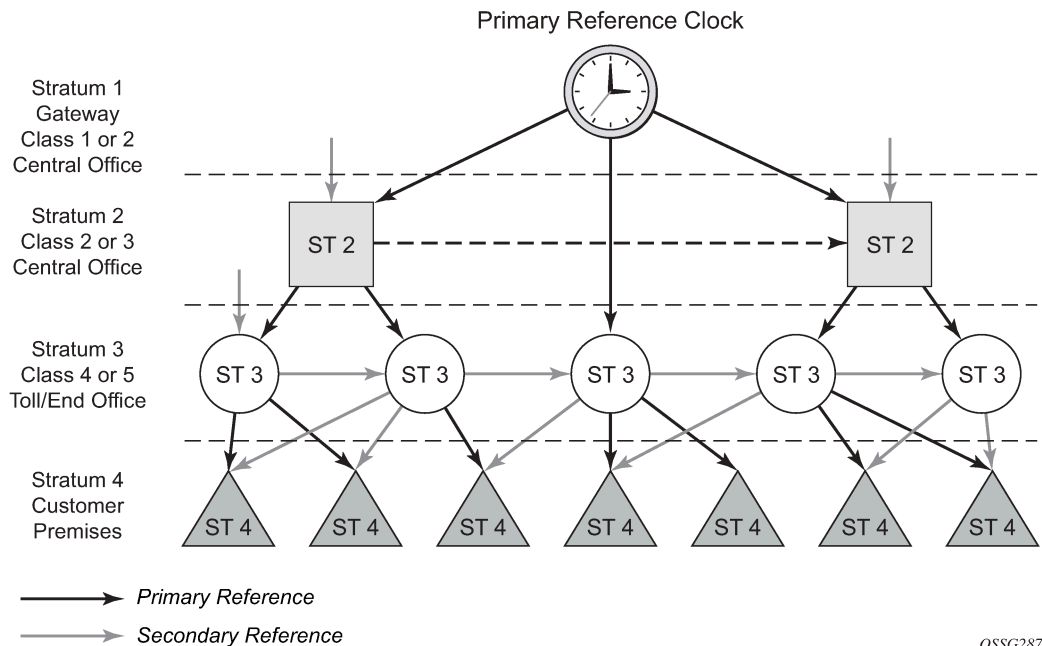
6.3 Network synchronization

This section describes network synchronization capabilities available on SR OS platforms. These capabilities involve multiple approaches to network timing; namely Synchronous Ethernet, BITS, and Adaptive clocking and a Precision Time Protocol (PTP) IEEE 1588v2. These features address barriers to entry by:

- provide synchronization quality required by the mobile space, such as radio operations and circuit emulation services (CES) transport
- augment and potentially replace the existing timing infrastructure and deliver high-quality network timing for frequency and time-sensitive wireline applications

The network time architecture in the following figure shows how network synchronization is commonly distributed in a hierarchical PTP topology at the physical layer.

Figure 10: Conventional network timing architecture (North American nomenclature)



The architecture shown in the preceding figure provides the following benefits:

- limits the need for high-quality clocks at each network element and only requires reliable and accurate replication of the input to remain traceable to its reference
- uses reliable physical media to provide transport of the timing signal. It does not consume any bandwidth and requires limited additional processing

The synchronization network is designed so a clock always receives timing from a clock of equal or higher stratum level or quality level. This ensures that if an upstream clock has a fault condition (for example, loses its reference and enters a holdover or free-run state) and begins to drift in frequency, the downstream clock is able to follow it. For greater reliability and robustness, most offices and nodes have at least two synchronization references that can be selected in priority order (such as primary and secondary).

Further levels of resiliency can be provided by designing a capability in the node clock that operates within prescribed network performance specifications in the absence of any reference for a specified period. A clock operating in this mode is said to hold the last known state over (or holdover) until the reference lock is once again achieved. Each level in the timing hierarchy is associated with minimum levels of network performance.

Each synchronization capable port can be independently configured to transmit data using the node reference timing or loop timing.

Specifically for synchronous Ethernet, transmission of a reference clock through a chain of Ethernet equipment requires that all equipment supports synchronous Ethernet. A single piece of equipment that is not capable of performing synchronous Ethernet breaks the chain. Ethernet frames still get through, but downstream devices should not use the recovered line timing because it is not traceable to an acceptable stratum source.

6.3.1 Central synchronization subsystem

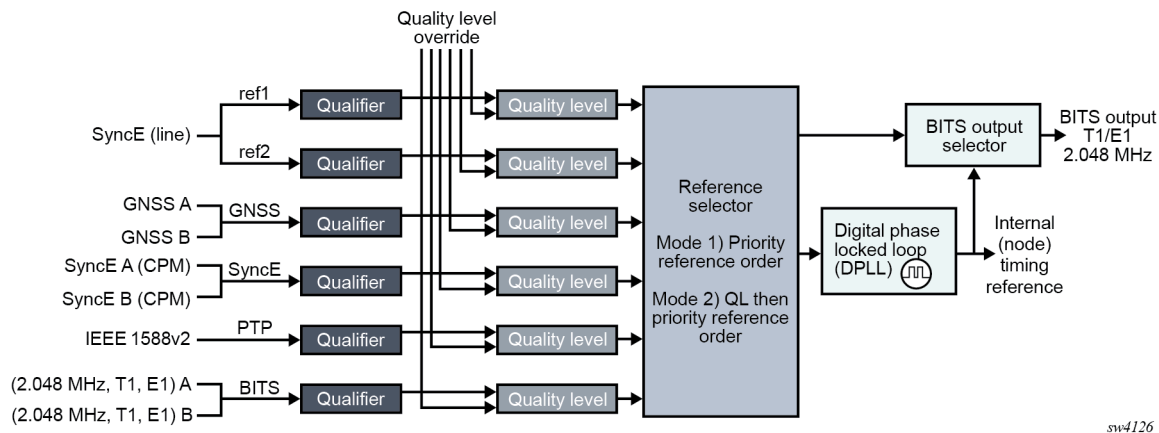
The timing subsystem for platforms has a central clock located on the CPM (motherboard). The timing subsystem performs many duties of the network element clock as defined by Telcordia (GR-1244-CORE) and ITU-T G.781.

The system can select from up to three (7950 XRS) or four (7450 ESS and 7750 SR) timing inputs to train the local oscillator. The priority order of these references must be specified. This is a simple ordered list of inputs: {bits, ref1, ref2, ptp}. The CPM clock output has the ability to drive the clocking for all line cards in the system. The routers support selection of the node reference using Quality Level (QL) indications. See [Figure 11: CPM clock synchronization reference selection](#) for more information about this selection process.



Note: Not all signals are available on all platforms.

Figure 11: CPM clock synchronization reference selection



The recovered clock can derive its timing from any of the following:

- Synchronous Ethernet ports
- BITS port on the CPM or CCM module
- GNSS RF ports (supported 7750 SR FP5 platforms)
- 10GE ports in WAN PHY mode
- IEEE 1588v2 timeReceiver port (PTP) (7450 ESS and 7750 SR)
- SyncE/1588 port on the CPM or the CCM

The BITS ports accept T1 or E1 signal formats. Some hardware also supports the 2048 kHz signal format. The format must be common between all BITSin and BITSout ports.

All settings of the signal characteristics for the BITS input apply to both ports. When the active CPM considers the BITS input as a possible reference, it first considers the BITS input port on the active CPM or CCM, followed by the BITS input port on the standby CPM or CCM, in that relative priority order. This relative priority order is in addition to the user-definable **ref-order**. For example, a **ref-order** of **bits ref1 ref2** would actually be BITS in (active CPM or CCM), followed by BITS in (standby CPM or CCM), followed by ref1, followed by ref2. When **ql-selection** is enabled, the QL of each BITS input port is viewed independently. The higher QL source is chosen.

When the active CPM considers the SyncE/1588 as a possible reference, the active CPM first considers the SyncE/1588 port on the active CPM or CCM, followed by the SyncE/1588 port on the standby CPM or CCM in that relative priority order. This relative priority order is in addition to the user-definable **ref-order**. For example, a **ref-order** of **sync ref1 ref2** would actually be SyncE/1588 (active CPM or CCM), followed by SyncE/1588 (standby CPM or CCM), followed by ref1, followed by ref2. When **ql-selection** is enabled, the QL of each SyncE/1588 input port is viewed independently. The higher QL source is chosen.

The following behavior applies to the platform architecture existing on 7750 SR-7/12/12e, 7750 SR-2s/7s/14s, 7750 SR-1e/2e/3e, 7750 SR-a4/a8, and 7450 ESS-7/12. When the BITS or SyncE port on the standby CPM is an option as input reference into the central clock of the active CPM, a display of the central clock data on the standby CPM indicates that it is locked to its local BITS or SyncE input. This is expected behavior and required to make the BITS input on the standby available to the active CPM as an option for reference selection.

The restrictions on the location for the source-port or source-bits for **ref1** and **ref2** are listed in [Table 15: Ref1 and ref2 timing references](#).

Table 15: Ref1 and ref2 timing references

Platform	Ref1 slots	Ref2 slots	Notes
7450 ESS-7	1 to 2	3 to 5	—
7450 ESS-12	1 to 5	6 to 10	—
7750 SR-1	1	1	Ref1 and ref2 cannot be on the same MDA
7750 SR-7	1 to 2	3 to 5	—
7750 SR-12	1 to 5	6 to 10	—
7750 SR-12e	1 to 5	6 to 9	—
7750 SR-a4	1	1	Ref1 and ref2 cannot be on the same MDA. Two CPMs must be installed to allow two references to be used
7750 SR-a8	1 to 2	1 to 2	Ref1 and ref2 cannot be on the same forwarding complex (IOM).
7750 SR-1e	1	1	Ref1 and ref2 cannot be on the same MDA
7750 SR-2e	1 to 2	1 to 2	Ref1 and ref2 cannot be on the same MDA

Platform	Ref1 slots	Ref2 slots	Notes
7750 SR-3e	1 to 3	1 to 3	Ref1 and ref2 cannot be on the same MDA
7750 SR-1s	1	1	Ref1 and ref2 cannot be on the same MAC chip. See the <i>7750 SR-1s Installation Guide</i> or use the show datapath command for the mappings When using XIOM and MDA, ref1 and ref2 cannot be on the same MDA
7750 SR-1x-48D 7750 SR-1-24D 7750 SR-1-48D 7750 SR-1x-92S 7750 SR-1-46S 7750 SR-1-96S 7750 SR-1se	1	1	Ref1 and ref2 cannot be on the same breakout connector
7750 SR-2s	1 to 2	1 to 2	Ref1 and ref2 cannot be on the same slot
7750 SR-2se	1 to 2	1 to 2	Ref1 and Ref 2 cannot be on the same slot
7750 SR-7s	1 to 6	1 to 6	Ref1 and ref2 cannot be on the same slot Slot 6 cannot be used if a CPM has been installed in that slot
7750 SR-14s	1 to 6	1 to 6	Ref1 and ref2 cannot be on the same slot
7950 XRS-20	1 to 10	1 to 10	Ref1 and ref2 cannot be on the same slot
7950 XRS-20e	1 to 10	1 to 10	Ref1 and ref2 cannot be on the same slot
7950 XRS-40	1 to 10	1 to 10	Ref1 and ref2 cannot be on the same slot

The BITS output ports can be configured to provided either the unfiltered recovered line clock from a line card port or the output of the central clock. The first case would be used if the port was connected to deliver an input reference directly to dedicated timing device in the facility (BITS or SASE device). The second case would be used to test the quality of the clocking used by the router.

When QL selection mode is disabled, the reversion setting controls when the central clock can reselect a previously failed reference.

The following table lists the selection operation for two references in both revertive and non-revertive modes:

Table 16: Revertive, non-revertive timing reference switching operation

Status of reference A	Status of reference B	Active reference non-revertive case	Active reference revertive case
OK	OK	A	A
Failed	OK	B	B
OK	OK	B	A
OK	Failed	A	A
OK	OK	A	A
Failed	Failed	holdover	holdover
OK	Failed	A	A
Failed	Failed	holdover	holdover
Failed	OK	B	B
Failed	Failed	holdover	holdover
OK	OK	A or B	A

6.3.2 7950 XRS-40 extension chassis central clocks

The central clock architecture previously described applies to each chassis of the 7950 XRS-40. There is a central clock located on each of the CPMs present in the extension chassis. However, there is no configuration for the central clocks on the CPMs of the extension chassis. The central clocks only use the BITS input ports of the extension chassis for their input reference. It is assumed that the quality of the reference provided into the BITS input ports of the extension chassis CPMs is equal to the quality of the Master chassis central clocks. See the *Installation Guide* for appropriate physical cabling to support this architecture.

6.3.3 Synchronization Status Messages



Note: See “Synchronization Status Messages (SSM)” in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Basic System Configuration Advanced Configuration Guide for Classic CLI* for information about advanced configurations.

Synchronization Status Messages (SSMs) allow the synchronization distribution network to determine the quality level of the clock sourcing a specific synchronization trail, and to allow a network element to select the best of multiple input synchronization trails. SSMs have been defined for various transport protocols

including T1/E1, and synchronous Ethernet, for interaction with office clocks, such as BITS or SSUs, and embedded network element clocks.

SSMs allow equipment to autonomously provision and reconfigure (by reference switching) their synchronization references, while helping to avoid the creation of timing loops. These messages are particularly useful to allow synchronization re-configurations when timing is distributed in both directions around a ring.

The following sections provide details about the SSM message functionality for different signal types. These functions apply to all platforms that support the signal type.

6.3.3.1 DS1 signals

DS1 signals can indicate the quality level of the source generating the timing information using the SSM transported within the 1544 kb/s signal Extended Super Frame (ESF) Data Link (DL), as specified in Recommendation G.704. No such provision is extended to SF formatted DS1 signals.

The format of the data link messages in ESF frame format is "0xxx xxx0 1111 1111", transmitted rightmost bit first. The six bits denoted "xxx xxx" contain the actual message; some of these messages are reserved for synchronization messaging. It takes 32 frames (such as 4 ms) to transmit all 16 bits of a complete DL.

6.3.3.2 E1 signals

E1 signals can indicate the quality level of the source generating the timing information using the SSM as specified in Recommendation G.704.

One of the Sa4 to Sa8 bits (the actual Sa bit is for user selection) is allocated for SSMs. To prevent ambiguities in pattern recognition, it is necessary to align the first bit (San1) with frame 1 of a G.704 E1 multi-frame.

A San bit is one of a 4-bit nibble, San1 to San4. San1 is the most significant bit; San4 is the least significant bit.

The message set in San1 to San4 is a copy of the set defined in SDH bits 5 to 8 of byte S1.

6.3.3.3 DS3/E3

DS3/E3 signals are not required to be synchronous. However, it is acceptable for their clocking to be generated from a synchronization source. The 7750 SR and the 7450 ESS allow E3/DS3 physical ports to be specified as a central clock input reference.

DS3/E3 signals do not support an SSM channel. QL-override should be used for these ports if ql-selection is enabled

6.3.4 Synchronous Ethernet



Note: See "Synchronous Ethernet" in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Basic System Configuration Advanced Configuration Guide for Classic CLI* for information about advanced configurations.

Traditionally, Ethernet-based networks employ the physical layer transmitter clock to be derived from an inexpensive +/-100ppm crystal oscillator and the receiver locks onto it. There is no need for long term

frequency stability because the data is packetized and can be buffered. For the same reason there is no need for consistency between the frequencies of different links. However, you can derive the physical layer transmitter clock from a high quality frequency reference by replacing the crystal with a frequency source traceable to a primary reference clock. This would not affect the operation of any of the Ethernet layers, for which this change would be transparent. The receiver at the far end of the link would lock onto the physical layer clock of the received signal, and therefore gain access to a highly accurate and stable frequency reference. Then, in a manner analogous to conventional hierarchical network synchronization, this receiver could lock the transmission clock of its other ports to this frequency reference and a fully time synchronous network could be established.

The advantage of using Synchronous Ethernet, compared with methods that rely on sending timing information in packets over an unclocked physical layer, is that it is not influenced by impairments introduced by the higher levels of the networking technology (packet loss, packet delay variation). Hence, the frequency accuracy and stability may be expected to exceed those of networks with unsynchronized physical layers.

Synchronous Ethernet allows users to gracefully integrate existing systems and future deployments into conventional industry-standard synchronization hierarchy. The concept behind synchronous Ethernet is analogous to SONET/SDH system timing capabilities. It allows the user to select any (optical) Ethernet port as a candidate timing reference. The recovered timing from this port is then used to time the system (for example, the CPM locks to this configured reference selection). The user then could ensure that any of system output would be locked to a stable traceable frequency source.

If the port is a fixed copper Ethernet port and in 1000BASE-T mode of operation, there is a dependency on the 802.3 link timing for the Synchronous Ethernet functionality (see ITU-T G.8262). The 802.3 link timing states must align with the wanted direction of Synchronous Ethernet timing flow. When a fixed copper Ethernet port is specified as an input reference for the node or when it is removed as an input reference for the node, an 802.3 link auto-negotiation is triggered to ensure the link timing aligns properly.

The SSM of Synchronous Ethernet uses an Ethernet OAM PDU that uses the slow protocol subtype. For a complete description of the format and processing, see ITU-T G.8264.

6.3.4.1 Timing reference selection based on quality level

For a BITS physical port, or for a synchronous Ethernet interface that supports Ethernet Synchronization Message Channel (ESMC), a timing input or PTP clock class provides a quality level value to indicate the source of timing of the far-end transmitter. These values provide input to the selection process on the nodal timing subsystem. This selection process determines which input to use to generate the signal on the SSM egress ports and the reference to use to synchronize the nodal clock, as follows:

- For the two reference inputs (ref1 and ref2) and for the BITS input ports, if the interface configuration supports the reception of a QL over SSM or ESMC, the quality level value is associated with the timing derived from that input.
- For the two reference inputs and for the BITS input ports, if the interface configuration is T1 with SF framing, the quality level associated with the input is QL-UNKNOWN.
- For the two reference inputs, if they are synchronous Ethernet ports and the ESMC is disabled, the quality level value associated with that input is QL-UNKNOWN.
- For the two reference inputs and for the BITS input ports, if the interface configuration supports the reception of a QL over SSM (and not ESMC), and no SSM value has been received, the quality level value associated with the input is QL-STU.

- For the two reference inputs and for the BITS input ports, if the interface configuration supports the reception of a QL over SSM or ESMC, but the quality level value received over the interface is not valid for the type of interface, the quality level value associated with that input is QL-INVALID.
- For the two reference inputs, if they are external synchronization ports, the quality level value associated with the input is QL-UNKNOWN.
- For the two reference inputs, if they are synchronous Ethernet ports and the ESMC is enabled, but no valid ESMC Information PDU has been received within the previous 5 s, the quality level value associated with that input is QL-FAILED.
- For GNSS reference input, the quality level is PRS if a frequency is successfully recovered; otherwise, the quality level is QL-FAILED.
- If the user has configured an override for the quality level associated with an input, the node displays both the received and override quality level value for the input. If no value has been received, the associated value is displayed instead.

After the quality level values have been associated with the system timing inputs, the two reference inputs and the external input timing ports are processed by the system timing module to select a source for the SSU. This selection process is as follows.

- Before an input can be used as a potential timing source, it must be enabled using the following command:

- **MD-CLI**

```
configure system central-frequency-clock ql-selection
```

- **classic CLI**

```
configure system sync-if-timing ql-selection
```

If the **ql-selection** command option is disabled, the priority order of the inputs for the Synchronous Equipment Timing Generator (SETG) is the priority order configured under the following command:

- **MD-CLI**

```
configure system central-frequency-clock ref-order
```

- **classic CLI**

```
configure system sync-if-timing ref-order
```

- If the **ql-selection** command is enabled, the priority of the inputs is calculated using the associated quality level value of the input and the priority order configured under the **ref-order** command. The inputs are ordered by the internal relative quality level based on their associated quality level values. If two or more inputs have the same quality level value, they are placed in order based on where they appear in the **ref-order** priority. The priority order for the SETG is based on both the reference inputs and the external synchronization input ports.
- After a prioritized list of inputs is calculated, the SETG and the external synchronization output ports are configured to use the inputs in their respective orders.
- After the SETG and external synchronization output ports priority lists are programmed, the highest-qualified priority input is used. To be qualified, the signal is monitored to ensure that it has the expected format and a frequency that is within the pull-in range of the SETG.

6.3.4.2 Extended QL TLV for ESMC messaging

SR OS supports the extended quality level (QL) TLV of the ESMC message as defined in ITU-T G.8264. This TLV allows the communication of newer QL values and some synchronization trail information up to the current clock. The newly added QL values include:

- QL_ePRTC
- QL_PRTC
- QL_PRC
- QL_eEEEC

Support for the TLV is included for 7250 IXR satellites, but is not supported for 7210 SAS satellites. The 7210 SAS satellite drops any received extended QL TLV. If one of the newer QLs is used in the **ql-override** commands for a 7210 SAS satellite central-frequency-clock, the configuration is not applied and the clock indicates "Unsupported Config".

6.3.5 Clock source quality level definitions

The following clock source quality levels have been identified for tracking network timing flow. These levels make up all the defined network deployment options described in Recommendation G.803 and G.781. The Option I network is a network developed on the original European SDH model. The Option II network is a network developed on the North American SONET model.

In addition to the QL values received over SSM of an interface, the standards also define additional codes for internal use. These include the following:

- QL INVx is generated internally by the system if and when an unallocated SSM value is received, where x represents the binary value of this SSM. All these independent values are assigned as the single value of QL-INVALID.
- QL FAILED is generated internally by the system if and when the terminated network synchronization distribution trail is in the signal fail state.

There is also an internal quality level of QL-UNKNOWN. This is used to differentiate from a received QL-STU code, but is equivalent for the purposes of QL selection.

The following table lists the synchronization message coding and source priorities for SSM received.

Table 17: Synchronization message coding and source priorities — SSM received

SSM value received on port				Internal relative quality level
SDH interface or SyncE interface in SDH mode	SONET interface or SyncE interface in SONET mode	E1 interface	T1 interface (ESF)	
0010 (prc)	0001 (prs)	0010 (prc)	00000100 11111111 (prs)	1. Best quality
	0000 (stu)		00001000 11111111 (stu)	2.

SSM value received on port				Internal relative quality level
SDH interface or SyncE interface in SDH mode	SONET interface or SyncE interface in SONET mode	E1 interface	T1 interface (ESF)	
	0111 (st2)		00001100 11111111 (ST2)	3.
0100 (ssua)	0100 (tnc)	0100 (ssua)	01111000 11111111 (TNC)	4.
	1101 (st3e)		01111100 11111111 (ST3E)	5.
1000 (ssub)		1000 (ssub)		6.
	1010 (st3/eec2)		00010000 11111111 (ST3)	7.
1011 (sec/eec1)		1011 (sec)		8. Lowest quality qualified in QL-enabled mode
	1100 (smc)		00100010 11111111 (smc)	9.
			00101000 11111111 (st4)	10.
	1110 (pno)		01000000 11111111 (pno)	11.
1111 (dnu)	1111 (dus)	1111 (dnu)	00110000 11111111 (dus)	12.
Any other	Any other	Any other	N/A	13. QL_INVALID
				14. QL-FAILED
				15. QL-UNC

The following table lists the synchronization message coding and source priorities for SSM transmitted.

Table 18: Synchronization message coding and source priorities — SSM transmitted

Internal relative quality level	SSM values to be transmitted by interface of type			
	SDH interface or SyncE interface in SDH mode	SONET interface or SyncE interface in SONET mode	E1 interface	T1 interface (ESF)
1. Best quality	0010 (prc)	0001 (PRS)	0010 (prc)	00000100 11111111 (PRS)

Internal relative quality level	SSM values to be transmitted by interface of type			
	SDH interface or SyncE interface in SDH mode	SONET interface or SyncE interface in SONET mode	E1 interface	T1 interface (ESF)
2.	0100 (ssua)	0000 (stu)	0100 (ssua)	00001000 11111111 (stu)
3.	0100 (ssua)	0111 (st2)	0100 (ssua)	00001100 11111111 (st2)
4.	0100 (ssua)	0100 (tnc)	0100 (ssua)	01111000 11111111 (tnc)
5.	1000 (ssub)	1101 (st3e)	1000 (ssub)	01111100 11111111 (st3e)
6.	1000 (ssub)	1010 (st3/eec2)	1000 (ssub)	00010000 11111111 (st3)
7.	1011 (sec/eec1)	1010 (st3/eec2)	1011 (sec)	00010000 11111111 (st3)
8. Lowest quality qualified in QL-enabled mode	1011 (sec/ eec1)	1100 (smc)	1011 (sec)	00100010 11111111 (smc)
9.	1111 (dnu)	1100 (smc)	1111 (dnu)	00100010 11111111 (smc)
10.	1111 (dnu)	1111 (dus)	1111 (dnu)	00101000 11111111 (st4)
11.	1111 (dnu)	1110 (pno)	1111 (dnu)	01000000 11111111 (pno)
12.	1111 (dnu)	1111 (dus)	1111 (dnu)	00110000 11111111 (dus)
13. QL_INVALID	1111 (dnu)	1111 (dus)	1111 (dnu)	00110000 11111111 (dus)
14. QL-FAILED	1111 (dnu)	1111 (dus)	1111 (dnu)	00110000 11111111 (dus)
15. QL-UNC	1011 (sec/eec1)	1010 (st3/eec2)	1011 (sec)	00010000 11111111 (st3)



Note: When the internal Quality Level is in the range of 9 through 14, the output codes shown in the preceding table, appear only if QL selection is disabled. If ql-selection is enabled, all the internal states are changed to internal state 15 (Holdover) and the ssm value generated reflects the holdover quality of the internal clock.

6.3.6 Advanced G.781 features

The central clock of the node supports several advanced features of the ITU-T G.781 standard. These include the specification of minimum acceptable QL values for the input references and the BITS output port, the ability to squelch the BITS output signal, and the specification of a Wait To Restore timer for input references. These features allow for more options in the management of the synchronization topology.

6.3.7 IEEE 1588v2 PTP



Note: See "IEEE 1588 for Frequency, Phase, and Time Distribution" in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Basic System Configuration Advanced Configuration Guide for Classic CLI* for information about advanced configurations.



Note: The IEEE 1588 Working Group has introduced the terms `timeTransmitter` and `timeReceiver` as alternatives to the former master/slave terminology. This document has been updated with these new terms.

Precision Time Protocol (PTP) is a timing-over-packet protocol defined in the IEEE 1588v2 standard 1588 PTP 2008.

PTP may be deployed as an alternative timing-over-packet option to Adaptive Clock Recovery (ACR). PTP provides the capability to synchronize network elements to a Stratum-1 clock or primary reference clock (PRC) traceable frequency source over a network that may or may not be PTP-aware. PTP has several advantages over ACR. It is a standards-based protocol, has lower bandwidth requirements, can transport both frequency and time, and can potentially provide better performance.

Support is provided for an ordinary clock in `timeReceiver` or `timeTransmitter` mode or a boundary clock. When configured as an ordinary clock `timeTransmitter`, PTP can only be used for the distribution of a frequency reference, not a time reference. The boundary clock and ordinary clock `timeReceiver` can be used for both frequency and time distribution.

The ordinary clock `timeTransmitter`, ordinary clock `timeReceiver`, and boundary clock communicate with neighboring IEEE 1588v2 clocks. These neighbor clocks can be ordinary clock `timeTransmitters`, ordinary clock `timeReceivers`, or boundary clocks. The communication can be based on either unicast IPv4/IPv6 sessions transported through IP interfaces or multicast Ethernet transported through Ethernet ports.



Note: The source address used for the originating IPv6 PTP messages must have an IPv6 address defined using the following commands:

- **MD-CLI**

```
configure system security source-address ipv6 address  
configure service vprn source-address ipv6 address
```

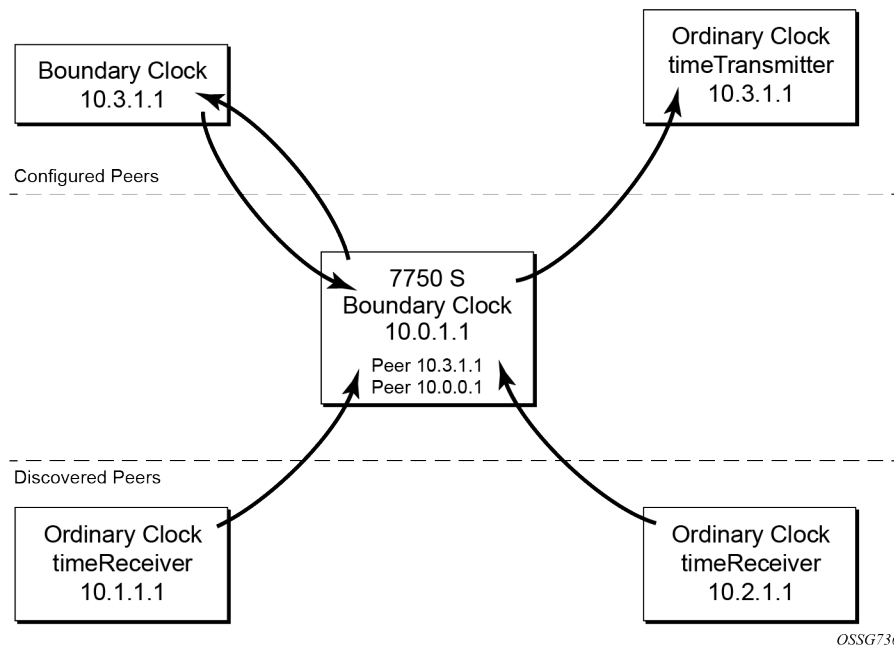
- **classic CLI**

```
configure system security source-address application6  
configure service vprn source-address application6
```

For the unicast IP sessions, the external clocks are labeled 'peers'. There are two types of peers: configured and discovered. An ordinary clock `timeReceiver` or a boundary clock should have configured peers for each PTP neighbor clock from which it may accept synchronization information. The router

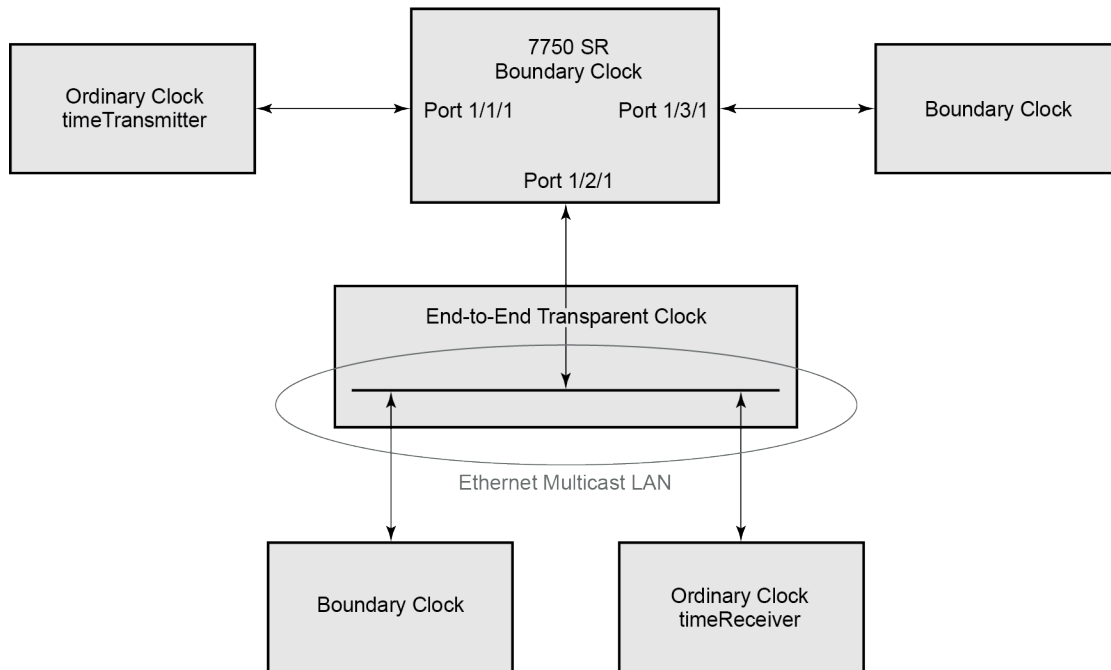
initiates unicast sessions with all configured peers. An ordinary clock timeTransmitter or boundary clock accepts unicast session requests from external peers. If the peer is not a configured peer, then it is considered a discovered peer. An ordinary clock timeTransmitter or boundary clock can deliver synchronization information toward discovered peers. [Figure 12: Peer clocks](#) shows the relationship of various neighbor clocks using unicast IP sessions to communicate with a 7750 SR configured as a boundary clock with two configured peers.

Figure 12: Peer clocks



For multicast Ethernet operation, the router listens for and transmits PTP messages using the configured multicast MAC address. Neighbor clocks are discovered via the reception of messages through an enabled Ethernet port. An ordinary clock timeTransmitter, ordinary clock timeReceiver, and a boundary clock support more than one neighbor PTP clock connecting into a single port. This may be encountered with the deployment of an Ethernet multicast LAN segment between the local clock and the neighbor PTP ports using an end-to-end transparent clock or an Ethernet switch. The Ethernet switch is not recommended because of the introduction of PDV and the potential degradation of performance but it can be used if appropriate to the application. [Figure 13: Ethernet multicast ports](#) shows the relationship of various neighbor clocks using multicast Ethernet sessions to a 7750 SR configured as a boundary clock. The 7750 SR has three ports configured for multicast Ethernet communications. Port 1/2/1 of the 7750 SR shows a connection where there are two neighbor clocks connecting to one port of the 7750 SR through an end-to-end transparent clock.

Figure 13: Ethernet multicast ports



al_0527

The ordinary clock timeTransmitter, ordinary clock timeReceiver, and boundary clock allow for PTP operation over both unicast IPv4/IPv6 and multicast Ethernet at the same time.

The IEEE 1588v2 standard includes the concept of PTP profiles. These profiles are defined by industry groups or standards bodies that define how IEEE 1588v2 is to be used for a particular application.

The following profiles are supported:

- IEEE 1588v2 (ieee1588-2008)
- G.8265.1 (g8265dot1-2010)
- G.8275.1 (g8275dot1-2014)
- G.8275.2 (g8275dot2-2016)

When an ordinary clock timeReceiver or a boundary clock receive Announce messages from one or more configured peers or multicast neighbors, it executes a Best TimeTransmitter Clock Algorithm (BTCA) to determine the state of communication between itself and the peers. The system uses the BTCA to create a hierarchical topology allowing the flow of synchronization information from the best source (the grandmaster clock) out through the network to all boundary and timeReceiver clocks. Each profile has a dedicated BTCA.

If the **profile** setting for the clock is ieee1588-2008, the precedence order for the BTCA is as follows:

1. priority1
2. clockClass
3. clockAccuracy
4. PTP variance (offsetScaledLogVariance)
5. priority2

6. clockIdentity

7. stepsRemoved from the grandmaster

The ordinary clock timeTransmitter, ordinary clock timeReceiver, and boundary clock set their local parameters as listed in [Table 19: Local clock parameters when profile is set to ieee1588-2008](#).

Table 19: Local clock parameters when profile is set to ieee1588-2008

Parameter	Value
clockIdentity	Chassis MAC address following the guidelines of 7.5.2.2.2 of IEEE 1588
clockClass	6 — local clock is configured using a time reference from a GNSS receiver 7 — local clock is in holdover after losing time reference from the local GNSS receiver for no more than ten minutes 13 — local clock configured as ordinary clock time Transmitter and is locked to an external reference 14 — local clock configured as ordinary clock time Transmitter and in holdover after having been locked to an external source 248 — local clock configured as ordinary clock time Transmitter and is in free run or the router is configured as a boundary clock 255 — local clock configured as ordinary clock timeReceiver
clockAccuracy	FE — unknown 21 — when using a time reference from a GNSS receiver
offsetScaledLogVariance	FFFF — not computed

If the **profile** setting for the clock is g8265dot1-2010, the precedence order for the best timeTransmitter selection algorithm is:

1. clockClass
2. priority

The ordinary clock timeTransmitter, ordinary clock timeReceiver, and boundary clock use local settings as listed in [Table 20: Local clock parameters when profile is set to g8265dot1-2010](#).

Table 20: Local clock parameters when profile is set to g8265dot1-2010

Parameter	Value
clockClass	80-110 — value corresponding to the QL out of the central clock as per Table 1/G.8265.1 255 — the clock is configured as ordinary clock timeReceiver

Parameter	Value
domain number	0 to 255 — configured domain value must be within this range when the G.8265.1 profile is used, and is 4 by default

The g8265dot1-2010 profile is for use in an environment with only ordinary clock timeTransmitters and timeReceivers for frequency distribution.

If the **profile** setting for the clock is g8275dot1-2014, the precedence order for the best timeTransmitter selection algorithm is very similar to that used with the default profile. It ignores **priority1**, includes a **localPriority** and includes the ability to force a port to never enter timeReceiver state (**timeTransmitter-only**).



Note: The **g8275dot1-2014** keyword is the CLI name of the profile. SR OS complies with the latest published version of the ITU-T G.8275.1 recommendation; the year in the keyword does not indicate the edition of the recommendation that is supported.

The precedence is as follows:

1. clockClass
2. clockAccuracy
3. PTP variance (offsetScaledLogVariance)
4. priority2
5. localPriority
6. clockIdentity (See Note)
7. stepsRemoved from the grandmaster



Note: If the two clocks being compared have a clockClass less than 128, this step is skipped. Skipping this step is the normal case because most clocks used as grandmasters advertise a clockClass less than 128.

The ordinary clock timeTransmitter, ordinary clock timeReceiver, and boundary clock use local settings listed in [Table 21: Local clock settings when profile is set to g8275dot1-2014](#).

Table 21: Local clock settings when profile is set to g8275dot1-2014

Parameter	Value
clockIdentity	Chassis MAC address following the guidelines of 7.5.2.2.2 of IEEE 1588
clockClass	165 — local clock configured to a boundary clock and the boundary clock was previously locked to a grandmaster with a clock class of 6 248 — local clock configured as boundary clock 255 — local clock configured as ordinary clock timeReceiver
clockAccuracy	FE — unknown
offsetScaledLogVariance	FFFF — not computed

Parameter	Value
clockMode	freerun acquiring locked holdover within specification holdover out of specification
sdold	0
profileIdentifier	The profile identifier assigned to the latest published version of the ITU-T G.8275.1 recommendation

The clock mode indicates the synchronization state of the local clock. The router reports both the current clock mode and the time that has elapsed since the clock entered that mode, which allows it to track the holdover performance of the clock. The values of the fields in the transmitted Announce messages follow the clock mode.

The **numberPorts** value in the defaultDS data set reports the number of PTP ports and peers on the local clock. If the node supports GNSS and is configured as a boundary clock, GNSS ports are also counted. Otherwise, GNSS inputs are not considered PTP ports are not counted.

The router transmits and accepts PTP messages that use either an IEEE 1588-2019 or an IEEE 1588-2008 message header. The majorSdold and minorSdold fields are set to 0 and the minorVersionPTP field is set to 1.

If the **profile** setting for the clock is *g8275dot2-2016*, the precedence order for the best master selection algorithm is very similar to that used with the *g8275dot1-2014* profile. It ignores the **priority1** parameter, includes a **localPriority** parameter, and includes the ability to force a port to never enter slave state (**master-only**). The precedence is as follows:

- clockClass
- clockAccuracy
- PTP variance (offsetScaledLogVariance)
- priority2
- localPriority
- clockIdentity
- stepsRemoved from the grandmaster



Note: If the two clocks being compared have a clockClass of less than 128, the comparison of the clockIdentity is skipped. Skipping the comparison of the clockIdentity is the normal case because the vast majority of clocks used as grandmasters advertise a clockClass of less than 128.

The following table describes the local parameter settings when the **profile** setting for the clock is *g8275dot2-2016*.

Table 22: Local Clock Parameters: g8275dot2-2016

Parameter	Value and Description
clockIdentity	Chassis MAC address following the guidelines of 7.5.2.2.2 of IEEE 1588
clockClass	6 — local clock is using a time reference from a GNSS receiver 165 — local clock is configured as a boundary clock in holdover; the boundary clock was previously locked to a grandmaster clock with a clock class of 6 248 — local clock is configured as a grandmaster clock or boundary clock in free-run mode 255 — local clock is configured as an ordinary clock slave
clockAccuracy	0xFE — unknown 0x21 — when using a time reference from a GNSS receiver
offsetScaledLogVariance	0xFFFF — not computed 0x4e5d (1.8E-15) — when using a time reference from a GNSS receiver

There is a limit on the number of external PTP clocks to which the ordinary clock timeReceiver or boundary clock requests unicast service (number of configured peers) and also a limit to the number of external PTP clocks to which the ordinary clock timeTransmitter or boundary clock grants unicast service (number of discovered peers). An association where the boundary clock has a symmetric relationship with another boundary clock (in other words, they both have the other as a configured peer) consumes a request and a grant unicast service in each router.

The number of configured Ethernet ports is not restricted.

There are limits to the maximum transmitted and received event message rates supported in the router. Each unicast IP service established consumes a portion of one of the unicast message limits. When either limit is reached, additional unicast service requests are refused by sending a grant response with zero in the duration field.

See the scaling guide for the appropriate release for the specific unicast message limits related to PTP.

Multicast messages are not considered when validating the unicast message limit. When multicast messaging on Ethernet ports is enabled, the PTP load needs to be monitored to ensure the load does not exceed the capabilities. There are several commands that you can use for this monitoring. If the capacity usage reaches 100%, the PTP software process on the router is at its limit of transmitting and receiving PTP packets.

Use the following command to identify the load of the PTP software process.

```
show system cpu
```

Because you cannot control the amount of PTP messages being received over the Ethernet ports, you can use statistics commands to identify the source of the message load.

Use the following command to display aggregate packet rates.

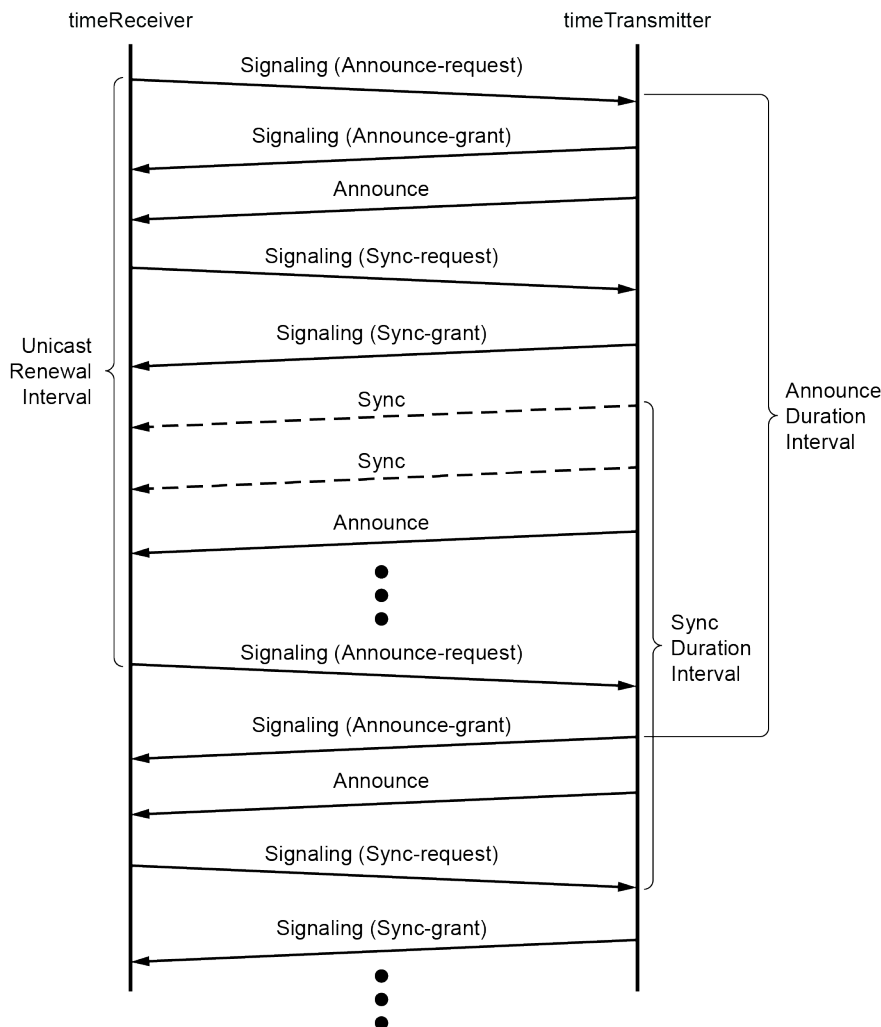
```
show system ptp statistics
```

Use the following commands to display received packet rates.

```
show system ptp port
show system ptp port detail
```

Figure 14: Messaging sequence between the PTP timeReceiver clock and PTP timeTransmitter clock shows the unicast negotiation procedure performed between a timeReceiver and a peer clock that is selected to be the timeTransmitter clock. The timeReceiver clock requests Announce messages from all peer clocks but only request Sync and Delay_Res messages from the clock selected to be the timeTransmitter clock.

Figure 14: Messaging sequence between the PTP timeReceiver clock and PTP timeTransmitter clock



OSSG666

6.3.7.1 PTP clock synchronization

The IEEE 1588v2 standard synchronizes the frequency and time from a timeTransmitter clock to one or more timeReceiver clocks over a packet stream. This packet-based synchronization standard defines

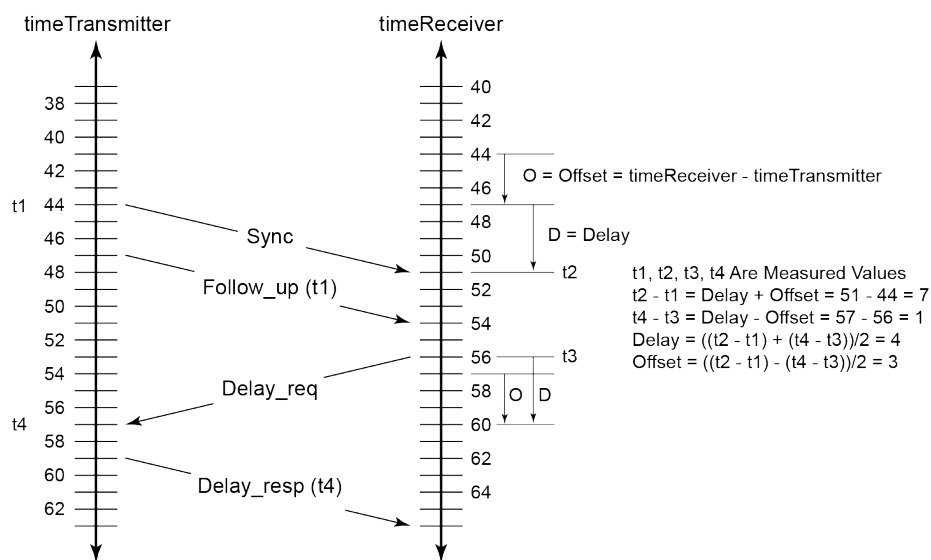
transport to use UDP/IP with unicast or Ethernet with multicast. See [PTP capabilities](#) for information about supported PTP encapsulation types.

As part of the basic synchronization timing computation, a number of event messages are defined for synchronization messaging between the PTP timeReceiver port and PTP timeTransmitter port. A one-step or two-step synchronization operation can be used, with the two-step operation requiring a follow-up message after each synchronization message. Ordinary clock timeTransmitter and boundary clock timeTransmitter ports use one-step operation; ordinary clock timeReceiver and boundary clock timeReceiver ports can accept messages from either one-step or two-step operation timeTransmitter ports.

The IEEE 1588v2 standard includes a mechanism to control the topology for synchronization distribution. The Best TimeTransmitter Clock Algorithm (BTCA) defines the states for the PTP ports on a clock. One port is set to the timeReceiver state and the other ports are set to the timeTransmitter (or passive) state. Ports in the timeReceiver state recover synchronization delivered by an external PTP clock, and ports in the timeTransmitter state transmit synchronization to external PTP clocks.

The following figure shows the basic synchronization timing computation between the PTP timeReceiver clock and PTP best timeTransmitter. This figure illustrates the offset of the timeReceiver clock referenced to the best timeTransmitter signal during startup.

Figure 15: PTP timeReceiver clock and timeTransmitter clock synchronization timing computation



OSSG732

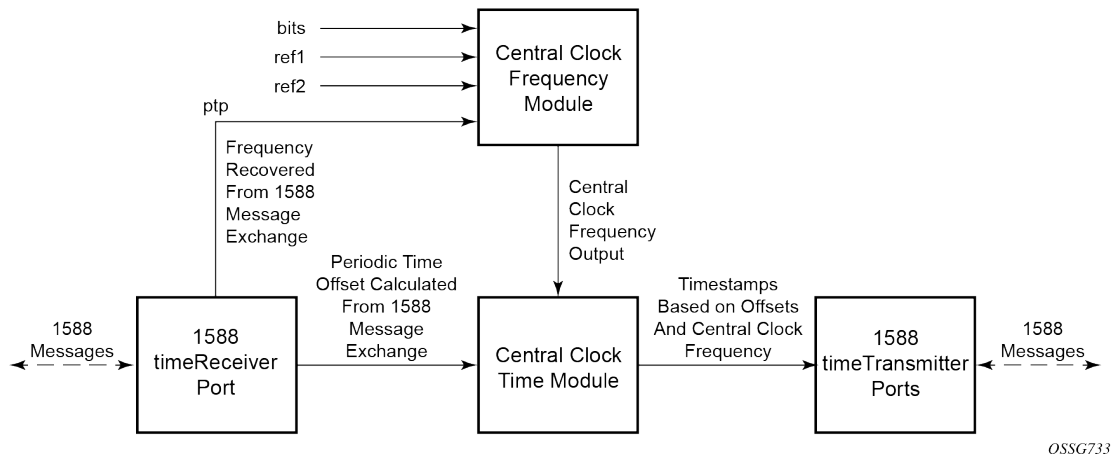
When using IEEE 1588v2 for distribution of a frequency reference, the timeReceiver calculates a message delay from the timeTransmitter to the timeReceiver based on the timestamps exchanged. A sequence of these calculated delays contain information of the relative frequencies of the timeTransmitter clock and timeReceiver clock but has a noise component related to the PDV experienced across the network. The timeReceiver must filter the PDV effects to extract the relative frequency data, then adjust the timeReceiver frequency to align with the timeTransmitter frequency.

When using IEEE 1588v2 for distribution of time, the 7750 SR and 7450 ESS use the four timestamps exchanged using the IEEE 1588v2 messages to determine the offset between the router time base and the external timeTransmitter clock time base. The router determines the offset adjustment and then in between these adjustments, the router maintains the progression of time using the frequency from the central clock of the router. This allows time to be maintained using a BITS input source or a Synchronous Ethernet input

source even if the IEEE 1588v2 communications fail. When using IEEE 1588v2 for time distribution, the central clock should at a minimum have a system timing input reference enabled.

The following figure shows a logical model for using PTP/1588 for network synchronization.

Figure 16: Logical model for using PTP/1588 for network synchronization



6.3.7.1.1 Synchronization uncertainty

The PTP protocol uses the BTCA to build the network topology from a PTP grandmaster, through one or more boundary clocks, and into timeReceiver clocks. This mechanism relies on the grandmaster information contained in the Announce messages sent between these clocks. While the BTCA is designed to create the topology quickly and without any timing loops, it does not address the fact that as each clock selects a new parent clock, it takes time for each clock to align its local time to that of its parent clock.

To address this, the IEEE and ITU-T adds the synchronizationUncertain bit to the header of Announce messages. A grandmaster or boundary clock sets this bit to true when it is calibrating. In addition, if this bit is set to true in the Announce messages from its parent clock, the boundary clock always transmits the bit as true. This allows the topology to settle quickly but also provides an indication to the final timeReceiver clocks as to when they can trust the time being delivered by the network.

The optional synchronizationUncertain bit (flagField octet 1, bit 6) defined in the G.8275.1, G.8275.2, and IEEE 1588 specifications in Announce messages indicates the synchronization uncertainty of the PTP clock to the user and downstream clocks. A synchronizationUncertain bit value of 1 indicates a SyncUncertain TRUE state, meaning the local clock is still in the transient phase and attempting to achieve stability; a synchronizationUncertain bit value of 0 indicates a SyncUncertain FALSE state, meaning the local clock has reached the steady state phase.

The SyncUncertain state transitions to FALSE when all of the following conditions are true for a number of consecutive, stable PTP time update windows:

- the parent PTP clock is in a SyncUncertain FALSE state
- PTP time recovery is locked
- the grandmaster PTP clock is Clock Class 6 or 7
- the central frequency clock is locked, and the quality level of the frequency is QL-PRC/QL-PRS

- PTP frequency recovery is locked if PTP is used for frequency and the profile is G.8275.2 or IEEE 1588v2

The SyncUncertain state transitions back to TRUE if any of the above conditions are not met, even momentarily, and the stable PTP time update window count resets to 0.

In some deployments, other PTP clocks in the network may not support the synchronizationUncertain bit. If the upstream clocks do not support synchronizationUncertain, the synchronizationUncertain bit is 0, indicating a SyncUncertain FALSE state regardless of the actual state of the local clock. If the downstream clocks do not support synchronizationUncertain, it may be desirable to stop the transmission of Announce messages while the local clock is in a state of SyncUncertain TRUE. Use the following command to stop this transmission:

- **MD-CLI**

```
configure system ptp tx-while-sync-uncertain false
```

- **classic CLI**

```
configure system ptp no tx-while-sync-uncertain
```

If this command is used to stop the transmission of Announce messages while the local clock is in a state of SyncUncertain TRUE, unicast negotiation grant requests are not granted and current grants are canceled.

6.3.7.2 Performance considerations

Although IEEE 1588v2 can be used on a network that is not PTP-aware, the use of PTP-aware network elements (boundary clocks) within the packet switched network improves synchronization performance by reducing the impact of PDV between the grandmaster clock and the timeReceiver clock. When IEEE 1588v2 is used to distribute high accuracy time, such as for mobile base station phase requirements, the network architecture requires the deployment of PTP awareness in every device between the grandmaster clock and the mobile base station timeReceiver.

In addition, performance is also improved by the removal of any PDV caused by internal queuing within the boundary clock or timeReceiver clock. This is accomplished with hardware that is capable of detecting and time stamping the IEEE 1588v2 packets at the Ethernet interface. This capability is referred to as port-based time stamping.

For a timeReceiver Clock or Boundary Clock, the maximum number of steps removed from the GM may be configured, as the higher the steps removed, the inaccuracy tends to increase. Any Announce message received with Steps Removed that is equal to or greater than the configured maximum steps removed value, are ignored. This is useful to ensure the desired level of PTP performance, as well as detected rogue messages around ring topologies.

6.3.7.2.1 Port-based timestamping of PTP messages

To meet the stringent performance requirements of PTP mobile network applications, the 1588 packets must be timestamped at the ingress and egress ports. This requires the use of 1588 port-based timestamping on the ports handling the PTP messages. This avoids any possible PDV that may be introduced between the port and the CPM. The ability to timestamp in the interface hardware is provided on a subset of the IMM and MDA assemblies of the routers. Generally, all FP4 and later generations of

the XMA, XMA-s, and MDA-e-XP modules support 1588 port-based timestamping. For other assemblies, contact your Nokia representative to verify the support for 1588 port-based timestamping.

When configuring the **ptp-hw-assist** command for PTP over IP, a loopback address must be used for PTP to ensure these message are timestamped at the port even after the routing topology changes. To configure the loopback address, use one of the following commands:

- **MD-CLI**

```
configure system security source-address ipv4 ptp address address
configure system security source-address ipv6 ptp address address
```

- **classic CLI**

```
configure system security source-address application ptp address
configure system security source-address application6 ptp address
```

Enabling the **ptp-hw-assist** command option within a Layer 3 interface is only supported if one of the following conditions is met:

- All physical ports contained in the interface support PBT for PTP over UDP/IPv4.
- All physical ports contained in the interface support PBT for PTP over UDP/IPv6 and a loopback IPv6 address is configured for PTP using the following commands:

- **MD-CLI**

```
configure system security source-address ipv6 ptp address address
configure service vprn source-address ipv6 ptp address address
```

- **classic CLI**

```
configure system security source-address application6 ptp address
configure service vprn source-address application6 ptp address
```

While the **ptp-hw-assist** feature supports port timestamping for a single router interface per physical port, the **ptp-timestamping** port-level configuration feature supports port timestamping for multiple router interfaces per physical port. The **ptp-timestamping** feature applies port timestamping to PTP messages using UDP/IPv4 or UDP/IPv6 transport. The destination IP address(es) specified under the port-level **ptp-timestamping** feature should be configured as one of the addresses defined for PTP use with the following commands:

- **MD-CLI**

```
configure system security source-address ipv4 ptp address address
configure service vprn source-address ipv4 ptp address address
configure system security source-address ipv6 ptp address address
configure service vprn source-address ipv6 ptp address address
```

- **classic CLI**

```
configure system security source-address application ptp address
configure service vprn source-address application ptp address
configure system security source-address application6 ptp address
configure service vprn source-address application6 ptp address
```

The **ptp-timestamping** feature enables the local PTP clock to use PTP over IP over multiple router interfaces per physical port. The feature can also be used for router interfaces across different router instances that have the same local PTP IP address.

When using the **ptp-timestamping** feature, all received messages that match the characteristics (PTP protocol, UDP over IP with the configured address, PTP event message) must be delivered to the PTP process of the node.



Caution: If PTP traffic on the port within a VLAN is forwarded across the node and does not end up at the PTP process of the node, the traffic may become corrupted. This traffic should be moved to a different physical port to avoid corruption.

6.3.7.3 PTP capabilities

For each PTP message type to be exchanged between the router and an external 1588 clock, a unicast session must be established using the unicast negotiation procedures. The router allows the configuration of the message rate to be requested from external 1588 clocks. The router also supports a range of message rates that it grants to requests received from the external 1588 clocks. The IEEE 1588 standard allows the grantor port to respond with a shorter duration than what was in the request. The router can accept such a grant and uses that duration. The router issues a renegotiation before the duration expires.

[Table 23: Message rates ranges and defaults](#) describes the ranges for both the rates that the router can request and grant.

Table 23: Message rates ranges and defaults

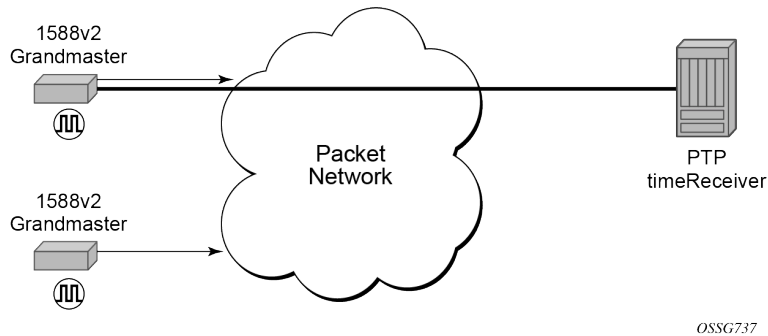
Message type	Rates requested by the 7450 ESS, 7750 SR, 7950 XRS, and VSR		Rates granted by the 7450 ESS, 7750 SR, 7950 XRS, and VSR	
	Min	Max	Min	Max
Announce	1 packet every 16 seconds	8 packets/second	packet every 16 seconds	8 packets/second
Sync	1 packet/second	64 packet/second	1 packet/second	128 packet/second
Delay_Resp	1 packet/second	64 packets/second	1 packet/second	128 packets/second
(Duration)	300	300	1	1000

State and statistics data for each PTP peer are available to assist in the detection of failures or unusual situations.

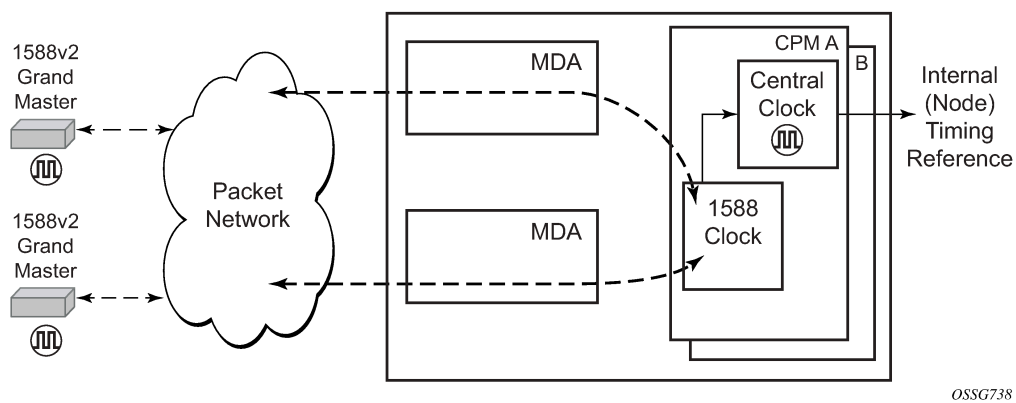
6.3.7.4 PTP ordinary timeReceiver clock for frequency

Traditionally, only clock frequency is required to ensure smooth transmission in a synchronous network. The PTP ordinary clock with timeReceiver capability on the router provides another option to reference a Stratum-1 traceable clock across a packet-switched network. The recovered clock can be referenced by the internal SSU and distributed to all slots and ports.

The following figure shows a PTP ordinary timeReceiver clock network configuration.

Figure 17: PTP ordinary timeReceiver clock for frequency

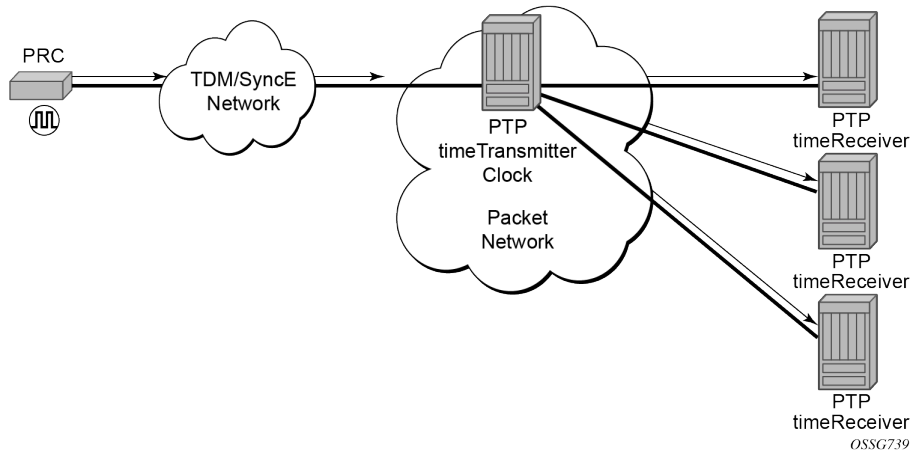
The PTP timeReceiver capability is implemented on the CPM, version 3 or later. The IEEE 1588v2 messages can ingress and egress the router on any line interface. The following figure shows the operation of an ordinary PTP clock in timeReceiver mode.

Figure 18: Ordinary timeReceiver clock operation

6.3.7.5 PTP ordinary timeTransmitter clock for frequency

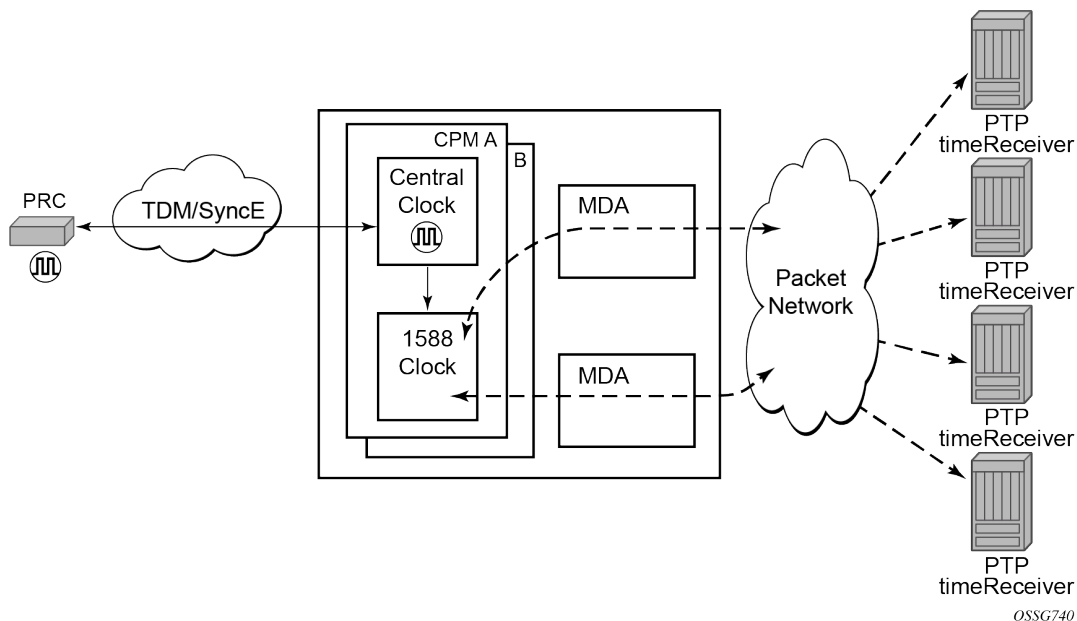
The router supports the PTP ordinary clock in timeTransmitter mode. Normally, an IEEE 1588v2 grandmaster is used to support many timeReceivers and boundary clocks in the network. In cases where only a small number of timeReceivers and boundary clocks exist and only frequency is required, a PTP integrated timeTransmitter clock can greatly reduce hardware and management costs to implement PTP across the network. It also provides an opportunity to achieve better performance by placing a timeTransmitter clock closer to the edge of the network, as close to the timeReceiver clocks as possible. The following figure shows a PTP timeTransmitter clock network configuration.

Figure 19: PTP timeTransmitter clock



All packets are routed to their destination via the best route as determined in the route table, as shown in the following figure. It does not matter which ports are used to ingress and egress these packets (unless port based time stamping is enabled for higher performance).

Figure 20: Ordinary timeTransmitter clock operation



6.3.7.6 PTP boundary clock for frequency and time

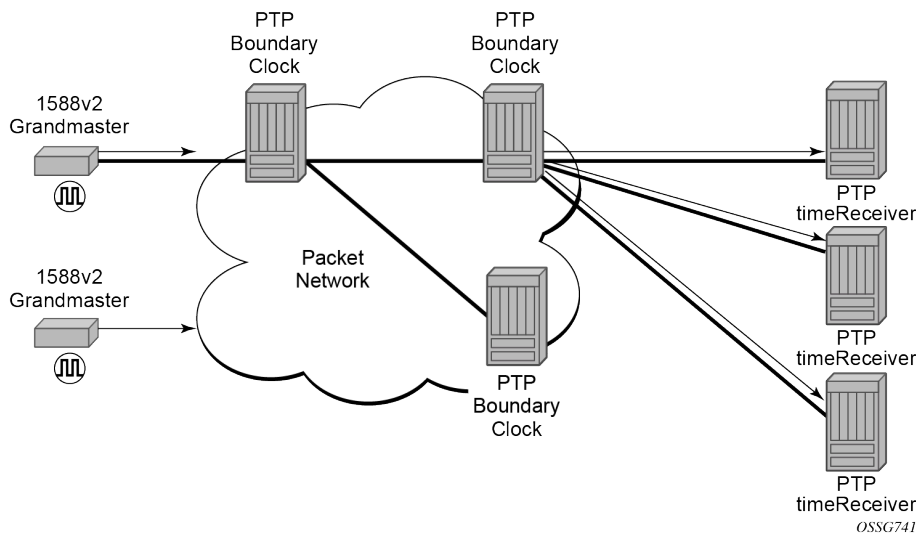
The router supports boundary clock (BC) PTP devices in both timeTransmitter and timeReceiver states.

IEEE 1588v2 can function across a packet network that is not PTP-aware. However, the performance may be unsatisfactory and unpredictable. PDV across the packet network varies with the number of hops, link speeds, utilization rates, and the inherent behavior of the routers. By using routers with boundary clock

functionality in the path between the grandmaster clock and the timeReceiver clock, one long path over many hops is split into multiple shorter segments, allowing better PDV control and improved timeReceiver performance. This allows PTP to function as a valid timing option in more network deployments and allows for better scalability and increased robustness in specific topologies, such as rings.

Boundary clocks can simultaneously function as a PTP timeReceiver of an upstream grandmaster (ordinary clock) or boundary clock, and as a PTP timeTransmitter of downstream timeReceivers (ordinary clock) and boundary clocks, as shown in the following figure.

Figure 21: PTP boundary clock for frequency and time



In addition, the use of port-based timestamping in every network element between the grandmaster and the end timeReceiver application is highly recommended for delivering time to meet one microsecond accuracies required of mobile applications.

The router always uses the frequency output of the central frequency clock to maintain the timebase within the router. When using the G.8275.1 profile, it is mandatory to have the central frequency clock configured to use a Layer 1 frequency source, such as a BITS input or a SyncE port. For other profiles, Nokia recommends using a Layer 1 frequency source. If a Layer 1 frequency source is unavailable, enable PTP as a source for the central frequency clock to have frequency for timestamping traceable to a high accuracy source.

When a router with an enabled GNSS port is configured with boundary clock functionality, the boundary clock acts as a grandmaster clock. The timeReceiver function stops and the timeTransmitter ports use frequency and time recovered from the GNSS port.

If it has lost reference, the boundary clock enters holdover mode and starts sending downstream clocks a downgraded clockClass according to the selected profile. If the lost reference was a grandmaster with clockClass of 6 (either directly or over the PTP path), the downgraded clockClass may depend on whether the profile has defined a "within holdover specifications". By default, the "within holdover specifications" is set to 10 minutes. Users can configure this time to any value between 0 and 3600 seconds.

6.3.7.7 PTP timeTransmitter clock for frequency and time distribution

PTP timeTransmitter clock capability for frequency and time distribution is implemented on the following 7750 SR equipment:

- 7750 FP5 SR-1x-48D
- 7750 FP5 SR-1-24D
- 7750 FP5 SR-1-48D
- 7750 FP5 SR-1x-92S
- 7750 FP5 SR-1-46S
- 7750 FP5 SR-1-92S
- 7750 FP5 SR-1se
- 7750 FP5 SR-2se

GNSS must be the active system timing and frequency reference for routers that are used as a grandmaster clock. The PTP timeTransmitter clock can be used for frequency and time distribution. See [Configuring system timing to use a GNSS RF port](#) for information about configuring the router to use GNSS as a system timing reference.

Use the following commands to configure the router as a grandmaster clock:

- **MD-CLI**

```
configure system ptp profile ieee1588-2008
configure system ptp clock-type master-only
configure system ptp admin-state enable
```

- **classic CLI**

```
configure system ptp profile ieee1588-2008
configure system ptp clock-type ordinary master
configure system ptp no shutdown
```

If it loses its reference, the timeTransmitter clock enters holdover and starts sending the downstream clock a downgraded clockClass according to the selected profile. If the lost reference was an integrated GNSS receiver, the downgraded clockClass may depend on the "within holdover specifications" specified by the profile. By default, the "within holdover specifications" is set to 10 minutes. Users can configure this time to any value between 0 and 3600 seconds.

6.3.7.8 ITU-T G.8275.2 profile and APTS

The 7750 SR supports Recommendation ITU-T G.8275.2, which, similar to Recommendation ITU-T G.8275.1, specifies the architecture that allows the distribution of time and phasing. Recommendation ITU-T G.8275.1 supports full-timing from the network and Recommendation ITU-T G.8275.2 supports partial-timing (PTS) and APTS.

When the 7750 SR is configured to use the G.8275.1 or G.8275.2 profile, it uses an alternate BTCA for best timeTransmitter clock selection. This BTCA includes a PTP dataset comparison that is defined in IEEE 1588-2008, but with the following differences:

- the **priority1** value is removed from the dataset comparison
- the **master-only** value must be considered
- multiple active grandmaster clocks are allowed; therefore, the BTCA selects the nearest clock of equal quality

- a port-level **local-priority** attribute value is used to select a timeReceiver port if two ports receive an Announce message. This attribute is used as a tiebreaker in the dataset comparison algorithm if all other previous attributes of the datasets being compared are equal.
- the **local-priority** value is considered for the default dataset

When the clock is configured as a boundary clock, the GNSS is treated as a virtual PTP port into the BTCA. On systems with redundant GNSS, the preference in the BTCA shall be the local GNSS followed by the standby GNSS. The GNSS receiver shall only be considered into the BTCA if it is in locked state. Also in these systems, when one of the GNSS is selected as the parent clock, there may still be a PTP port running frequency and time recovery from a remote PTP timeTransmitter port as a backup. The ptp statistics reflect this backup session.

When the PTP clock is configured to use the G.8275.2 profile without GNSS configured, the clock operates using PTS. When the PTP clock is configured to use the G.8275.2 profile and the internal GNSS is configured and operationally up, GNSS is the preferred reference by default and is selected as the source of time for PTP. For extra resilience, APTS can be deployed by configuring a PTP backup over a network with PTP support to a remote grandmaster clock.

When the clock is configured as a boundary clock, the GNSS is treated as a virtual PTP port into the BTCA. On systems with redundant GNSS, the preference in the BTCA is the local GNSS followed by the standby GNSS. The GNSS receiver is considered into the BTCA when it is in locked state. Also in these systems, when one of the GNSS is selected as the parent clock, there may still be a PTP port running frequency and time recovery from a remote PTP timeTransmitter port as a backup. The PTP statistics reflect this backup session.

During normal operation, the local GNSS source is the reference for time and frequency. If the GNSS source fails, the PTP backup is automatically used to keep the clocks synchronized and stable. The delay offset value, which is calculated while the GNSS source was up, is applied to the PTP backup to keep time and phase as accurate as possible. See [GNSS failure with APTS](#) for more information about APTS functionality during a GNSS failure.

The following table describes the mapping between ITU-T G.8275.2 and PTP clock types. T-BC-A and T-TSC-A clocks are applicable to APTS.

Table 24: Mapping between ITU-T G.8275.2 and PTP clock types

Clock type from ITU-T G.8275.2	Description	Clock type from IEEE 1588
T-GM	timeTransmitter ordinary clock (clock with a single PTP port; cannot be a timeReceiver from another PTP clock)	Ordinary clock
	timeTransmitter boundary clock (clock with multiple PTP ports; cannot be a timeReceiver from another PTP clock)	Boundary clock ¹
T-BC-P (partial)	Boundary clock (may become a grandmaster clock, or may be a timeReceiver from another PTP clock)	Boundary clock

¹ As defined by IEEE 1588, a clock with multiple PTP ports is a boundary clock.

Clock type from ITU-T G.8275.2	Description	Clock type from IEEE 1588
T-BC-A (assisted partial)	Boundary clock assisted by a local time reference that is used as a primary source of time (may become a grandmaster clock, or may be a time Receiver to another PTP clock)	Boundary clock ²
T-TSC-P (partial)	Always timeReceiver; single-port ordinary clock	Ordinary clock
	PTP clock at the end of the PTP synchronization chain; multiple port clock	Boundary clock ¹
T-TSC-A (assisted partial)	Always timeReceiver; single-port ordinary clock assisted by a local time reference that is used as a primary source of time	Ordinary clock ²
	PTP clock at the end of the PTP synchronization chain; multiple-port clock assisted by a local time reference that is used as a primary source of time	Boundary clock ^{1, 2}

6.3.7.9 PTP clock redundancy

The PTP module in the router exists on the CPM. The PTP module on the standby CPM is kept synchronized to the PTP module on the active CPM. All sessions with external PTP peers are maintained over a CPM switchover.

6.3.7.10 PTP message encapsulations

For physical ports located on an MDA/IMM/XMA/MDA-s of a 7750 SR or a 7950 XRS platform, PTP messages are supported in the following encapsulations:

- PTP within raw Ethernet (no VLAN header)
- PTP within UDP/IPv4 or UDP/IPv6 within raw Ethernet (no VLAN header)
- PTP within UDP/IPv4 or UDP/IPv6 within dot1q Ethernet (one VLAN header)

For the SyncE/1588 ports located on the CPM or CCM of a 7750 SR or a 7950 XRS platform, PTP supports PTP messages within raw Ethernet (no VLAN header).

For physical ports located on a 7210 SAS satellite, PTP supports PTP messages within raw Ethernet (no VLAN header).

No other encapsulations are supported.

6.3.7.11 PTP time for system time and OAM time

PTP has the potential to provide much more accurate time into the router than can be obtained with NTP. This PTP recovered time can be made available for system time and OAM packet timestamping to improve

² Examples of a local time reference include PRTC or a GNSS-based time source.

the accuracy of logged events and OAM delay measurements. To activate PTP as the source for these internal time bases, allocate PTP as a local server into NTP. This allows the NTP time recovery to use PTP as a time source and then distribute it to system time and the OAM process within the router. This activation also affects the operation of the NTP server within the SR OS. The PTP server appears as an NTP stratum 0 server. Consequently the SR OS advertises itself as an NTP Stratum 1 server to external peers and clients. This activation may impact the NTP topology.

6.3.7.12 PTP within routing instances

PTP is supported over direct Ethernet encapsulation (that is, PTP ports) and UDP/IP encapsulation (that is, PTP peers). PTP ports operate below the routing plane. They can be used on appropriate ports irrespective of any type of router interface also on the port. PTP peers operate at the routing plane and have restrictions based on and across the following router instances.

Transmission and reception of PTP messages using PTP peers is supported in the following contexts:

- Network interface in the Base routing instance (**configure router interface**)
- IES interface (**configure service ies interface**)
- VPRN interface (**configure service vprn interface**)

Transmission and reception of PTP messages using PTP peers is not supported in the following contexts:

- IES spoke SDP interface (**configure service ies spoke-sdp interface**)
- VPRN spoke SDP interface (**configure service vprn spoke-sdp interface**)
- VPRN transport tunnel (**configure service vprn auto-bind-tunnel** or **configure service vprn spoke-sdp**)
- Any interface of the management router instance
- Any interface of the vpls-management router instance
- Any interface of a user created CPM router instance

It is important to note that there is only one PTP clock within the router. All PTP ports and PTP peers communicate into one clock instance. Only one router instance may have PTP peers configured, which means that only that router instance (or PTP port) can run the timeReceiver functionality and recover time from an external PTP clock. All other router instances only support the dynamic PTP peers. The PTP process in the router only includes outward server time toward the dynamic PTP peers. The dynamic PTP peers are shared across all router instances. If it is needed to control the number of dynamic peers that can be consumed by a routing instance, then it must be configured for that routing instance.

6.3.7.13 PTSF conditions for G.8275.1

The PTP clock in the router monitors the Sync, Follow_Up (if present), and Delay_Resp messages received from external neighbor ports. If a high variation is detected in the network path between the external neighbor port and the local port, that neighbor port is considered unusable (PTSF-unusable as defined in the ITU-T G.8275.1 recommendation). When a neighbor is unusable, all Announce messages from that neighbor are discarded on reception and excluded from the BTCA. If the neighbor is the parent clock to the local clock, the local clock must either select a new parent clock or go into holdover. In addition, any neighbor clock marked as unusable cannot act as the parent to the local PTP clock until underlying condition is investigated and resolved, and the unusable state is cleared.

Use the commands in the following context to enable monitoring for the PTSF-unusable condition.

```
configure system ptp ptsf monitor-ptsf-unusable
```

The unusable state is cleared when PTP, PTSF-unusable monitoring, or the local PTP port is administratively disabled, the PTP port is deleted, or the external neighbor port stops sending messages to the node. It can also be cleared by using the appropriate **clear** command.

The PTP clock also monitors whether the timing messages that it expects from an external neighbor port continue to arrive. If those messages stop arriving, the neighbor port is considered to have experienced loss of timing messages (PTSF-LOTM, defined as PTSF-lossOfTimingMessages in the ITU-T G.8275.1 recommendation).

Use the **sync-receipt-timeout** and **delay-resp-receipt-timeout** commands in the following contexts to configure the number of intervals after receiving Sync and Delay_Resp messages to declare a neighbor as PTSF-LOTM.

- **MD-CLI**

```
configure system ptp router peer
configure system ptp port
```

- **classic CLI**

```
configure service vprn ptp peer
configure system ptp peer
configure system ptp port
```

Expiry of either timeout sets the read-only PTSF-LOTM state for the affected PTP port or peer. A port or peer in the PTSF-LOTM state is excluded from the BTCA in the same way as a neighbor that is marked unusable, as previously described.

Upon receiving the expected messages from the neighbor again, the system clears the PTSF-LOTM state and restarts the corresponding receipt timeout. The state is also cleared when PTP or the local PTP port is administratively disabled, when the PTP port or configured peer is deleted, or when the corresponding receipt timeout command is disabled.

6.3.7.14 Profile interworking

There is one PTP clock within an SR OS system. The clock runs a BTCA and can set a port into timeReceiver state to recover frequency, time, or both. The recovery process is controlled by the rules of a primary profile. The SR OS also allows frequency, time, or both to be distributed outward from the clock using PTP messages that conform to the rules of an alternate profile. The primary profile and alternate profiles include a parameter to configure the standard profile that defines these rules. The following guidelines apply to profile interworking:

- Alternate profiles can only be configured if the primary profile is using standard profile **g8275dot1-2014** or **g8275dot2-2016**.
- Alternate profiles can use standard profiles **g8275dot1-2014**, **g8275dot2-2016**, **g8265dot1-2010**, or **ieee1588-2008**.
- The primary profile and an alternate profile can use the same standard profile.
- Multiple alternate profiles can also use the same standard profile.



Note: The last two cases described in the preceding list may be useful if external clocks require different domain numbers or announce message rates.

Alternate profiles are associated with PTP ports and peers. Alternate profile associations are configured for PTP ports and learned by PTP peers. PTP peer alternate profiles are learned by matching the domain number of received unicast messaging with the domain of the configured alternate profile. Configured peers always use the primary profile.



Caution: When a PTP port is configured, the **log-delay-interval** and **log-sync-interval** commands are automatically configured based on the primary PTP profile that is configured. When an alternate profile is assigned to the PTP port, these values are not changed. The user must configure the **log-delay-interval** and **log-sync-interval** to align with the requirements of the attached equipment. Additionally, if the **log-delay-interval** and **log-sync-interval** commands are not changed from their default values, the values change if the primary profile changes. This behavior may result in an unexpected message rate if the default values are retained.

The following table lists the profile interworking between the primary and alternate profiles depending on the standard profile used in the primary.

Table 25: Allowed standard profiles in alternate profile

Standard profile used in primary profile	Standard profile allowed in alternate profile
g8275dot1-2014	g8275dot1-2014 g8275dot2-2016 g8265dot1-2010 ieee1588-2008
g8275dot2-2016	g8275dot1-2014 g8275dot2-2016 g8265dot1-2010 ieee1588-2008
ieee1588-2008	not allowed
g8265dot1-2010	not allowed

6.3.7.15 Annex J performance monitoring statistics

Support is provided for the collection of performance monitoring statistics for the time recovery algorithm based on Annex J/IEEE1588-2019. The following table describes the record index values.

Table 26: Performance monitor record index values

Record index value	Record shown
0	current 15-minute interval

Record index value	Record shown
1-96	15-minute interval within the last 24 hours
97	current 24-hour interval
98	previous 24-hour interval
501	current minute interval
502-516	one-minute interval within the last 15 minutes

Each record includes the average, minimum, maximum, and standard deviation for the following statistics:

- offset-from-master
- mean-path-delay
- timeTransmitter-to-timeReceiverDelay (master-to-slave-delay)
- timeReceiver-to-timeTransmitter delay (slave-to-master-delay)

These performance statistics are available in the following scenarios:

- local active timeReceiver port to remote PTP parent
- local active GNSS to backup PTP port
- local active timeReceiver port to other selected local PTP ports

In the local active timeReceiver port to remote PTP parent scenario, the local active timeReceiver port and the remote parent timeTransmitter port are compared. In a system where the PTP has stabilized, the offset-from-master tends toward 0. A high value indicates that the PTP path has high packet delay variation or high asymmetry.

In the local active GNSS to backup PTP port scenario, when a platform includes a GNSS receiver that is being used as a PTP time source with G.8275.1 or G.8275.2 as the PTP profile, the PTP clock runs a PTP timeReceiver port with an external timeTransmitter port, if one is available. If the PTP clock runs a PTP timeReceiver port, the Annex J statistics are computed as a comparison between the timeReceiver computed time and the GNSS time. Computing the Annex J statistics this way can provide a good indication for PTP performance should the GNSS lose lock and the node switches to use PTP input as its source of time.

In the local active timeReceiver port to other selected local PTP ports scenario, the local active timeReceiver port and another timeReceiver port in the router are compared. Up to four timeReceiver ports can be monitored simultaneously. In this case, these timeReceiver port are designated to be monitored by configuring the PTP port (or PTP peer) to be a monitorReceiver. At the remote end of these monitored PTP ports, the timeTransmitter port can be configured to be monitorSender so that the monitor process can proceed correctly, no matter the PTP state of the port.

6.3.7.16 PTP path trace

SR OS supports PTP path trace of IEEE1588-2019 clause 16.2. This feature records and displays the chain of PTP clocks from the local clock up to the grandmaster clock.

A node-level configuration option allows the user to include or exclude the PATH_TRACE TLV. If this feature is enabled and the Announce messages received from the parent clock does not include the

TLV, the SR OS boundary clock creates the TLV with the local clock identity and includes that TLV in the Announce message transmitted by the node.

SR OS supports a maximum length chain of 40 clocks in the PATH_TRACE TLV. If this number is exceeded, SR OS includes only the closest 40 clock identifiers in the transmitted PATH_TRACE TLV.

6.3.8 Synchronization with Ethernet satellites

Synchronous Ethernet (SyncE) is supported on Ethernet satellites. Support is provided for both the distribution of frequency from the central clock of the host outward via client ports and also the option to use one or two client ports of the satellite as options to feed inward to the central frequency clock of the host.

Configuration of the central frequency clock of the satellite shall be either automatic or manual. When the satellite is first configured for SyncE support, the mode shall be automatic where the clock is configured to use its uplink ports as the two references and lock to one of these for feeding frequency synchronization outward through the client ports. As soon as any configuration is changed, for example, changing to use one of the client ports for input, the mode is changed to manual and it remains in that mode until the configuration is removed.

If **sync-e** is configured, under the following contexts, the settings for **ref1** and **ref2** can be created:

- **MD-CLI**

```
configure satellite ethernet-satellite sync-e
```

- **classic CLI**

```
configure system satellite eth-sat sync-e
```

The **ref1** and **ref2** commands are configured under the following contexts:

- **MD-CLI**

```
configure satellite ethernet-satellite central-frequency-clock
```

- **classic CLI**

```
configure satellite ethernet-satellite sync-if-timing
```

The source ports of **ref1** and **ref2** of the satellite clock can be configured to both be uplink ports (frequency synchronization flows from host system into satellite) or client ports (frequency synchronization always derived from local client ports, never the host) or one host port and the other a client port. The satellite central clock always uses **ql-selection** so Nokia strongly recommends having ESMC frames enabled on the devices feeding SyncE toward the satellite client ports.

There are two settings for a satellite when using PTP from the host across a satellite. The satellite must be enabled to support updating the PTP event message correction field. Setting **ptp-tc** to enabled turns on support for use of PTP over Ethernet from the host across the satellite.

Use the following command to enable **ptp-tc**:

- **MD-CLI**

```
configure satellite ethernet-satellite feature ptp-tc
```

- **classic CLI**

```
configure system satellite eth-sat feature ptp-tc
```

When used with PTP over Ethernet from the host, the performance is optimized. This allows for the inclusion of the host and satellite system in a network chain targeting one microsecond accurate time in the end application (see ITU-T recommendation series G.827x). PTP over IP can be used with this solution, but the performance is much less than with PTP over Ethernet and one microsecond accurate time cannot be guaranteed.

When a satellite is enabled for **ptp-tc**, all PTP messages encapsulated in the raw Ethernet frames that any of the satellite client ports receive must be destined for the 7750 SR host PTP process. If the PTP messages are not extracted for the host PTP process, those messages are corrupted.

To use PTP over IP from the host across a satellite, in addition to enabling **ptp-tc**, the command options in **ptp-ip** also need to be configured. This subtree allows the PTP over IP updating to be enabled and defines the specific IPv4 address, IPv6 address, or both IPv4 and IPv6 addresses that are used by the PTP over IP packets. Any PTP messages that use IPv4 must match the IPv4 address and any PTP messages that use IPv6 must match the IPv6 address. PTP over IP messages can enter the satellite client ports with raw or dot1q encapsulation; QinQ encapsulation for these messages is not supported. The IPv4 and IPv6 address used for PTP must be common for all PTP messages for the host PTP process. The routing interfaces can exist in different routing instances, but the address used for PTP across those routing instances must be the same.

6.4 QinQ network interface support

Use the following command to enable the creation of network interfaces on a QinQ-encapsulated VLAN on a system-wide level.

```
configure system ip allow-qinq-network-interface
```

When enabled, the egress IOM limits are changed to allow a maximum of 11 MPLS labels instead of 12.

[Table 27: QinQ combination \(✓\) and restriction \(x\) table](#) lists the allowed and restricted QinQ combinations.

Table 27: QinQ combination (✓) and restriction (x) table

	SAP x.0	SAP x.*	SAP x.y	Nw interface x.0	Nw interface x.*	Nw interface x.y	SAP *.*	SAP *.NULL	SAP 0.*	Inverse SAP
SAP x.0	x	✓	✓	x	x	x	✓	✓	✓	x
SAP x.*	✓	x	✓	x	x	x	✓	✓	✓	x
SAP x.z	✓	✓	✓	x	x	✓	✓	✓	✓	✓
Nw interface x.0	x	x	x	x	x	✓	✓	✓	✓	x

	SAP x.0	SAP x.*	SAP x.y	Nw interface x.0	Nw interface x.*	Nw interface x.y	SAP *.*	SAP *.NULL	SAP 0.*	Inverse SAP
Nw interface x.*	x	x	x	x	x	x	✓	✓	✓	x
Nw interface x.z	x	x	✓	✓	x	✓	✓	✓	✓	x
SAP *.*	✓	✓	✓	✓	✓	✓	x	✓	✓	✓
SAP *.NULL	✓	✓	✓	✓	✓	✓	✓	x	✓	x
SAP 0.*	✓	✓	✓	✓	✓	✓	✓	✓	x	x
Inverse SAP	x	x	✓	x	x	x	✓	x	x	x

6.5 LLDP

The IEEE 802.1ab Link Layer Discovery Protocol (LLDP) is a unidirectional protocol that uses the MAC layer to transmit specific information related to the capabilities and status of the local device. LLDP can send and receive information from a remote device stored in the related MIBs.

LLDP does not contain a mechanism for soliciting information received from other LLDP agents, nor does it provide the means to confirm the receipt of information. LLDP provides the flexibility of enabling a transmitter and receiver separately. The following LLDP configurations are allowed.

- An LLDP agent can only transmit information.
- An LLDP agent can only receive information.
- An LLDP agent can transmit and receive information.

The information fields in each LLDP frame are contained in an LLDP Data Unit (LLDPDU) as a sequence of variable length information elements, which each include type, length, and value fields (TLVs).

- Type indicates the nature of information being transmitted.
- Length indicates the length of the information string in octets.
- Value is the information that is transmitted (for example, a binary bit map or an alphanumeric string that can contain one or more fields).

Each LLDPDU contains four mandatory TLVs and can contain optional TLVs as selected by the NMS, as follows:

- Chassis ID TLV
- Port ID TLV
- Time-To-Live TLV
- End of LLDPDU TLV

- Zero or more optional TLVs, depending on the maximum size of the LLDPDU allowed.

An LLDP agent or port is identified by a concatenated string formed by the Chassis ID TLV and the Port ID TLV. This string is used by a recipient to identify an LLDP port or agent. The combination of the Port ID and Chassis ID TLVs remains unchanged until the port or agent is operational.

The TTL (Time To Live) field of a Time-To-Live TLV can be either zero or a non-zero value. A zero value in the TTL field notifies the receiving LLDP agent to immediately discard all information related to the sending LLDP agent. A non-zero value in the TTL field indicates the time duration for which the receiving LLDP agent should retain the information related to the sending LLDP agent. The receiving LLDP agent discards all information related to the sending LLDP agent after the time interval indicated in the TTL field is complete.



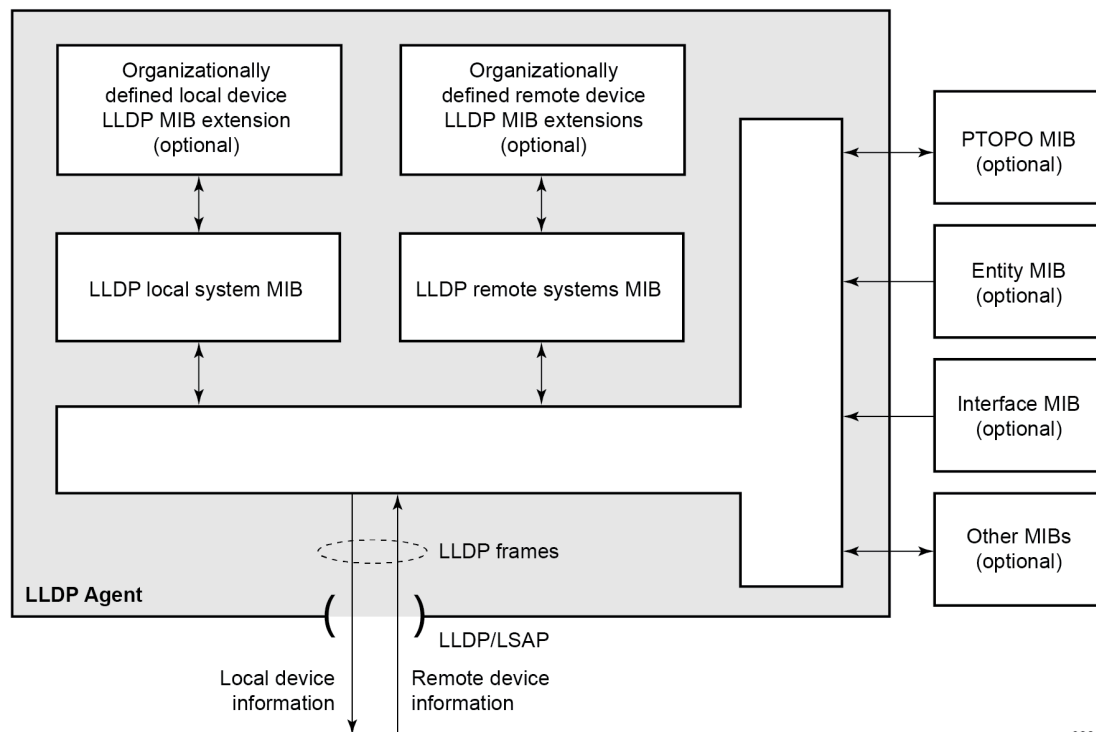
Note: A TTL value of zero can be used to signal that the sending LLDP port has initiated a port shutdown procedure.

The End Of LLDPDU TLV indicates the end of the LLDPDU.

The IEEE 802.1ab standard defines a protocol that:

- advertises connectivity and management information about the local station to adjacent stations on the same IEEE 802 LAN
- receives network management information from adjacent stations on the same IEEE 802 LAN
- operates with all IEEE 802 access protocols and network media
- establishes network management information schema and object definitions that are suitable for storing connection information about adjacent stations
- Provides compatibility with a number of MIBs as depicted in [Figure 22: LLDP internal architecture for a network node](#).

Figure 22: LLDP internal architecture for a network node

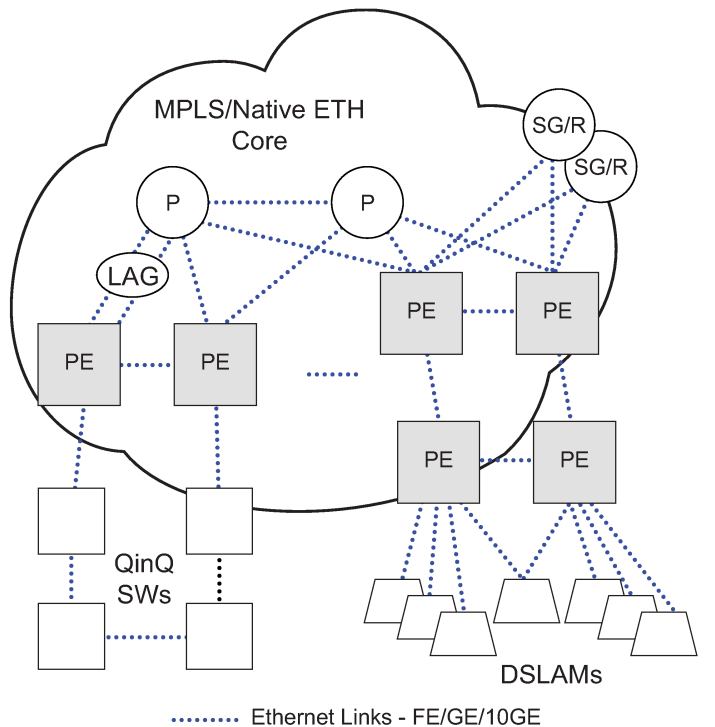


0981

Network operators must be able to discover the topology information to detect and address network problems and inconsistencies in the configuration. Moreover, standard-based tools can address the complex network scenarios where multiple devices from different vendors are interconnected using Ethernet interfaces.

The example displayed in [Figure 23: Customer use example for LLDP](#) depicts a MPLS network that uses Ethernet interfaces in the core or as an access/hand off interfaces to connect to different kind of Ethernet enabled devices such as service gateway/routers, QinQ switches, DSLAMs or customer equipment.

Figure 23: Customer use example for LLDP



OSSG263

IEEE 802.1ab LLDP running on each Ethernet interfaces in between all the above network elements may be used to discover the topology information.

6.6 IP hashing as an LSR

It is now possible to include IP header in the hash routine at an LSR for the purpose of spraying labeled-IPv4 and labeled-IPv6 packets over multiple equal cost paths in ECMP in an LDP LSP and over multiple links of a LAG group in all types of LSPs.

A couple of configurable options are supported. The first option is referred to as the Label-IP Hash option and is designated in the CLI as **lbl-ip**. When enabled, the hash algorithm parses down the label stack and after it hits the bottom of the stack, it checks the next nibble. If the nibble value is four or six then it assumes it is an IPv4 or IPv6 packet. The result of the hash of the label stack, along with the incoming port and system IP address, is fed into another hash along with source and destination address fields in the IP packet's header. The second option is referred to as IP-only hash and is enabled in CLI using **ip-only**. It operates the same way as the Label-IP Hash method except the hash is performed exclusively on the source and destination address fields in the IP packet header. This method supports both IPv4 and IPv6 payload.

By default, MPLS packet hashing at an LSR is based on the whole label stack, along with the incoming port and system IP address. This method is referred to as the Label-Only Hash option and is enabled by entering **lbl-only**.

Use the following context to configure **lbl-only**, **lbl-ip**, and **ip-only** on a system-wide basis or override them on a per-IP-interface basis.

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

6.7 Satellites

The 7210 SAS and 7250 IXR satellites are supported on the 7750 SR and 7950 XRS as Ethernet satellites.

Satellites act as port extenders to the 7750 SR or 7950 XRS through an external chassis, without requiring management of the external chassis. Satellites are a logically integrated part of the 7750 SR chassis and share a single IP address. They are configured on the 7750 SR or 7950 XRS and use the SR OS host to:

- apply all service-level QoS policies
- configure deep buffering and granular queuing
- shape all egress traffic to the configured satellite access command options

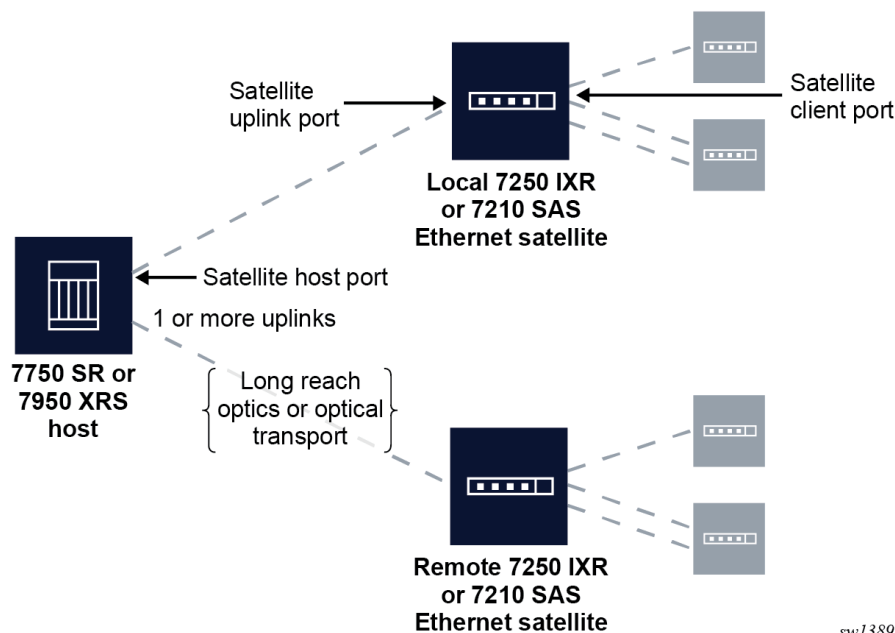


Note: If a user configures CRC and symbol monitoring, the system uses the CRC and symbol monitoring Signal Degrade (SD) state to pick the best available host port for communicating with the management plane of the downstream satellite.

Satellites can be colocated with the 7750 SR or 7950 XRS or remotely located within the same building, campus, or within a several-kilometer range. This range is determined by the pluggable optics. The connection between the host and satellite must be a direct Ethernet connection.

The following figure shows the satellite solution capabilities.

Figure 24: Satellite solution capabilities



Perform the following primary tasks to configure a satellite:

1. Create a software repository that specifies where the satellite obtains its correct software image.
2. Create an Ethernet satellite association that binds a chassis to a set of uplinks and a software repository.
3. Configure the satellite ports to specify port configuration and service association.

6.7.1 Ethernet satellites

The Ethernet satellite support feature allows the following chassis to act as a port extension for the 7750 SR or 7950 XRS host:

- 7210 SAS-Sx
- 7210 SAS-S
- 7250 IXR-X1/7250 IXR-Xs
- 7250 IXR-s
- 7250 IXR-e 24SFP+ 8SFP28 2QSFP28

In this configuration, the host node performs all configuration and management functions. Management of the satellite node is not required when it is configured in an Ethernet satellite operations mode. A direct, non-switched, Ethernet connection between the 7750 SR or 7950 XRS host and the Ethernet satellite must be provided. The use of active Layer 2 switching devices in the path between the host and satellite is not supported.

[Table 28: Supported 7210 SAS Ethernet satellite chassis](#) and [Table 29: Supported 7250 IXR Ethernet satellite chassis](#) list the supported Ethernet satellite chassis.

Table 28: Supported 7210 SAS Ethernet satellite chassis

Chassis type	SAT-type string
7210 SAS-Sx 24-port fiber	es24-1gb-sfp
7210 SAS-Sx 48-port fiber	es48-1gb-sfp
7210 SAS-S 24F4SFP+	es24-sass-1gb-sfp
7210 SAS-S 48F4SFP+	es48-sass-1gb-sfp
7210 SAS-Sx 24-port copper 7210 SAS-S 24-port copper	es24-1gb-tx
7210 SAS-Sx 48-port copper 7210 SAS-S 48-port copper	es48-1gb-tx
7210 SAS-Sx 24-port copper + PoE 7210 SAS-S 24-port copper + PoE	es24-1gb-tx
7210 SAS-Sx 48-port copper + PoE 7210 SAS-S 48-port copper + PoE	es48-1gb-tx

Chassis type	SAT-type string
7210 SAS-Sx 64-port 10GE + 4-port QSFP28	es64-10gb-sfpp+4-100gb-qsfp28
7210 SAS-Mxp	es24-sasmxp-1gb-sfp

Table 29: Supported 7250 IXR Ethernet satellite chassis

Chassis type	SAT-type string
7250 IXR-Xs	es6-qsfpdd+48-sfp56
7250 IXR-X1	es32-qsfp28+4-qsfpdd
7250 IXR-s	es48-sfpp+6-qsfp28
7250 IXR-e 24SFP+ 8SFP28 2QSFP28	es24-sfpp+8-sfp28+2-qsfp28

**Note:**

- The 7210 SAS-Sx 64-port 10GE Ethernet satellite supports both 10GE and 1GE optics. See the *7210 Optics Guide* for a list of supported modules.
- The 7210 SAS-Mxp, 64x10GE + 4xQSFP28 7210 SAS-Sx, and 7250 IXR satellites do not support the local-forwarding feature.
- PoE functionality is not supported when the 7210 SAS PoE-capable switches are used in satellite mode.
- For traffic sent by the 7750 SR or 7950 XRS host to the 7210 SAS satellite, the satellite Q-tag P-bits and DEI bits are set based on the forwarding class and profile associated with the traffic through the 7750 SR or 7950 XRS system.
- For CRC monitoring on Ethernet satellite ports, limit the setting of signal-failure thresholds to only a subset of the uplinks. Setting the signal-failure threshold on all the links could result in satellite isolation if CRC errors are seen simultaneously on all uplinks.
- Dual-homing or multihoming of satellites to two or more hosts is not supported.

6.7.2 Software repositories for satellites

The software repositories define the locations from where the host can obtain software for subcomponents including Ethernet satellites. The software repository is also used to upgrade an existing subcomponent by changing the location of the image to be served to the remote device. The software repositories are not used for management of the host router software, which is managed using the standard procedures described in the *SR OS R26.x.Rx Software Release Notes*.

Each software repository supports up to three locations to search for the software. A location may be a URL or a directory on a compact flash. When an upgrade operation is initiated, each of the three locations is checked in sequence to locate the required software. The upgrade operation fails if the software is not located in any of the configured locations. The satellite booting operation also fails if the software cannot be located.

At least one software repository must be configured to support a satellite connected to the local host as follows:

1. Create a software repository using a unique repository name.

2. Specify the primary location for the SAS/IXR image.
3. Optionally, specify a secondary or tertiary image location and a description:

- **MD-CLI**

```
configure system software-repository
configure satellite ethernet-satellite software-repository
```



Note: First configure the software repository in the **system** context and then reference it in the **satellite software-repository** context.

- **classic CLI**

```
configure system software-repository
configure system satellite eth-sat software-repository
```



Caution: Software for TDM satellites and Ethernet satellites should be stored in separate software repositories. There is one file that has the same name for both types of software, that is overwritten if they are placed in the same repository.

6.7.3 Upgrading satellite software

About this task

This procedure describes how to change or upgrade the satellite software.

Procedure

Step 1. Copy the new satellite software images to a local compact flash card. It is recommended that the new image files be placed in a different directory.

Although you can store the satellite software on a remote server and use a URL to reference the remote location, Nokia recommends that the primary image location is locally accessible.

Step 2. Create a new software repository using a new name and at least a primary-location for the 7210 SAS/7250 IXR image.

Step 3. Use the following contexts to modify the satellite configuration such that the **software-repository** references the newly created software repository.

- **MD-CLI**

```
configure satellite ethernet-satellite
```

- **classic CLI**

```
configure system satellite eth-sat
```

Step 4. Reboot the satellite to load the new software. Depending on whether a firmware update is needed, perform one of the following steps to reboot the satellite.

- If a satellite firmware update is not required:
 - a. The satellite loads the new software the next time it reboots.
 - b. If required, use the following administrative command to reset the satellite:

- **MD-CLI**

```
admin satellite ethernet-satellite reboot now
```

- **classic CLI**

```
admin satellite eth-sat reboot now
```

- If a satellite firmware update is required:

- a. Use one of the following commands to continue the upgrade to the 7210 firmware image and allow it to execute completely:

- **MD-CLI**

```
admin satellite ethernet-satellite synchronize
```

- **classic CLI**

```
admin satellite eth-sat sync-boot-env
```

- b. Use the following command to reboot the satellite to update the firmware image. This causes the 7210 SAS-Sx to upgrade the included firmware images. This process takes longer than a normal reboot.

- **MD-CLI**

```
admin satellite ethernet-satellite reboot upgrade
```

- **classic CLI**

```
admin satellite eth-sat reboot upgrade now
```

6.7.4 Provisioning a 7250 IXR satellite

Prerequisites

- Compact flash with the ZTP software kit for the 7250 IXR chassis (Release 22.10 or later)
- 7250 IXR MAC address printed on the chassis label or from the console during the ZTP autoboot

About this task

This task describes the recommended provisioning model for 7250 IXR satellites using the 7250 IXR ZTP boot mechanism.

Procedure

- Step 1.** Insert the compact flash with the ZTP software kit in the 7250 IXR chassis (Release 22.10 or later).
- Step 2.** Connect the associated uplinks between the 7750 SR or 7950 XRS host and the 7250 IXR chassis and power on the 7250 IXR chassis.
- Step 3.** Obtain the MAC address from the 7250 IXR.
This information is printed on the chassis label or it displays on the console during ZTP autoboot.

Step 4. Configure a software repository containing the 7250 IXR images.

Step 5. Use the commands in the following context to configure and enable the satellite on the host:

- **MD-CLI**

```
configure satellite ethernet-satellite
```

- **classic CLI**

```
configure system satellite eth-sat
```

Step 6. Configure and enable the satellite uplinks (breakout, RS-FEC, and admin status enabled) and establish the port topology for the uplink-to-host port.

6.7.5 Synchronization features with satellites

See [Synchronization with Ethernet satellites](#) for restrictions on SyncE and PTP when using satellites.

6.7.6 Satellite configuration

After creating the software repositories, configure the satellite. The satellite configuration is required to create a satellite binding to a satellite ID, and to provide more information that uniquely identifies the satellite chassis, chassis type, and the software repository to be used to boot the remote satellite.

The following can be specified for a satellite:

- **MAC address**

The satellite chassis MAC address must be specified. This is used to bind a specific chassis to the associated satellite ID. (The local host router boots only satellites with configured MAC addresses.) This is mandatory.

- **satellite type**

The satellite chassis type must be specified and must match the chassis type that the satellite advertises during the boot process. This is mandatory.

- **software repository**

A preconfigured software repository must be specified in the satellite configuration. This defines the location of the software image to boot the associated 7210 SAS-Sx. This is mandatory.

- **enabled state**

By default, a new satellite is in the disabled state and must be administratively enabled. This is mandatory.

- **description**

Configure an optional description string associated with the satellite.

- **sync-e**

Enable **sync-e** for an Ethernet satellite.



Note: For some Ethernet satellite platforms, before you can reference the uplinks you must configure the breakout connection on the Ethernet satellite uplink ports (and any other uplink

ports). You can only configure the connector after you configure the satellite on the host. Only specific platforms require this command option, for example, the 7210 SAS-Sx 10/100GE.

6.7.6.1 Satellite client and uplink port ID formats

When referencing satellite ports, always use 1 for the slot number.
Use the following format to reference Ethernet satellite client ports.

```
configure port esat-sat-id/1/portNum
```

Example: Ethernet satellite client ports

```
configure port esat-4/1/2
```

Use the following format to reference Ethernet satellite uplink ports.

```
configure port esat-sat-id/1/uportNum
```

Example: Ethernet satellite uplink ports

```
configure port esat-5/1/u2
```

Ethernet satellite client ports support all port modes (access, network, and hybrid).
Configuring services associated with satellite client ports is the same as configuring services on local 7750 SR ports, except that satellite client ports are referenced with the syntax for the Ethernet satellite port described in the preceding information. Users must create a **port-scheduler-policy** to ensure that the 7750 SR is able to shape traffic for the egress satellite port type and speed.

6.7.6.2 Port naming conventions

Consider the following information when configuring and naming satellite client and uplink ports.

Client ports

Client ports connect the satellite to end devices.

Table 30: client ports

Port format	Satellites where supported
esat-x / y / port-id	Satellite client ports that do not support connectorized optics
esat-x / y / c-z / connector-id	Satellite client ports that accept connectorized optics (for instance, QSFP type optics)

Where:
x = the satellite ID
y = reserved for possible satellite with multiple slots, currently always 1

port-id = the physical port number on the satellite chassis

z = ports that accept connectors are designated by *c-port-id*, the *port-id* is the physical port or cage number

connector-id = the breakout connector from the QSFP-type optic

Example: client port naming

The following example displays a client port with satellite ID 3 and port 64 that has a 4x100 breakout referred to as connector 2.

esat-3/1/c64/2

Uplink ports

Uplink ports connect the satellite to the host router. Uplink port IDs use the same format as client ports, but have an additional modifier "u" to designate that the associated port is an uplink.



Note: A connectorized port cannot mix client and uplink ports.

Table 31: uplink ports

Port format	Satellites where supported
esat-<i>x</i> / <i>y</i> / <i>uport-id</i>	Satellite uplink ports that do not support connectorized optics
esat-<i>x</i> / <i>y</i> / <i>c-z</i> / <i>uconnector-id</i>	Satellite uplink ports that accept connectorized optics (for instance, QSFP type optics)

Where:

x = the satellite ID

y = reserved for possible satellite with multiple slots, currently always 1

port-id = the physical port number on the satellite chassis

z = ports that accept connectors are designated by *c-port-id*, the *port-id* is the physical port or cage number

u = indicates an uplink port

connector-id = the breakout connector from the QSFP-type optic



Note:

On a satellite that supports QSFP-type connectors for uplinks, the format is the same as client ports, with the addition of the "u" designator.

Example: uplink port naming

The following example displays an uplink port with a satellite ID of 2 and port 65 as an uplink.

esat-2/1/u65

Example: uplink port using a connector naming

The following example displays an uplink port with a satellite ID of 4 and connector 34 (4x100 breakout). The associated uplinks are as follows.

esat-4/1/c34/u[1..4]

6.7.6.3 Local forwarding



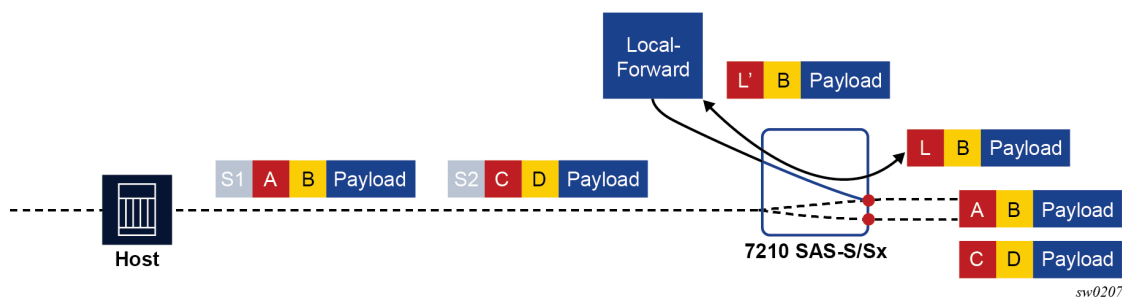
Note: This information applies to the classic CLI.

The local forwarding capability allows traffic to be forwarded between two client satellite ports without going through the SR host, which allows for optimal forwarding by preserving uplink bandwidth:

- Locally forwarded traffic is identified based on the ingress VLAN tag.
- The outer VLAN tag used to identify the traffic to be locally forwarded can be different at the two bypass endpoints. In that case, as traffic is forwarded from the ingress to the egress, the outer VLAN tag is modified.
- The bypass paths are bidirectional, so only a single local-forwarding path needs to be defined to allow for traffic flow in both directions.

Figure 25: Local forwarding shows an example of local forwarding.

Figure 25: Local forwarding



A local-forward bypass is created by using the following commands to create a local-forward bypass, then associating a set of two satellite access points as endpoints for the local-forward bypass:

- The two endpoints must be ports on the same Ethernet satellite chassis.
- If a LAG is used as an endpoint, all member links must be ports on the same Ethernet satellite.
- All satellite ports must be client ports by default, or must be configured as a client port using the port-template command.

The following example shows the commands to configure a local-forward bypass between client ports esat-2/1/1:66 and esat-2/1/50:101.

Example: classic CLI

```
A:node-2>config>system>satellite# info
-----
    local-forward 10 create
      description "local-forward to offload router"
      sap esat-2/1/1:66 create
      exit
      sap esat-2/1/50:101 create
      exit
      no shutdown
    exit
-----
```

6.7.6.4 Port template

The **port-template** command hierarchy allows the creation of a satellite template that reconfigures the port role and uplink association for one or more satellite ports. This template can then be applied to one or more Ethernet satellite instances, in which case those satellites inherit the specified port role and uplink associations.

The port template is necessary when reconfiguring a satellite uplink as a client port for use as part of a local-forward bypass path. Use the following command to create satellite templates:

- **MD-CLI**

```
configure satellite port-template
```

- **classic CLI**

```
configure system satellite port-template
```

6.7.6.5 7210 SAS 10GE client ports

Ports 51 and 52 on the 48xGE + 4x10GE satellite chassis can be reassigned as client ports instead of uplink ports. This provides the flexibility to offer 10GE services from these satellite chassis. These two 10GE ports can be reconfigured as client ports using the **port-template** configuration commands described above. The port template configuration must be done before SAPs, interfaces, or services can be applied to the associated satellite ports.

6.7.6.6 7210 SAS 100GE client ports

Connectors 3 and 4 on the 64x10GE+4xQSFP28 (sat-type es64-10gb-sfpp+4-100gb-qsfp28) can be reassigned as client ports instead of uplinks. This provides the flexibility to offer 100GE services from these satellite chassis. These two 100GE ports can be reconfigured as client ports using the **port-template** configuration commands. The port template must be configured before port topology bindings are configured as well as before SAPs, interfaces, or services can be applied to the associated satellite ports.

6.7.6.7 10GE uplinks on the 64x10GE+4xQSFP28 satellite

On the 7210 SAS-Sx 64x10GE+4xQSFP28 (sat-type es64-10gb-sfpp+4-100gb-qsfp28) satellite, selected 10GE ports can be reconfigured and used as satellite uplinks to the host router running SR OS.

Up to 16 10GE interfaces can be used as uplinks for the associated satellite. You must create a new satellite template that configures the needed 10GE interfaces as uplinks. In addition, use a port template to specify the uplink association between the remaining client ports and configured uplinks.

Apply the new template to the satellite using the **template-name** command, where the **template-name** is the name configured in the **port-template** context.

This feature requires the 7210 SAS-Sx to be running Release 9.0.R10 or later for the 7210 SAS-Sx 64x10GE+4x100GE and 7210 SAS Release 10.0 or later for the 64x10GE+4xQSFP28 satellite.

The following restrictions apply:

- The 10GE ports used as satellite uplinks must start at port 1 and be sequential, up to the maximum of 16 10GE uplinks.
- When 10GE ports are used as uplinks, the 4x100GE ports are not available for use and should be configured as **role none**.

The following example shows the 10GE port satellite uplink configuration.

Example: MD-CLI

```
[ex:/configure satellite]
A:admin@node-2# info
  port-template "10gUp" {
    admin-state enable
    sat-type es64-10gb-sfpp+4-100gb-qsfp28
    port "1/1/1" {
      role uplink
    }
    port "1/1/2" {
      role uplink
    }
    port "1/1/3" {
      role uplink
    }
    ...
    port "1/1/9" {
      uplink 1/1/1
    }
    port "1/1/10" {
      uplink 1/1/1
    }
    ...
    port "1/1/16" {
      uplink 1/1/2
    }
    port "1/1/17" {
      uplink 1/1/2
    }
    port "1/1/65" {
      role none
    }
    port "1/1/66" {
      role none
    }
    ...
  }
  ethernet-satellite 20 {
    admin-state enable
    mac-address d0:99:d5:96:ee:41
    sat-type es64-10gb-sfpp+4-100gb-qsfp28
    software-repository repl
    port-template "10gUp"
  }
```

Example: classic CLI

```
A:node-2>config>system>satellite# info
-----
      port-template "10gUp" sat-type "es64-10gb-sfpp+4-100gb-qsfp28" create
      port 1/1/1
        role uplink
        uplink none
      exit
```

```

    port 1/1/2
      role uplink
      uplink none
    exit
    port 1/1/3
      role uplink
      uplink none
    exit
    ...
    port 1/1/9
      uplink 1/1/1
    exit
    port 1/1/10
      uplink 1/1/1
    exit
    ...
    port 1/1/16
      uplink 1/1/2
    exit
    port 1/1/17
      uplink 1/1/2
    exit
    ...
    port 1/1/65
      role none
    exit
    port 1/1/66
      role none
    exit
    ...
    no shutdown
  exit
eth-sat 20 create
  mac-address d0:99:d5:96:ee:41
  sat-type "es64-10gb-sfpp+4-100gb-qsfp28"
  port-template "10gUp"
  software-repository "rep1"
  no shutdown
exit
-----

```

6.7.6.8 Satellite uplink resiliency

An option in the **port-map** configuration allows a secondary uplink to be assigned to enable uplink resiliency. A secondary uplink is used to carry the traffic associated with the client port if the primary uplink becomes unavailable. If traffic is switched to the secondary uplink, when the primary uplink becomes available, traffic is reverted to the primary as soon as possible.

Use the following command to configure a secondary uplink per client port:

- **MD-CLI**

```
configure satellite ethernet-satellite port-map secondary
```

- **classic CLI**

```
configure system satellite eth-sat port-map secondary
```

To configure a secondary uplink, after the primary uplink is specified, the **secondary** keyword should be included, followed by the intended uplink to be used as the secondary uplink.

Example: MD-CLI

```
[ex:/configure satellite]
A:admin@node-2# info
    ethernet-satellite 1 {
        port-map esat-1/1/2 {
            primary esat-1/1/u1
            secondary esat-1/1/u3
        }
    }
```

Example: classic CLI

```
A:node-2>config>system>satellite# info
-----
    eth-sat 1 create
        port-map esat-1/1/2
        primary esat-1/1/u1
        secondary esat-1/1/u3
    exit
-----
```

- If there are no SAPs or interfaces bound to a client port, then any change can be made to the uplinks.
- If a SAP or interface is bound to a client port, or the client port is member of a LAG or ETH tunnel, then only one uplink change per configuration command is allowed (see below).
- The primary cannot be changed directly, this requires multiple steps.
 1. swap primary and secondary
 2. remove secondary
 3. add new secondary
 4. do a second swap of primary and secondary

The following are basic actions allowed with a single command:

- add or delete secondary uplink
- swap primary and secondary
- add a secondary uplink and swap secondary with primary

Uplink mapping can be changed, but a client uplink must be maintained throughout the process. For example, client-10 is mapped to uplink-1 (U-1), but must move to uplink-2 (U-2). To do this, add U-2 as the secondary uplink, then swap the primary and secondary, making U-2 the primary uplink for client-10 and switching traffic to U-2. After the switch is complete, remove U-1. U-1 cannot be directly replaced with U-2, as the client port would have no uplink during the switch.

6.7.6.9 Dynamic uplink resiliency

The host system can use the dynamic uplink resiliency mechanism to automatically manage uplink resiliency assignments. Through this mechanism, the host dynamically assigns a primary and secondary uplink for each satellite client port. Depending on the configuration options specified, the uplinks (primary and secondary) can be distributed over one or two FP forwarding engines.

Uplink distribution behavior

When using dynamic uplink resiliency, the primary and secondary uplink assignments are reevaluated each time an uplink on a satellite becomes operationally up or goes operationally down. This rebalances uplink assignments for optimal distribution of client ports. If a new client port is created, as a result of applying or modifying a connector breakout, the router runs the algorithm and assigns the client port primary and secondary uplinks. If the client port is deleted, the port is just removed; the algorithm is not rerun.

Uplinks are assigned to achieve fair distribution of client ports based on the following criteria:

- FP distribution
- MDA distribution
- connector distribution



Note: Dynamic uplink resiliency requires that the Ethernet satellite is running SR OS Release 20.9.R1 or later for the 7210 SAS satellites and Release 25.7.R1 or later for the 7250 IXR satellites.

Dynamic uplink resiliency configuration

The following example shows the dynamic uplink resiliency configuration.

Example: MD-CLI

```
[ex:/configure satellite]
A:admin@node-2# info
    ethernet-satellite 1 {
        admin-state enable
        sat-type es48-lgb-sfp
        dynamic-uplink true
        uplink-distribution dual-complex
    }
```

Example: classic CLI

```
A:node-2>config>system>satellite# info
-----
    eth-sat 1 create
        sat-type es48-lgb-sfp
        dynamic-uplink true
        uplink-distribution dual-complex
        no shutdown
    exit
-----
```

6.7.6.10 Ethernet LAGs with satellite member links

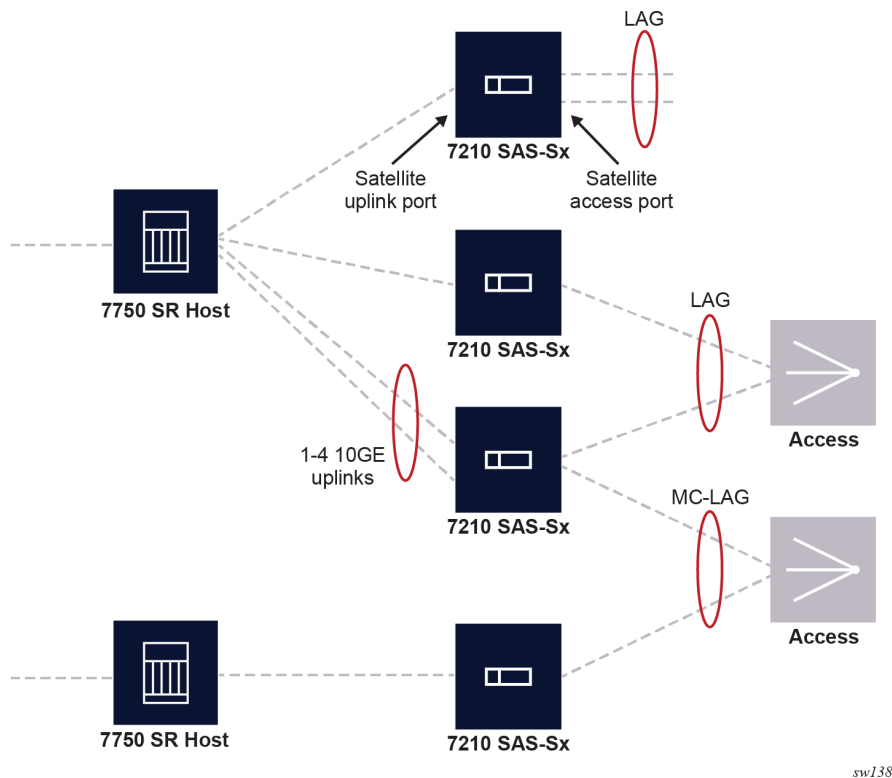
Ethernet LAGs can use a mixture of physical Ethernet ports (including connector-based ports) and satellite access ports. However, satellite access ports are not supported within mixed speed LAGs.



Note: All LAG member links must share the same QoS profile; however, this is not possible with a mixture of physical Ethernet ports and satellite access ports in a single LAG. As a result, Nokia recommends users do not use LAG in a mixed LAG configuration for an extended period of time.

Satellite client ports can also be used in MC-LAGs as both primary and standby link members, either using LACP or power-off signaling modes. MC-LAGs can also use a combination of physical and satellite client ports as link members.

Figure 26: Ethernet LAGs with satellite member links



All SR OS service access capabilities are supported. Where satellite access ports are used, the service entry point begins at the satellite access port.

6.7.6.11 CRC monitoring

This topic describes CRC monitoring for Ethernet satellite ports.

Use the commands in the following context to configure Ethernet CRC monitoring for Ethernet satellite hosts and uplink ports.

```
configure port ethernet crc-monitor
```

For CRC monitoring on Ethernet satellite ports, limit the setting of signal-failure thresholds to only a subset of the uplinks. Setting the signal-failure threshold on all the links could result in satellite isolation if CRC errors are seen simultaneously on all the uplinks. If a port enters a signal-failed state, it must be administratively reset to be re-enabled. If this occurs on the satellite uplink side, console access to the satellite must be available or the satellite must be rebooted.

Example: Configuration of signal-degrade and signal-failure thresholds on Ethernet ports (MD-CLI)

```
[ex:/configure port 1/1/1 ethernet crc-monitor]
A:admin@node-2# info
  port esat 2/1/r/u {
    admin-state enable
    ethernet {
      crc-monitor {
        signal-degrade {
          threshold 9
        }
        signal-failure {
          threshold 9
        }
      }
    }
  }
  port 1/1/c35/1 {
    admin-state enable
    ethernet {
      crc-monitor {
        signal-degrade {
          threshold 9
        }
        signal-failure {
          threshold 9
        }
      }
    }
  }
}
```

Example: Configuration of signal-degrade and signal-failure thresholds on Ethernet ports (classic CLI)

```
A:node-2>config# info
-----
  port esat2/1/r/u
    ethernet
      crc-monitor
        sd-threshold 9
        sf-threshold 9
      exit
    exit
    no shutdown
  exit
  port 1/1/c35/1
    ethernet
      crc-monitor
        sd-threshold 9
        sf-threshold 9
      exit
    exit
    no shutdown
  exit
```

6.8 Auto-provisioning

Auto-provisioning is used to provision a node using an external DHCP server and file server. It is used to obtain a configuration file and an image file from an external server using an in-band mechanism. Auto-provisioning is not compatible with an out-of-band management port.

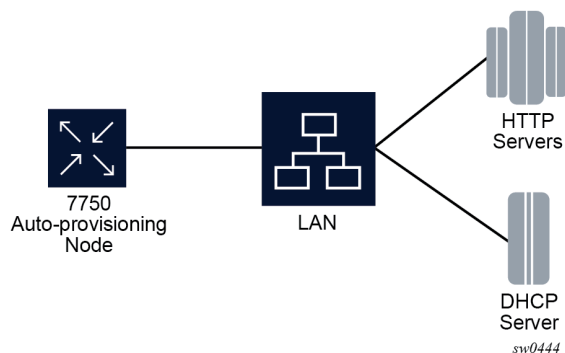
Before using auto-provisioning, the SR OS must be booted up and running the application image. In addition, it needs to have some minimum configuration before the auto-provision script is executed by the user.

After the auto-provision application is triggered using a tools command, SR OS checks all operationally up ports without IP addresses and send DHCP discovery to these interfaces. The DHCP server needs to be configured with Option 67 and the user must provide the SR OS with the URL of a file server and the corresponding directory for the image.

Figure 27: Example of a network with no DHCP relay to Figure 29: Example of a network with multiple subnets describe scenarios in which auto-provisioning are used.

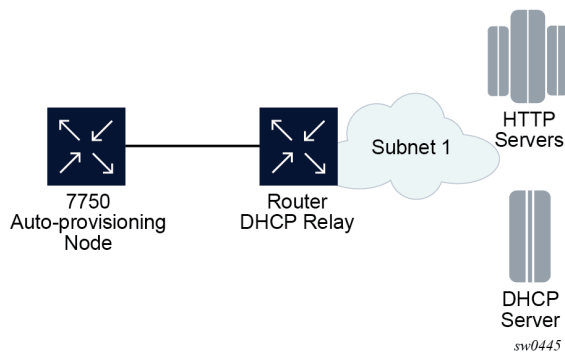
In Figure 27: Example of a network with no DHCP relay, there is no DHCP relay and all IP addresses are assigned from a single pool.

Figure 27: Example of a network with no DHCP relay



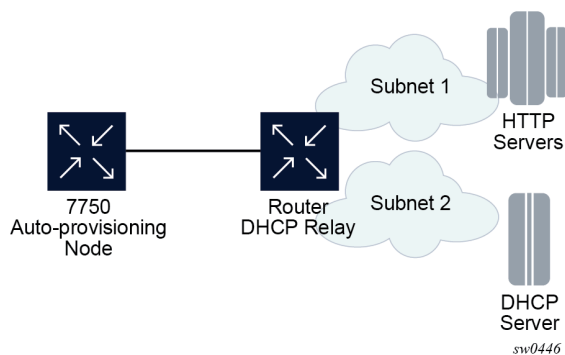
In Figure 28: Example of a network with a DHCP relay, there is a DHCP relay which injects the Option 82 as a gateway address. The DHCP server is assigned the IP address from the pool dictated by the gateway address option 82. The DHCP server and HTTP server are in the same subnet. The DHCP offer has option 3 "router" which is used for a default gateway creation on the 7750 SR.

Figure 28: Example of a network with a DHCP relay



In [Figure 29: Example of a network with multiple subnets](#), all components are in different subnets. The DHCP relay adds Option 82 to the DHCP request as the gateway address which is used for pool selection. The DHCP server must add option 3 configured with the gateway address of the HTTP server.

Figure 29: Example of a network with multiple subnets



6.8.1 Auto-provisioning limits

The following are some configuration limits for auto-provisioning:

- A maximum of 12 Layer 3 interfaces are supported for auto-provisioning.
- Only IPv4 auto-provisioning is supported.
- It is highly recommended to only have a basic card, MDA, port, and interface configuration as described in this document and no additional static routes or IGP or BGP protocols when performing auto-provisioning because auto-provisioning installs default static routes that may be affected by any extra routing configuration.
- A maximum of 255 characters is supported for the remote URL (200 character maximum for the filepath, the rest for the main URL consisting of the protocol, login credentials, and host IP). A maximum of 200 characters is supported for the local URL. The local file or folder name must not exceed 99 characters.
- The maximum number of file pairs for each image/config record is 10.

6.8.2 Auto-provisioning process

In this process, the node detects operational ports, attempts to discover its IP address, and downloads the relevant files for provisioning.

1. The node sends a DHCP discovery request to the DHCP server using the out-of-band management port. If DHCP discovery is unsuccessful, the node reattempts it using the in-band management ports.
2. After DHCP discovery is successful, the DHCP server returns an IPv4 or IPv6 FTP or HTTP URL of a file server from which the node can retrieve provisioning information.
3. The node downloads the provisioning information and performs the auto-provisioning according to the specifications in the files.
4. After the node is successfully provisioned, it automatically reboots and becomes operationally up.

See [Provisioning files](#) for more information about the auto-provisioning process.

The SR OS can also initiate the auto-provisioning process using a **tools** command.

6.8.3 Auto-provisioning DHCP rules

The following are the DHCP rules in the auto-provisioning stage:

1. First, auto-provisioning walks through the interfaces with a configured port, where the port is in operational status up, one by one.
2. It sends a DHCP request to the first configured interface with a port up and no IP address configured.
 - If, on this interface, multiple DHCP offers arrives, only the first offer is sent to the auto-provisioning task and the other offers are ignored. This could occur if the node is on a LAN and multiple DHCP servers are connected to the interface.
 - The DHCP client has an exponential retry mechanism. If the DHCP offer does not arrive from the server, the client resends a DHCP request at 2, 4, 8, 32 and 64 s, with 64 s being the maximum timeout. If the 64 s timeout interval is reached, the DHCP client keeps retrying every 64 s. The user can configure a timeout value. If no DHCP offer has arrived by this timeout value, the auto-provisioning process moves to the next interface.
 - If the DHCP offer arrives on the port and the DHCP client task does not acknowledge the DHCP offer, for any reason, it disables the DHCP client and remove the IP from the port.
 - If the DHCP offer arrives on the port and the DHCP client acknowledges the offer, it sends the information to auto-provisioning. If auto-provisioning does not like the offer, because there is no Option 67, Option 67 is malformed, or for any other reason listed in [Auto-provisioning failure](#), the auto-provisioning process deconfigures the DHCP client and the DHCP client sends a DHCP release, and unassigns the IP address.
 - In case of failure, more information is displayed by the auto-provisioning process and the process moves to the next port that is up and does not have an IP address.
3. If auto-provisioning is successful using the offer and its option, the provisioning file download starts though the protocol dictated by Option 67.

The **auto-provisioning** command is CLI blocking. All information about the auto-provisioning process is displayed on the CLI and logged.

6.8.4 Auto-provisioning failure

Auto-provisioning fails for the following reasons:

- There is no Option 67.
- The Option 67 format is not acceptable to auto-provisioning.
- The format is a URL or DNS is not supported. There is a failure in the download provisioning file or the server is not reachable.
- There is failure in the download of the image or config file using the provisioning file information, for example, the server is not available, the wrong directory is listed, or the wrong credentials are given.
- The image or config fails to copy to the compact flash.
- The image or config fails to sync to the inactive CPM.
- The BOF does not point to the compact flash, for example, it is pointing to the network.

If the auto-provisioning procedure on this interface fails, then auto-provisioning does the following:

1. Displays information about the blocked CLI and in the log, describing the failure in detail.
2. Updates the DHCP task so the DHCP task can take the appropriate actions to release the IP address on the interface. This is done by sending a DHCP release for the DHCP ack received from the server.
3. Goes to the next interface with port up and no IP address.



Note: If no other interface with port up is found, the auto-provisioning task stops and a failure error is displayed on the CLI and in the log.

6.9 Administrative tasks

This section contains information to perform administrative tasks.

6.9.1 Saving configurations

Whenever configuration changes are made, the modified configuration must be saved so they are not lost when the system is rebooted.

Configuration files are saved by executing explicit command syntax which includes the file URL location to save the configuration file as well as options to save both default and non-default configuration parameters. Boot option file (BOF) parameters specify where the system should search for configuration and image files as well as other operational parameters during system initialization.

For more information about boot option files, see the Boot Options section.

6.9.2 Specifying post-boot configuration files

Two post-boot configuration extension files are supported and are triggered when either a successful or failed boot configuration file is processed. The **boot-bad-exec** and **boot-good-exec** commands specify

URLs for the CLI scripts to be run following the completion of the bootup configuration. A URL must be specified or no action is taken.

For example, after a configuration file is successfully loaded, the specified URL can contain a nearly identical configuration file with specific commands enabled or disabled, or particular parameters specified and according to the script which loads that file.

6.9.3 Network timing

In Time Domain Multiplexed networks, the concept of network timing is used to prevent over-run or under-run issues where circuits are groomed (rebundled) and switched. Hardware exists in each node that takes a common clock derived from an internal oscillator, a specific receive interface, or special BITS interface and provides it to each synchronous interface in the system. Usually, each synchronous interface is allowed to choose between using the chassis-provided clock or the clocking recovered from the received signal on the interface. The clocking is used to drive the transmit side of the interface. The appropriate configuration at each node which defines how interface clocking is handled must be considered when designing a network that has a centralized timing source so each interface is operating in a synchronous manner.

The effect of timing on a network is dependent on the nature of the type of traffic carried on the network. With bit-wise synchronous traffic (traditional circuit-based voice or video), non-synchronous transmissions cause a loss of information in the streams affecting performance. With packet-based traffic, the applications expect and handle jitter and latency inherent to packet-based networks. When a packet-based network is used to carry voice or video traffic, the applications use data compression and elasticity buffering to compensate for jitter and latency. The network itself relies on appropriate Quality of Service (QoS) definitions and network provisioning to further minimize the jitter and latency the application may experience.

6.9.4 Power supplies

SR OS supports a **power-supply** command to configure the type and number of power supplies present in the chassis. The operational status of a power source is always displayed by the LEDs on the Control Processor/Switch Fabric Module (CP/SFM) front panel. However the power supply information must be explicitly configured to generate a power supply alarm if a power source becomes operationally disabled.

The following example configures power supply settings for the 7750 SR, 7950 XRS, and 7450 ESS.

Example: MD-CLI

```
[ex:/configure chassis router chassis-number 1]
A:admin@node-2# info
  power-supply 1 {
    power-supply-type dc-single
  }
```

Example: classic CLI

```
A:node-2>config>system# info
-----
  power-supply 1 dc
-----
```

6.9.5 CPM file synchronization

In redundant systems, file synchronization ensures that the active and standby CPMs have identical operational configuration, including the active configuration, CPM, XCM, and IOM images in the event of a failure or reset of the active CPM.

You can configure automatic synchronization to occur when you execute the following commands to save the configuration:

- **MD-CLI**

```
admin save
admin save bof
```

- **classic CLI**

```
admin save
bof save
```

See [File synchronization](#) for information about which files are synchronized with automatic synchronization.

6.9.6 Forcing a switchover

The **admin redundancy force-switchover now** command forces an immediate switchover to the standby CPM.

If the active and standby are not synchronized, users can manually synchronize the standby CPM by rebooting the standby by issuing the **admin reboot standby** command on the active CPM.

6.10 System router instances

SR OS supports multiple Layer 3 router instances. These instances have their own IP addressing spaces and configuration options. Router instances are isolated from each other.

The following are the different types of router instances in SR OS:

- **Base**

All SR OS routers have the Base router instance: the system created default router instance used to forward user IP traffic among router line card ports. Router interfaces (that is, network interfaces configured under **configure router [Base]**) and IES services and interfaces exist in the Base router instance. The Base router instance is identified in SNMP as vRtrType = baseRouter (1) and has a vRtrID of 1.

- **VPRN instances**

Another type of router instance is the set of user-configured VPRN services. Each VPRN service has a unique router instance. For more information about VPRN services and their associated router instances, see the *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN*. VPRN router instances are identified in SNMP as vRtrType = vprn (2), and the vRtrID is dynamically allocated.

- **Special system router instances**

SR OS routers also support the following special router instances:

- **management**

The management router instance is a system created router instance that is used for management of the router. The management router instance is bound to CPM/CCM ports A/1 and B/1. This is a CPM router instance which cannot be renamed or deleted by a user. The management router instance is identified in SNMP as `vrRtrType = vr(3)`, and the `vrRtrID` is 4095.

- **vpls-management**

The vpls-management router instance is used for management of VPLS services. It is identified in SNMP as `vrRtrType = vr(3)`, and the `vrRtrID` is 4094.

- **User created CPM router instances**

User created CPM router instances are user defined router instances that are mainly used with Ethernet ports on the CPM/CCM cards: CPM router instances only use CPM/CCM Ethernet ports as interfaces. CPM router instances have a user-defined name and are the only types of non-VRPN router instances that can be created by the user. User created CPM router instances are identified in SNMP as `vrRtrType = vr(3)`, and the `vrRtrID` is dynamically allocated.

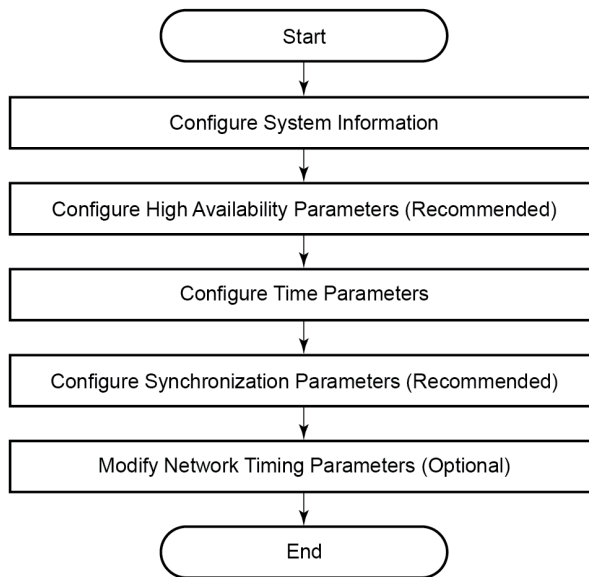
Some management protocols can use either the base routing instance (in-band) or the management routing instance (out-of-band). A listing of these protocols can be found in the CPM Filter: Protocols and Ports section of the *7450 ESS, 7750 SR, 7950 XRS, and VSR System Management Guide*. Unless otherwise stated in the detailed description of the protocol, when the server or client for the protocol is reachable via the management routing instance, those protocol messages use the management interface for the protocol communication.

If BOF is set up with autoconfiguration and the DHCP server provides a general default route such as 0.0.0.0/0, with some protocols (like PCEP, TACACS+, RADIUS, and LDAP), Authentication, Authorization, Accounting (AAA) always prefers OOB over in-band connectivity. This is because these protocols prefer to use the OOB management port first. If a matching route is not found, in-band is attempted. The static route provided by DHCP must be properly set to ensure the correct route preference is made by these protocols.

6.11 System configuration process overview

The following figure shows the process for basic system provisioning.

Figure 30: System configuration and implementation flow



7750_SR_Basics_27

6.12 Saving configurations manually

Whenever configuration changes are made, the modified configuration must be saved so the changes are persistent when the system is rebooted. The system uses the configuration and image files, as well as other operational parameters necessary for system initialization, according to the locations specified in the boot option file (BOF) parameters. For more information about BOFs, see the [System initialization and boot options](#) chapter of this manual.

Configuration files can be manually saved in two ways with the **admin save** command:

- When a location is specified as a parameter after the command, the system saves the configuration to the file URL that is entered.
- When a location is not specified, the system saves the configuration to the current BOF primary configuration location. If the primary configuration location (path or filename) has changed since the last boot, the new configuration location is used.

If the save attempt fails to the primary location, the system does not try to save the file to the secondary or tertiary configuration locations unless the path and filename are explicitly specified with the **admin save** command.

The classic CLI provides additional options for saving the configuration that are not needed in the MD-CLI. See the specific command descriptions in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide* for more information.

6.13 Common configuration tasks

This section provides an overview of the CLI commands used to configure system parameters:

- [System information](#)
- [System time elements](#)

6.13.1 System information

This section describes the basic system information commands that configure the system name of the router, contact information, location (such as an address, floor, room number, and so on), CLLI code, and global positioning system (GPS) coordinates.

6.13.1.1 System name

The device's system name is used in the prompt string. Only one system name can be configured; if multiple system names are configured, the last one overwrites the previous entry.

Use the following command to configure the system name.

```
configure system name
```

6.13.1.2 Contact

Use the **contact** command to specify the name of a system administrator, IT staff member, or other administrative entity.

Use the following command to configure the contact.

```
configure system contact
```

6.13.1.3 Location

Use the **location** command to specify the location of the device. For example, enter the city, building address, floor, room number, and so on, where the router is located.

Use the following command to configure the location.

```
configure system location
```

6.13.1.4 CLLI code

The Common Language Location Code (CLLI code) is an 11-character standardized geographic identifier that is used to uniquely identify the geographic location of a router.

Use the following command to configure the CLLI code.

```
configure system clli-code
```

6.13.1.5 GPS coordinates

Use the optional **coordinates** command to specify the GPS location of the device. If the string contains special characters (#, \$, spaces, and so on), the entire string must be enclosed within double quotes.

Use the following command to configure the coordinates.

```
configure system coordinates
```

6.13.2 System time elements

The system clock maintains time according to Coordinated Universal Time (UTC). Configure the time zone and summer time (daylight saving time) command options to correctly display time according to the local time zone.

6.13.2.1 Zone

The **zone** command sets the time zone and time zone UTC offset for the router. SR OS supports system-defined and user-defined time zones. The system-defined time zones and offsets are listed in [Table 14: System-defined time zones and UTC offsets](#).

Use the following command to set the time zone and time zone UTC offset.

```
configure system time zone
```

6.13.2.2 Summer (daylight saving) time

Configure the start and end dates and offset for summer (daylight saving) time to override system defaults or for user-defined time zones. When configured, the time will be adjusted by changing to the configured offset when summer time starts and returning to the configured offset when summer time ends.

Use commands in the following context to configure the start day, end day, and offset of the summer.

```
configure system time dst-zone
```

If the time zone configured is listed in [Table 14: System-defined time zones and UTC offsets](#), the start and end command options and offset do not need to be configured with this command unless there is a need to override the system defaults. The command will return an error if the start and end dates and times are not available either in [Table 14: System-defined time zones and UTC offsets](#) or entered as optional command options in this command.

6.13.2.3 NTP

NTP is the Network Time Protocol defined in RFC 1305, *Network Time Protocol (Version 3) Specification, Implementation and Analysis* and RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*. It allows for the participating network nodes to keep time more accurately and more importantly they can maintain time in a more synchronized fashion between all participating network nodes.

SR OS uses an NTP process based on a reference build provided by the Network Time Foundation. Nokia strongly recommends that the users review RFC 8633, *Network Time Protocol Best Current Practices*, when they plan to use NTP with the router. The RFC section "Using Enough Time Sources" indicates that using only two time sources (NTP servers) can introduce instability if they provide conflicting information. To maintain accurate time, Nokia recommends configuring three or more NTP servers.

NTP uses stratum levels to define the number of hops from a reference clock. The reference clock is considered to be a stratum-0 device that is assumed to be accurate with little or no delay. Stratum-0 servers cannot be used in a network. However, they can be directly connected to devices that operate as stratum-1 servers. A stratum-1 server is an NTP server with a directly-connected device that provides Coordinated Universal Time (UTC), such as a GPS or atomic clock.

The higher stratum levels are separated from the stratum-1 server over a network path, therefore, a stratum-2 server receives its time over a network link from a stratum-1 server. A stratum-3 server receives its time over a network link from a stratum-2 server.

SR OS routers normally operate as a stratum-2 or higher device. The router relies on an external stratum-1 server to source accurate time into the network. However, SR OS also allows for the use of the local PTP recovered time to be sourced into NTP. In this latter case, the local PTP source appears as a stratum-0 server and SR OS advertises itself as a stratum-1 server. Activation of the PTP source into NTP may impact the network NTP topology because the SR OS router is promoted to stratum-1.

SR OS router runs a single NTP clock which then operates NTP message exchanges with external NTP clocks. Exchanges can be made with external NTP clients, servers, and peers. These exchanges can be through the base, management, or VPRN routing instances.

NTP operates associations between clocks as either client or server, symmetric active and symmetric passive, or broadcast modes. These modes of operation are applied according to which elements are configured on the router. To run server mode, the user must enable NTP server mode for the base and each needed VPRN routing instance. To run client mode, the user must configure external servers. If both the local router and remote router are configured with each other as peers, then the router operates in symmetric active mode. If only one side of the association has peering configured, then the modes are symmetric passive. To operate using broadcast mode, interfaces must be configured to transmit as broadcast servers or receive as broadcast clients.

NTP server operation for both unicast and broadcast communication within a VPRN is configured within the VPRN (see the NTP Within a VPRN Service section in *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN*).



Note: NTP provides lightweight synchronization across a network for alignment of system time for logging purposes. NTP does not provide the high accuracy time needed for the on-air applications of the mobile base stations. The more recent PTP protocol has been developed for these applications (see [Network synchronization](#)).

The following NTP elements are supported:

- **server mode**

In this mode, the node advertises the ability to act as a clock source for other network elements. The node, by default, transmits NTP packets in NTP version 4 mode.

- **authentication keys**

Authentication keys implement increased security support in carrier and other networks. Both DES and MD5 authentication are supported, as well as multiple keys.

- **operation in symmetric active mode**

This capability requires that NTP be synchronized with a specific node that is considered more trustworthy or accurate than other nodes carrying NTP in the system. This mode requires that a specific peer is set.

- **server and peer addressing using IPv6**

Both external servers and external peers may be defined using IPv6 or IPv4 addresses. Other features (such as multicast, broadcast) use IPv4 addressing only.

- **broadcast or multicast modes**

When operating in these modes, the node receives or sends using either a multicast (default 224.0.1.1) or a broadcast address. Multicast is supported only on the CPM MGMT port.

- **alert when NTP server is not available**

When none of the configured servers are reachable on the node, the system reverts to manual timekeeping and issues a critical alarm. When a server becomes available, a trap is issued indicating that standard operation has resumed.

- **NTP and SNTP**

If both NTP and SNTP are enabled on the node, then SNTP transitions to an operationally down state. If NTP is removed from the configuration or shut down, then SNTP resumes an operationally up state.

- **gradual clock adjustment**

As several applications (such as Service Assurance Agent (SAA)) can use the clock, and if determined that a major (128 ms or more) adjustment needs to be performed, the adjustment is performed by programmatically stepping the clock. If a minor (less than 128 ms) adjustment must be performed, then the adjustment is performed by either speeding up or slowing down the clock.

- To avoid the generation of too many events/trap the NTP module rates limit the generation of events/traps to three per second. At that point a single trap is generated that indicates that event/trap squashing is taking place.

6.13.2.3.1 Authentication-check

NTP supports an authentication mechanism to provide some security and access control to servers and clients. The authentication check feature provides the option to skip the rejection of NTP PDUs that do not match the authentication key or authentication type requirements.

The default behavior when authentication is configured is to reject all NTP PDUs that have a mismatch in either the authentication key ID, type, or key.

When authentication check is configured, NTP PDUs are authenticated on receipt. However, mismatches cause a counter to be increased, one counter for key ID, one for type, and one for key value mismatches.

Use commands in the following context to enable authentication check.

```
configure system time ntp authentication-check
```

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
    admin-state enable
    authentication-check true
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
authentication-check
no shutdown
-----
```

6.13.2.3.2 Authentication-key

The **authentication-key** command configures an authentication key ID, key type, and key used to authenticate NTP PDUs sent to and received from other network elements participating in the NTP. For authentication to work, the authentication key ID, authentication type, and authentication key value must match.

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
admin-state enable
authentication-key 1 {
    key "0AwgNULbzgI hash2"
    type des
}
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
shutdown
authentication-key 1 key "0AwgNULbzgI" hash2 type des
-----
```

6.13.2.3.3 Broadcast

The **broadcast** command is used to transmit broadcast packets on a given interface. Interfaces in the base routing context or the management interface may be specified. Due the relative ease of spoofing of broadcast messages, it is strongly recommended to use authentication with broadcast mode. The messages are transmitted using a destination address that is the NTP Broadcast address. The following example enables NTP and configures the broadcast interface.

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
admin-state enable
broadcast "Base" interface-name "int11" {
    version 4
    ttl 127
}
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
```

```

-----
broadcast interface int11 version 4 ttl 127
no shutdown
-----

```

6.13.2.3.4 Broadcastclient

The **broadcastclient** command enables listening to NTP broadcast messages on the specified interface. Interfaces in the base routing context or the management interface may be specified. Due the relative ease of spoofing of broadcast messages, it is strongly recommended to use authentication with broadcast mode. The messages must have a destination address of the NTP Broadcast address. The following example enables NTP and configures the broadcast client.

Example: MD-CLI

```

[ex:/configure system time ntp]
A:admin@node-2# info
    admin-state enable
    broadcast-client "Base" interface-name "int11" {
    }

```

Example: classic CLI

```

A:node-2>config>system>time>ntp# info
-----
broadcastclient interface int11
no shutdown
-----

```

6.13.2.3.5 Multicast

When configuring NTP the node can be configured to transmit or receive multicast packets on the CPM MGMT port (CPM applies to the 7450 ESS and 7750 SR). Broadcast and multicast messages can easily be spoofed; therefore, authentication is strongly recommended. Multicast is used to configure the transmission of NTP multicast messages. When transmitting multicast NTP messages the default address of 224.0.1.1 is used. The following example enables NTP and multicast.

Example: MD-CLI

```

[ex:/configure system time ntp]
A:admin@node-2# info
    admin-state enable
    multicast {
    }

```

Example: classic CLI

```

A:node-2>config>system>time>ntp# info
-----
multicast
no shutdown
-----

```

6.13.2.3.6 Multicastclient

The **multicastclient** command is used to configure an address to receive multicast NTP messages on the CPM MGMT port (7450 ESS and 7750 SR). Broadcast and multicast messages can easily be spoofed, therefore, authentication is strongly recommended. If **multicastclient** is not configured, all NTP multicast traffic is ignored. The following example enables NTP and configures the address to receive multicast NTP messages.

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
    admin-state enable
    multicast-client {
        authenticate true
    }
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
                multicastclient
                no shutdown
-----
```

6.13.2.3.7 NTP-server

The **ntp-server** command configures the node to assume the role of an NTP server. Unless the **server** command is used, this node will function as an NTP client only and will not distribute the time to downstream network elements. If the **authenticate** command option is specified, the NTP server requires client packets to be authenticated.

The following is an example of a configuration output of NTP enabled with the **ntp-server** command configured.

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
    admin-state enable
    ntp-server {
        authenticate true
    }
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
                no shutdown
                ntp-server
-----
```

6.13.2.3.8 Peer

Configuration of an NTP peer configures symmetric active mode for the configured peer. Although any system can be configured to peer with any other NTP node, Nokia recommends configuring authentication and to configure known time servers as their peers. Administratively disable this command to remove the configured peer.

Use commands in the following context to configure symmetric active mode.

```
configure system time ntp peer
```

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
  admin-state enable
  peer 192.168.1.1 router-instance "Base" {
    key-id 1
  }
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
      no shutdown
      peer 192.168.1.1 key-id 1
-----
```

6.13.2.3.9 Server

The **server** command is used when the node should operate in client mode with the NTP server specified in the address field. Up to ten NTP servers can be configured. The following example enables NTP and configures the server.

Example: MD-CLI

```
[ex:/configure system time ntp]
A:admin@node-2# info
  admin-state enable
  server 192.168.1.1 router-instance "Base" {
    key-id 1
  }
```

Example: classic CLI

```
A:node-2>config>system>time>ntp# info
-----
      no shutdown
      server 192.168.1.1 key 1
-----
```

6.13.2.3.10 Configuring system time to use a GNSS RF port

A 7750 SR FP5 equipped with an integrated GNSS RF port that is connected to an active GNSS antenna can be configured to receive a system time reference from the port.

When the GNSS RF port is enabled and configured for system timing or frequency, the PTP timeTransmitter clock uses GNSS for frequency and time distribution. See *7450 ESS, 7750 SR, 7950 XRS, and VSR Interface Configuration Guide* for information about enabling ports and configuring system timing to use a GNSS RF port.

Use the commands in the following context to specify PTP as an NTP server and source of system time.

```
configure system time ntp
```

6.13.2.4 SNTP

SNTP is a compact, client-only version of NTP. SNTP can only receive the time from SNTP/NTP servers; it cannot be used to provide time services to other systems. SNTP can be configured in either broadcast or unicast client mode.

SNTP time elements include the [Broadcast-client](#) and [Server-address](#).

Use the commands in the following context to configure the SNTP.

```
configure system time sntp
```

6.13.2.4.1 Broadcast-client

Use the following command to enable listening at the global device level to SNTP broadcast messages on interfaces with broadcast client configured:

- **MD-CLI**

```
configure system time sntp sntp-state broadcast
```

- **classic CLI**

```
configure system time sntp broadcast-client
```

The following example shows SNTP enabled with the broadcast client.

Example: MD-CLI

```
[ex:/configure system time sntp]
A:admin@node-2# info
    admin-state enable
    sntp-state broadcast
```

Example: classic CLI

```
A:node-2>config>system>time>sntp# info
-----
    broadcast-client
```

```
no shutdown
-----
```

6.13.2.4.2 Server-address

Use the following command to configure an SNTP server for SNTP unicast client mode:

- **MD-CLI**

```
configure system time sntp server
```

- **classic CLI**

```
configure system time sntp server-address
```

The following is an example of a configuration output of SNTP enabled with the **server-address** command configured.

Example: MD-CLI

```
[ex:/configure system time sntp]
A:admin@node-2# info
  admin-state enable
  server 10.10.0.94 {
    version 1
    prefer true
    interval 100
  }
```

Example: classic CLI

```
A:node-2>config>system>time>sntp# info
-----
      server-address 10.10.0.94 version 1 preferred interval 100
      no shutdown
-----
```

6.13.2.5 CRON

The CRON feature supports periodic and date and time-based scheduling in SR OS. CRON can be used, for example, to schedule Service Assurance Agent (SAA) functions. CRON functionality includes the ability to specify scripts that need to be run, when they are scheduled, including one-time only functionality (one-shot), interval and calendar functions. Scheduled reboots, peer turn ups, service assurance agent tests and more can all be scheduled with CRON, as well as OAM events, such as connectivity checks, or troubleshooting runs.

CRON supports the schedule element. The schedule function configures the type of schedule to run, including one-time only (one-shot), periodic, or calendar-based runs. All runs are determined by month, day of month or weekday, hour, minute, and interval (seconds).

6.13.2.5.1 Schedule

The **schedule** command configures the type of schedule to run, including one-time only (oneshot), periodic, or calendar-based runs. All runs are determined by month, day of the month or weekday, hour, minute, and interval (seconds). If the **end-time** and **interval** command options are both configured, whichever condition is reached first is applied.

The following is an example of a configuration output that schedules a script named "test" to run every 15 minutes on the 17th of each month and every Friday until noon on July 17, 2007.

Example: MD-CLI

```
[ex:/configure system cron]
A:admin@node-2# info
  schedule "test" owner "TiMOS CLI" {
    day-of-month [17]
    minute [0 15 30 45]
    weekday [friday]
    end-time {
      date-and-time 2007-07-17T12:00:00.0+00:00
    }
  }
```

Example: classic CLI

```
*A:node-2>config>system>cron# info
-----
  schedule "test"
    shutdown
    day-of-month 17
    minute 0 15 30 45
    weekday friday
    end-time 2007/07/17 12:00
  exit
-----
```

6.13.3 ANCP enhancements

Persistency is available for subscriber ANCP attributes and is stored on the on-board compact flash card. ANCP data stays persistent during an ISSU as well as node reboots. During recovery, ANCP attributes are first restored fully from the persistence file, and incoming ANCP sessions are temporarily on hold. Afterwards, new ANCP data can overwrite any existing values. This new data is then stored into the compact flash in preparation for the next event.

6.13.4 Configuration backups

You can specify the maximum number of configuration and classic CLI index backups stored in the configuration primary location.

For example, the maximum number of backups is set to 5 and the configuration file is called `xyz.cfg`. When the configuration is saved, the most recent file is renamed with a `.1` extension. Each subsequent file increments the numeric extension until the maximum count is reached. The oldest file (5) is deleted as more recent files are saved.

- config.cfg
- config.cfg.1
- config.cfg.2
- config.cfg.3
- config.cfg.4
- config.cfg.5

Each persistent index file is updated at the same time as the associated configuration file. When the index file is updated, the save is performed to config.cfg and the classic CLI index file is created as config.ndx. Synchronization between the active and standby SF/CPMSF/CPM is performed for all configurations and their associated persistent index files.

Use the following commands to specify the maximum number of backup versions of the configuration and index files kept in the primary location:

- **MD-CLI**

```
configure system management-interface configuration-save configuration-backups
```

- **classic CLI**

```
configure system config-backup
```

The following example shows the command usage to set the maximum number of backup versions of the configuration and index files kept in the primary location to 7.

Example: MD-CLI

```
[ex:/configure]
A:admin@node-2# system management-interface configuration-save configuration-backups 7
```

Example: classic CLI

```
A:node-2# system config-backup 7
```

Example: Configuration backup configuration output (MD-CLI)

```
[ex:/configure system]
A:admin@node-2# info
  contact "Fred Information Technology"
  name "node-2"
  location "Bldg.1-floor 2-Room 201"
  coordinates "N 45 58 23, W 34 56 12"
  cli-code "abcdefg1234"
...
  management-interface {
    configuration-save {
      configuration-backups 7
    }
  }
...
```

Example: Configuration backup configuration output (classic CLI)

```
A:node-2>config>system>time# info
```

```
#-----  
echo "System Configuration"  
#-----  
    name "node-2"  
    contact "Fred Information Technology"  
    location "Bldg.1-floor 2-Room 201"  
    cli-code "abcdefg1234"  
    coordinates "N 45 58 23, W 34 56 12"  
    config-backup 7  
...  
#-----
```

6.14 System timing

When network timing is required for the synchronous interfaces in the router, a timing subsystem provides a clock to all synchronous interfaces within the system.

This section describes the commands used to configure and control the timing subsystem.

6.14.1 Entering edit mode



Note: This applies to the classic CLI.

Use the following command to enter the edit mode and edit timing references.

```
configure system sync-if-timing begin
```

Example

The following is an example of an error message that is displayed if you try to modify **sync-if-timing** parameters without entering the **begin** keyword.

```
A:node-2>config>system>sync-if-timing>ref1# source-port 2/1/1  
MINOR: CLI The sync-if-  
timing must be in edit mode by calling begin before any changes can be made.  
MINOR: CLI Unable to set source port for ref1 to 2/1/1.  
A:node-2>config>system>sync-if-timing>ref1#
```

6.14.2 Configuring timing references

Use commands in the following context to configure timing reference settings:

- **MD-CLI**

```
configure system central-frequency-clock
```

- **classic CLI**

```
configure system sync-if-timing
```

The source port specified for **ref1** and **ref2** is dependent on the router model type and chassis slot. See the details in the specific command descriptions in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*.

6.14.3 Using the revert command

The revert function allows the clock to revert to a higher-priority reference if the current reference goes offline or becomes unstable.

When mode is non-revertive, a failed clock source is not selected again. If a node enters holdover because of the references being in previous failed state, then the node selects one of the previously failed references instead of going into holdover.

If revertive switching is enabled, the highest-priority valid timing reference is used. If a reference with a higher priority becomes valid, a switchover to that reference is initiated. If a failure on the current reference occurs, the next highest reference takes over. Use the following command to enable revert mode:

- **MD-CLI**

```
configure system central-frequency-clock revert true
```

- **classic CLI**

```
configure system sync-if-timing revert
```

If non-revertive switching is enabled, the active reference always remains selected while its valid, even if a higher priority reference becomes available. If the active reference becomes invalid, a reference switchover to a valid reference with the highest priority is initiated. The failed reference is eligible for selection when it becomes operational. Use the following command to turn off revert mode:

- **MD-CLI**

```
configure system central-frequency-clock revert false
```

- **classic CLI**

```
configure system sync-if-timing no revert
```

6.14.4 Committing and discarding changes

Use the following command to save changes made to the timing references during a session. Modifications are not persistent across system boots unless you enter this command.

- **MD-CLI**

```
configure system central-frequency-clock commit
```

- **classic CLI**

```
configure system sync-if-timing commit
```

Use the following command to discard changes that have been made to the timing references during a session:

- **MD-CLI**

```
configure system central-frequency-clock discard
```

- **classic CLI**

```
configure system sync-if-timing abort
```

6.14.5 Forcing a specific reference

This applies to the classic CLI.

The **debug sync-if-timing force-reference** command should only be used to test and debug problems. Network synchronization problems may appear if network elements are left with this manual override setting. When the system timing reference input has been forced, it may be cleared using the **no force-reference** command.

Force the CPM clock to use a specific input reference using the **force-reference** command.

When the command is executed, the CPM clock on the active CPM immediately switches its input reference to that specified by the command. If the specified input is not available (shutdown), or in a disqualified state, the CPM clock shall use the next qualified input reference based on the selection rules.

This command also affects the BITS output port. If the BITS output port selection is set to line-reference and the reference being forced is not the BITS input port, then the system uses the forced reference to generate the signal out the BITS output port. If the BITS output port selection is set to internal-clock, then the system uses the output of the CPM clock to generate the signal for the BITS output port.

On a CPM activity switch, the **force** command is cleared and normal reference selection is determined.

6.14.6 Configuring system timing to use a GNSS RF port

Use the following command to configure a GNSS RF port as a system timing reference:

- **MD-CLI**

```
configure system central-frequency-clock gnss admin-state enable
```

- **classic CLI**

```
configure system sync-if-timing gnss no shutdown
```

Example: Timing reference configuration (MD-CLI)

```
[ex:/configure system central-frequency-clock]
A:admin@node-2# info
  gnss {
    admin-state enable
  }
```

Example: Timing reference configuration (classic CLI)

```
A:node-2>config>system>sync-if-timing# info
-----
```

```
gnss
  no shutdown
exit
-----
```

6.15 Configuring synchronization and redundancy

6.15.1 Configuring subscriber management persistence

The following example shows how to configure subscriber management persistence.

Example: MD-CLI

```
[ex:/configure system persistence]
A:admin@node-2# info
  subscriber-mgmt {
    description "cf3:SubMgmt-Test"
    location cf3
  }
```

Example: classic CLI

```
A:node-2>config>system>persistence# info
-----
  subscriber-mgmt
    description "cf3:SubMgmt-Test"
    location cf3:
  exit
-----
```

6.15.2 Configuring a CLI script file for synchronization

You can specify the location and name of the CLI script file executed following a redundancy switchover from the previously active CPM card. Use the following command to configure the file URL:

- **MD-CLI**

```
configure redundancy switchover-exec file-url
```

- **classic CLI**

```
configure system switchover-exec file-url
```

6.15.3 Configuring synchronization options

You can specify the type of synchronization operation to perform between the primary and secondary CPMs after a change is made to the configuration files or the boot environment information in the boot options file (BOF).

Use the following commands to configure which types of changes cause automatic synchronization:

```
configure redundancy synchronize boot-env
configure redundancy synchronize config
```

See [File synchronization](#) for information about which files are synchronized with automatic synchronization.

6.15.4 Displaying synchronization options

Use the following command to display the synchronization information.

```
show redundancy synchronization
```

Example: Synchronization information with boot-env option configured

```
=====
Synchronization Information
=====
Standby Status           : disabled
Last Standby Failure     : N/A
Standby Up Time          : N/A
Standby Version           : N/A
Failover Time            : N/A
Failover Reason          : N/A
Boot/Config Sync Mode    : Boot Environment
Boot/Config Sync Status  : No synchronization
Last Config File Sync Time : Never
Last Boot Env Sync Time  : Never
Rollback Sync Mode       : None
Rollback Sync Status     : No Rollback synchronization
Last Rollback Sync Time  : Never
Certificate Sync         : Enabled
Cert Sync Status         : unknown
Last Cert Sync Time      : Never
=====
```

6.16 Configuring multichassis redundancy for LAG

When configuring the associated LAG ID, the LAG must be in access mode and LACP must be enabled. The following example configures multichassis redundancy features.

Example: MD-CLI

```
[ex:/configure redundancy multi-chassis]
A:admin@node-2# info
  peer 10.10.10.2 {
    admin-state enable
    description "Mc-Lag peer 10.10.10.2"
    mc-lag {
      admin-state enable
      lag "lag-1" {
        lacp-key 32666
        system-id 00:00:00:33:33:33
        system-priority 32888
      }
    }
  }
```

```
    }
}
```

Example: classic CLI

```
A:node-2>config>redundancy>multi-chassis# info
-----
    peer 10.10.10.2 create
      description "Mc-Lag peer 10.10.10.2"
      mc-lag
        no shutdown
      exit
      no shutdown
    exit
-----
```

6.17 Post-boot configuration extension files

Two post-boot configuration extension files are supported and are triggered when either a successful or failed boot configuration file is processed. The commands specify URLs for the CLI scripts to be run following the completion of the startup configuration. A URL must be specified or no action is taken. The commands are persistent between router reboots and are included in the configuration saves.

Use the following commands to specify the CLI scripts that are run following the completion of the boot-up configuration.

```
configure system boot-bad-exec
configure system boot-good-exec
```

6.17.1 Show command output and console messages

Use the following command to show the current value of the bad and good exec URLs and indicate whether a post-boot configuration extension file was executed when the system was booted.

```
show system information
```

If an extension file was executed, the command also indicates whether it completed successfully.

The following example shows the show output for the 7750 SR.

Output example: Show system information output for the 7750 SR

```
=====
System Information
=====
System Name           : node-2
...
BOF Source            : cfl:
Image Source          : primary
Config Source         : primary
Last Booted Config File: ftp://test:test@192.168.xx.xxx/./12.cfg
Last Boot Cfg Version  : MON MAR 07 16:58:46 2022 UTC
Last Boot Config Header: # TiMOS-B-22.2.R1 both/x86_64 Nokia 7750 SR
                        Copyright (c) 2000-2022 Nokia.
```

```

# All rights reserved. All use subject to applicable license
agreements.
# Built on Sat Feb 26 15:31:00 PST 2022 by builder in /
builds/c/222B/R1/panos/main/sros
# Configuration format version 22.2 revision 0
# Generated MON MAR 07 16:58:46 2022 UTC
Last Boot Index Version: N/A
Last Boot Index Header : N/A
Last Saved Config      : N/A
Time Last Saved       : N/A
Changes Since Last Save: Yes
Time Last Modified    : 2004/03/06 03:30:45
Max Cfg/BOF Backup Rev : 7
Cfg-OK Script         : ftp://test:test@192.168.xx.xxx/./ok.cfg
Cfg-OK Script Status  : not used
Cfg-Fail Script        : ftp://test:test@192.168.xx.xxx/./fail.cfg
Cfg-Fail Script Status : not used
...
=====

```

When executing a post-boot configuration extension file, status messages are displayed on the console before the "Login" prompt.

The following example shows a failed bootup configuration that caused a boot-bad-exec file containing another error to be executed.

Example: Failed start-up configuration error message

```

Attempting to exec configuration file:
'ftp://test:test@192.168.xx.xxx/./12.cfg' ...
System Configuration
Log Configuration
MAJOR: CLI #1009 An error occurred while processing a CLI command -
File ftp://test:test@192.168.xx.xxx/./12.cfg, Line 195: Command "log" failed.
CRITICAL: CLI #1002 An error occurred while processing the configuration file.
The system configuration is missing or incomplete.
MAJOR: CLI #1008 The SNMP daemon is disabled.
If desired, enable SNMP with the 'config>system>snmp no shutdown' command.
Attempting to exec configuration failure extension file:
'ftp://test:test@192.168.xx.xxx/./fail.cfg' ...
Config fail extension
Enabling SNMP daemon
MAJOR: CLI #1009 An error occurred while processing a CLI command -
File ftp://test:test@192.168.xx.xxx/./fail.cfg, Line 5: Command "abc log" failed.
TiMOS-B-x.0.Rx both/hops ALCATEL Copyright (c) 2000-2001 Alcatel-Lucent.
All rights reserved. All use subject to applicable license agreements.
Built on Thu Nov 207 19:19:11 PST 2008 by builder in /rel5x.0/b1/Rx/panos/main

Login:

```

6.18 Configuring system monitoring thresholds

This section provides information about configuring system monitoring thresholds.

6.18.1 Creating events

The **event** command controls the generation and notification of threshold crossing events configured with the **alarm** and **hc-alarm** (high capacity) commands. When a threshold crossing event is triggered, the **rmon event** configuration optionally specifies whether an entry in the alarm table is created to record the occurrence of the event. It can also specify whether an SNMP trap be generated for the event. There are two notifications for threshold crossing events: a rising alarm and a falling alarm.

Creating an event entry in the alarm table does not create a corresponding entry in the event logs. However, when the event is set to trap, the generation of a rising alarm or falling alarm notification creates an entry in the event logs and that is distributed to whatever log destinations are configured: console, session, memory, file, syslog, or SNMP trap destination. The log message includes a rising or falling threshold crossing event indicator, the sample type (absolute or delta), the sampled value, the threshold value, the RMON alarm ID, the associated RMON event ID, and the sampled SNMP object identifier.

The **alarm** command configures an entry in the RMON-MIB::alarmTable. The **hc-alarm** command configures an entry in the HC-ALARM-MI::hcAlarmTable. These commands control the monitoring and triggering of threshold crossing events. For notification or logging of a threshold crossing event to occur, there must be at least one associated **rmon event** configured.

The agent periodically takes statistical sample values from the MIB OID specified for monitoring and compares them to thresholds that have been configured. The **alarm** and **hc-alarm** commands configure the MIB OID to be monitored, the polling period (interval), sampling type (absolute or delta value), and rising and falling threshold parameters. If a sample has crossed a threshold value, the associated event is generated.

Preconfigured CLI threshold commands are available. Preconfigured commands hide some of the complexities of configuring RMON alarm and event commands and perform the same function. The preconfigured commands do not require the user to know the SNMP object identifier to be sampled. The preconfigured threshold configurations include memory warnings and alarms and compact flash usage warnings and alarms.

Example: MD-CLI

```
[ex:/configure system thresholds]
A:admin@node-2# info
  cflash-cap-warn-percent "cf1-B:" {
    rising-threshold 100
    falling-threshold 50
    interval 240
    startup-alarm either
  }
  kb-memory-use-alarm {
    rising-threshold 50000000
    falling-threshold 45999999
    interval 500
    startup-alarm either
  }
  rmon {
    event 5 {
      description "alarm testing"
      owner "Timos CLI"
    }
  }
}
```

Example: classic CLI

```
A:node-2>config>system>thresholds# info
-----
      rmon
        event 5 description "alarm testing" owner "Timos CLI"
      exit
      cflash-cap-warn cfl-B: rising-threshold 2000000 falling-threshold
1999900 interval 240 trap
      memory-use-alarm rising-threshold 50000000 falling-threshold
45999999 interval 500
-----
```

6.19 Configuring LLDP

Use the commands in the following context to configure LLDP.

```
configure system lldp
```

Example: LLDP default configuration (MD-CLI)

```
[ex:/configure system lldp]
A:admin@node-2# info detail
## apply-groups
## apply-groups-exclude
  admin-state enable
  tx-credit-max 5
  message-fast-tx 1
  message-fast-tx-init 4
  tx-interval 30
  tx-hold-multiplier 4
  reinit-delay 2
  notification-interval 5
```

Example: LLDP default configuration (classic CLI)

```
A:node-2>config>system>lldp# info detail
-----
      no tx-interval
      no tx-hold-multiplier
      no reinit-delay
      no notification-interval
      no tx-credit-max
      no message-fast-tx
      no message-fast-tx-init
      no shutdown
-----
```

Example: LLDP port configuration (MD-CLI)

```
[ex:/configure port 1/1/1 ethernet lldp]
A:admin@node-2# info
  dest-mac nearest-bridge {
    receive true
    transmit true
  }
  tx-tlvs {
    port-desc true
```

```

        sys-cap true
    }
    tx-mgmt-address system {
        admin-state enable
    }
}

```

Example: LLDP port configuration (classic CLI)

```

A:node-2>config>port>ethernet>lldp# info
-----
dest-mac nearest-bridge
      admin-status tx-rx
      tx-tlvs port-desc sys-cap
      tx-mgmt-address system
exit
-----

```

Example: Global system LLDP configuration (MD-CLI)

```

[ex:/configure system lldp]
A:admin@node-2# info
tx-interval 10
tx-hold-multiplier 2
reinit-delay 5
notification-interval 10

```

Example: Global system LLDP configuration (classic CLI)

```

A:node-2>config>system>lldp# info
-----
tx-interval 10
tx-hold-multiplier 2
reinit-delay 5
notification-interval 10
-----

```

6.20 Configuring low-power mode features

This section provides information about configuring low-power mode features to reduce power consumption. The following types of low-power mode features are supported:

- operational features - reduce system forwarding capacity by disabling active components, such as SFMs, during periods of lower network traffic to lower power consumption
- provisioning features - disable inactive or provisioned components that are not in use, such as line cards, ports, and optics to lower power consumption

6.20.1 7750 SR-s low-power switch fabric mode

The SR OS low-power switch fabric mode provides a set of commands to monitor switch fabric utilization and place a configurable number of SFMs in power save mode, which significantly reduces their power utilization. Low-power fabric mode can be enabled by the operator or management system during periods

of lower network traffic, such as from midnight to 5:00 a.m., allowing the router to provide the required forwarding capacity with fewer active SFMs.

Switch fabric utilization should first be monitored for a period of time, typically several weeks or months, to determine the baseline hourly utilization of the router. This helps identify time periods when the full switching capacity of the router is not needed, and the user can enable low-power fabric mode to reduce power consumption. The switch fabric utilization can be monitored using:

- the following CLI command

```
show system switch-fabric utilization
```

Output example: show system switch-fabric utilization

```
=====
Switch Fabric Utilization (Sample period: 10 seconds)
=====
Slot/FP          Utilization
-----
1/1              74%
1/2              75%
1/3              73%
1/4              79%
=====
```

- NETCONF or gRPC to monitor the following YANG state paths

```
state system switch-fabric utilization
```

- SNMP to monitor the tmnxSysSwFabResMonTable OIDs in the TIMETRA-SYSTEM-MIB

Monitor the chassis power utilization while low-power fabric mode is disabled to determine the power savings when it is enabled using the following command.

```
show chassis power-management utilization
```

Output example: Chassis power utilization

```
=====
Chassis Power Zone 1 Utilization
=====
SUPPLY                                PEAK DEMAND
Power Capacity   :    0.00 Watts      Chassis/Fan    :   953.62 Watts
                                           IO Module     :  1195.18 Watts
                                           CPM Module    :   540.32 Watts
                                           Fabric Module :   959.33 Watts
                                           MDA Module    :  3660.18 Watts
                                           XIOM Module   :   475.89 Watts
                                           Peak Util.    :  7784.52 Watts

Mode           :    none              Current Util. :  7496.55 Watts
Reserved Power :    0.00 Watts        Safety Level   :    0.00 Watts (  0%)
Util. + Reserve:  7496.55 Watts        Safety Alert   :    N/A
Remaining Power: -7496.00 Watts        Alert Level    :    N/A
Active Conditions: safety-level
=====
```

Use the following command to administratively enable low-power fabric mode:

- **MD-CLI**

```
configure system power-management power-save low-power-fabric admin-state enable
```

- **classic CLI**

```
configure system power-management power-save low-power-fabric no shutdown
```

Low-power mode only needs to be administratively enabled once, and remains enabled until disabled in the configuration. When low-power fabric mode is administratively enabled, all SFMs run at full power until low-power fabric mode is operationally enabled. Use the following command to display the administrative state.

```
show chassis power-management power-save
```

Output example: Administratively enabled low-power fabric mode

```
=====
Power Save Information
=====
Low Power Switch Fabric Mode
-----
Admin State           : Up
Oper State            : Down
Switch Fabric Capacity : 100.00%
=====
```

Select a switch fabric capacity percentage that provides the required forwarding capacity. Execute the following command at the scheduled time to operationally enable low-power fabric mode, with the needed fabric capacity specified, in this example 50%.

```
tools perform system power-management power-save low-power-fabric capacity 50
```

When low-power fabric mode is operationally enabled, the operational switch fabric capacity percentage is displayed using the following command.

```
show chassis power-management power-save
```

Output example: show chassis power-management power-save

```
=====
Power Save Information
=====
Low Power Switch Fabric Mode
-----
Admin State           : Up
Oper State            : Up
Switch Fabric Capacity : 50.00%
=====
```

When low-power fabric mode is operationally enabled, the SFM operational state is displayed as “power save” using the following command.

```
show sfm
```

Output example: show sfm

```
=====
```

SFM Summary				
Slot	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Comments
1	sfm-s	up	power save	
2	sfm-s	up	power save	
3	sfm-s	up	power save	
4	sfm-s	up	power save	
5	sfm-s	up	up	
6	sfm-s	up	up	
7	sfm-s	up	up	
8	sfm-s	up	up	

```
=====
```

Monitor the switch fabric utilization to verify that adequate switch fabric capacity is still available using the following command.

```
show system switch-fabric utilization
```

Output example: show system switch-fabric utilization

```
=====
```

Switch Fabric Utilization (Sample period: 10 seconds)	
Slot/FP	Utilization
1/1	35%
1/2	33%
1/3	34%
1/4	39%

```
=====
```

Use the following command to determine how much power is saved.

```
show chassis power-management utilization
```

Output example: show chassis power-management utilization

```
=====
```

Chassis Power Zone 1 Utilization			
SUPPLY		PEAK DEMAND	
Power Capacity	: 0.00 Watts	Chassis/Fan	: 945.20 Watts
		I/O Module	: 1192.24 Watts
		CPM Module	: 540.32 Watts
		Fabric Module	: 472.44 Watts
		MDA Module	: 3652.08 Watts
		XIOM Module	: 475.36 Watts
		Peak Util.	: 7277.64 Watts
Mode	: none	Current Util.	: 6561.27 Watts
Reserved Power	: 0.00 Watts	Safety Level	: 0.00 Watts (0%)
Util. + Reserve	: 6561.27 Watts	Safety Alert	: N/A
Remaining Power	: -6561.00 Watts	Alert Level	: N/A
Active Conditions:	safety-level		

```
=====
```

Execute the following command at the scheduled time to operationally disable low-power fabric mode, by setting the switch fabric capacity to 100%.

```
tools perform system power-management power-save low-power-fabric capacity 100
```

The low-power fabric mode administrative state does not need to be disabled each time the operational state is disabled. Use the commands below to verify that the full switch fabric capacity is available.

Use the following command to verify the switch fabric capacity percentage.

```
show chassis power-management power-save
```

Output example: show chassis power-management power-save

```
=====
Power Save Information
=====
Low Power Switch Fabric Mode
-----
Admin State           : Up
Oper State            : Down
Switch Fabric Capacity : 100.00%
=====
```

Use the following command to verify the administrative state of the SFM.

```
show sfm
```

Output example: show sfm

```
=====
SFM Summary
=====
```

Slot	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Comments
1	sfm-s	up	up	
2	sfm-s	up	up	
3	sfm-s	up	up	
4	sfm-s	up	up	
5	sfm-s	up	up	
6	sfm-s	up	up	
7	sfm-s	up	up	
8	sfm-s	up	up	

```
=====
```

Low-power fabric mode can also be operationally enabled and disabled using the following:

- CRON scheduling on the router to execute the following command:

```
tools perform system power-management power-save low-power-fabric capacity
```

- NETCONF to execute the following YANG-modeled action:

```
perform system power-management power-save low-power-fabric capacity
```

- SNMP to set the tChassisPwrMgmtLowPwrFabCapacity OID (not supported in model-driven configuration mode)

The following usage guidelines apply to low-power fabric mode:

- This feature is supported on the 7750 SR-2s, 7750 SR-2se, 7750 SR-7s, and 7750 SR-14s platforms. On the 7750 SR-2se, this feature is only supported with "xcm-2se" IOMs and cannot be operationally enabled with "xcmc-2se" IOMs.
- Operationally enabling the feature takes several seconds to put the SFMs into power save mode. Operationally disabling low-power fabric mode, however, can take several minutes as the SFMs must boot and become operational again.
- A few packets that are in transit through the switching fabric may be dropped when SFMs are put into power save mode.
- A few packets may be dropped when SFMs are brought back to operational mode.
- All components in the router, such as SFMs and IOMs, must be operational or unprovisioned before low-power fabric mode can be operationally enabled.
- If any component in the router fails while SFMs are in power save mode, low-power fabric mode is operationally disabled.
- Each SFM has additional fabric links for resiliency that provide slightly more capacity to the system. For example, enabling low-power mode on one SFM in the 7750 SR-7s, which has four total SFMs, would provide a forwarding capacity of slightly more than 75%.
- Unexpected traffic bursts that exceed the operational low-power fabric capacity may be dropped.

6.20.2 7750 SR low-power optic mode

The SR OS low-power optic mode provides a per-port command to reduce optic and port power consumption when a port is populated with an optic, is administratively disabled, and does not need to be fully powered. For example, when a card is provisioned and populated with optics in every port, and ports are then incrementally deployed to meet capacity demands over a period of time. When low-power optic mode is enabled and the port is administratively disabled, the optic is placed into low-power mode and the associated links between the ASICs are disabled to save power. The port does not forward traffic, but the optic can still be managed by the system.

Low-power optic mode can be configured when a port is administratively enabled, but does not take effect until the port is administratively disabled.

Use the following command to administratively enable or disable low-power optic mode.

```
configure port transceiver power-save
```

When the optic is operationally in low-power mode, the transceiver status displays "low-power-mode". Use the following command to display the operational mode.

```
show port optical
```

Example: Optic in low-power mode

```
=====
Optical Interface : 5/1/c5
=====
```

Transceiver Data

```

Transceiver Status : low-power-mode
Transceiver Type   : QSFP28                      DCO           : Disabled
OLS                : Disabled
Model Number       : 3HE10551AARA01  NOK  IPU3BFVEAA
TX Laser Wavelength: 850 nm                      Diag Capable   : yes
Number of Lanes    : 4
Connector Code     : MP0 1x12                    Vendor OUI      : 00:90:65
Manufacture date   : 2017/12/06                  Media           : Ethernet
Serial Number      : XYK0J2E
Part Number        : FTL9551REPM-A5
Optical Compliance : 100GBASE-SR4
Link Length support: 70m for OM3; 100m for OM4

```

The following usage guidelines apply to the low-power optic mode:

- This feature is not supported on the following types of ports:
 - CPM 1588/sync ports (for example, port A/3) with SFPs
 - FlexE client ports
 - Port Cross-Connect (PXC) ports
 - satellite uplink and client ports
- When using connector ports, the **power-save** command must be configured on the connector port (for example, port 1/1/c1) and cannot be configured on each breakout port (port 1/1/c1/1, and so on). The optic enters low-power mode when all breakout ports are administratively disabled, even when the connector port is administratively enabled.

6.20.3 Low-power card mode

The SR OS low-power card mode provides a per-card command to reduce card power consumption when a card is inserted in the system and does not need to be powered on. For example, when a card is provisioned for future capacity requirements.

When low-power card mode is enabled, the card is placed in an idle state that consumes minimal power. The card is not counted in the intelligent power management budget and does not forward traffic.

The card must be administratively disabled before low-power card mode can be enabled. Use the following commands to configure a card for low-power card mode:

- **MD-CLI**

```

configure card admin-state disable
configure card power-save

```

- **classic CLI**

```

configure card shutdown
configure card power-save

```

When a card is in low-power card mode, the operational status displays “provisioned” and the equipped type is “(not equipped)”. Use the following command to display the operational status.

```
show card
```

Example: Low-power card mode

=====				
Card Summary				
=====				
Slot	Provisioned Type	Admin Operational		Comments
	Equipped Type (if different)	State	State	

1	xcm2-7s	down	provisioned	
	(not equipped)			
2	xcm2-7s	up	up	
3	xcm2-7s	up	up	
4	xcm2-7s	up	up	
6	xcm2-7s	up	up	
A	cpm2-s	up	up/active	
B	cpm2-s	up	up/standby	
=====				

7 CPM redundancy

This chapter provides information about configuring CPM redundancy parameters.

7.1 File synchronization

7450 ESS, 7750 SR, and 7950 XRS routers that support CPM redundancy use a 1:1 redundancy scheme. CPM redundancy requires file synchronization between the active and standby CPM compact flash drives so that they maintain identical system files to prevent inconsistencies in the event of a CPM failure.

If synchronization fails, alarms and log messages that indicate the type of error that caused the failure of the synchronization operation are generated. When the error condition ceases to exist, the alarm is cleared.

Only files stored on the router are synchronized. If a configuration file or image is stored in a location other than on a local compact flash, the file is not synchronized (for example, storing a configuration file on an FTP server).

Two types of synchronization are supported:

- **automatic**

Automatic synchronization is set to **config** by default on systems with CPM redundancy. You can configure automatic synchronization for all files required for the boot process, or you can configure it for only the configuration files. Use the following command to configure which type of automatic synchronization to perform.

```
configure redundancy synchronize
```

When automatic synchronization is set to **config**, the following files are synchronized if they have been modified:

- the configuration files (`config.cfg`) and their backups in the primary, secondary, and tertiary configuration locations
- the LI configuration file (`li.cfg`) and its backups
- the model-driven commit history and incremental saved configuration files in the `.commit-history` directory when the system is in model-driven configuration mode
- the persistent index file (`config.ndx`) and its backups when the system is in classic or mixed configuration mode
- SSH key files in the `.ssh` directory
- the password history file (`passwd/history`)

When automatic synchronization is set to **boot-env**, the following files are synchronized if they have been modified and are applicable to the platform:

- the configuration files (`config.cfg`) and their backups in the primary, secondary, and tertiary configuration locations
- the BOF configuration file (`bof.cfg`) and its backups

- the LI configuration file (`li.cfg`) and its backups
- the model-driven commit history and incremental saved configuration files in the `.commit-history` directory when the system is in model-driven configuration mode
- the persistent index file (`config.ndx`) and its backups when the system is in classic or mixed configuration mode
- the boot loader file (`boot.ldr`)
- all SR OS image (`.tim`) files in the primary, secondary, and tertiary image locations
- signature (`.txt`) files (for example, `sha256sum.txt`)
- the nvsys file (`nvsys.info`)
- SSH key files in the `.ssh` directory
- the password history file (`passwd/history`)
- classic CLI rollback files (`.rb`) and their backups
- the license file (`licence.txt`) in the license location

BOF configuration file (`bof.cfg`) and file synchronization also occurs when the following BOF parameter values are changed:

- the configuration files (`config.cfg`) and their backups in the primary, secondary, and tertiary configuration locations
- all image (`.tim`) files in the primary, secondary, and tertiary image locations
- the license file (`licence.txt`) in the license location

Configuration file (`config.cfg`) synchronization also occurs when the configuration is committed in model-driven interfaces if the **auto-config-save** command is enabled, or is saved with the **admin save** command.

The debug configuration file (`debug.cfg`) and its backups are not synchronized, because the debug configuration is not automatically loaded after a CPM switchover. These files must be manually copied to standby CPM using the following command, if needed:

- **MD-CLI**

```
file copy
```

- **classic CLI**

```
file cp
```

LI configuration file (`li.cfg`) synchronization also occurs when the LI configuration is committed in model-driven interfaces or is saved using the following command:

- **MD-CLI**

```
admin save li
```

- **classic CLI**

```
li save
```

- **manual**

You can manually synchronize all files required for the boot process, the configuration files, or PKI certificate files in the **system-pki** directory. Manual synchronization synchronizes all files regardless whether or not they have been modified, which takes a significantly longer time than automatic synchronization. Use the following commands to perform manual synchronization:

– MD-CLI

```
admin redundancy synchronize boot-environment
admin redundancy synchronize configuration
admin redundancy synchronize certificate
```

– classic CLI

```
admin redundancy synchronize boot-env
admin redundancy synchronize config
admin redundancy synchronize cert
```



Note:

- See "Configuration rollback" in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide* for more information about classic CLI configuration rollback file synchronization.
- See the **configure redundancy cert-sync** command in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide* and *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI Command Reference Guide* for more information about PKI certificate synchronization.

7.2 Active and standby designations

Typically, the first Switch Fabric (SF)/CPM card installed in a redundant 7450 ESS, 7750 SR, 7950 XRS, and VSR router chassis assumes the role as active, regardless of whether it is inserted in Slot A or B. The next CPM installed in the same chassis then assumes the role as the standby CPM. If two CPM are inserted simultaneously (or almost simultaneously) and are booting at the same time, then preference is given to the CPM installed in Slot A.

If only one CPM is installed in a redundant router device, then it becomes the active CPM regardless of the slot it is installed in.

The active and standby designations can be visually determined by LEDs on the CPM/CCM faceplate. See the appropriate platform *Installation Guide* for LED indicator details.

The following example shows the output when the CPM installed in Slot A is acting as the active CPM and the CPM installed in Slot B is acting as the standby.

Output example: Show card command output on the 7950 XRS

Card Summary				
Slot	Provisioned Type Equipped Type (if different)	Admin State	Operational State	Comments
1	xcm-x20	up	provisioned	
A	cpm-x20	up	up/active	
B	cpm-x20	up	up/standby	

=====

The following example shows the console message when a CPM boots, sees an active CPM, and becomes the standby CPM.

Example: Console message for CPM switch to standby

```
...
Slot A contains the Active CPM
This CPM (Slot B) is the Standby CPM
```

7.3 When the active CPM goes offline

When an active CPM goes offline (because of reboot, removal, or failure), the standby CPM takes control without rebooting or initializing itself. It is assumed that the CPMs are synchronized, therefore, there is no delay in operability. When the CPM that went offline boots and then comes back online, it becomes the standby CPM.

When the standby CPM comes online, the following output is shown.

Example

```
Active CPM in Slot A has stopped
Slot B is now active CPM

Attempting to exec configuration file:
'cf3:/config.cfg' ...

...

Executed 49,588 lines in 8.0 seconds from file cf3:\config.cfg
```

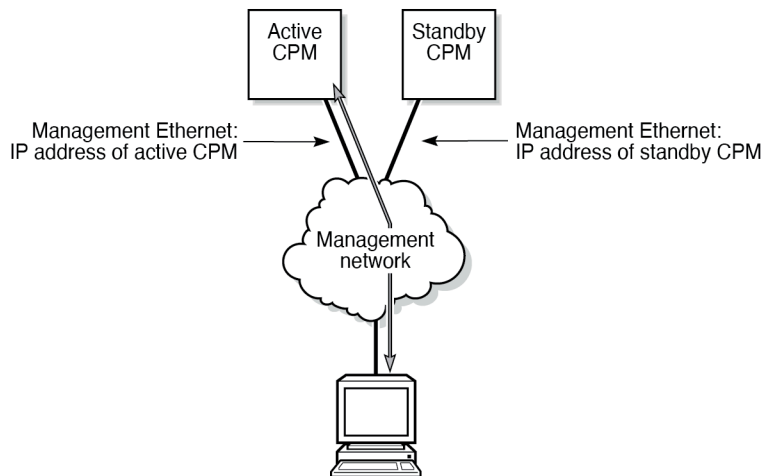
7.4 OOB management Ethernet port redundancy

SR OS provides a resilient out-of-band (OOB) management Ethernet redundancy mode for system management.

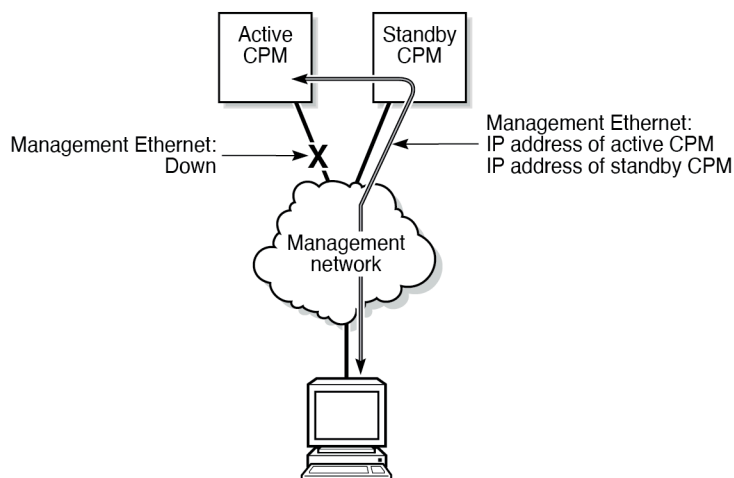
Use the following command to enable OOB management Ethernet port redundancy.

```
configure redundancy mgmt-ethernet
```

When the management Ethernet port is down on the active CPM, the OOB Ethernet redundancy feature allows the active CPM to use the management Ethernet port of the standby CPM, as shown in the following figures.

Figure 31: Management Ethernet: normal mode

25169

Figure 32: Management Ethernet: redundancy mode

25168

7.5 DHCP persistence

The DHCP persistence feature on the 7750 SR retains information learned through DHCP snooping across reboots. This information can include data such as the IP address, MAC binding information, lease length information, and ingress SAP information (required for VPLS snooping to identify the ingress interface). This information is referred to as the DHCP lease-state information.

When a DHCP message is snooped, the system performs the following steps to make the data persistent. For systems with only one CPM, only step 1 applies; for those with dual CPMs, all steps apply.

1. When a DHCP ACK is received from a DHCP server, the entry information is written to the active CPM compact flash. If writing is successful, the ACK is forwarded to the DHCP client. If persistency

fails completely (for example, because of a bad compact flash), a trap is generated indicating that persistency can no longer be guaranteed.

If the complete persistency system fails, the DHCP ACKs are still forwarded to the DHCP clients. DHCP ACKs may get dropped and not forwarded to the DHCP clients only during small persistency interruptions, or in overload conditions of the compact flash.

2. DHCP message information is sent to the standby CPM, where the DHCP information is also logged on the compact flash. If persistency fails on the standby CPM, a trap is generated.

7.5.1 DDP access optimization for DHCP leases

A high rate of DHCP renewals can create a load on the compact flash file system when subscriber management and DHCP server persistence is enabled. To optimize the access to the Dynamic Data Persistency (DDP) files on the compact flash, a lease-time threshold can be specified that controls the eligibility of a DHCP lease for persistency updates when no other data other than the lease expiry time is to be updated.

The following example shows the configuration of a DHCP lease-time threshold.

Example: MD-CLI

```
[ex:/configure system persistence]
A:admin@node-2# info
    dhcp-server {
        location cf2
    }
    subscriber-mgmt {
        location cf2
    }
    options {
        dhcp-leasetime-threshold 90061
    }
```

Example: classic CLI

```
A:node-2>config>system>persistence# info
-----
    subscriber-mgmt
        location cf2:
    exit
    dhcp-server
        location cf2:
    exit
    options
        dhcp-leasetime-threshold days 1 hrs 1 min 1 sec 1
    exit
-----
```

When the offered lease time of the DHCP lease is less than the configured threshold, the lease is flagged to skip persistency updates and is installed with its full lease time upon a persistency recovery after a reboot.

The **dhcp-leasetime-threshold** command controls persistency updates for:

- DHCPv4 and DHCPv6 leases for a DHCP relay or proxy (enabled with **persistence subscriber-mgmt**)
- DHCPv4 leases for DHCP snooping in a VPLS service (enabled with **persistence subscriber-mgmt**)

- DHCPv4 and DHCPv6 leases for a DHCP server (enabled with **persistence dhcp-server**)

To check if a DHCP relay or proxy lease is flagged to skip persistency updates, use the **tools dump persistence submgt record record-key** CLI command. When flagged to skip persistency updates, the persistency record output includes "Skip Persistency Updates: true".

To check if a DHCP server lease is flagged to skip persistency updates, use the **tools dump persistence dhcp-server record record-key** CLI command. When flagged to skip persistency updates, the persistency record output includes "lease mode : LT" (LT = Lease Time) and a "lease time : ..." field. When not flagged to skip persistency updates, the persistency record output includes "lease mode : ET" (ET = Expiry Time) and an "expires : ..." field.

8 Zero touch provisioning

Traditional deployment of new nodes in a network is a multi-step process in which the user connects to the hardware to provision global and local configuration options. ZTP automatically configures the node by obtaining the required information from the network and automatically provisioning the node with minimal manual intervention and configuration. When nodes that support ZTP are installed in the rack, connected to the network, and powered on, the nodes are auto-provisioned.



Note: To support ZTP, make sure the new nodes are purchased with the **auto-boot** flag enabled in the factory-loaded BOF.

8.1 ZTP overview

ZTP is used to automatically install and provision new nodes in the field. For out-of-band management, the nodes can be installed and powered up with network connectivity on the management (Mgmt) port. For in-band management, the first two connectors on the first two slots can be used for ZTP.



Note: For breakout connectors, only the first breakout port on the first two connectors can be used for ZTP.

ZTP VLAN discovery is enabled by default.

After network connectivity is established, the ZTP process starts automatically. The node sends a DHCP discovery request to the DHCP server using a ZTP-capable port and the DHCP server returns an IPv4/IPv6 FTP or HTTP URL from which the provisioning information can be retrieved. The provisioning information is in a file called the provisioning file, which contains the URL of the image, config, and other files to be downloaded. After downloading these files and successfully provisioning, the node automatically reboots and comes back up in normal mode.

Secure ZTP (SZTP), which is an extension of ZTP, is also supported. See [Legacy SZTP \(non RFC 8572-compliant\)](#) for information about SZTP.



Note: ZTP and SZTP support TLS 1.2 and TLS 1.3 for HTTPS.

8.1.1 Network requirements

ZTP requires the following network components:

- **DHCP server (IPv4 or IPv6)**

The DHCP server supports assignment of IP addresses through DHCP requests and offers.

- **file server**

The FTP or HTTP file server is used for staging and transfer of RPMs, configurations, images, and scripts.

- **DHCP relay**

A DHCP relay is required if the servers are across a Layer 3 network.

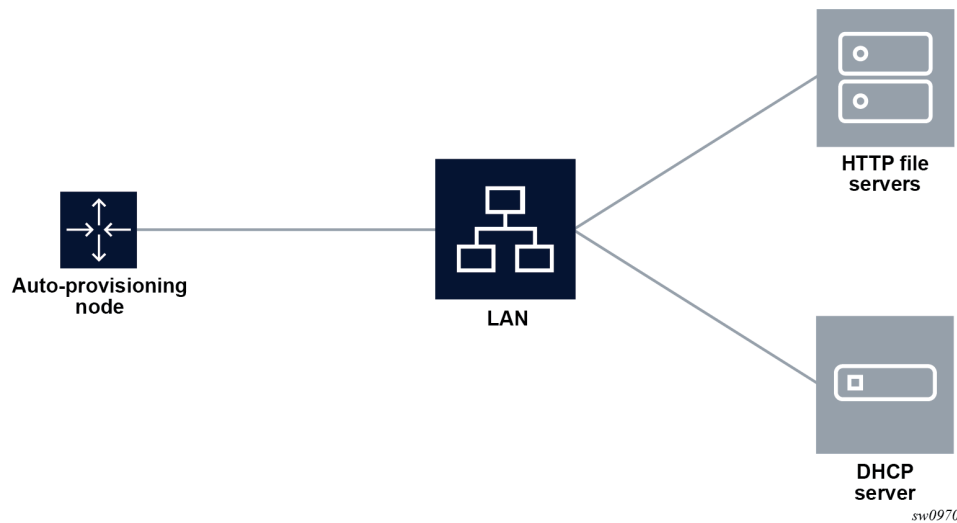
8.1.2 Network support

ZTP operates in the following network environments:

- **node, file servers, and DHCP server in the same subnet**

The following figure shows the scenario where all components are in a Layer 2 broadcast domain. There is no DHCP relay and all IP addresses are assigned from a single pool.

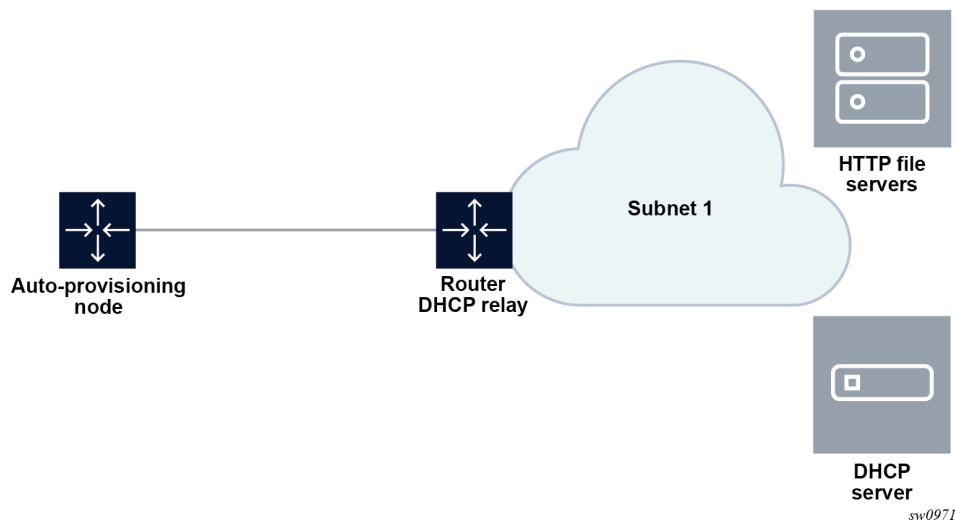
Figure 33: Auto-provisioning with all components in the same subnet



- **file servers and DHCP server in the same subnet, separate from the nodes**

The following figure shows the scenario where only the file servers and DHCP server are in the same subnet. The DHCP relay is used to fill option 82 as the gateway address. The gateway address is used to find the appropriate pool in the DHCP server to assign the correct subnet IP address to the system.

Figure 34: Auto-provisioning with only file and DHCP servers in the same subnet

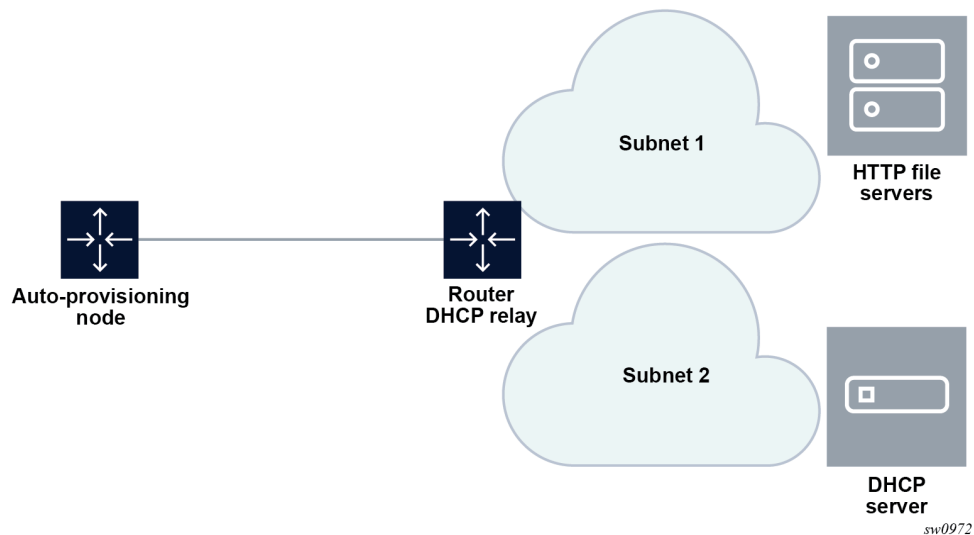


DHCP allows the Option 3 router to define the default gateway. If multiple addresses are provided using Option 3, the first address is used for the default gateway.

- **node, file servers, and DHCP server in different subnets**

The following figure shows the scenario where all components are in different subnets. The DHCP relay adds the option 82 gateway address to the DHCP request, and the DHCP server adds the option 3 with the gateway address of the file server.

Figure 35: Auto-provisioning with all components in different subnets



8.2 ZTP process overview

ZTP consists of the following processes:

- [Auto-boot process](#)
- [Auto-provisioning process](#)

8.2.1 Auto-boot process

In this process, the node discovers and provisions the chassis and installed cards.

1. The node is first connected to the network and powered on.
2. The out-of-band management port is checked for link connectivity. If a link is not found, the system checks the in-band management ports for a link.
3. The first two card or MDA slots are auto-provisioned based on the installed card types. See [ZTP overview](#) for information about the specific card or MDA slots that are used.
4. The auto-boot process switches control to the auto-provisioning process.

See [Auto-boot process details](#) for more information about the auto-boot process.

8.2.2 Auto-provisioning process

In this process, the node detects operational ports, attempts to discover its IP address, and downloads the relevant files for provisioning.

1. The node sends a DHCP discovery request to the DHCP server using the out-of-band management port. If DHCP discovery is unsuccessful, the node reattempts it using the in-band management ports.
2. After DHCP discovery is successful, the DHCP server returns an IPv4 or IPv6 FTP or HTTP URL of a file server from which the node can retrieve provisioning information.
3. The node downloads the provisioning information and performs the auto-provisioning according to the specifications in the files.
4. After the node is successfully provisioned, it automatically reboots and becomes operationally up.

See [Provisioning files](#) for more information about the auto-provisioning process.

The SR OS can also initiate the auto-provisioning process using a **tools** command.

8.3 DHCP support for ZTP

This section provides information about DHCP messages, DHCP clients, and DHCP servers that are supported by ZTP.

8.3.1 DHCP server offer options

Options 66, 67, and 43 are supported for indicating the location of the provisioning file. If both Options 66 and 67 are present in the DHCP offer, they take precedence over Option 43.

Option 66 contains the server URL or IP address, and Option 67 contains the URL of the provisioning file location.

Options 66 and 67 are meant for use by PXE TFTP, but are also used for HTTP and FTP. If an offer arrives with Options 66 and 67, Option 66 should resolve the server IP address and Option 67 should resolve the file location. Option 66 can be omitted by the provider, in which case Option 67 is used for both the server IP address and provider file URL. If an offer arrives with Option 67 only, it should resolve both the server IP address and file URL.

The auto-provisioning process distinguishes the host part of the URL and can resolve it using DHCP DNS.

8.3.1.1 Nokia-specific TLV

The Nokia-specific TLV is NOKIA-DCTOR-AUTOCONFIG. The location of the BOF for each system to use is configured in the optional **autoboot** file parameter, which is a standard Option 43 value initialized at the beginning of the process. The BOF location is sent in Option 43 as part of the DHCP offer and Ack messages from the DHCP server to the system. The system uses the location specified in Option 43 to initiate an FTP download of the BOF.

8.3.2 Supported DHCP client options for ZTP

The following table lists the supported DHCP client options for ZTP.

Table 32: Supported DHCP client options for ZTP

Options	DHCP IPv4 option	IPv4 comments	DHCP IPv6 option	IPv6 comments
Lease time	Option 51	Always infinite	—	—
Requested option list	Option 55	—	—	—
Client ID	Option 61	Default is chassis serial ID	Option 1 (DUID)	Type 2 — vendor-assigned unique ID (default with chassis serial ID) Type 3 — link-layer address
User class	Option 77	"platform;timos-release;ztp"	Option 15	"platform;timos-release;ztp"
Class ID	Option 60	"NOKIA: Fmt ChassisType Strings"	—	—

8.3.3 Supported DHCP server options for ZTP

The following table lists the supported DHCP server options for ZTP.

Table 33: Supported DHCP server options for ZTP

Options	DHCP IPv4 option	IPv4 comments	DHCP IPv6 option	IPv6 comments
Subnet mask	Option 1	—	—	—
Router	Option 3	Default gateway	—	—
DNS server	Option 6	DNS server	Option 23	DNS server
Syslog server	Option 7	Server IP address	—	—
Lease time	Option 51	Must be infinite	—	—
Server address	Option 54	Identifies the DHCP server	—	—
Classless static route	Option 121	Used to install static routes	—	—
NTP server ³	Option 42	—	Option 56	—
TFTP server name	Option 66	Server IP address	—	—
Bootfile name	Option 67	URL of the file This option can be used without Option 66, in which case it contains the server name and the URL	Option 59	Server name and URL of the file
Vendor-specific options (See Nokia-specific TLV)	Option 43	Nokia proprietary file location; can be used instead of Options 66 or 67, but Options 66 and 67 take precedence over Option 43	Option 17	Nokia proprietary file location; can be used instead of Option 59, but Option 59 takes precedence over Option 17

³ When the node is running in ZTP mode, the date and time are set by NTP. This information is required for HTTPs certificate verification, and to record date and time stamps in events and logs.

8.3.4 DHCP discovery and solicitation

IPv4 DHCP discovery and IPv6 DHCP solicitation are supported.

IPv4 DHCP discovery messages and IPv6 DHCP solicitation messages are sent from out-of-band and in-band management ports with active links. The first valid DHCP offer for the address family that arrives on the node is used.

In the BOF, the **auto-boot** option can be configured to send out IPv4, IPv6, or both IPv4 and IPv6 DHCP requests.

8.3.4.1 DHCP discovery (IPv4 and IPv6)

This section describes DHCP discovery options.

8.3.4.1.1 DHCP discovery Options 61 and 77

SR OS supports both Option 61 (client ID) and Option 77 (user class) DHCP discovery options.

Option 61 is used for DHCP server pool selection. Option 61 provides the client ID; the serial ID of the chassis with a type of 0 is used by default. This option is configurable using commands in the **bof auto-boot** context.

Option 77 provides the user class, describing what the device is and other information, such as the OS version. This option is set automatically, but can be removed using the BOF configuration. For example, the user can delete **include-user-class** in the BOF auto-boot configuration to avoid sending Option 77.

For ZTP, the DHCP discovery message should be sent with Option 77; the following information is automatically configured:

```
platform;timos-release;ztp
```

For auto-provisioning, Option 77 should use the following information:

```
platform;timos-release;AP
```

8.3.4.1.2 DHCP discovery Option 1 DUID (IPv6)

By default, the node uses RFC 3315 DUID Type 2 vendor-assigned unique IDs. The value for *enterprise-id* is 6527 and the identifier is the chassis serial number.



Note: The system uses the chassis serial number for ZTP pool selection and auto-provisioning.

The option to use Type 3 is configured in the BOF. For MAC, the chassis MAC address is configured in a string format.

Type 1 is not supported.

8.3.4.2 DHCP solicitation (IPv6)

Unlike IPv4 DHCP offers, which contain the prefix and default route, IPv6 DHCP offers only contain the IP address assignment. The IPv6 route advertisement (RA) provides the default router and the prefix is set to /128 for the IP address supplied by the DHCP server.

For further information about RA support, see [IPv6 DHCP/RA details](#). For further information about DHCP server offers, see [DHCP server offer options](#).

8.3.5 IPv4 and IPv6 DHCP support

The ZTP process supports the use of IPv4 and IPv6 DHCP clients to obtain the provisioning file.

For ZTP processes, the node transmits both IPv4 and IPv6 discovery and solicitation messages. If offers arrive from both IPv4 and IPv6 servers, both offers are cached and the first offer received is processed. If the first offer does not fulfill the ZTP requirements and is rejected, the second offer is processed and accepted or rejected. If both offers received on an interface are rejected, ZTP goes to the next interface.

The provisioning file only allows file transfer in the address family of the DHCP offer that is used. If the offer is IPv4, the provisioning files are downloaded using IPv4. If the offer is IPv6, the provisioning files are downloaded using IPv6.

8.3.5.1 IPv4 route installation details

Option 3 (default route) and Option 121 (classless static route) are supported for IPv4 DHCP.

For identical routes with different next hops, only the first route is installed and the second route is kept as a backup route. ECMP is not supported.

There is no route limit for Option 121.

8.3.5.2 IPv6 DHCP/RA details

IPv6 DHCP offers do not contain an IP prefix. The IPv6 prefix is usually obtained from the IPv6 RA arriving from the upstream router. Because the SR OS router is the host for the ZTP process, the system assigns a /128 prefix to the IPv6 address obtained from the DHCP offer.

SR OS supports the use of an IPv6 RA to install IPv6 default and static routes from upstream routers. The system installs all the routes advertised using the RA. If the same route has been advertised from multiple upstream routers (next hops), the system installs the route with the highest preference. SR OS does not support ECMP if the same route is advertised from multiple next hops by multiple RAs.

In accordance with RFC 4861 recommendations, SR OS ensures that all RAs are obtained before the auto-provisioning process is started for IPv6. RFC 4861 recommends that the host (in this case, the SR OS router) send a minimum of three route solicitations to increase the likelihood of at least one route solicitation being received by the upstream routers. Each route solicitation is followed by a 4-second timeout, so the third route solicitation is sent 8 seconds after the first. The upstream routers must respond within 0.5 seconds. As a result, the SR OS router receives all RAs and routes within 8.5 seconds of the first route solicitation, and waits a maximum of 9 seconds to receive all RAs; ZTP always waits 20 seconds to receive all RAs, however, only the first RA received is used.

8.3.5.3 ZTP and DHCP timeouts

The ZTP timeout is user-configurable with a default value of 30 minutes. See [Options and option modification](#) and [Configuring the ZTP timeout in the provisioning file](#) for more information. After each ZTP timeout, the node reboots and reattempts the ZTP process. If the ZTP timeout interval expires while the node is executing a DHCP offer or downloading files, the node does not reboot. The DHCP offer is executed until it succeeds or fails, at which point the node reboots. If the offer is successful, the node comes up in normal operation mode.

The DHCP timeout interval is 20 seconds. If a DHCP offer is not received within the DHCP timeout interval, the auto-provisioning process is reattempted using the next valid interface.

8.4 ZTP procedure details

This section describes ZTP procedures including node bootup, BOF, auto-provisioning, logs, and events.

8.4.1 Node bootup

After the node is powered up, the BOF is examined for the **auto-boot** flag status. If the **auto-boot** flag is set in the `bof.cfg` file, the node goes into ZTP mode. If the **auto-boot** flag is not set in the `bof.cfg` file, the node continues booting normally.

If it is in ZTP mode, the node provisions all hardware necessary for the ZTP process. This includes the fabric, the first two card slots, and the MDAs for the first two card slots. The node then checks for links on the management (Mgmt) port and valid Ethernet ports.



Note: A `bof.cfg` file with the **auto-boot** flag enabled can be shipped as an orderable part with the applicable software license. The **auto-boot** flag can also be set using the **bof auto-boot** command.

For more information about the BOF, see [BOF](#).

8.4.1.1 Reinitiating ZTP during normal node bootup

ZTP can be reinitiated any time by setting the **auto-boot** flag and configuring the flag options in the BOF. After the auto-boot flag is set, any reboot forces the node into ZTP mode, including DHCP discovery, and downloading and reprocessing the provisioning file. The old BOF is kept in the storage medium until the ZTP process is successful, then the old BOF information is overwritten. If an unsuccessful ZTP process is interrupted and the **auto-boot** flag is removed, the node boots using the old BOF.

8.4.2 BOF

Two versions of each supported 7750 SR platform software license are currently available: one for non-ZTP bootup, and one for ZTP bootup. Software packages for ZTP bootup contain a `bof.cfg` file with the **auto-boot** flag set, which causes the node to automatically boot up in ZTP mode and execute ZTP processes.

The **auto-boot** flag contains the following information:

- **client ID**

The client ID is sent to the DHCP server to identify the chassis or node and to find a pool for the DHCP offer. If no client ID is configured, the chassis serial number is sent.

This option is used for both IPv4 client ID and IPv6 DUID Type 2.

- **port (port:vlan)**

The port is used to send DHCP discovery; the port number must be configured manually in the BOF.

For more information about the BOF, see [System initialization and boot options](#).

8.4.2.1 SD card and compact flash support

Nokia recommends that the provisioning file should download all files to the SD card, and the BOF should point to the SD card for imaging and configuration.

The BOF itself does not support loading from the network using HTTP or HTTPS.

8.4.3 Auto-boot process details

This section describes the ZTP auto-boot process.

8.4.3.1 Options and option modification

By default, the auto-boot process scans all ZTP-enabled ports to find a port with an operational link. The scanned ports include:

- out-of-band management port (Mgmt port)
- Ethernet ports on the first two card or MDA slots (used for in-band management)



Note: For breakout connectors, only the first breakout port in the connector can be used for ZTP.

ZTP attempts to discover the node IP via DHCP and identifies the node using DHCP client ID Option 61 (IPv4) or Option 1 (IPv6). The client ID uses the chassis serial number by default. The chassis serial number is visible on the shipping box of the chassis.

[Table 32: Supported DHCP client options for ZTP](#) lists the default DHCP client options for ZTP. Some client options can be manually configured in the BOF using the **bof auto-boot** command.

The optional **auto-boot** configuration options are as follows:

- **management port**

Specify that ZTP should only be performed using the out-of-band management port (Mgmt port).

- **in-band VLAN**

Specify ZTP should only be performed using Ethernet ports on the first two card or MDA slots. The VLAN ID can be used to specify an in-band VLAN to use for the auto-boot process.

- **IPv4, IPv6**

Specify that IPv4 discovery, IPv6 discovery, or both, should be performed. If both are specified, the system dual-stacks.

- **client identifier**

Identify the node to the DHCP server and find a pool for DHCP offers. This information is sent using Option 61 (IPv4) or Option 1 (IPv6). If the **client-identifier** options are not configured, the chassis serial number is sent by default. This option is used for both IPv4 client ID and IPv6 DUID Type 2.

- **include user class**

Specify to include Option 77.

- **timeout**

Specify in minutes the timeout for the ZTP process to be executed successfully before the node is rebooted and ZTP is retried because of an unsuccessful ZTP completion. The default ZTP timeout is 30 minutes.

See [Configuring the ZTP timeout in the provisioning file](#) for information about how to configure the ZTP timeout in a ZTP provisioning file.

The **auto-boot** options can be modified using the **bof auto-boot** command, or by interrupting the bootup process and manually modifying the `bof.cfg` file.



Caution: Manually modifying the `bof.cfg` file is not recommended. When modifying **auto-boot** options using CLI, all required options must be explicitly configured because the default cases are no longer used. When modifying the `bof.cfg` file manually, the format must be correct.

8.4.3.2 CLI access

The auto-boot process is executed in the background and does not block the CLI. The user can enter CLI commands while the auto-boot process runs in the background. A warning message is displayed to notify the user that the auto-boot process is being executed. Any configurations performed using the CLI may be lost when the node reboots following successful auto-boot and auto-provisioning processes. After the node has finished booting and if the **auto-boot** flag is set in the BOF, the node displays the login prompt.

The user can access the CLI using a console and can change and save the BOF configuration; as such, the user can remove or modify the **auto-boot** option in the BOF.

8.4.3.3 Interrupting auto-boot

The auto-boot process can be interrupted using the **tools perform auto-boot terminate** command. After the auto-boot process is terminated, use the **bof auto-boot** command to modify the **auto-boot** flag.



Note: The **auto-boot** flag can also be modified without interrupting the auto-boot process.

8.4.4 Auto-provisioning process

In this process, the node detects operational ports, attempts to discover its IP address, and downloads the relevant files for provisioning.

1. The node sends a DHCP discovery request to the DHCP server using the out-of-band management port. If DHCP discovery is unsuccessful, the node reattempts it using the in-band management ports.
2. After DHCP discovery is successful, the DHCP server returns an IPv4 or IPv6 FTP or HTTP URL of a file server from which the node can retrieve provisioning information.

3. The node downloads the provisioning information and performs the auto-provisioning according to the specifications in the files.

4. After the node is successfully provisioned, it automatically reboots and becomes operationally up.

See [Provisioning files](#) for more information about the auto-provisioning process.

The SR OS can also initiate the auto-provisioning process using a **tools** command.

8.4.4.1 VLAN discovery

The node can perform VLAN discovery if it is shipped in ZTP mode. VLAN discovery is supported only for the in-band management port. It is not supported for the out-of-band management ports.

After the node is installed and powered up:

1. ZTP is attempted on the null (untagged) port first, including the out-of-band management port, and then on all in-band management ports with operational links.
 - a. SR OS scans each port with an operational link and sends IPv4 DHCP discovery messages.
 - b. SR OS waits for the DHCP offer within the DHCP timeout.
2. The first VLAN with a valid offer that includes the IPv4 DHCP Options 66 and 67, or Option 67 or 43, or IPv6 DHCP Option 59 or 17 is selected as the working VLAN and the ZTP process is executed on this VLAN.



Note: If there is no offer or the offer does not have the relevant or correct options, SR OS floods the network with DHCP discovery messages on all remaining non-reserved VLANs (1 to 4094).

3. When a VLAN is discovered, the ZTP process is executed on the respective VLAN as described in the following sections.



Note: If there is no offer on any VLAN or the offer does not have the relevant or correct options, the node starts over from step 1.

8.4.4.1.1 VLAN discovery option

By default, the auto-boot flag in the `bof.cfg` file has the VLAN discovery option enabled. The option can be disabled manually in the `bof.cfg` file or implicitly from the CLI BOF menu, using the command **bof auto-boot inband**. When the VLAN discovery option is disabled, the node executes the ZTP process using the untagged method only.

8.4.4.2 Auto-provisioning procedure

After the node enters ZTP mode, the auto-discovery process is executed to provision the necessary hardware for node discovery.

The following are the operational steps of the auto-discovery process.

1. DHCP is used to discover the IP address of the node.
2. Options 66 and 67, or Option 43 is used to find and download the provisioning file.

The provisioning file includes the location of necessary files, such as configuration information, system image, and licenses, along with the DNS needed to resolve these location URLs. The file also includes BOF information required to boot the node into operational mode.

3. The provisioning file is executed to download the named files to the node.
4. After all files are successfully downloaded, the node is rebooted and the **auto-boot** flag is cleared from the BOF.

After the node reboots, it comes up in normal operational mode.

The node can be put back into ZTP mode by editing the BOF to include the **auto-boot** flag and saving the BOF. Doing this causes the node to enter ZTP mode after it is rebooted.

Use one of the following methods to run the auto-provisioning process.

- **automatic execution**

The auto-boot process automatically executes the auto-provisioning process if the **auto-boot** flag is set in the BOF.

- **manual execution**

The auto-provisioning process can be executed manually using the following command.

```
tools perform system auto-node-provisioning
```

If the auto-provisioning process is executed manually, only interfaces without IP addresses are considered part of the discovery mechanism. Additionally, while the process is running, it attempts to discover DHCP servers using all card or MDA slots and ports with Layer 3 interfaces that do not have IP addresses.



Note: Using the following command while the auto-boot process is running is not allowed.

```
tools perform system auto-node-provisioning
```

8.4.4.3 Out-of-band management versus in-band management

The auto-provisioning process can use the out-of-band management port (Mgmt port), or in-band management on Ethernet ports.

The node attempts the auto-provisioning process using any port with an operational link, starting with the out-of-band management port. If the node cannot be discovered using the out-of-band management port, either because the port is down or the port is not receiving a DHCP offer from the DHCP server, the process is reattempted using Ethernet ports. If the node cannot be discovered using the Ethernet ports, the process is reattempted using the out-of-band management port and the cycle repeats.

The following operational guidelines apply to in-band and out-of-band management ports:

- Out-of-band management and in-band management support untagged frames.
- Out-of-band management does not support dot1q (VLAN tags).
- In-band management supports dot1q interfaces if the VLAN is correctly configured in the BOF.
- In-band ports support VLAN discovery for IPv4 by default, if not disabled in the BOF.

If out-of-band management is used, no card or MDA provisioning is necessary and the auto-provisioning process executes as soon as an active link is detected on the Mgmt port.

To use out-of-band management exclusively, use the following command:

```
bof auto-boot management-port
```

To use in-band management exclusively, use the following command:

```
bof auto-boot inband vlan
```

8.4.4.3.1 Supported in-band management ports

See [ZTP overview](#) for information about which ports support in-band management for ZTP.

8.4.5 Provisioning files

Provisioning files are created by the user based on requirements and the locations of the necessary files. A provisioning file contains the locations and URLs of critical files such as the system image, configuration files, and necessary licenses, and can also contain DNS server information used to resolve these locations.

A provisioning file consists of two main parts:

- **locations of files**

Contains locations of the following file types:

- system image
- configuration files
- licenses

These items can be downloaded using HTTP, HTTPS, or FTP; DNS server information can also be included.



Note: If classic configuration mode is required when booting with ZTP, configuration files must have **exit all** as the first executable line.

- **BOF information**

Contains BOF information to be loaded on the node after the ZTP processes are completed; the BOF section of the file must be formatted correctly.



Caution: Ensure that the **auto-boot** flag is not set on the BOF that will be downloaded by the auto-provisioning process; failure to do so will cause the node to go back into ZTP mode after it reboots.

The provisioning file is stored on the SD card and can be executed using the following command to re-download the named files:

```
tools perform system auto-node-provisioning file
```

8.4.5.1 Provisioning file download

The provisioning file location is discovered using DHCP offer Options 66 and 67 or Option 43, and is downloaded using HTTP or FTP.

The provisioning file URL can be resolved using DNS, in which case the IP addresses for up to three DNS servers should be present in the DHCP offer using Option 6 (IPv4) or Option 23 (IPv6). The DHCP DNS is only used for resolving the provisioning file URL, and not for resolving the URLs of the files named within the provisioning file.

ZTP does not support Option 15 domain names; the URL of the provisioning file should be in "*host/domain*" format, or a simple IP address should be used.

8.4.5.2 Provisioning file resolution using DNS

If the downloaded provisioning file includes a DNS IP in the DNS section of the file, the URLs of the files in the provisioning file must be resolved using this DNS server or the DNS server listed in the DHCP offer.

Up to three DNS addresses (primary, secondary, tertiary) can be listed in the DNS sections of the provisioning file. If all three DNS addresses are listed, they are attempted in the listing order to resolve the file URLs.

8.4.5.3 File download and redundancy

Up to three locations can be set for each file type, using the `primary-url`, `secondary-url`, and `tertiary-url` fields. The auto-provisioning process attempts to download all files using the `primary-url` information for each file. If this attempt is unsuccessful, the process reattempts using the `secondary-url` information for each file. If this attempt is not successful, the process reattempts using the `tertiary-url` information.

A ZTP operation is considered successful when all files named in the provisioning file are downloaded. If all file locations are attempted and all named files are not successfully downloaded, the auto-provisioning process fails and ZTP reattempts the provisioning process using the next valid interface.

8.4.5.4 Configuring the ZTP timeout in the provisioning file

The ZTP timeout is the total time allowed for the ZTP process to execute successfully. If the ZTP process fails to run successfully within the ZTP timeout duration, the node is rebooted and the ZTP process is retried. The default timeout is 30 minutes.

See [Options and option modification](#) for information about how to configure the ZTP timeout using CLI commands.

ZTP timeout information can be included in the provisioning file to configure a value different from the default. For a provisioning file example, see [Example provisioning file](#).

Example

The following example shows how the timeout, in hours, is added to the provisioning file to change the default value.

```
set {
  timeout {
    hours 1
```

```
}
}
```

Example

The following example shows how the timeout, in minutes, is added to the provisioning file to change the default value.

```
set {
  timeout {
    minutes 90
  }
}
```

8.4.5.5 Downloading the image file

The image file can be downloaded in the following ways:

- Extract the images from the OLCS CFLASH file and explicitly list them in the provisioning file. Each of the .tim files is validated and the version of the software is checked. The hash-value must be explicitly listed.

```
download {
  image "${B00T-PATH}/both.tim" {
    primary-url "${FILESERVER}/both.tim"
    verification {
      hash-algorithm sha256
      hash-value "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
  }
  .....
}
```

- Zip the image files and perform a single image download. ZTP can extract the images and verify the .tim file and software version. An additional file can also be listed for all files to be verified using sha256 or md5.

```
download {
  image "${B00T-PATH}/images.zip" {
    primary-url "${FILESERVER}/images.zip"
    unzip
    verification {
      hash-algorithm sha256
      hash-value "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
  }
  # SHA-256 Checksum Files - File of a list of SHAR256 checksum values and file names (must
  be absolute)
  sha256 "${B00T-PATH}/somefile.txt" {
    primary-url "${FILESERVER}/sha256-checksums.txt"
    verification {
      hash-algorithm sha256
      hash-value "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
  }
}
```

8.4.5.6 Example provisioning file

This section provides examples of provisioning file information.

Example: Provisioning file information

```
set {
  timeout {
    hours 1
  }
}
dns {
  primary 192.0.2.1
  secondary 192.0.2.2
  tertiary 192.0.2.3
  domain sample.domain.com
}
download {
  image "cf3:/both.tim" {
    primary-url "http://192.168.40.140:81/both.tim"
    secondary-url "http://192.168.40.140:81/both.tim"
    tertiary-url "http://192.168.40.140:81/both.tim"
  }
  image "cf3:/support.tim" {
    primary-url "http://192.168.40.140:81/support.tim"
    secondary-url "http://192.168.40.140:81/support.tim"
    tertiary-url "http://192.168.40.140:81/support.tim"
  }
  config "cf3:/config.cfg" {
    primary-url "ftp://ftpserv:name@192.168.194.50/./images/dut-a.cfg"
    secondary-url "http://192.168.41.140:81/dut-a.cfg"
    tertiary-url "http://192.168.42.140:81/dut-a.cfg"
  }
  file "cf3:/license.txt" {
    primary-url "ftp://ftpserv:name@192.168.194.50/./images/provision_example.cfg"
    secondary-url "http://192.168.41.140:81/dut-a.cfg"
    tertiary-url "http://192.168.42.140:81/dut-a.cfg"
  }
}
bof {
  primary-image cf3:/both.tim
  primary-config cf3:/config.tim
  address 192.168.100.1 active
  autonegotiate
  duplex full
  speed 100
  wait 3
  persist off
  console-speed 115200
}
```

For an HTTPS URL, the trust anchor needs to be referenced in the provisioning file. The trust anchor name references the entry in the `import trust-anchor` section of the file. The provisioning file can also specify the TLS version to use for HTTPS. If the TLS version is not specified, "all" will be considered as the default.

The following example shows a provisioning file with three trust anchors, TRUST_ANCHOR_12, TRUST_ANCHOR_13, and TRUST_ANCHOR_ALL. Each anchor includes the `tls-version` which enforces the use of TLS 1.2, 1.3, or both, respectively. A different configuration download is bound to each anchor.

Example: Trust anchors for HTTPS URL in provisioning file

```

import {
  trust-anchor TRUST_ANCHOR_12 {
    cert "cf3:/ca12.crt" {
      format pem
      tls-version "1.2"
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca12.crt
    }
    crl "cf3:/ca12.crl" {
      format der
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca12.crl
    }
  }
  trust-anchor TRUST_ANCHOR_13 {
    cert "cf3:/ca13.crt" {
      format pem
      tls-version "1.3"
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca13.crt
    }
    crl "cf3:/ca13.crl" {
      format der
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca13.crl
    }
  }
  trust-anchor TRUST_ANCHOR_ALL {
    cert "cf3:/ca-all.crt" {
      format pem
      tls-version "all"
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca-all.crt
    }
    crl "cf3:/ca-all.crl" {
      format der
      primary-url ftp://user-name:password@10.10.10.66//fileserver-1/ca-all.crl
    }
  }
}
download {
  config "cf3:/ztp/ztp_dut-a.cfg.12" {
    primary-url https://10.10.10.64:81/ztp_dut-a.cfg
    primary-trust-anchor "TRUST_ANCHOR_12"
  }
  config "cf3:/ztp/ztp_dut-a.cfg.13" {
    primary-url https://10.10.10.64:81/ztp_dut-a.cfg
    primary-trust-anchor "TRUST_ANCHOR_13"
  }
  config "cf3:/ztp/ztp_dut-a.cfg.all" {
    primary-url https://10.10.10.64:81/ztp_dut-a.cfg
    primary-trust-anchor "TRUST_ANCHOR_ALL"
  }
}
}

```

8.4.5.7 Proxy support

HTTP and HTTPS can connect to public servers using a proxy. The proxy is in URL format and the URL must be resolved using the provisioning file DNS.

The proxy can include a username and password. Proxy Auto-Configuration (PAC) is not supported.

Proxy information formatting is as follows:

`http://user@hostname:file-path`

`https://user@hostname:file-path`

`proxy http://ip-or-url user@hostname:port`

The HTTP (or HTTPS) proxy support information is included in **file** commands and in the ZTP provisioning file. The following example shows HTTP proxy information in the provisioning file.

Example

```
image "cf3:/both.tim" {
    primary-url "http://200.150.40.140:81/both.tim"
    secondary-url "http://200.150.40.140:81/both.tim"
    tertiary-url "http://200.150.40.140:81/both.tim"
    primary-proxy http://132.2.3.1:8080
    secondary-proxy http://133.3.4.1:8080
}
```

8.4.6 Day 0 configuration

To improve initial router functionality and reuse information discovered by ZTP in the configuration file, the user can include an optional day 0 configuration template within the provisioning file. This day 0 configuration template can obtain configuration details for specific modules that can be used to provision the node and establish node connectivity. Predefined symbols are used to add parameters discovered by ZTP, such as the port and VLAN, to the configuration template.

Any supported configuration can be included in the day 0 configuration template, including configuration of port types and services. However, Nokia recommends that day 0 configuration details should be kept to the minimum necessary to ensure that basic network connectivity is available for the ZTP process to run and complete.



Caution: Before using the configuration, verify the configuration details in the day 0 template and check for syntax, context, and other errors. The ZTP process does not verify the configuration details before implementing them.

When the provisioning file is received through the ZTP process, the day 0 configuration template section of the provisioning file is converted to a configuration file and stored on the SD card of the node, as specified by the **Write** `cf3:/configuration-file-name.cfg` entry within the template. The BOF section of the provisioning file must specify the configuration file generated from the day 0 configuration template using the `primary-config cf3:/configuration-file-name.cfg` entry.

Some unknown elements within the day 0 configuration template can be entered as symbols, which are then converted to discovered information as the node is provisioned; see [Day 0 symbols](#).

See [Sample day 0 configuration template](#) for a configuration example, including the day 0 configuration template and associated BOF section of the provisioning file.

8.4.6.1 Day 0 configuration for multi-slot routers

On multi-slot routers, the discovered IOMs, MDAs, IMMs, and fabric (that is, the discovered hardware) are part of the day 0 configuration. ZTP automatically adds the CLI configuration to the correct location in the day 0 configuration in the provisioning file.

This process ensures that the nodes for different sites are populated with the correct hardware, regardless of what physical hardware is used for a specific site.

The day 0 configuration process for multi-slot routers is as follows:

1. The provider creates a day 0 configuration template with the required services and interfaces.
2. ZTP provisions the IOMs, MDAs, IMMs, and fabric.
3. ZTP discovers the port and VLAN on which the DHCP server is connected.
4. ZTP adds the discovered and provisioned hardware to the day 0 configuration in the provisioning file
5. ZTP uses the day 0 configuration in provisioning file to create an initial configuration file and saves it to the specified location on the SD card of the node.

The BOF section of the provisioning file must specify the configuration file generated from the day 0 configuration.

6. After rebooting, the node boots up using the day 0 configuration file.

8.4.6.2 Day 0 symbols

Use symbols to generalize and reuse a single provisioning file across multiple ZTP deployments and for value substitutions and conditions in the day 0 configuration template of the provisioning files. The same symbol can appear as many times as required and can be used for both conditional statements and substitutions.

8.4.6.2.1 Symbol use for substitution

A day 0 configuration template can include symbols that are replaced with discovered information when the day 0 configuration is implemented. Symbols for substitution must be entered using the format `$(<symbol-name>)`. The following example shows a supported use case of a symbol for substitution.

Example

```
card $(ztp.bootstrap.slot)
  card-type $(ztp.bootstrap.card-type)
    mda $(ztp.bootstrap.mda)
      mda-type $(ztp.bootstrap.mda-type)
  exit
exit
```

8.4.6.2.2 Symbol use for conditions

A day 0 configuration template can include symbols that are replaced with discovered information when the day 0 configuration is implemented. Symbols for conditions must be entered using the format `@(<symbol-name>)` at the beginning of a line. If the condition is not met (that is, the symbol does not exist), the rest of the line is skipped and not written into the day 0 configuration file on the compact flash. The following example shows a supported use case of a symbol for conditions.

Example

```
@(ztp.uplink.connector)    port (ztp.uplink.connector)
@(ztp.uplink.connector)    connector
```

```
@(ztp.uplink.connector) breakout ${ztp.uplink.breakout}
@(ztp.uplink.connector) exit
@(ztp.uplink.connector) exit
```

8.4.6.2.3 Supported symbols

The following table describes the supported symbols for a day 0 configuration template.

Table 34: Supported symbols

Symbol name	Description	Example value	Conditional	Discovered
sys.platform	Platform name	"7750"	No. Always available.	Configured for the system and is exposed at system boot-up
sys.type	Chassis type	"SR-7"	No. Always available.	Configured for the system and is exposed at system bootup
sys.serial-number	Serial number of the chassis	"NS1234567"	No. Always available.	Configured for the system and is exposed at system bootup
sys.mac-address	Base chassis MAC of chassis	"01:01:01:01:01:01"	No. Always available.	Configured for the system and is exposed at system bootup
sys.sw-version	Software version	"TIMOS-B-23.10.R2"	No. Always available.	Configured for the system and is exposed at system boot-up
ztp.bootstrap.uplink	Full uplink used for bootstrap (including VLAN). Intended to be used as interface port binding in the day 0 configuration.	"1/1/1" "1/1/1:1000" "A/1"	No. Always available.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.uplink.card-type	Card-type of uplink slot used for bootstrap	"xcm-1s"	Yes. Available when uplink is inband.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.uplink.slot	Slot of uplink used for bootstrap	"1"	Yes. Available when uplink is inband.	Discovered at VLAN discover or ZTP discovery time

Symbol name	Description	Example value	Conditional	Discovered
ztp.uplink.xiom	Xiom identifier of uplink used for bootstrap	"x1"	Yes. Available when uplink is inband and has an xiom.	Discovered at VLAN discover or ZTP discovery time
ztp.uplink.xiom-type	Xiom-type of xiom used for bootstrap	lom2-se-6.0t"	Yes. Available when uplink is inband and has an xiom.	Discovered at VLAN discover or ZTP discovery time
ztp.uplink.mda	The MDA number of the MDA used for bootstrap (This will either be the MDA under card or MDA under xiom if uplink has xiom)	"1"	Yes. Available when uplink is inband.	Discovered at VLAN discover or ZTP discovery time
ztp.uplink.mda-type	The MDA number of the MDA used for bootstrap (This will either be the MDA under card or MDA under xiom if uplink has xiom)	"m4-sfp"	Yes. Available when uplink is inband.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.uplink.port	Uplink port without encapsulation value (VLAN)	1/2/c1/1	No. Always available.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.uplink.vlan	Uplink VLAN (encapsulation value)	"1000"	Yes. Available when uplink has a vlan.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.connector	Uplink connector ID	1/1/c1	Yes. Available when uplink is a connector-port.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.breakout	Connector breakout type	c1-10g	Yes. Available when uplink is a connector port.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.rs-fec	Connector RS-FEC mode setting		Yes. If connector has a non-default rs-fec mode setting.	Discovered at VLAN discover or ZTP discovery time

Symbol name	Description	Example value	Conditional	Discovered
ztp.bootstrap.uplink.rs-fec	Uplink port RS-FEC mode setting		Yes. If uplink port has a non-default rs-fec mode setting.	Discovered at VLAN discover or ZTP discovery time
ztp.bootstrap.uplink.speed	Uplink port speed setting		Yes. If uplink port has a non-default speed setting	Discovered at VLAN discover or ZTP discovery time
Ztp.bootstrap.ip	IP address used during bootstrap (IPv4 or IPv6)	1.2.3.4 1::344:2	No. Always available.	Discovered via DHCP offer
Ztp.bootstrap.prefixlen	The prefix length of the IP address used under interface during bootstrap		No. Always available.	Discovered via DHCP offer

8.4.6.3 Sample day 0 configuration template

The following is a sample of a day 0 configuration template included in a provisioning file.

Example

```
# Generate Day-0 Configuration
# -----
Write "cf3:/config-template.cfg" {
#####
# !!DAY0 CONFIG START!! #
#####
exit all
configure
#-----
echo "System Configuration"
#-----
    system
        name "chassis-name"
        snmp
        exit
        login-control
            idle-timeout disable
        exit
        time
            ntp
                server 192.168.194.202
                no shutdown
            exit
        exit
        thresholds
            rmon
            exit
        exit
```

```

    exit
#-----
echo "System Security Configuration"
#-----
    system
        security
            telnet-server
            ftp-server
            snmp
                community "private" rwa version both
                community "public" r version both
            exit
        exit
    exit
#-----
echo "Log Configuration"
#-----
    log
        snmp-trap-group 90
            trap-target ""
        exit
        log-id 90
            from main change
            to snmp
        exit
    exit
exit all
configure
    router
#-----
echo "IP Configuration"
#-----
        interface "IF-1/1/c1/1"
            port ${ztp.bootstrap.uplink}
            address 192.168.0.1/31
            ipv6
            exit
            no shutdown
        exit
    exit
exit all
#####
# !!DAY0 CONFIG END!!  #
#####
}
# Generate New BOF File
# -----
bof {
    primary-image  cf3:/image.both
    primary-config cf3:/config-template.cfg
    autonegotiate
    duplex full
    speed 100
    wait 3
    persist off
    console-speed 115200
}

```

8.4.7 Logs and events

ZTP displays detailed events about all stages of the auto-boot and auto-provisioning processes. All events are saved in a log file on the node SD card at the end of the ZTP process.

ZTP also supports a direct connection to a syslog server, where the syslog server IP address is communicated using IPv4 DHCP option 7. SR OS only supports the syslog server DHCP option for IPv4 DHCP as described in section 3.9 of RFC 2132, *DHCP Options and BOOTP Vendor Extensions*.

8.4.7.1 Syslog

ZTP supports transmitting event logging information to an IPv4 syslog server, as follows:

- DHCP Option 7**
The user can include Option 7 in the DHCP server offer to specify the IPv4 address of the destination syslog server. RFC 2132 describes the syslog server IPv4 address encoding as follows:

The log server option specifies a list of MIT-LCS UDP log servers available to the client. Servers SHOULD be listed in order of preference. The code for the log server option is 7. The minimum length for this option is 4 octets, and the length MUST always be a multiple of 4.

Code	Len	Address 1				Address 2			
7	n	a1	a2	a3	a4	a1	a2	...	

Logging information is transmitted over UDP in cleartext.

- secure-syslog**
After downloading the provisioning file, the user can specify the **syslog** option in the file to use TLS for syslog.

 **Note:** Option 7 must be omitted from the DHCP offer to use TLS for syslog.

8.5 Legacy SZTP (non RFC 8572-compliant)

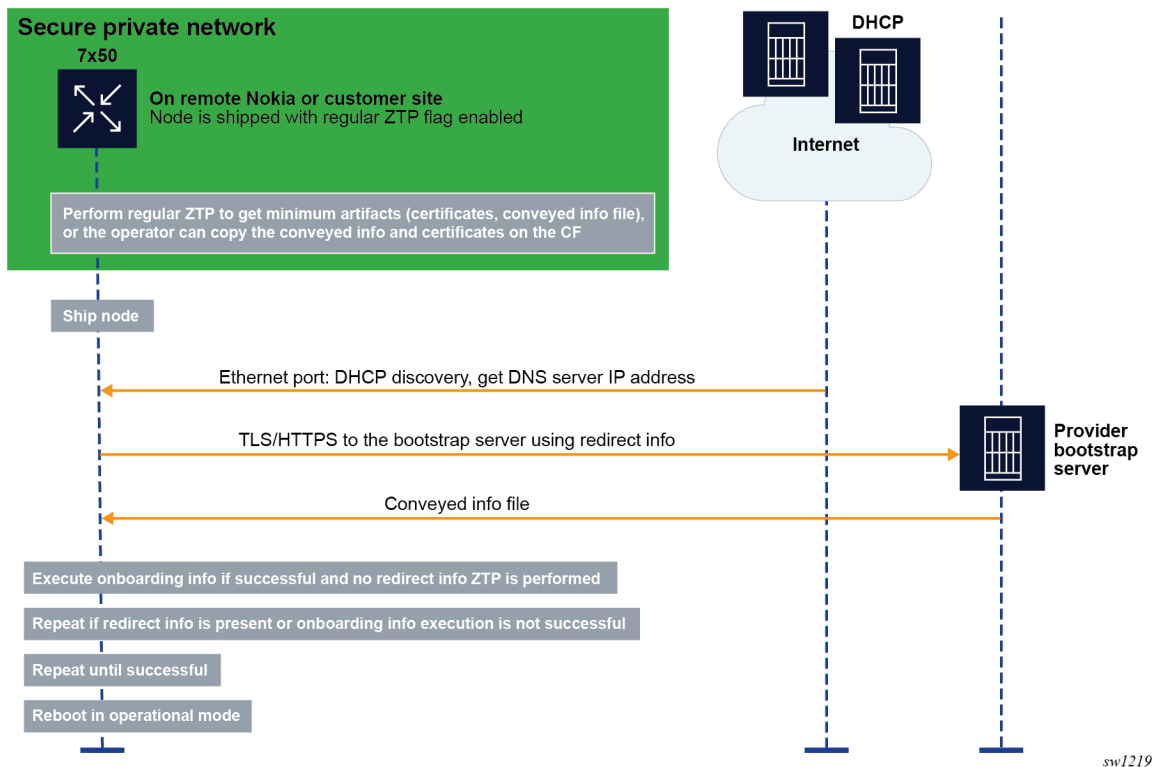
The SR OS implementation of SZTP is a partial application of RFC 8572 and is evolving to meet all RFC 8572 aspects. SZTP is an extension of ZTP as follows:

- The provisioning file and the node discovery of the MDAs, IOMs, and ports with links up are supported in both ZTP and SZTP.
- The ports that are ZTP-capable are also SZTP-capable.

SZTP securely bootstraps the node and provides it with the information required to boot up the node in an operational mode; this information includes all the initial artifacts required to create a mutual trust relationship between the node and the bootstrap server. After the node boots, it discovers the bootstrap server IP address, communicates with the server, and authenticates both the server and itself. Finally, the node securely downloads the encrypted boot image and initial configuration information.

SR OS uses different bootstrapping methods to obtain the required TLS certificates, trust anchors, and redirect information to connect securely to the server and download all the necessary information to boot in an operational mode.

Figure 36: SZTP process



In the example shown in the preceding figure, one of the following methods can be used to bootstrap the node securely:

- The user stages the node at their own site and bootstraps it using ZTP through a secure or private network. The node obtains the TLS certificates, trust anchors, and keys from a conveyed information file, which must be copied into the compact flash (CF). The Uniform Resource Identifier (URI) of this file is included in the ZTP provisioning file.
- If the node has CF, the user copies manually to the CF the certificates, trust anchors, and conveyed information file. Optionally, the user can also include redirect information in the conveyed information file.

After the node is bootstrapped securely, it is shipped to the installation site, where it boots.

If the node has redirect information, it tries to connect the bootstrap server specified in the redirect information and establish a TLS session to create mutual trust between the node and the server.

If the node does not have redirect information, it performs a DHCP discovery and tries to obtain the redirect information using DHCP option 143 (IPv4) or 136 (IPv6). After obtaining the redirect information, the node tries to connect to the bootstrap server using TLS.

The node uses option 67 from the DHCP server or the URI from the file field of the redirect information to locate the conveyed information from the bootstrap server. The conveyed information provides the node with one of the following:

- more redirect information for a new file server and other required resources to connect to the file server to download all the required information and files
- onboarding information containing the URI of the boot image and the initial configuration
- both redirect information and onboarding information, in which case the node executes the onboarding information first and then executes the redirect information

8.5.1 Staging the secure environment

The following artifacts are required:

- node client certificates and keys
- bootstrap server certificates for the first redirect
- security artifacts file, which contains the trust anchor definitions and import instructions. Optionally, this file can also contain conveyed information, which consists of redirect, if applicable, and onboarding information.

The staging options are the following:

- Copy the artifacts to a CF that can be installed in cf1:, cf2:, or cf3: when the node is at the installation site.
- Alternatively, use ZTP to pre-stage the node and download the root security artifacts using a trusted DHCP server and provisioning file.



Note: Nokia recommends that the TLS client should have a certificate so that it is authenticated by the ZTP server. Although the client certification can be omitted for TLS configuration, this is not recommended.

8.5.2 Bootstrapping methods

The following bootstrapping methods are supported:

- Use DHCP option 143 (IPv4) or 136 (IPv6), as described in RFC 8572. Optionally, the user can obtain the specific node URI (server and directory) by providing the DHCP server option 61 for the DHCP server, which in turn provides option 67 for the file directory, or option 143 (IPv4) or 136 (IPv6) with the server IP and file directory information. In this case, the TLS certificates, trust anchors, and keys must be installed on the node at the user premises.
- Copy the following information to the CF:
 - redirect information for the bootstrap server
 - TLS certificates and trust anchors, and private keys
 - onboarding information
- Use ZTP to provide the following information to the node:
 - redirect information for the bootstrap server
 - TLS certificates and trust anchors, and private keys
 - onboarding information

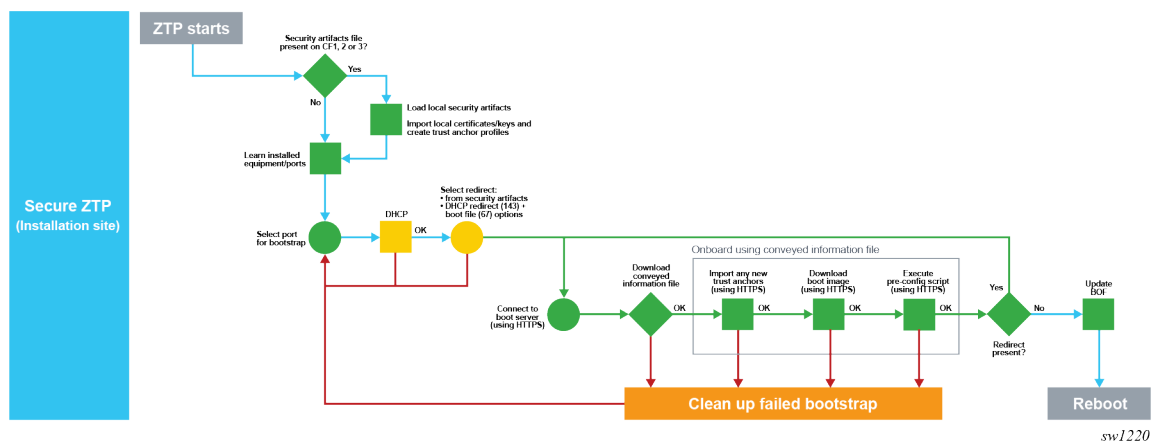
- Redirect the node from the first bootstrap server to consecutive bootstrap servers. The bootstrap server can provide the node with additional redirect information in a secure encrypted manner. The redirect and onboarding information are provided in the conveyed information file.
 - redirect information to another bootstrap server
 - required TLS certificates and trust anchors, and private keys
 - onboarding information

8.5.3 Installation site process

At the installation site, the **auto-boot** flag in the BOF signals the ZTP process. The presence of a conveyed information file on the node signals to the node that it is a secure ZTP procedure.

The following figure shows the SZTP process at the installation site.

Figure 37: Installation site SZTP process



After staging, the port that has the link up is selected and SZTP is executed on it.

The node loads the security artifacts to install the TLS certificates and trust anchors. DHCP discovery messages are sent out on each port in sequence. If no DHCP offer is received, SZTP moves to the next port with the link up. The OOB port is examined first, followed by the untagged in-band ports. If no DHCP offer is received, VLAN discovery is performed on the in-band ports only by flooding VLAN 0 to 4196 with DHCP discovery.

After DHCP discovery completes, the node obtains an IP address and can optionally obtain option 143 (IPv4) or option 136 (IPv6) for redirect information. If the redirect information is present in the conveyed information file, it is preferred over the DHCP redirect information.

The node is connected to the bootstrap server as indicated by the redirect information and a TLS mutual authentication is established using the certificates. The bootstrap server must have the correct certificates, keys, and trust anchors to create the mutual TLS trust.

After the node authenticates the server and authenticates itself to the server, it downloads the conveyed information file from the server using HTTPS. The node obtains the server location of the conveyed information from DHCP option 67 or the file field in the redirect information.

If the conveyed information file contains redirect information, the node tries to connect to the new bootstrap server indicated in the new redirect information. The node can download the new certificates indicated in the conveyed information.

If the conveyed information file contains only onboarding information, the node downloads the onboarding file.

If the conveyed information file contains both onboarding and redirect information to the next bootstrap server, the node executes the onboarding information first, then the redirect information.

The process is successful if the node executes the onboarding information without errors.

8.5.3.1 Initial conveyed information file

The conveyed information file (also referred to as `conveyed-info.ztp` file) contains the certificates, keys, and trust anchors required to establish the TLS connection. This is the minimum information that the node requires to start SZTP after staging at the installation site. The initial file must be added to `cf3`: by copying it on the CF manually or using regular ZTP procedures and the provisioning file.

Example: Contents of a conveyed information file

```
import {
  client {
    cert "cf3:/artifacts/node.cert"
    key "cf3:/artifacts/node.key" {
      format der
    }
  }
  trust-anchor BOOTSERVER {
    cert "cf3:/artifacts/bootserver.cert"
  }
}
```

The certificates, keys, and trust anchor information can be encrypted using the **encrypt** command, as shown in the following example. When the **encrypt** keyword is present, the information is downloaded from the URI and encrypted using AES256.

Example: Using the encrypt command

```
import {
  client {
    encrypt
    cert "cf3:/artifacts/node.cert"
    key "cf3:/artifacts/node.key" {
      format der
    }
  }
  trust-anchor BOOTSERVER {
    encrypt
    cert "cf3:/artifacts/bootserver.cert"
  }
}
```

Optionally, the file can contain the redirect information as shown in the following example. It is not mandatory to include the redirect information in the file because the preliminary redirect information can be obtained using DHCP.



Note: The redirect information in the file is preferred over the DHCP redirect information because it is trusted.

Example: Redirect information in the file

```
import {
  client {
    encrypt
    cert "cf3:/artifacts/node.cert"
    key "cf3:/artifacts/node.key" {
      format der
    }
  }
  trust-anchor BOOTSERVER {
    encrypt
    cert "cf3:/artifacts/bootserver.cert"
  }
}

redirect-information {
  boot-server "https://mybootserver.com/" {
    port 50
    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
  boot-server "https://backupserver.com/" {
    port 50
    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
}
```

The following example shows a file containing the entire conveyed information, including redirect and onboarding information. See [Onboarding information](#).

Example: File with entire conveyed information

```
import {
  client {
    encrypt
    cert "cf3:/artifacts/node.cert"
    key "cf3:/artifacts/node.key" {
      format der
    }
  }
  trust-anchor BOOTSERVER {
    encrypt
    cert "cf3:/artifacts/bootserver.cert"
  }
}

redirect-information {
  boot-server "https://mybootserver.com/" {
    port 50
    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
  boot-server "https://backupserver.com/" {
    port 50
    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
}
```

```

}
onboarding-information {
  boot-image
    download-uri https://images.com/$(sys.platform).zip
  pre-configuration-script "https://config.com/provisioning.cfg"
}

```

8.5.3.2 Onboarding information

The onboarding information is required to obtain all critical resources to boot the node in the normal mode of operation with the latest boot image and configuration. When processing the onboarding information, the device must first process the boot image information (if any), then execute the preconfiguration script (if any).

The onboarding information is present in the conveyed information file only. See [Conveyed information](#).

Example: Onboarding information

```

onboarding-information {
  boot-image
    download-uri https://images.com/$(sys.platform).zip
  pre-configuration-script "https://config.com/provisioning.cfg"
}

```

The download-uri is the URI to the boot image in ZIP format only. A URI list can exist, each pointing to the primary, secondary, or tertiary images . zip file. SR OS has multiple images, for example, CPM image and IOM image, and these images can be downloaded in a ZIP file. The URI can point to the ZIP file bundle to download all the images from a primary source. If the image is downloaded using download-uri, the destination of the image is always cf3:.

Example: Using download-uri information

```

onboarding-information {
  boot-image {
    # Download-URI(Mandatory): URL to ZIP file of images. Up to 3 for primary,
    secondary, tertiary
    # -> Base directory is "cf3:/"
    download-uri "https://server.download.com/images.zip"
    download-uri "https://server2.download.com/images.zip"
    download-uri "https://server3.download.com/images.zip"
    # VERIFICATION (Optional): File within ZIP file to verify images
    image-verification {
      hash-algorithm sha256
      hash-value "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
  }

  # Another provisioning file that can be loaded. Downloaded as "cf3:/autoboot.cfg.1,2,3"
  based on chaining
  pre-configuration-script "https://server.file.com/someotherprovisioningfile.txt"
}

```

Optionally, the node can run a checksum on the downloaded ZIP file using a hash to ensure there are no download errors. The hash algorithm and the hash value are noted in the onboarding information, as shown in the previous onboarding information example. The supported hash algorithms are SHA256 and MD5.

The preconfiguration script can be used to download the provisioning file. The provisioning file and the ZTP provisioning file have the same format. The provisioning file has to be executed to completion for the ZTP process to be successful. The provisioning file can also contain the location of the image. The image can be downloaded using the download URI or the provisioning file. When downloading the image using the provisioning file, the destination of the image can be dictated. The BOF must be configured accordingly to boot from the image destination. The image destination must always be cf3: or a folder in cf3:.



Note: The preconfiguration script is always required to clear the **auto-boot** flag from the BOF. A minimal BOF configuration is required in the provisioning file.

8.5.3.2.1 Preconfiguration script

The preconfiguration script is the actual ZTP provisioning file. SZTP supports all features of the ZTP provisioning file.

The preconfiguration script and provisioning file must be executed by the onboarding information to update the BOF configuration and remove the **auto-boot** flag. This ensures that the node comes back up in a normal mode of operation. For examples of the preconfiguration script and provisioning file information included in the onboarding information, see [Onboarding information](#).

8.5.3.2.2 Additional capabilities in the ZTP provisioning file

The following optional capabilities of the provisioning file are supported for both ZTP and SZTP:

- **checksum for file download**

SHA256 and MD5 are supported. The hash algorithm and hash value can be specifically configured for the file as shown in the following example. The file checksum is checked against the hash algorithm and hash value.

```
# List of Configuration Files to download
config "${(BOOT-PATH)}/somefile.txt" {
    primary-url "${(FILESERVER)}/somefile.txt"
    verification {
        hash-algorithm sha256
        hash-value
        "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
}
```

- **file encryption using AES256**

When a file is downloaded and if the file has the **encrypt** keyword enabled in the provisioning file, the file is encrypted using AES256 and placed on the CF. This is useful when downloading certificates, keys, and trust anchors for TLS. The following example shows an excerpt from the provisioning file. In this example, in the certificate import the keyword **encrypt** is used to encrypt each file on the CF after the file was downloaded. In addition, the checksum is calculated on each file and checked to ensure the files are downloaded without errors.

```
import {
    client {
        cert "${(CERT-PATH)}/device.crt" {
            primary-url "${(FILESERVER)}/device.crt"
            encrypt
            verification {
                hash-algorithm sha256
            }
        }
    }
}
```

```

        hash-value
        "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
    key "${CERT-PATH}/device.key" {
      format pem
      primary-url "${FILESERVER}/device.key"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
        "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
  }
  trust-anchor "NokiaSvcs" {
    cert "${CERT-PATH}/owner-ca.crt" {
      format pem
      primary-url "${FILESERVER}/owner-ca.crt"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
        "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
    crt "${CERT-PATH}/owner-ca.crl"
    format der
    primary-url "${FILESERVER}/owner-ca.crl"
    encrypt
    verification {
      hash-algorithm sha256
      hash-value
      "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
    }
  }
}

```

- **downloading the image files in ZIP file format**

The files can be downloaded using the provisioning file with an optional checksum. The ZIP file is automatically unzipped and each individual file is placed on the CF. The checksum is checked across the entire ZIP file before it is unzipped.



Note: The specified directory path must end with a forward slash (/).

```

image "${BOOT-PATH}/images.zip" {
  primary-url "${FILESERVER}/images.zip"
  unzip {
    directory "cf3:/ztp/"
  }
  verification {
    hash-algorithm sha256
    hash-value
    "53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
  }
}

```

- **downloading a SHA256 or MD5 checksum file**

The user can download a SHA256 or MD5 checksum file from Nokia servers for the ZIP image files. After the image file is unzipped, each *.tim file has its checksum checked against these checksum files.

The following are examples of checksum files.

```
17717f13ecf19179e367d990277e943993d53d771b44d19fddf5d349b1e7e7c4 cf3:/ztp/cpm.tim
616656b2224e65e29d71355ea41d9dc548c281e9f729c97fe46f3a5c643acb09 cf3:/ztp/iom.tim
dd9455158dc2dfbf937eb372bbef73ebab97f951efa742eec16a4ed4edd3ca9b cf3:/ztp/support.tim
```

Checksum or file decryption errors cause the failure of the ZTP or SZTP procedure, in which case the node generates an error and logs the event.

8.5.3.2.3 Certificate chaining

The following is an example of certificate chaining using a root and intermediate CA in the provisioning file.

Example: Certificate chaining

```
import {
  trust-anchor CMP_ROOT {
    cert "cf3:/root.crt" {
      format pem
      primary-url "http://ztp-server.com/pki/certificates/Classified_Plattform_Root_
CA-1.pem"
      secondary-url "http://ztp-server.com/pki/certificates/Classified_Plattform_
Root_CA-1.pem"
    }
  }
  trust-anchor CMP_ISSUING {
    cert "cf3:/issuing.crt" {
      format pem
      primary-url "http://ztp-server.com/pki/certificates/Classified_Plattform_
Issuing_CA_1.pem"
      secondary-url "http://ztp-server.com/pki/certificates/Classified_Plattform_
Issuing_CA_1.pem"
    }
    crl "cf3:/issuing.crl" {
      format der
      primary-url "http://ztp-server.com/pki/crls/Classified_Plattform_Issuing_CA_
1.crl"
      secondary-url "http://ztp-server.com/pki/crls/Classified_Plattform_Issuing_CA_
1.crl"
    }
  }
}

download {
  config "cf3:/config.cfg" {
    primary-url "https://ztp-server.com/config-ztp-ne21.cfg"
    secondary-url "https://ztp-server.com/config-ztp-ne21.cfg"
    primary-trust-anchor "CMP_ISSUING"
    secondary-trust-anchor "CMP_ISSUING"
  }
  config "cf3:/snmp.cfg" {
    primary-url "https://ztp-server.com/snmp.cfg"
    secondary-url "https://ztp-server.com/snmp.cfg"
    primary-trust-anchor "CMP_ISSUING"
    secondary-trust-anchor "CMP_ISSUING"
  }
}
```

```

}
bof {
  primary-image cf3:\TiMOS-SR-24.7.R1
  primary-config cf3:\config.cfg
  wait 3
  persist on
}

```

8.5.3.3 Conveyed information

After SR OS authenticates successfully to the bootstrap server, the node can download the conveyed information using HTTPS. The user can choose the name of the conveyed information file.

The SR OS conveyed information is trusted and does not require an additional signature verification.

Example: File with only onboarding information

The following conveyed information file example contains only onboarding information.

```

onboarding-information {
  boot-image
    download-uri https://images.com/$(sys.platform).zip
  pre-configuration-script "https://config.com/provisioning.cfg"
}

```

The conveyed information can also contain redirect information, in which case a recursive redirect can happen to another bootstrap server. If the conveyed information contains onboarding information and redirect information, the node executes the onboarding information first, then the redirect to the next bootstrap server.

Example: File with onboarding and redirect information

The following conveyed information file example contains onboarding and redirect information, and the certificates required for the second redirect.

```

onboarding-information {
  boot-image
    download-uri https://images.com/$(sys.platform).zip
  pre-configuration-script "https://config.com/provisioning.cfg"
}
import {
  client {
    cert "cf3:/artifacts/node.cert"
    key "cf3:/artifacts/node.key" {
      format der
    }
  }
  trust-anchor BOOTSERVER {
    cert "cf3:/artifacts/bootserver.cert"
  }
}

redirect-information {
  boot-server "https://mybootserver.com/" {
    port 50
    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
  boot-server "https://backupserver.com/" {
    port 50
  }
}

```

```

    trust-anchor BOOTSERVER
    file "conveyed.info"
  }
}

```

After the conveyed information is executed successfully, the BOF is loaded in the provisioning file to which the preconfiguration script is pointing and the **auto-boot** flag is cleared.

8.6 RFC 8572-compliant SZTP

SR OS supports the following ZTP modes, flags, and features in the `bof.cfg` file:

- normal ZTP – auto-boot without prestaged conveyed information
- legacy SZTP (non RFC 8572-compliant) – auto-boot with prestaged conveyed information
- RFC 8572 SZTP – auto-boot securely (does not support prestaging)

SR OS supports SZTP compliant with RFC 8572. To use SZTP in accordance with RFC 8572, order the correct SR OS software part number so that the package includes a `bof.cfg` file that contains the following flag:

- **MD-CLI**

```
bof auto-boot dhcp secure
```

- **classic CLI**

```
bof auto-boot secure
```

When you power on the router in SZTP mode, SR OS uses the RESTCONF client and server, along with YANG models, to exchange artifacts in accordance with RFC 8572. SZTP runs on the same ports that support ZTP on each platform.

SZTP bootstraps a node securely. The process requires the following:

- initial artifacts that establish mutual trust between the device and the bootstrap server
- the bootstrap server IP address, which the device obtains through DHCP discovery
- a secure and encrypted download of critical data such as the image and configuration file from the servers named in the onboarding information

SR OS uses a dedicated bootstrapping method to obtain TLS certificates, trust anchors, and redirect information to connect securely to the bootstrap server and download everything needed to boot in operational mode.



Note: In IEEE 802.1AR, a Device Identifier (DevID) is the general term for a Secure Device Identifier. The Initial Device Identifier (IDevID) is a unique, manufacturer-provided identity bound to the device, typically stored in secure hardware, that represents the device's core identity. A Local Device Identifier (LDevID) is a locally significant device identity for short-term, environment-specific access.

SZTP performs the following steps:

- The ZTP bootstrap server (owned and managed by the user, and optionally Nokia CNDO) authenticates the router as Nokia-manufactured equipment. The router presents the IDevID certificate installed on the router.
- The router authenticates data from the ZTP bootstrap server as coming from its owner (the user who purchased it). The bootstrap server sends an Ownership Voucher (OV) to the router, and the router validates it.
- The router transfers ownership or trust to the customer by installing customer certificates and using them for later communication with the bootstrap server.

Initially, the customer bootstrap server stores:

- DevID trust anchor certificates
- an ownership voucher for each purchased router (device), available from [Nokia sZTP Portal](#)
- optional redirect bootstrap server certificates and trust anchors if the customer redirects the router to an alternate server to download the image and configuration (the customer generates these)
- a list of artifacts (configurations, images, scripts, and so on)

On the CPM, the following resides in the TPM:

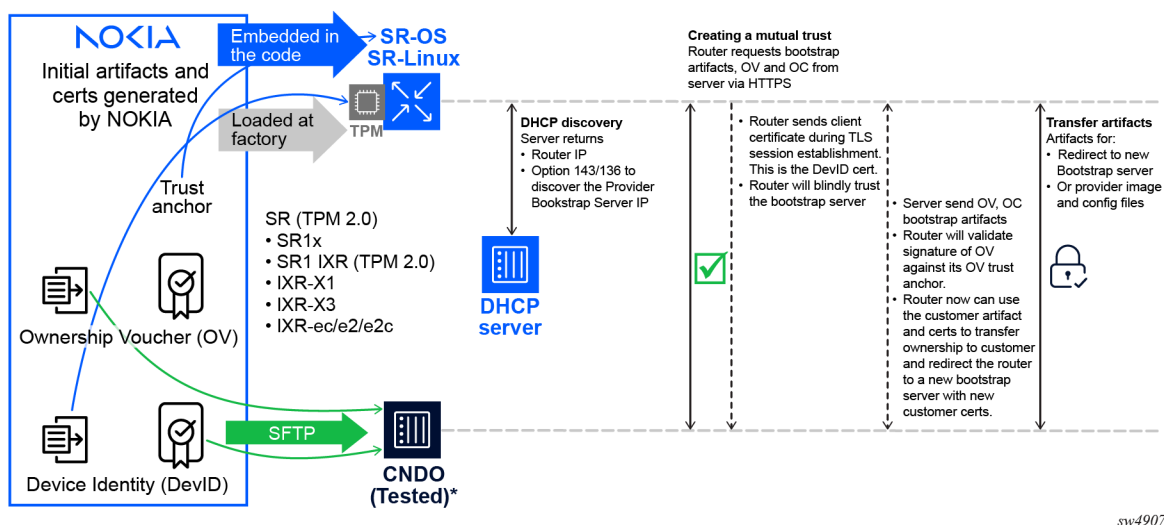
- a per-CPM private key that generates the IDevID, valid for the lifetime of the CPM
- the DevID public certificate
- in the software, a Nokia CA trust anchor certificate that validates the signature of the ownership voucher



Note: The IDevID certificate subject matches the serial ID.

The following figure shows the SZTP user workflow.

Figure 38: SZTP user workflow



The detailed workflow displayed in [Figure 38: SZTP user workflow](#) is as follows:

1. At the Nokia factory, Nokia creates and signs a unique DevID certificate for each CPM and loads it onto the router.

2. The user obtains the DevID trust anchor from [Nokia sZTP Portal](#).
3. The OV trust anchor is stapled to the Nokia image.
4. The user obtains the OV from [Nokia sZTP Portal](#).
 - a. Each OV is unique per router (by serial ID) and contains the pinned-domain-cert that the owner certificate (OC) must chain to.
 - b. Nokia CAs sign each OV. Because each OV contains a unique node serial ID, each OV has its own signature.
5. The DevID certificate and ownership voucher establish mutual trust between the router and the first bootstrap server. Each party holds the appropriate certificate, trust anchor, and key pairs.
6. To connect to [Nokia sZTP Portal](#), the customer registers with Nokia IT, uploads a certificate, and establishes an HTTPS connection to download the certificates and material for steps 1 through 4.

These artifacts enable the node to bootstrap securely. Users can ship the node to the installation site, install it, and power it on. The node then performs the following sequence of events.

1. Performs DHCP discovery and obtains redirect information from DHCP option 143 (IPv4) or 136 (IPv6). After the node receives redirect information, it connects to the bootstrap server over HTTPS to establish mutual trust.
2. Uses HTTPS to establish trust with the bootstrap server:
 - a. The router initially trusts the bootstrap server without verifying the TLS server identity.
 - b. The router sends its DevID certificate so the bootstrap server can authenticate the router.
 - c. The bootstrap server sends the router an OV, OC, and the appropriate artifacts. The router authenticates the OC using the appropriate trust anchor. After the router authenticates the OC, it can use the artifacts, redirect information, or onboarding information.



Note: The OV and OC validate the artifacts, not the TLS connection; the TLS connection remains unverified.

- i. If signed redirect information is present and the router validates it using the OV and OC, the artifacts include the address of another bootstrap server and, optionally, the trust anchor so the router can connect to that bootstrap server over TLS and continue the bootstrapping process there.
- ii. If signed onboarding information is present and the router verifies it with the OV and OC, the router uses the image and configuration URLs to download the image and configuration. The router does not send a progress report in this case.



Note: If a provisioning file is present on the router, SR OS ignores it and runs the SZTP process instead.

8.6.1 TLS details

SZTP can use TLS 1.2 and TLS 1.3. At the time of the TLS handshake, all SR OS supported ciphers and signatures are advertised.

TLS 1.2 supports the following ciphers and signatures:

- <cipher-suite-code>:
 - tls-rsa-with-aes128-cbc-sha256

- tls-rsa-with-aes256-cbc-sha256
- tls-rsa-with-aes128-gcm-sha256
- tls-rsa-with-aes256-gcm-sha384
- tls-ecdh-eccdsa-aes128-gcm-sha256
- tls-ecdh-eccdsa-aes256-gcm-sha384
- tls-ecdh-rsa-aes128-gcm-sha256
- tls-ecdh-rsa-aes256-gcm-sha384
- SHA1 ciphers are excluded because they are outdated and not secure:
 - tls-rsa-with3des-ede-cbc-sha
 - tls-rsa-with-aes128-cbc-sha
 - tls-rsa-with-aes256-cbc-sha

TLS 1.3 supports the following ciphers and signatures:

- <cipher-suite-code>:
 - tls-aes128-gcm-sha256
 - tls-aes256-gcm-sha384
 - tls-chacha20-poly1305-sha256
 - tls-aes128-ccm-sha256
- <group-suite-code>:
 - tls-ecdh-256
 - tls-ecdh-384
 - tls-ecdh-521
 - tls-x25519
 - tls-x448
- <signature-suite-code>:
 - tls-rsa-pss-rsa-sha256
 - tls-ecdsa-secp384r1-sha384



Note: SR OS cannot use other signature algorithms because OpenSSL uses this list for the server certificate signature, client certificate signature, and the client certificate that has a TPM2 private key. The current implementation only supports this signature algorithm.

8.6.2 Bootstrap methods

The following bootstrap methods are supported for SZTP (RFC 8572-compliant):

- DHCP – DHCP is only used for option 143 (IPv4) or 136 (IPv6) as per RFC 8572 and is not used as a bootstrapping method. DHCP provides option 143/136 with the bootstrap server IP address
- Bootstrap server – Nokia Service Node Automation (SNA) and third-party servers. Nokia has not tested ZTP with any other non-Nokia bootstrap server.

Additional redirects from the first bootstrap server to consecutive bootstrap servers are possible. The bootstrap server can provide the node with additional redirect information in a secure, encrypted manner.

Redirect information to another bootstrap server

The redirect information can include a trust anchor for the next bootstrap server. See [Conveyed, redirect, and onboarding information support](#) for information about how the trust anchor affects the connection.



Note: Redirect information is provided as part of the conveyed information file. The conveyed information must not contain onboarding information.

8.6.3 Secure ZTP BOF flag

Because all TPM 2.0 nodes are shipped with appropriate certificates, it is impossible to determine whether the ZTP method is SZTP or normal ZTP from the presence of the certificates alone.

On SR OS, there is a specific SZTP BOF flag (separate from the ZTP flag) and a specific ordering part number.

The SZTP flag in the `bof.cfg` file is configured with the following command:

- **MD-CLI**

```
bof auto-boot dhcp secure
```

- **classic CLI**

```
bof auto-boot secure
```

8.6.4 Conveyed, redirect, and onboarding information support

When the router initiates communication with the bootstrap server, it uses the DevID to establish TLS communication. Until the OV is validated, the router considers this an untrusted server and accepts only the redirect information within the conveyed-information YANG data in the `ietf-sztp-conveyed-info` module.

SR OS supports signed or unsigned redirect artifacts:

- If the redirect is unsigned, the provided TLS certificates (trust anchor) from that unsigned redirect are not used.
- TLS certificates are accepted only when the redirect is signed.
- At any point in the redirect process, SR OS can accept signed onboarding information. Progress reports are sent only over a trusted (TLS-verified) connection. Onboarding received over an untrusted connection produces no progress report.

SR OS mandates that the trust anchor must be provided for each bootstrap server within the redirect information. If a signed redirect arrives without a trust anchor, SR OS treats this redirect as unsigned and requires a signed redirect from the next server.

Additionally, the CRL must be stapled to the redirect certificate. This certificate is used to promote the connection from untrusted to trusted as described in appendix B of RFC 8572.

Example: Redirect information YANG structure

```

+--:(redirect-information)
+-- redirect-information
  +-- bootstrap-server* [address]
    +-- address          inet:host
    +-- port?            inet:port-number
    +-- trust-anchor?    cms

```

Example: Onboarding information YANG structure

After a trusted connection is established, SR OS supports the following onboarding information:

```

+--:(onboarding-information)
+-- onboarding-information
  +-- boot-image
    | +-- os-name?          string
    | +-- os-version?      string
    | +-- download-uri*    inet:uri
    | +-- image-verification* [hash-algorithm]
    |   +-- hash-algorithm identityref
    |   +-- hash-value     yang:hex-string
    +-- configuration-handling? enumeration
    +-- pre-configuration-script? script
    +-- configuration?      binary
    +-- post-configuration-script? script

```

The pre-configuration-script is supported; its format is described in [Configuration script](#).

The boot-image container indicates the download-uri of the software image ZIP file containing all SROS.tim files, with a maximum of three URIs. The image-verification leaf is mandatory when a download-uri is supplied.

Download URI supports HTTPS, FTP, and HTTP, with HTTPS blindly accepting the server's end-entity certificate.



Note: Nokia recommends using the boot-image leaf in the onboarding information structure to download the image.

The following onboarding options are initially not supported:

- configuration-handling
- configuration
- post-configuration-script

8.6.5 Image download rules

Although Nokia does not recommend omitting the boot-image container, this method is supported to provide flexible image downloads. In this scenario, SR OS runs the pre-configuration-script (including the provisioning script or file), and the image object is processed. SR OS sends a boot-image-installed-rebooting message to the bootstrap server after successful processing.

Image URL in the boot-image container

If the `os-version` and `os-name` are not set, SR OS times out and moves to the next bootstrap server, or to the next port if no other server exists.

If `os-version` and `os-name` are set:

1. SR OS checks `os-version` against the running OS version.
 - If they match, the image URL setting does not matter; SR OS sends a `boot-image-complete` message.
 - If they do not match, `download-uri` and `image-verification` must be set. Otherwise, SR OS errors out, sends a `boot-image-error` message, and moves to the next bootstrap server or port.
2. If the current image version is not equal to `os-version`, SR OS sends `boot-image-mismatch`, downloads the image, verifies that the downloaded image version matches the `os-version`, and verifies the hash. If any step fails, SR OS sends a `boot-image-error` message and moves to the next bootstrap server.
3. SR OS processes the preconfiguration script and the provisioning script or file, including updating the BOF portion.
 - If the `boot-image` is set and the provisioning file is parsed with an image object in the provisioning script or file, SR OS errors out with `pre-script-error`.
4. On completion, SR OS sends the `boot-image-installed-rebooting` message to the bootstrap server.

8.6.6 Configuration script

The configuration script is a base64 encoded JSON structure defined as follows:

Example

```
yang-data config-script:
+--rw config-script
  +--rw type?          enumeration
  +--rw (location)
    +--:(embedded)
      | +--rw embedded?    binary
      +--:(download-url)
        +--rw download-url* types-sros:remote-file-url
```

Currently, one type of `config-script` is supported: `pre-boot-config-script`.

The pre-boot configuration script can be administered in two formats:

- provisioning script – embedded provisioning file in a preconfiguration script
- provisioning file – embedded URL in a preconfiguration script to download the provisioning file

8.6.6.1 Provisioning file format

Provisioning files do not support the legacy non-RFC 8572-compliant SZTP format, including onboarding information and redirect information. If onboarding or redirect information is present in the provisioning file, parsing fails while **auto-boot secure** is enabled in the BOF.



Note: For legacy secure ZTP, use the **auto-boot** flag in the BOF and not the **auto-boot secure** flag.

The `download-url` leaf-list supports up to three URL locations for downloading the provisioning file. SR OS supports FTP, HTTP, and HTTPS download protocols.



Note: Nokia recommends obtaining the provisioning file from the connected bootstrap server using TLS mutual authentication.

The following ZTP provisioning file components are supported:

- configuration file URL (primary, secondary, tertiary)
- file download
- use the DNS in the provisioning file if present; otherwise, use the DHCP DNS
- generate a new BOF file to point it to the correct directory when the configuration and image are downloaded
- day-zero configuration

The following ZTP provisioning file components are not supported:

- onboarding information
- redirect information

It is not recommended to put the image file URL in the provisioning file.



Note: Nokia recommends using the `boot-image` leaf at the `onboarding-information` structure to download the image. Although the user can also use the provisioning file in the `pre-configuration-script` to download the image, this method is not recommended. Do not configure both options. If the `boot-image` container is set and the provisioning file also contains an image download, SR OS rejects the provisioning script with a pre-script error. See [Image download rules](#) for more information.

8.6.6.2 Provisioning script format

Day-zero configuration is not supported in the `pre-configuration-script` provisioning script format, but it is supported in the provisioning file format.

When the provisioning file contents are included with the provisioning script, they are supplied in the embedded leaf as base64-encoded.

The provisioning script supports a maximum of 10,000 characters.

When SZTP receives the `pre-configuration-script` with the provisioning script, it writes the received script to the CF. Because the script format is not checked until it is loaded on the router, Nokia recommends testing the script before deploying it in a production environment.

The provisioning script is saved in the form of a ZTP provisioning file and has limitations when the following command is enabled:

- **MD-CLI**

```
bof auto-boot dhcp secure
```

- **classic CLI**

```
bof auto-boot secure
```

8.6.6.3 Examples of the pre-configuration-script

Example: Preconfiguration script with download URLs

```

"pre-configuration-script" : "base64 encoded"
{
  "config-script": {
    "type": "pre-boot-config-script",
    "location": {
      "download-url": [
        "https://<ip-address1>/path-to-pre-boot-script",
        "https://<ip-address2>/path-to-pre-boot-script",
        "https://<ip-address3>/path-to-pre-boot-script"
      ]
    }
  }
}

```

Example: Preconfiguration script with embedded provisioning

```

"pre-configuration-script" : "base64 encoded"
{
  "config-script": {
    "type": "pre-boot-config-script",
    "location": {
      "embedded": "base64-encrypted"
    }
  }
}

```

Example: Embedded provisioning script content

The following example displays embedded content (base64-encoded provisioning script body before encoding).



Note: Day zero configuration is not supported for embedded pre-configuration-script content.

```
{
"config-script": {
  "type": "pre-boot-config-script",
  "location": {
    "embedded": "setvar FILESERVER      = \"https://1.1.1.1/web/${sys.platform}/
${sys.type}\"\"nsetvar BOOT-PATH      = \"cf3:/ztp\"\"nsetvar CERT-PATH      = \"
\"$(BOOT-PATH)/certs\"\"nsetvar DAY0-CONFIG  = \"\"$(BOOT-PATH)/day0.cfg\"\"nsetvar
NSP_IP          = \"192.3.4.5\"\"n\nimport {\n  client {\n    cert \"$(CERT-
PATH)/device.crt\" {\n      primary-url \"$(FILESERVER)/device.crt\"\"n\t
\t encrypt\n\t\t verification {\n\t\t\t hash-algorithm sha256\n\t\t\t hash-value \"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849\
\n\t\t\t }\n\t\t\t key \"$(CERT-PATH)/device.key\" {\n
format pem\n\t\t\t primary-url \"$(FILESERVER)/device.key\"\"n\t\t\t encrypt
\n\t\t\t verification {\n\t\t\t\t hash-algorithm sha256\n\t\t\t\t hash-value \"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849\
\n\t\t\t\t }\n
```

```
} \n } \n trust-anchor \"NokiaSvc\" { \n cert \"$(CERT-PATH)/owner-ca.crt\" { \n format pem \n primary-url \"$(FILESERVER)/owner-ca.crt\" \n encrypt \n verification { \n hash-algorithm sha256 \n hash-value \"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849\" \n } \n url \"$(CERT-PATH)/owner-ca.crl\" { \n format der \n primary-url \"$(FILESERVER)/owner-ca.crl\" \n encrypt \n verification { \n hash-algorithm sha256 \n hash-value \"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849\" \n } \n } \n } \n }
```

Example: Provisioning script written into a file

```

setvar FILESERVER      = "https://1.1.1.1/web/$(sys.platform)/$(sys.type)"
setvar BOOT-PATH       = "cf3:/ztp"
setvar CERT-PATH       = "$(BOOT-PATH)/certs"
setvar DAY0-CONFIG     = "$(BOOT-PATH)/day0.cfg"
setvar NSP_IP          = "192.3.4.5"

import {
  client {
    cert "$(CERT-PATH)/device.crt" {
      primary-url "$(FILESERVER)/device.crt"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
    key "$(CERT-PATH)/device.key" {
      format pem
      primary-url "$(FILESERVER)/device.key"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
  }
  trust-anchor "NokiaSvcS" {
    cert "$(CERT-PATH)/owner-ca.crt" {
      format pem
      primary-url "$(FILESERVER)/owner-ca.crt"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
    crl "$(CERT-PATH)/owner-ca.crl" {
      format der
      primary-url "$(FILESERVER)/owner-ca.crl"
      encrypt
      verification {
        hash-algorithm sha256
        hash-value
"53473e2727caf55f3a38fa466622af2147762e26a8587e9248240a572cdee849"
      }
    }
  }
}

```

```
}  
}
```

8.6.7 Installation site

At the installation site, the following command in the BOF signals the SZTP process:

- **MD-CLI**

```
bof auto-boot dhcp secure
```

- **classic CLI**

```
bof auto-boot secure
```

The following sequence of events occurs:

1. After the node is installed, the port with link up is selected and SZTP is executed on it. Port support for SZTP is the same as ZTP on a different chassis.
2. The node loads the IDevID certificates and OV trust anchor.
3. DHCP discovery is sent on each port in order. If there is no DHCP offer, SZTP moves to the next port with link up.
 - a. The OOB port is examined first, then untagged in-band ports.
 - b. If there is no offer, VLAN discovery is performed on in-band ports only (flood VLAN 0 through 4096 with DHCP discovery).
4. After DHCP discovery obtains an IP address, the node obtains option 143 (IPv4) or 136 (IPv6) for redirect information.
5. The node connects to the bootstrap server indicated by the redirect information from the DHCP option and establishes mutual authentication and trust using IDevID and OV certificates and trust anchors.
 - a. The bootstrap server must have the correct OV, keys, and DevID trust anchor to create this mutual trust.
 - b. Mutual trust uses HTTPS.
6. After the bootstrap server authenticates the node, the router blindly trusts the server and requests the conveyed information using RESTCONF from the server.
7. If the conveyed information from RESTCONF contains signed redirect information, the node connects to the new bootstrap server indicated in the redirect. The node can download a new trust anchor indicated by the redirect information.
8. If the conveyed information YANG contains only onboarding information, the node first processes the `boot-image` container (version check, download, hash verification), then runs the provisioning script carried in or downloaded using the `pre-configuration-script` leaf.
9. SZTP succeeds if the node executes onboarding information with no redirect information and the onboarding information executes without error.

8.6.8 Non-TPM 2.0 platforms

Non-TPM 2.0 platforms do not support RFC 8572-compliant SZTP. They support a legacy SZTP with the **auto-boot** flag, in which the user must stage the router by adding redirect, onboarding, and certificate files on the router directly.

8.6.9 Image download directory

Currently, SZTP onboarding information has no way to create a folder for a specific image download; all images are downloaded to the root directory of CF3.

Nokia recommends downloading the image as a ZIP file, which allows the OS to unzip the file and expand the file directory under the CF3 root directory.

Include the image version in the ZIP filename. This prevents the download from accidentally overwriting existing versions.

In addition, after the root directory of the ZIP file is uncompressed, ensure the file is prefixed with the image software version. This naming convention prevents folder collisions or overwriting previously uncompressed image folders.

Example: Example ZIP filename

Images-26.10.r1.zip

The following example displays a root directory after uncompressing the file.

```
Images-26.10.r1
| ---image
|   ...
```

8.6.10 Logs

SZTP supports the following log methods:

- console – when the user logs in to the router and the router is in SZTP processing mode, logs are displayed on the console CLI
- log files – logging files are generated automatically; use **tools dump auto-boot log** to view logs
- syslog – configured using the provisioning file (may result in lost logs for a period) or DHCPv4 Option 7

8.6.11 NTP and OV certificate date and time validation

If the DHCP offer includes NTP DHCP option 42 (for IPv4) or option 56 (for IPv6), the router validates the OV certificate date and time. If these NTP options are not present, the router bypasses this verification.

9 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.

9.1 Access Node Control Protocol (ANCP)

draft-ietf-ancp-protocol-02, *Protocol for Access Node Control Mechanism in Broadband Networks*

RFC 5851, *Framework and Requirements for an Access Node Control Mechanism in Broadband Multi-Service Networks*

9.2 Bidirectional Forwarding Detection (BFD)

draft-ietf-lsr-ospf-bfd-strict-mode-10, *OSPF BFD Strict-Mode*

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)*

RFC 5883, *Bidirectional Forwarding Detection (BFD) for Multihop Paths*

RFC 7130, *Bidirectional Forwarding Detection (BFD) on Link Aggregation Group (LAG) Interfaces*

RFC 7880, *Seamless Bidirectional Forwarding Detection (S-BFD)*

RFC 7881, *Seamless Bidirectional Forwarding Detection (S-BFD) for IPv4, IPv6, and MPLS*

RFC 7883, *Advertising Seamless Bidirectional Forwarding Detection (S-BFD) Discriminators in IS-IS*

RFC 7884, *OSPF Extensions to Advertise Seamless Bidirectional Forwarding Detection (S-BFD) Target Discriminators*

RFC 9247, *BGP - Link State (BGP-LS) Extensions for Seamless Bidirectional Forwarding Detection (S-BFD)*

9.3 Border Gateway Protocol (BGP)

draft-gredler-idr-bgplu-epe-14, *Egress Peer Engineering using BGP-LU*

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-flowspec-oid-03, *Revised Validation Procedure for BGP Flow Specifications*
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-flowspec-interfaceset-03, *Applying BGP flowspec rules on a specific interface set*
draft-ietf-idr-flowspec-path-redirect-05, *Flowspec Indirection-id Redirect – localised ID*
draft-ietf-idr-flowspec-redirect-ip-02, *BGP Flow-Spec Redirect to IP Action*
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 7674, *Clarification of the Flowspec Redirect Extended Community*
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 8571, *BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions*
RFC 8950, *Advertising IPv4 Network Layer Reachability Information (NLRI) with an IPv6 Next Hop*
RFC 8955, *Dissemination of Flow Specification Rules*
RFC 8956, *Dissemination of Flow Specification Rules for IPv6*
RFC 9086, *Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing BGP Egress Peer Engineering*
RFC 9294, *Application-Specific Link Attributes Advertisement Using the Border Gateway Protocol - Link State (BGP LS)*
RFC 9351, *Border Gateway Protocol - Link State (BGP-LS) Extensions for Flexible Algorithm Advertisement*
RFC 9494, *Long-Lived Graceful Restart for BGP*
RFC 9552, *Distribution of Link-State and Traffic Engineering Information Using BGP*

9.4 Bridging and management

IEEE 802.1AB, *Station and Media Access Control Connectivity Discovery*
IEEE 802.1ad, *Provider Bridges*
IEEE 802.1ag, *Connectivity Fault Management*
IEEE 802.1ah, *Provider Backbone Bridges*
IEEE 802.1ak, *Multiple Registration Protocol*
IEEE 802.1aq, *Shortest Path Bridging*
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*
IEEE 802.1X, *Port Based Network Access Control*

9.5 Broadband Network Gateway (BNG) Control and User Plane Separation (CUPS)

3GPP TS 23.003, *Numbering, addressing and identification*
3GPP TS 23.007, *Restoration procedures*
3GPP TS 23.402, *Architecture enhancements for non-3GPP accesses – S2a roaming based on GPRS*
3GPP TS 23.501, *System architecture for the 5G System (5GS)*
3GPP TS 23.502, *Procedures for the 5G System (5GS)*
3GPP TS 23.503, *Policy and charging control framework for the 5G System (5GS)*
3GPP TS 24.501, *Non-Access-Stratum (NAS) protocol for 5G System (5GS)*
3GPP TS 29.244, *Interface between the Control Plane and the User Plane nodes*
3GPP TS 29.281, *General Packet Radio System (GPRS) Tunneling Protocol User Plane (GTPv1-U)*
3GPP TS 29.500, *Technical Realization of Service Based Architecture*
3GPP TS 29.501, *Principles and Guidelines for Services Definition*
3GPP TS 29.502, *Session Management Services*
3GPP TS 29.503, *Unified Data Management Services*
3GPP TS 29.512, *Session Management Policy Control Service*
3GPP TS 29.518, *Access and Mobility Management Services*
3GPP TS 32.255, *5G data connectivity domain charging*
3GPP TS 32.290, *Services, operations and procedures of charging using Service Based Interface (SBI)*
3GPP TS 32.291, *5G system, charging service*
BBF TR-459, *Control and User Plane Separation for a Disaggregated BNG*
BBF TR-459.2, *Multi-Service Disaggregated BNG with CUPS: Integrated Carrier Grade NAT function*
RFC 8300, *Network Service Header (NSH)*
RFC 8910, *Captive-Portal Identification in DHCP and Router Advertisements (RAs)*

9.6 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*

9.7 Circuit emulation

RFC 4553, *Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)*
RFC 5086, *Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)*
RFC 5287, *Control Protocol Extensions for the Setup of Time-Division Multiplexing (TDM) Pseudowires in MPLS Networks*

9.8 Ethernet

IEEE 802.3ah, *Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks*
IEEE 802.3x, *Ethernet Flow Control*
ITU-T G.8031/Y.1342, *Ethernet Linear Protection Switching*
ITU-T G.8032/Y.1344, *Ethernet Ring Protection Switching*
ITU-T Y.1731, *OAM functions and mechanisms for Ethernet based networks*

9.9 Ethernet VPN (EVPN)

draft-ietf-bess-evpn-ip-aliasing-03, *EVPN Support for L3 Fast Convergence and Aliasing/Backup Path*
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*
draft-ietf-bess-evpn-unequal-lb-16, *Weighted Multi-Path Procedures for EVPN Multi-Homing – section 9*
draft-ietf-bess-evpn-vpws-gateway-01, *Ethernet VPN Virtual Private Wire Services Gateway Solution*
RFC 7432, *BGP MPLS-Based Ethernet VPN*
RFC 7623, *Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)*
RFC 8214, *Virtual Private Wire Service Support in Ethernet VPN*
RFC 8317, *Ethernet-Tree (E-Tree) Support in Ethernet VPN (EVPN) an Provider Backbone Bridging EVPN (PBB-EVPN)*
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 9014, *Interconnect Solution for Ethernet VPN (EVPN) Overlay Networks*
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 9541, *Flush Mechanism for Customer MAC Addresses Based on Service Instance Identifier (I-SID) in Provider Backbone Bridging EVPN (PBB-EVPN)*
RFC 9625, *EVPN Optimized Inter-Subnet Multicast (OISM) Forwarding – ingress replication and mLDP*
RFC 9784, *Virtual Ethernet Segments for EVPN and Provider Backbone Bridge EVPN*
RFC 9785, *Preference-Based EVPN Designated Forwarder (DF) Election*
RFC 9786, *EVPN Port-Active Redundancy Mode*
RFC 10039, *Interconnecting EVPN and IPVPN Domains*

9.10 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*

9.11 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 6119, *IPv6 Traffic Engineering in 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 6329, *IS-IS Extensions Supporting IEEE 802.1aq Shortest Path Bridging*

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*

9.12 Internet Protocol (IP) Fast Reroute (FRR)

draft-ietf-rtgwg-lfa-manageability-08, *Operational management of Loop Free Alternates*

RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*

RFC 7431, *Multicast-Only Fast Reroute*

RFC 7490, *Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)*

RFC 8518, *Selection of Loop-Free Alternates for Multi-Homed Prefixes*

9.13 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 2347, *TFTP Option Extension*

RFC 2348, *TFTP Blocksize Option*

RFC 2349, *TFTP Timeout Interval and Transfer Size Options*

RFC 2428, *FTP Extensions for IPv6 and NATs*

RFC 2617, *HTTP Authentication: Basic and Digest Access Authentication*

RFC 2784, *Generic Routing Encapsulation (GRE)*

RFC 2818, *HTTP Over TLS*

RFC 2890, *Key and Sequence Number Extensions to 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 7011, *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*
RFC 7012, *Information Model for IP Flow Information Export*
RFC 7230, *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*
RFC 7231, *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*
RFC 7232, *Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests*
RFC 7301, *Transport Layer Security (TLS) Application Layer Protocol Negotiation Extension*
RFC 7616, *HTTP Digest Access Authentication*
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*

9.14 Internet Protocol (IP) multicast

cisco-ipmulticast/pim-autorp-spec01, *Auto-RP: Automatic discovery of Group-to-RP mappings for IP multicast* – version 1
draft-ietf-bier-pim-signaling-08, *PIM Signaling Through BIER Core*
draft-ietf-idmr-traceroute-ipm-07, *A "traceroute" facility for IP Multicast*
draft-ietf-l2vpn-vpls-pim-snooping-07, *Protocol Independent Multicast (PIM) over Virtual Private LAN Service (VPLS)*
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 3306, *Unicast-Prefix-based IPv6 Multicast Addresses*
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 3956, *Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address*

RFC 3973, *Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised) – auto-RP groups*

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 4607, *Source-Specific Multicast for IP*

RFC 4608, *Source-Specific Protocol Independent Multicast in 232/8*

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 5496, *The Reverse Path Forwarding (RPF) Vector TLV*

RFC 6037, *Cisco Systems' Solution for Multicast in MPLS/BGP IP VPNs*

RFC 6512, *Using Multipoint LDP When the Backbone Has No Route to the Root*

RFC 6513, *Multicast in MPLS/BGP IP VPNs*

RFC 6514, *BGP Encodings and Procedures for Multicast in MPLS/IP VPNs*

RFC 6515, *IPv4 and IPv6 Infrastructure Addresses in BGP Updates for Multicast VPNs*

RFC 6516, *IPv6 Multicast VPN (MVPN) Support Using PIM Control Plane and Selective Provider Multicast Service Interface (S-PMSI) Join Messages*

RFC 6625, *Wildcards in Multicast VPN Auto-Discover Routes*

RFC 6826, *Multipoint LDP In-Band Signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Path*

RFC 7246, *Multipoint Label Distribution Protocol In-Band Signaling in a Virtual Routing and Forwarding (VRF) Table Context*

RFC 7385, *IANA Registry for P-Multicast Service Interface (PMSI) Tunnel Type Code Points*

RFC 7716, *Global Table Multicast with BGP Multicast VPN (BGP-MVPN) Procedures*

RFC 7761, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*

RFC 8279, *Multicast Using Bit Index Explicit Replication (BIER)*

RFC 8296, *Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks – MPLS encapsulation*

RFC 8401, *Bit Index Explicit Replication (BIER) Support via IS-IS*

RFC 8444, *OSPFv2 Extensions for Bit Index Explicit Replication (BIER)*

RFC 8487, *Mtrace Version 2: Traceroute Facility for IP Multicast*

RFC 8534, *Explicit Tracking with Wildcard Routes in Multicast VPN* – (C-*,C-*) wildcard
RFC 8556, *Multicast VPN Using Bit Index Explicit Replication (BIER)*
RFC 9573, *MVPN/EVPN Tunnel Aggregation with Common Labels* – DCB and static service labels

9.15 Internet Protocol (IP) version 4

RFC 791, *Internet Protocol*
RFC 792, *Internet Control Message Protocol*
RFC 826, *An Ethernet Address Resolution Protocol*
RFC 951, *Bootstrap Protocol (BOOTP)* – relay
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 1534, *Interoperation between DHCP and BOOTP*
RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*
RFC 1812, *Requirements for IPv4 Routers*
RFC 1918, *Address Allocation for Private Internets*
RFC 2003, *IP Encapsulation within IP*
RFC 2131, *Dynamic Host Configuration Protocol*
RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
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

9.16 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 3646, *DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*
RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*
RFC 3971, *SEcure Neighbor Discovery (SEND)*
RFC 3972, *Cryptographically Generated Addresses (CGA)*
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 4862, *IPv6 Stateless Address Autoconfiguration – router functions*
RFC 4890, *Recommendations for Filtering ICMPv6 Messages in Firewalls*
RFC 4941, *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*
RFC 5007, *DHCPv6 Leasequery*
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 6092, *Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service – Internet Control and Management, Upper-Layer Transport Protocols, UDP Filters, IPsec and Internet Key Exchange (IKE), TCP Filters*
RFC 6106, *IPv6 Router Advertisement Options for DNS Configuration*
RFC 6164, *Using 127-Bit IPv6 Prefixes on Inter-Router Links*
RFC 6221, *Lightweight DHCPv6 Relay Agent*
RFC 6437, *IPv6 Flow Label Specification*
RFC 6603, *Prefix Exclude Option for DHCPv6-based Prefix Delegation*
RFC 8021, *Generation of IPv6 Atomic Fragments Considered Harmful*
RFC 8200, *Internet Protocol, Version 6 (IPv6) Specification*
RFC 8201, *Path MTU Discovery for IP version 6*

9.17 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*

9.18 Label Distribution Protocol (LDP)

draft-pdutta-mpls-ldp-adj-capability-00, *LDP Adjacency Capabilities*

draft-pdutta-mpls-ldp-v2-00, *LDP Version 2*

draft-pdutta-mpls-ldp-up-redundancy-00, *Upstream LSR Redundancy for Multi-point LDP Tunnels*

draft-pdutta-mpls-multi-ldp-instance-00, *Multiple LDP Instances*

draft-pdutta-mpls-ldp-hello-reduce-04, *Targeted LDP Hello Reduction*

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*

RFC 6388, *Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths*

RFC 6512, *Using Multipoint LDP When the Backbone Has No Route to the Root*

RFC 6826, *Multipoint LDP in-band signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths*

RFC 7032, *LDP Downstream-on-Demand in Seamless MPLS*

RFC 7473, *Controlling State Advertisements of Non-negotiated LDP Applications*

RFC 7552, *Updates to LDP for IPv6*

9.19 Layer Two Tunneling Protocol (L2TP) Network Server (LNS)

draft-mammoliti-l2tp-accessline-avp-04, *Layer 2 Tunneling Protocol (L2TP) Access Line Information Attribute Value Pair (AVP) Extensions*

RFC 2661, *Layer Two Tunneling Protocol "L2TP"*

RFC 2809, *Implementation of L2TP Compulsory Tunneling via RADIUS*

RFC 3438, *Layer Two Tunneling Protocol (L2TP) Internet Assigned Numbers: Internet Assigned Numbers Authority (IANA) Considerations Update*

RFC 3931, *Layer Two Tunneling Protocol - Version 3 (L2TPv3)*
RFC 4719, *Transport of Ethernet Frames over Layer 2 Tunneling Protocol Version 3 (L2TPv3)*
RFC 4951, *Fail Over Extensions for Layer 2 Tunneling Protocol (L2TP) "failover"*

9.20 Multiprotocol Label Switching (MPLS)

draft-ietf-mpls-lsp-ping-ospfv3-codepoint-02, *OSPFv3 CodePoint for MPLS LSP Ping*
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 4023, *Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)*
RFC 4182, *Removing a Restriction on the use of MPLS Explicit NULL*
RFC 4950, *ICMP Extensions for Multiprotocol Label Switching*
RFC 5332, *MPLS Multicast Encapsulations*
RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*
RFC 6374, *Packet Loss and Delay Measurement for MPLS Networks – Delay Measurement, Channel Type 0x000C*
RFC 6424, *Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels*
RFC 6425, *Detecting Data Plane Failures in Point-to-Multipoint Multiprotocol Label Switching (MPLS) - Extensions to LSP Ping*
RFC 6790, *The Use of Entropy Labels in MPLS Forwarding*
RFC 7308, *Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)*
RFC 7510, *Encapsulating MPLS in UDP*
RFC 7746, *Label Switched Path (LSP) Self-Ping*
RFC 7876, *UDP Return Path for Packet Loss and Delay Measurement for MPLS Networks – Delay Measurement*
RFC 8029, *Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures*

9.21 Multiprotocol Label Switching - Transport Profile (MPLS-TP)

RFC 5586, *MPLS Generic Associated Channel*
RFC 5921, *A Framework for MPLS in Transport Networks*
RFC 5960, *MPLS Transport Profile Data Plane Architecture*
RFC 6370, *MPLS Transport Profile (MPLS-TP) Identifiers*
RFC 6378, *MPLS Transport Profile (MPLS-TP) Linear Protection*
RFC 6426, *MPLS On-Demand Connectivity and Route Tracing*

RFC 6427, *MPLS Fault Management Operations, Administration, and Maintenance (OAM)*
RFC 6428, *Proactive Connectivity Verification, Continuity Check and Remote Defect indication for MPLS Transport Profile*
RFC 6478, *Pseudowire Status for Static Pseudowires*
RFC 7213, *MPLS Transport Profile (MPLS-TP) Next-Hop Ethernet Addressing*

9.22 Network Address Translation (NAT)

draft-ietf-behave-address-format-10, *IPv6 Addressing of IPv4/IPv6 Translators*
draft-ietf-behave-v6v4-xlate-23, *IP/ICMP Translation Algorithm*
draft-miles-behave-l2nat-00, *Layer2-Aware 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*
RFC 6146, *Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers*
RFC 6333, *Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion*
RFC 6334, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite*
RFC 6887, *Port Control Protocol (PCP)*
RFC 6888, *Common Requirements For Carrier-Grade NATs (CGNs)*
RFC 7753, *Port Control Protocol (PCP) Extension for Port-Set Allocation*
RFC 7915, *IP/ICMP Translation Algorithm*

9.23 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*

9.24 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 5187, *OSPFv3 Graceful Restart – helper mode*

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*

9.25 Path Computation Element Protocol (PCEP)

draft-alvarez-pce-path-profiles-04, *PCE Path Profiles*

draft-dhs-spring-pce-sr-p2mp-policy-00, *PCEP extensions for p2mp sr policy*

draft-ietf-pce-binding-label-sid-15, *Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks*. – MPLS binding SIDs

draft-ietf-pce-multipath-18, *Path Computation Element Communication Protocol (PCEP) Extensions for Signaling Multipath Information*

draft-ietf-pce-pceps-tls13-04, *Updates for PCEPS: TLS Connection Establishment Restrictions*

RFC 5440, *Path Computation Element (PCE) Communication Protocol (PCEP)*

RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*

RFC 8233, *Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs) – Path Delay Metric*

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*

RFC 9862, *Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing (SR) Policy Candidate Paths*

9.26 Point-to-Point Protocol (PPP)

RFC 1332, *The PPP Internet Protocol Control Protocol (IPCP)*

RFC 1661, *The Point-to-Point Protocol (PPP)*

RFC 1877, *PPP Internet Protocol Control Protocol Extensions for Name Server Addresses*

RFC 1990, *The PPP Multilink Protocol (MP)*

RFC 1994, *PPP Challenge Handshake Authentication Protocol (CHAP)*

RFC 2516, *A Method for Transmitting PPP Over Ethernet (PPPoE)*

RFC 4638, *Accommodating a Maximum Transit Unit/Maximum Receive Unit (MTU/MRU) Greater Than 1492 in the Point-to-Point Protocol over Ethernet (PPPoE)*

RFC 5072, *IP Version 6 over PPP*

9.27 Policy management and credit control

3GPP TS 29.212 Release 11, *Policy and Charging Control (PCC); Reference points – Gx support as it applies to wireline environment (BNG)*

RFC 4006, *Diameter Credit-Control Application*

RFC 6733, *Diameter Base Protocol*

9.28 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*

9.29 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)*

9.30 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 2867, *RADIUS Accounting Modifications for Tunnel Protocol Support*
RFC 2868, *RADIUS Attributes for Tunnel Protocol Support*
RFC 2869, *RADIUS Extensions*
RFC 3162, *RADIUS and IPv6*
RFC 4818, *RADIUS Delegated-IPv6-Prefix Attribute*
RFC 5176, *Dynamic Authorization Extensions to RADIUS*
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*
RFC 6911, *RADIUS attributes for IPv6 Access Networks*

9.31 Resource Reservation Protocol - Traffic Engineering (RSVP-TE)

draft-newton-mpls-te-dynamic-overbooking-00, *A Diffserv-TE Implementation Model to dynamically change booking factors during failure events*
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 3477, *Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)*
RFC 3564, *Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering*
RFC 3906, *Calculating Interior Gateway Protocol (IGP) Routes Over Traffic Engineering Tunnels*
RFC 4090, *Fast Reroute Extensions to RSVP-TE 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 4875, *Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)*
RFC 5712, *MPLS Traffic Engineering Soft Preemption*

RFC 5817, *Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks*

9.32 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*

9.33 Segment Routing (SR)

draft-bashandy-rtgwg-segment-routing-uloop-15, *Loop avoidance using Segment Routing*

draft-filsfils-spring-net-pgm-extension-srv6-usid-15, *Network Programming extension: SRv6 uSID instruction*

draft-filsfils-spring-srv6-net-pgm-insertion-08, *SRv6 NET-PGM extension: Insertion*

draft-ietf-bess-mvpn-evpn-sr-p2mp-07, *Multicast and Ethernet VPN with Segment Routing P2MP and Ingress Replication – MVPN*

draft-ietf-idr-segment-routing-te-policy-23, *Advertising Segment Routing Policies in BGP*

draft-ietf-idr-ts-flowspec-srv6-policy-03, *Traffic Steering using BGP FlowSpec with SR Policy*

draft-ietf-pim-p2mp-policy-ping-03, *P2MP Policy Ping*

draft-ietf-pim-sr-p2mp-policy-06, *Segment Routing Point-to-Multipoint Policy – MPLS*

draft-ietf-rtgwg-segment-routing-ti-lfa-11, *Topology Independent Fast Reroute using Segment Routing*

draft-ietf-spring-conflict-resolution-05, *Segment Routing MPLS Conflict Resolution*

draft-ietf-spring-sr-replication-segment-16, *SR Replication segment for Multi-point Service Delivery – MPLS*

draft-voyer-6man-extension-header-insertion-10, *Deployments With Insertion of IPv6 Segment Routing Headers*

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 8663, *MPLS Segment Routing over IP – BGP SR with SR-MPLS-over-UDP/IP*

RFC 8665, *OSPF Extensions for Segment Routing*

RFC 8666, *OSPFv3 Extensions for Segment Routing*

RFC 8667, *IS-IS Extensions for Segment Routing*

RFC 8669, *Segment Routing Prefix Segment Identifier Extensions for BGP*
RFC 8754, *IPv6 Segment Routing Header (SRH)*
RFC 8814, *Signaling Maximum SID Depth (MSD) Using the Border Gateway Protocol - Link State*
RFC 8986, *Segment Routing over IPv6 (SRv6) Network Programming*
RFC 9085, *Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing*
RFC 9088, *Signaling Entropy Label Capability and Entropy Readable Label Depth Using IS-IS – advertising ELC*
RFC 9089, *Signaling Entropy Label Capability and Entropy Readable Label Depth Using OSPF – advertising ELC*
RFC 9252, *BGP Overlay Services Based on Segment Routing over IPv6 (SRv6)*
RFC 9256, *Segment Routing Policy Architecture*
RFC 9259, *Operations, Administration, and Maintenance (OAM) in Segment Routing over IPv6 (SRv6)*
RFC 9350, *IGP Flexible Algorithm*
RFC 9352, *IS-IS Extensions to Support Segment Routing over the IPv6 Data Plane*
RFC 9514, *Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing over IPv6 (SRv6)*
RFC 9800, *Compressed SRv6 Segment List Encoding*
RFC 9819, *Argument Signaling for BGP Services in Segment Routing over IPv6 (SRv6)*

9.34 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-mboned-msdp-mib-01, *Multicast Source Discovery protocol MIB*
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 2213, *Integrated Services Management Information Base using SMIv2*
RFC 2494, *Definitions of Managed Objects for the DS0 and DS0 Bundle Interface Type*
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 3165, *Definitions of Managed Objects for the Delegation of Management Scripts*
RFC 3231, *Definitions of Managed Objects for Scheduling Management Operations*
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 3498, *Definitions of Managed Objects for Synchronous Optical Network (SONET) Linear Automatic Protection Switching (APS) Architectures*

RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

RFC 3592, *Definitions of Managed Objects for the Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) Interface Type*

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 3637, *Definitions of Managed Objects for the Ethernet WAN Interface Sublayer*

RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*

RFC 3877, *Alarm Management Information Base (MIB)*

RFC 3895, *Definitions of Managed Objects for the DS1, E1, DS2, and E2 Interface Types*

RFC 3896, *Definitions of Managed Objects for the DS3/E3 Interface Type*

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*

SFLOW-MIB revision 200309240000Z, *sFlowMIB*

9.35 Storage sanitization

IEEE 2883-2022, *IEEE Standard for Sanitizing Storage* – clear sanitization method for CF, MMC, SSD, SD, USB

9.36 Timing

GR-1244-CORE Issue 3, *Clocks for the Synchronized Network: Common Generic Criteria*
GR-253-CORE Issue 3, *SONET Transport Systems: Common Generic Criteria*
IEEE 1588-2008, *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*
ITU-T G.781, *Synchronization layer functions*
ITU-T G.811, *Timing characteristics of primary reference clocks*
ITU-T G.813, *Timing characteristics of SDH equipment slave clocks (SEC)*
ITU-T G.8261, *Timing and synchronization aspects in packet networks*
ITU-T G.8262, *Timing characteristics of synchronous Ethernet equipment slave clock (EEC)*
ITU-T G.8262.1, *Timing characteristics of an enhanced synchronous Ethernet equipment slave clock (eEEEC)*
ITU-T G.8264, *Distribution of timing information through packet networks*
ITU-T G.8265.1, *Precision time protocol telecom profile for frequency synchronization*
ITU-T G.8272, *Timing characteristics of primary reference time clocks – PRTC-A, PRTC-B*
ITU-T G.8275.1, *Precision time protocol telecom profile for phase/time synchronization with full timing support from the network*
ITU-T G.8275.2, *Precision time protocol telecom profile for phase/time synchronization with partial timing support from the network*
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*

9.37 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*
RFC 9534, *Simple Two-Way Active Measurement Protocol Extensions for Performance Measurement on a Link Aggregation Group*

9.38 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 7041, *Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging*

RFC 7117, *Multicast in Virtual Private LAN Service (VPLS)*

9.39 Voice and video

DVB BlueBook A86, *Transport of MPEG-2 TS Based DVB Services over IP Based Networks*

ETSI TS 101 329-5 Annex E, *QoS Measurement for VoIP - Method for determining an Equipment Impairment Factor using Passive Monitoring*

ITU-T G.1020 Appendix I, *Performance Parameter Definitions for Quality of Speech and other Voiceband Applications Utilizing IP Networks - Mean Absolute Packet Delay Variation & Markov Models*

ITU-T G.107, *The E Model - A computational model for use in planning*

ITU-T P.564, *Conformance testing for voice over IP transmission quality assessment models*

RFC 3550, *RTP: A Transport Protocol for Real-Time Applications – Appendix A.8*

RFC 4585, *Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)*

RFC 4588, *RTP Retransmission Payload Format*

9.40 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*

9.41 Yet Another Next Generation (YANG) OpenConfig Models

openconfig-aaa.yang version 0.4.0, *OpenConfig AAA Model*

openconfig-aaa-radius.yang version 0.3.0, *OpenConfig AAA RADIUS Model*

openconfig-aaa-tacacs.yang version 0.3.0, *OpenConfig AAA TACACS+ Model*

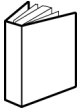
openconfig-acl.yang version 1.0.0, *OpenConfig ACL Model*

openconfig-alarms.yang version 0.3.2, *OpenConfig Alarms Model*

openconfig-bfd.yang version 0.2.2, *OpenConfig BFD Model*
openconfig-bgp.yang version 6.1.0, *OpenConfig BGP Model*
openconfig-bgp-common.yang version 6.0.0, *OpenConfig BGP Common Model*
openconfig-bgp-common-multiprotocol.yang version 6.0.0, *OpenConfig BGP Common Multiprotocol Model*
openconfig-bgp-common-structure.yang version 6.0.0, *OpenConfig BGP Common Structure Model*
openconfig-bgp-global.yang version 6.0.0, *OpenConfig BGP Global Model*
openconfig-bgp-neighbor.yang version 6.1.0, *OpenConfig BGP Neighbor Model*
openconfig-bgp-peer-group.yang version 6.1.0, *OpenConfig BGP Peer Group Model*
openconfig-bgp-policy.yang version 4.0.1, *OpenConfig BGP Policy Model*
openconfig-if-aggregate.yang version 2.4.3, *OpenConfig Interfaces Aggregated Model*
openconfig-if-ethernet.yang version 2.12.2, *OpenConfig Interfaces Ethernet Model*
openconfig-if-ip.yang version 3.9.0, *OpenConfig Interfaces IP Model*
openconfig-if-ip-ext.yang version 2.3.1, *OpenConfig Interfaces IP Extensions Model*
openconfig-igmp.yang version 0.3.1, *OpenConfig IGMP Model*
openconfig-interfaces.yang version 3.8.0, *OpenConfig Interfaces Model*
openconfig-isis.yang version 1.1.0, *OpenConfig IS-IS Model*
openconfig-isis-policy.yang version 0.5.0, *OpenConfig IS-IS Policy Model*
openconfig-isis-routing.yang version 1.1.0, *OpenConfig IS-IS Routing Model*
openconfig-lacp.yang version 2.1.0, *OpenConfig LACP Model*
openconfig-lldp.yang version 0.1.0, *OpenConfig LLDP Model*
openconfig-local-routing.yang version 1.2.0, *OpenConfig Local Routing Model*
openconfig-messages.yang version 0.0.1, *OpenConfig Messages Model*
openconfig-mpls.yang version 2.3.0, *OpenConfig MPLS Model*
openconfig-mpls-ldp.yang version 3.0.2, *OpenConfig MPLS LDP Model*
openconfig-mpls-rsvp.yang version 2.3.0, *OpenConfig MPLS RSVP Model*
openconfig-mpls-te.yang version 2.3.0, *OpenConfig MPLS TE Model*
openconfig-network-instance.yang version 1.1.0, *OpenConfig Network Instance Model*
openconfig-network-instance-l3.yang version 0.11.1, *OpenConfig L3 Network Instance Model – static routes*
openconfig-ospfv2.yang version 0.4.0, *OpenConfig OSPFv2 Model*
openconfig-ospfv2-area.yang version 0.4.0, *OpenConfig OSPFv2 Area Model*
openconfig-ospfv2-area-interface.yang version 0.4.0, *OpenConfig OSPFv2 Area Interface Model*
openconfig-ospfv2-common.yang version 0.4.0, *OpenConfig OSPFv2 Common Model*
openconfig-ospfv2-global.yang version 0.4.0, *OpenConfig OSPFv2 Global Model*
openconfig-packet-match.yang version 1.1.0, *OpenConfig Packet Match Model*
openconfig-pim.yang version 0.4.3, *OpenConfig PIM Model*
openconfig-platform.yang version 0.32.0, *OpenConfig Platform Model*

openconfig-platform-common.yang version 0.32.0, *OpenConfig Common Platform Model*
openconfig-platform-cpu.yang version 0.1.1, *OpenConfig Platform CPU Module*
openconfig-platform-fan.yang version 0.1.1, *OpenConfig Platform Fan Model*
openconfig-platform-linecard.yang version 0.1.2, *OpenConfig Platform Linecard Model*
openconfig-platform-port.yang version 0.4.2, *OpenConfig Port Model*
openconfig-platform-psu.yang version 0.2.1, *OpenConfig Platform PSU Module*
openconfig-platform-transceiver.yang version 0.18.0, *OpenConfig Transceiver Model*
openconfig-procmon.yang version 0.4.0, *OpenConfig Process Monitoring Model*
openconfig-qos.yang version 0.11.2, *OpenConfig QoS Model*
openconfig-qos-elements.yang version 0.11.2, *OpenConfig QoS Elements Model*
openconfig-qos-interfaces.yang version 0.11.2, *OpenConfig QoS Interfaces Model*
openconfig-qos-mem-mgmt.yang version 0.11.2, *OpenConfig QoS Memory Management Model*
openconfig-relay-agent.yang version 0.1.0, *OpenConfig Relay Agent Model*
openconfig-routing-policy.yang version 3.0.0, *OpenConfig Routing Policy Model*
openconfig-rsvp-sr-ext.yang version 0.1.0, *OpenConfig RSVP-TE and SR Extensions Model*
openconfig-system.yang version 3.0.0, *OpenConfig System Model*
openconfig-system-grpc.yang version 1.0.0, *OpenConfig System gRPC Model*
openconfig-system-logging.yang version 0.3.1, *OpenConfig System Logging Model*
openconfig-system-terminal.yang version 0.3.0, *OpenConfig System Terminal Model*
openconfig-telemetry.yang version 0.5.0, *OpenConfig Telemetry Model*
openconfig-terminal-device.yang version 1.9.0, *OpenConfig Terminal Device Model*
openconfig-vlan.yang version 3.2.2, *OpenConfig VLAN Model*

Customer document and product support



Customer documentation

[Customer documentation welcome page](#)



Technical support

[Product support portal](#)



Documentation feedback

[Customer documentation feedback](#)