



System Management Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.9.x

First Published: 2023-03-31

Americas Headquarters

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

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

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

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

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

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

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

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

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

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

© 2023 Cisco Systems, Inc. All rights reserved.



Preface

From Release 6.1.2 onwards, Cisco introduces support for the 64-bit Linux-based IOS XR operating system. Extensive feature parity is maintained between the 32-bit and 64-bit environments. Unless explicitly marked otherwise, the contents of this document are applicable for both the environments. For more details on Cisco IOS XR 64 bit, refer to the [Release Notes](#) for Cisco ASR 9000 Series Routers, Release 6.1.2 document.

This guide describes the System Management configuration details for Cisco IOS XR software. This chapter contains details on the changes made to this document.

- [Changes to This Document, on page iii](#)
- [Communications, Services, and Additional Information, on page iii](#)

Changes to This Document

This table lists the changes made to this document since it was first released.

Table 1: Changes to This Document

Date	Summary
None	None

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business results you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco DevNet](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).

Cisco Bug Search Tool

[Cisco Bug Search Tool](#) (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER

1

New and Changed System Management Features

This chapter lists all the features that have been added or modified in this guide. The table also contains references to these feature documentation sections.

- [System Management Features Added or Modified in IOS XR Release 7.9.x, on page 1](#)

System Management Features Added or Modified in IOS XR Release 7.9.x

Feature	Description	Changed in Release	Where Documented
FQDN for NTP Server on Non-default VRF	This feature was introduced	Release 7.9.1	FQDN for NTP Server, on page 58



CHAPTER 2

YANG Data Models for System Management Features

This chapter provides information about the YANG data models for System Management features.

- [Using YANG Data Models, on page 3](#)

Using YANG Data Models

Cisco IOS XR supports a programmatic way of configuring and collecting operational data of a network device using YANG data models. Although configurations using CLIs are easier and human-readable, automating the configuration using model-driven programmability results in scalability.

The data models are available in the release image, and are also published in the [Github](#) repository. Navigate to the release folder of interest to view the list of supported data models and their definitions. Each data model defines a complete and cohesive model, or augments an existing data model with additional XPathS. To view a comprehensive list of the data models supported in a release, navigate to the **Available-Content.md** file in the repository.

You can also view the data model definitions using the [YANG Data Models Navigator](#) tool. This GUI-based and easy-to-use tool helps you explore the nuances of the data model and view the dependencies between various containers in the model. You can view the list of models supported across Cisco IOS XR releases and platforms, locate a specific model, view the containers and their respective lists, leaves, and leaf lists presented visually in a tree structure. This visual tree form helps you get insights into nodes that can help you automate your network.

To get started with using the data models, see the *Programmability Configuration Guide*.



CHAPTER 3

Configuring Object Tracking

This module describes the configuration of object tracking on your Cisco IOS XR network. For complete descriptions of the commands listed in this module, see [Related Documents, on page 20](#). To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco ASR 9000 Series Aggregation Services Router Commands Master List*.

Table 2: Feature History Table

Feature Name	Release Information	Description
Enhanced Object Tracking	Release 6.4.2	The Enhanced Object Tracking feature is introduced. The ability to error-disable interfaces is added based on the state of objects that are tracked.
Enhanced Object Tracking	Release 4.2.1	The ability to create a tracked list based on a threshold percentage or weight was added.
Enhanced Object Tracking	Release 4.0.0	This feature was introduced.

This module contains the following topics:

- [Prerequisites for Implementing Object Tracking, on page 5](#)
- [Information about object tracking, on page 6](#)
- [Restrictions for Enhanced Object Tracking, on page 7](#)
- [How to Implement Object Tracking, on page 7](#)
- [Configure Enhanced Object Tracking, on page 17](#)
- [Configuration Examples for Configuring Object Tracking, on page 20](#)
- [Additional References, on page 20](#)

Prerequisites for Implementing Object Tracking

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information about object tracking

Object tracking enables the system to monitor a specific object and perform a client-defined action on another object. The object receiving the action may not have any direct relationship with the tracked object. The system bases these actions on changes in the properties of the tracked object.

- **Boolean AND function** — When you assign a Boolean AND function to a tracked list, the system requires every object in the subset to be in the "up" state for the tracked object to be considered "up."
- **Boolean OR function** — When you assign a Boolean OR function to a tracked list, the system requires at least one object in the subset to be in the "up" state for the tracked object to be considered "up."

Assign a tracked object

You can assign each tracked object a **unique name** using the `track` command in **configuration mode**.

Track process

The tracking process operates as follows:

- **Polling**: The system periodically polls the tracked object.
- **State Monitoring**: It detects and reports any changes in the object's state.
- **State Values**: The tracked object can be in either an "up" or "down" state.
- **Polling Timing**: Polling occurs either:
 - **Immediately**, or
 - **After a configured delay**

Track multiple objects

You can track multiple objects using a **list**. The system supports flexible combinations using **Boolean logic**.

Boolean AND Function

- When you assign a **Boolean AND** function to a tracked list:
 - **All** objects in the subset must be in the "up" state.
 - This condition sets the tracked object to the "up" state.

Boolean OR Function

- When you assign a **Boolean OR** function to a tracked list:
 - **At least one** object in the subset must be in the "up" state.
 - This condition sets the tracked object to the "up" state.

Restrictions for Enhanced Object Tracking

- You can perform Enhanced Object Tracking only on physical interfaces and not on virtual interfaces.
- The only action you can perform is error-disabling interfaces based on the state of a track (up/down).
- The maximum number of action interfaces that can be added under a single track is 1024.

How to Implement Object Tracking

This section describes the various object tracking procedures.

Tracking the Line Protocol State of an Interface

Perform this task in global configuration mode to track the line protocol state of an interface.

A tracked object is considered up when a line protocol of the interface is up.

After configuring the tracked object, you may associate the interface whose state should be tracked and specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type** **line-protocol** **state**
4. **interface** *type* *interface-path-id*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example:	Enters track configuration mode.

	Command or Action	Purpose
	<pre>RP/0/RSP0/CPU0:router(config)# track track1</pre>	<ul style="list-style-type: none"> <i>track-name</i>—Specifies a name for the object to be tracked. <p>Note Special characters are not allowed in a <i>track-name</i>.</p>
Step 3	<p>type line-protocol state</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track)# type line-protocol state</pre>	Creates a track based on the line protocol of an interface.
Step 4	<p>interface type interface-path-id</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track-line-prot)# interface atm 0/2/0/0.1</pre>	<p>Specifies the interface to track the protocol state.</p> <ul style="list-style-type: none"> <i>type</i>—Specifies the interface type. For more information, use the question mark (?) online help function. <i>interface-path-id</i>—Identifies a physical interface or a virtual interface. <p>Note Use the show interfaces command to see a list of all possible interfaces currently configured on the router.</p> <p>Note The loopback and null interfaces are always in the up state and, therefore, cannot be tracked.</p>
Step 5	<p>exit</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track-line-prot)# exit</pre>	Exits the track line protocol configuration mode.
Step 6	<p>(Optional) delay {up seconds down seconds}</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track)# delay up 10</pre>	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> end commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-track)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IP Route Reachability

When a host or a network goes down on a remote site, routing protocols notify the router and the routing table is updated accordingly. The routing process is configured to notify the tracking process when the route state changes due to a routing update.

A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type route reachability**
4. Use one of the following commands:
 - **vrf** *vrf-table-name*
 - **route ipv4** *IP-prefix/mask*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example:	Enters track configuration mode. <ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked.

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config)# track track1	Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type route reachability Example: RP/0/RSP0/CPU0:router(config-track)# type route reachability vrf internet	Configures the routing process to notify the tracking process when the state of the route changes due to a routing update.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • vrf <i>vrf-table-name</i> • route ipv4 <i>IP-prefix/mask</i> Example: RP/0/RSP0/CPU0:router(config-track-route)# vrf vrf-table-4 or RP/0/RSP0/CPU0:router(config-track-route)# route ipv4 10.56.8.10/16	Configures the type of IP route to be tracked, which can consist of either of the following, depending on your router type: <ul style="list-style-type: none"> • <i>vrf-table-name</i>—A VRF table name. • <i>IP-prefix/mask</i>—An IP prefix consisting of the network and subnet mask (for example, 10.56.8.10/16).
Step 5	exit Example: RP/0/RSP0/CPU0:router(config-track-line-prot)# exit	Exits the track line protocol configuration mode.
Step 6	(Optional) delay { up <i>seconds</i> down <i>seconds</i> } Example: RP/0/RSP0/CPU0:router(config-track)# delay up 10	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use the commit or end command.	commit —Saves the configuration changes, and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration mode, without committing the configuration changes.

Building a Track Based on a List of Objects

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a Boolean expression to determine the state of the list.

A tracked list contains one or more objects. The Boolean expression enables two types of calculations by using either AND or OR operators. For example, when tracking two interfaces, using the AND operator, up means that *both* interfaces are up, and down means that *either* interface is down.



Note An object must exist before it can be added to a tracked list.

The NOT operator is specified for one or more objects and negates the state of the object.

After configuring the tracked object, you must associate the interface whose state should be tracked and you may optionally specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list boolean { and | or }**
4. **object** *object-name* [**not**]
5. **exit**
6. (Optional) **delay {up seconds | down seconds}**
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example: RP/0/RSP0/CPU0:router(config)# track track1	Enters track configuration mode. • <i>track-name</i> —Specifies a name for the object to be tracked. Note Special characters are not allowed in a <i>track-name</i> .

	Command or Action	Purpose
Step 3	type list boolean { and or } Example: <pre>RP/0/RSP0/CPU0:router(config-track-list)# type list boolean and</pre>	Configures a Boolean list object and enters track list configuration mode. <ul style="list-style-type: none"> • boolean—Specifies that the state of the tracked list is based on a Boolean calculation. • and—Specifies that the list is up if all objects are up, or down if one or more objects are down. For example when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down. • or—Specifies that the list is up if at least one object is up. For example, when tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.
Step 4	object object-name [not] Example: <pre>RP/0/RSP0/CPU0:router(config-track-list)# object 3 not</pre>	Specifies the object to be tracked by the list <ul style="list-style-type: none"> • <i>object-name</i>—Name of the object to track. • not—Negates the state of the object.
Step 5	exit Example: <pre>RP/0/RSP0/CPU0:router(config-track-line-prot)# exit</pre>	Exits the track line protocol configuration mode.
Step 6	(Optional) delay {up seconds down seconds} Example: <pre>RP/0/RSP0/CPU0:router(config-track)# delay up 10</pre>	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre> or <pre>RP/0/RSP0/CPU0:router(config-track)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Percentage

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold percentage to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold percentage**
4. **object** *object-name*
5. **threshold percentage up percentage down percentage**
6. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example: RP/0/RSP0/CPU0:router(config)# track track1	Enters track configuration mode. <ul style="list-style-type: none"> <i>track-name</i>—Specifies a name for the object to be tracked. Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type list threshold percentage Example: RP/0/RSP0/CPU0:router(config-track-list)# type list threshold percentage	Configures a track of type threshold percentage list.

	Command or Action	Purpose
Step 4	object <i>object-name</i> Example: <pre>RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 1 RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 2 RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 3 RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 4</pre>	Configures object 1, object 2, object 3 and object 4 as members of track type track1.
Step 5	threshold <i>percentage up percentage down percentage</i> Example: <pre>RP/0/RSP0/CPU0:router(config-track-list-threshold)# threshold percentage up 50 down 33</pre>	<p>Configures the percentage of objects that need to be UP or DOWN for the list to be considered UP or Down respectively.</p> <p>For example, if object 1, object 2, and object 3 are in the UP state and object 4 is in the DOWN state, the list is considered to be in the UP state.</p>
Step 6	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-track)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Weight

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold weight to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold weight**
4. **object** *object-name* **weight** *weight*
5. **threshold** **weight up** *weight down* *weight*
6. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RSP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	track <i>track-name</i> Example: <pre>RP/0/RSP0/CPU0:router(config)# track track1</pre>	Enters track configuration mode. • <i>track-name</i> —Specifies a name for the object to be tracked. Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type list threshold weight Example: <pre>RP/0/RSP0/CPU0:router(config-track-list)# type list threshold weight</pre>	Configures a track of type, threshold weighted list.
Step 4	object <i>object-name</i> weight <i>weight</i> Example: <pre>RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 1 weight 10 RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 2 weight 5 RP/0/RSP0/CPU0:router(config-track-list-threshold)# object 3 weight 3</pre>	Configures object 1, object 2 and object 3 as members of track t1 and with weights 10, 5 and 3 respectively.
Step 5	threshold weight up <i>weight down</i> <i>weight</i> Example: <pre>RP/0/RSP0/CPU0:router(config-track-list-threshold)#</pre>	Configures the range of weights for the objects that need to be UP or DOWN for the list to be considered UP or DOWN respectively. In this example, the list is considered to be in the DOWN state because objects 1 and 2 are in the

	Command or Action	Purpose
	<code>threshold weight up 10 down 5</code>	UP state and the cumulative weight is 15 (not in the 10-5 range).
Step 6	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-track)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-track)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IPSLA Reachability

Use this task to enable the tracking of the return code of IP service level agreement (SLA) operations.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type rtr** *ipsla-no reachability*
4. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	<p>configure</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router# configure</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	track <i>track-name</i> Example: RP/0/RSP0/CPU0:router(config)# track t1	Enters track configuration mode. Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type rtr <i>ipsla-no</i> reachability Example: RP/0/RSP0/CPU0:router(config-track)# type rtr 100 reachability	Specifies the IP SLA operation ID to be tracked for reachability. Values for the <i>ipsla-no</i> can range from 1 to 2048.
Step 4	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Configuring IPSLA Tracking: Example

This example shows the configuration of IPSLA tracking:

```
RP/0/RSP0/CPU0:router(config)# track track1
RP/0/RSP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RSP0/CPU0:router(config-track)# delay up 5
RP/0/RSP0/CPU0:router(config-track)# delay down 10
```

Configure Enhanced Object Tracking

You can configure tracks with the **action** command to enable Enhanced Object Tracking. As a prerequisite, configure the track type that is to be tracked.

The following example shows how to configure the **action** command on a track based on the change in state of the track:

```
/* Configure track1 to track line-protocol state of the interface FourHundredGigE0/0/0/1
*/
Router#configure
Router(config)#track track1
Router(config-track)#type line-protocol state
Router(config-track-line-prot)#interface FourHundredGigE0/0/0/1
Router(config-track-line-prot)#exit
Router(config-track)#exit
```

```

/* Configure track2 to track line-protocol state of the interface FourHundredGigE0/1/0/1
*/
Router(config)#track track2
Router(config-track)#type line-protocol state
Router(config-track-line-prot)#interface FourHundredGigE0/1/0/1
Router(config-track-line-prot)#exit
Router(config-track)#exit

/* Configure track3 with boolean AND of track1 state and track2 state. Specify actions to
take when track3 state changes. */
Router(config)#track track3
Router(config-track)#type list boolean and
Router(config-track-list-boolean)#object track1
Router(config-track-list-boolean)#object track2
Router(config-track-line-boolean)#exit
Router(config-track)#action
Router(config-track-action)#track-down error-disable interface FourHundredGigE0/0/0/0
auto-recover
Router(config-track-action)#track-down error-disable interface FourHundredGigE0/1/0/0

```

The following running configuration example shows you how to configure the **action** command for the scenario described in Figure 1.

```

track track1
  type line-protocol state
  interface FourHundredGigE0/0/0/1
  !
  !
track track2
  type line-protocol state
  interface FourHundredGigE0/1/0/1
  !
  !
track track3
  type list boolean and
  object track1
  object track2
  !
  action
    track-down error-disable interface FourHundredGigE0/0/0/0 auto-recover
    track-down error-disable interface FourHundredGigE0/1/0/0

```

Verification

To view the state of the track, use the **show track** command.

Initially, let us assume the line-protocol state of FourHundredGigE0/0/0/1 (track1 interface) and FourHundredGigE0/1/0/1 (track2 interface) are up and HundredGigE0/0/0/35 (track4 interface) is down.

```

Router#show track
Track track3
  List boolean and is UP
  7 changes, last change 16:04:28 IST Mon Jul 02 2018
    object track2 UP
    object track1 UP
Track track1
  Interface FourHundredGigE0/0/0/1 line-protocol
  Line protocol is UP
  7 changes, last change 16:04:28 IST Mon Jul 02 2018
Track track2
  Interface FourHundredGigE0/1/0/1 line-protocol

```

```

Line protocol is UP
7 changes, last change 16:02:41 IST Mon Jul 02 2018

```

To verify if the interface configured for tracking is error-disabled, use the **show error-disable** command. As none of the track states match the track-action state, there are no error-disabled interfaces.

```

Router#show error-disable
Interface          Error-Disable reason          Retry (s)  Time disabled
-----
There are no interfaces error-disabled matching the given criteria

```

To view the status of all the interfaces of the tracked object, use the **show interface brief** command.

```

Router#show interface brief
Intf Name          Intf State LineP State  Encap Type  MTU (byte) BW (Kbps)
-----
FourHundredGigE0/0/0/0    up        up          ARPA        1514        100000000
FourHundredGigE0/0/0/1    up        up          ARPA        1514        100000000
FourHundredGigE0/1/0/0    up        up          ARPA        1514        100000000
FourHundredGigE0/1/0/1    up        up          ARPA        1514        100000000

```

When a track state changes, the corresponding track action happens and the status of the interfaces configured in the action changes. The state of track3 becomes "down" when either track1 state or track2 state becomes "down". The following **show error-disable** command displays the corresponding output.

```

Router#show error-disable
Interface          Error-Disable reason          Retry (s)  Time disabled
-----
FH0/0/0/0          ot-track-state-change        ---        08:42:08
FH0/1/0/0          ot-track-state-change        ---        08:42:01

```

When the state of track3 is "down", the **show interface brief** command displays the following output.

```

Router#show interface brief
Intf Name          Intf State  LineP State  Encap Type  MTU (byte) BW (Kbps)
-----
FourHundredGigE0/0/0/0    admin-down  admin-down  ARPA        1514        100000000
FourHundredGigE0/0/0/1    admin-down  admin-down  ARPA        1514        100000000
FourHundredGigE0/1/0/0    admin-down  admin-down  ARPA        1514        100000000
FourHundredGigE0/1/0/1    up          up          ARPA        1514        100000000

```

When track3 state comes back up, the error-disable status on the interface FourHundredGigE0/0/0/0 clears. This is because of the **auto-recover** configuration for FourHundredGigE0/0/0/0. However, interface FourHundredGigE0/1/0/0 remains in the error-disable status because **auto-recover** isn't configured on this interface.

The change reflects in the output of the **show interface brief** command.

```

RP/0/0/CPU0:ios#show interface brief
Intf Name          Intf State  LineP State  Encap Type  MTU (byte) BW (Kbps)
-----
FourHundredGigE0/0/0/0    up          up          ARPA        1514        100000000
FourHundredGigE0/0/0/1    up          up          ARPA        1514        100000000
FourHundredGigE0/1/0/0    err-disable admin-down  ARPA        1514        100000000
FourHundredGigE0/1/0/1    up          up          ARPA        1514        100000000

```

Configuration Examples for Configuring Object Tracking

Configuring IPSLA Tracking: Example

This example shows the configuration of IPSLA tracking, including the ACL and IPSLA configuration:

ACL configuration:

```
RP/0/RSP0/CPU0:router(config)# ipv4 access-list abf-track
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 permit any any nexthop track track1 1.2.3.4
```

Object tracking configuration:

```
RP/0/RSP0/CPU0:router(config)# track track1
RP/0/RSP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RSP0/CPU0:router(config-track)# delay up 5
RP/0/RSP0/CPU0:router(config-track)# delay down 10
```

IPSLA configuration:

```
RP/0/RSP0/CPU0:router(config)# ipsla
RP/0/RSP0/CPU0:router(config-ipsla)# operation 1
RP/0/RSP0/CPU0:router(config-ipsla-op)# type icmp echo
RP/0/RSP0/CPU0:router(config-ipsla-icmp-echo)# source address 2.3.4.5
RP/0/RSP0/CPU0:router(config-ipsla-icmp-echo)# destination address 1.2.3.4
RP/0/RSP0/CPU0:router(config-ipsla-icmp-echo)# frequency 60
RP/0/RSP0/CPU0:router(config-ipsla-icmp-echo)# exit
RP/0/RSP0/CPU0:router(config-ipsla-op)# exit
RP/0/RSP0/CPU0:router(config-ipsla)# schedule operation 1
RP/0/RSP0/CPU0:router(config-ipsla-sched)# start-time now
RP/0/RSP0/CPU0:router(config-ipsla-sched)# life forever
```

Additional References

The following sections provide references related to implementing object tracking for IPSec network security.

Related Documents

Related Topic	Document Title
IP SLA configuration information	<i>Implementing IP Service Level Agreements on the Cisco ASR 9000 Series Router module in System Monitoring Configuration Guide for Cisco ASR 9000 Series Routers</i>
IP SLA commands	<i>IP Service Level Agreement Commands on the Cisco ASR 9000 Series Router module in System Monitoring Command Reference for Cisco ASR 9000 Series Routers</i>

Related Topic	Document Title
Object tracking commands	<i>Object Tracking Commands on the Cisco ASR 9000 Series Router</i> module in <i>System Management Command Reference for Cisco ASR 9000 Series Routers</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC 2401	<i>Security Architecture for the Internet Protocol</i>

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 4

In Service Software Upgrade for Cisco IOS XR 64 Bit

This module contains the following topics:

- [Overview of ISSU on ASR 9000 with IOS XR 64 Bit](#) , on page 23
- [Restrictions and Usage Guidelines](#), on page 24
- [Pre-Installation Tasks](#), on page 25
- [ISSU on ASR 9000 with IOS XR 64 Bit: Single Phase Installation](#) , on page 28
- [ISSU on ASR 9000 with IOS XR 64 Bit: Multi Step Installation](#), on page 32
- [Installing Packages Using ISSU : Related Commands](#), on page 33

Overview of ISSU on ASR 9000 with IOS XR 64 Bit

In-Service Software Upgrade (ISSU) provides the ability to upgrade the IOS XR 64 Bit version on ASR 9000 with minimal disruption on the control plane and forwarding plane. ISSU supports upgrading an image from a lower to a higher version. ISSU supports zero topology loss (ZTL) and causes only a minimal packet loss of less than six seconds.

You can perform ISSU installation in a single step or as multiple phases. You need to perform the pre-installation tasks before executing ISSU. During the pre-installation tasks and ISSU execution, V1 refers to the image currently running on the router and V2 will be the upgraded image.

ISSU execution contains the following phases:

- **Prepare phase:** The installable files are pre-checked and loaded on the router before activation. This phase is optional.
- **Activate phase:** The new image (V2) is downloaded to all nodes in the router replacing the old image (V1). This phase can be run in step-by-step phases like *Load*, *Run*, and *Cleanup* or by using a one-shot *Activate* phase.



Note The *Prepare* phase is optional and can be skipped because the *Load* phase prepares the package if *Prepare* phase was not performed before the *Load* phase.

- **Commit phase:** The ISSU installation is complete with V2 on all nodes.

ISSU supports upgrading the System Admin VM and XR VM individually. Using ISSU, the System Admin VM and XR VM can also be upgraded sequentially. The upgrade sequence is System Admin ISSU followed by XR ISSU. Committing the upgrade from XR VM commits both the System Admin and XR software. But, committing the upgrade from System Admin VM commits only the System Admin software.



Note When RSP1 is the active RP and System Admin VM ISSU is triggered on Cisco ASR 9000 with IOS XR 64 bit, there is an additional VM switch over compared to performing System Admin VM ISSU from RSP0. This is an expected behaviour.

Restrictions and Usage Guidelines

- ISSU on ASR 9000 with IOS XR 64 Bit is supported only on few third generation ASR 9000 Series Ethernet line cards, RSPs and RPs.

The third generation of line cards include:

- A9K-8X100G-LB-SE
- A9K-8X100G-LB-TR
- A9K-8X100GE-SE
- A9K-8X100GE-TR
- A9K-4X100GE-SE
- A9K-4X100GE-TR
- A9K-MOD400-SE
- A9K-MOD400-TR
- A9K-MOD200-SE
- A9K-MOD200-TR
- A9K-4X100GE
- A99-12X100GE
- A99-12X100GE-CM
- A9K-24X10GE-1G-SE
- A9K-24X10GE-1G-TR
- A9K-48X10GE-1G-SE
- A9K-48X10GE-1G-TR
- A99-48X10GE-1G-SE
- A99-48X10GE-1G-TR

The third generation of RSP and RP cards include:

- A9K-RSP880-SE
 - A9K-RSP880-TR
 - A9K-RSP880-LT-SE
 - A9K-RSP880-LT-TR
 - A99-RSP-SE
 - A99-RSP-TR
 - A99-RP2-SE
 - A99-RP2-TR
- ISSU is not supported on the ASR 9000 fourth generation line cards and fifth generation line cards.
 - ISSU is not supported on the single RP system of ASR 9901, ASR 9902, and ASR 9903.

Pre-Installation Tasks

Pre-Installation Tasks

Before performing ISSU on ASR 9000 with IOS XR 64 Bit, complete the following tasks.

1. Configure NTP in XR VM. Once you configure NTP on XR VM, System Admin VM automatically synchronizes with NTP running on RSP. If NTP server is not available, configure clock on both XR VM and System Admin VM in configuration mode and make sure that your clock is set to the correct location and timezone.

```
RP/0/RSP0/CPU0:Router# show ntp associations
      address      ref clock      st  when  poll reach  delay  offset  disp
*~172.27.131.19    171.68.38.65      2   13    64   77    2.05  -1.174  191.09
~172.27.130.34    171.68.38.65      2   35    64    0    0.00   0.000  15937
+~172.27.130.33    171.68.38.65      2    9    64   77    2.41  10.370  189.47
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

```
sysadmin-vm:0_RSP0:Router# show ntp associations
Wed Oct 31 15:18:59.728 UTC-07:00
      remote      refid      st t when poll reach  delay  offset  jitter
=====
external:
*rsp0_xr          172.27.131.19    3 u  12  128  347    0.171   0.989   0.456
  rsp1_xr          .STEP.          16 u   -  256    0    0.000   0.000   0.000
internal:
*192.0.16.4        172.27.131.19    3 u  12  128  347    0.171   0.989   0.456
  192.0.20.4        .STEP.          16 u   -  256    0    0.000   0.000   0.000
```

2. Ensure that the dual RP and RSP systems are synchronized and they are in active and standby roles respectively using the **show redundancy summary** command. The line card status should be *Final Band* or *Running*.

```
RP/0/RSP0/CPU0:Router# show redundancy summary

Active Node      Standby Node
```

```

-----
0/RSP0/CPU0      0/RSP1/CPU0 (Node Ready, NSR:Ready)

RP/0/RSP0/CPU0:Router# show platform vm
Node name      Node type      Partner name    SW status      IP address
-----
0/3/CPU0       LC (ACTIVE)     NONE           FINAL Band     192.0.12.3
0/7/CPU0       LC (ACTIVE)     NONE           FINAL Band     192.0.36.3
0/RSP1/CPU0    RP (STANDBY)    0/RSP0/CPU0    FINAL Band     192.0.20.4
0/4/CPU0       LC (ACTIVE)     NONE           FINAL Band     192.0.24.3
0/RSP0/CPU0    RP (ACTIVE)     0/RSP1/CPU0    FINAL Band     192.0.16.4

```

3. Ensure that firmware on linecards, RSP, and RP is upgraded to the latest version. You can upgrade the router cards in a single step by using the **upgrade hw-module location all fpd all** command. Use the **show hw-module location location fpd** command to verify the firmware versions.

```
RP/0/RSP0/CPU0:Router# upgrade hw-module location all fpd all
```

This example shows verifying the firmware versions for a node.

```
RP/0/RSP0/CPU0:Router# show hw-module location 0/rsp1 fpd
```

Location	Card type	HWver	FPD device	ATR Status	FPD Versions	
					Running	Programd
0/RSP1	A9K-RSP880-SE	1.0	Alpha-FPGA	CURRENT	0.16	0.16
0/RSP1	A9K-RSP880-SE	1.0	CBC	CURRENT	34.39	34.39
0/RSP1	A9K-RSP880-SE	1.0	Cha-FPGA	CURRENT	0.08	0.08
0/RSP1	A9K-RSP880-SE	1.0	IPU-FPGA	CURRENT	0.66	0.66
0/RSP1	A9K-RSP880-SE	1.0	IPU-FSBL	CURRENT	1.108	1.108
0/RSP1	A9K-RSP880-SE	1.0	IPU-Linux	CURRENT	1.108	1.108
0/RSP1	A9K-RSP880-SE	1.0	Omega-FPGA	CURRENT	0.16	0.16
0/RSP1	A9K-RSP880-SE	1.0	Optimus-FPGA	CURRENT	0.12	0.12
0/RSP1	A9K-RSP880-SE	1.0	Primary-BIOS	CURRENT	10.60	10.60
0/RSP1	A9K-RSP880-SE	1.0	SSDa-SMART	N/A	7.05	7.05
0/RSP1	A9K-RSP880-SE	1.0	SSDb-SMART	N/A	7.05	7.05

4. Check the disk storage space on both System Admin VM and XR VM and ensure that sufficient disk space is available. Remove files like show-tech, cores, kernel dumps, manually created text, log, debug information and so on.

This example shows verifying the disk storage space for System Admin VM and XR VM on RSP0. You also need to verify the disk space on the standby RSP (RSP1). If required you can verify the disk storage on line cards using the **show media location** command.

```
RP/0/RSP0/CPU0:Router# show media
```

Media Information for local node.

```

-----
Partition      Size      Used      Percent    Avail
rootfs:        3.8G      1.3G      34%        2.4G
/run           14G       340K      1%         14G
harddisk:      5.5G      1.6G      31%        3.6G
/run/netns     14G       340K      1%         14G
log:           469M      33M       8%         401M
config:        469M      1.7M      1%         432M
disk0:         968M      5.3M      1%         897M
harddiska:     3.6G      11M       1%         3.5G
/misc/app_host 2.4G      61M       3%         2.2G
-----

```

```

rootfs: = root file system (read-only)
log:    = system log files (read-only)
config: = configuration storage (read-only)

```

```
RP/0/RSP0/CPU0:Router# admin show media
```

```
-----
Partition                Size      Used  Percent   Avail
-----
rootfs:                   2.4G     838M    38%     1.4G
harddisk:                 7.6G     1.2G    17%     6.0G
log:                      469M      22M     5%     412M
config:                   469M      3.3M     1%     431M
disk0:                    968M      1.6M     1%     900M
a9ksys:                   736M      4.3M     1%     695M
harddiskb:                3.0G      1.2G    41%     1.8G
install:                  4.8G     932M    21%     3.6G
install:/tmp              7.6G      1.2G    17%     6.0G
install:/cache            7.6G      1.2G    17%     6.0G
rootfs:/install/tmp       7.6G      1.2G    17%     6.0G
-----

rootfs: = root file system (read-only)
log:    = system log files (read-only)
config: = configuration storage (read-only)
install: = install repository (read-only)
a9ksys: = ASR9K system storage (read-only)
```

5. Populate the repository with RPMs and SMUs. You can pick and install individual RPMs, SMUs, one by one, or make a tarball and install one tarball or break it down with multiple tarballs.



Note You cannot include a tarball within another tarball. However, multiple tarballs can be specified at once.

```
RP/0/RSP0/CPU0:Router# install add source tftp://172.27.131.19/xrimages/e652/
asr9k-mini-x64-6.5.2.13I.iso ASR9K-RPMS-65213I.tar

RP/0/RSP0/CPU0:Router# show install request
The install add operation 4 is 80% complete
RP/0/RSP0/CPU0:Router#

RP/0/RSP0/CPU0:Router#
RP/0/RSP0/CPU0:Oct 31 19:44:34.076 : sdr_instmgr[1156]:
%INSTALL-INSTMGR-2-OPERATION_SUCCESS : Install operation 4 finished successfully
Oct 31 19:44:35 Install operation 4 finished successfully
```

6. Check the repository to validate that packages, images, or SMUs are populated properly in the router's repository by using the **show install repository** command. There should be a one to one relationship between V1 and V2 images and SMUs. For example, if you install a SMU on V1, you also need the corresponding V2 version in the repository to execute ISSU.

```
RP/0/RSP0/CPU0:Router# show install repository | i mini
asr9k-mini-x64-6.2.3          <--V1 iso image currently running
asr9k-mini-x64-6.5.2.13I     <--V2 iso image to upgrade to
```

7. Extract the ISO image in System Admin VM or XR VM depending on the version of the image.
 - For IOS XR versions prior to 6.5.1: You should extract the ISO image in XR VM as well as in System Admin VM because the system can only use packages in RPM format.
 - For IOS XR version 6.5.1 and onwards: System automatically extracts the system admin package from the ISO file once you mentioned the file name while executing ISSU. You only need to extract the XR package separately.

```
RP/0/RSP0/CPU0:Router# install extract asr9k-mini-x64-6.5.2.13I
Oct 31 20:50:30 Install operation 9 started by root:
```

```

install extract asr9k-mini-x64-6.5.2.13I
Oct 31 20:50:30 Package list:
Oct 31 20:50:30 asr9k-mini-x64-6.5.2.13I
Oct 31 20:50:31 Install operation will continue in the background

RP/0/RSP0/CPU0:Router# show install repository | i xr-

asr9k-xr-6.5.2.13I                                <--V2 extracted image to upgrade to
asr9k-xr-6.2.3

sysadmin-vm:0_RSP0:Router# install extract asr9k-mini-x64-6.5.2.13I
result Wed Oct 31 20:36:34 2018 Install operation 1 (install extract) started by user
'root' will continue asynchronously.
sysadmin-vm:0_RSP0:MYISSU#

sysadmin-vm:0_RSP0:Router# show install repository | i sys
asr9k-sysadmin-6.2.3
asr9k-sysadmin-6.5.2.13I                            <--V2 extracted image to upgrade to

```

ISSU on ASR 9000 with IOS XR 64 Bit: Single Phase Installation

This section shows how to perform ISSU on ASR 9000 with IOS XR 64 Bit in a single step. You can either upgrade the system or install a patch in a single step. The system upgrade is done using an ISO image file, while the patch installation is done using packages and SMUs.

You should perform the following steps before performing this task:

- Copy the package to be installed either on the router's hard disk or on a network server to which the router has access.
- Ensure that dual route processor (RP) system with standby is in "is ready" state.

Perform the following steps to upgrade the system or install a patch in a single step.



Note Depending on whether you are installing a System Admin package or a XR package, execute these commands in the System Admin EXEC mode or XR EXEC mode respectively

1. (Optional) Prepare the installable files by using the **install prepare issu** *package_name* command. During the prepare phase, pre-activation checks are made, and the components of the installable files are loaded on to the router setup.

For System Admin VM:

```
sysadmin-vm:0_RSP0# install prepare issu asr9k-xr-<release-version>
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install prepare issu asr9k-xr-<release-version>
```

2. Activate the ISSU installation in XR VM or System Admin VM in a single step by using the **install activate issu** command.

For System Admin VM:

```
sysadmin-vm:0_RSP0# install activate issu asr9k-xr-<release-version>
```

For XR VM:


```
RP/0/RSP0/CPU0:router# install activate issu asr9k-xr-<release-version>
```



Note ISSU operation takes about 30 minutes to complete. If the ISSU operation is not concluded in 40 minutes, the ISSU may timeout or abort.

3. Commit the newly active software by using the **install commit** command.

For System Admin VM:

```
sysadmin-vm:0_RSP1# install commit
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install commit
```

Examples: Install Packages Using ISSU Single Step Installation on ASR 9000 with IOS XR 64 Bit

This example shows performing System Admin VM upgrade using ISSU and how to verify the installation using show commands.

!# Verify packages in the repository

```
sysadmin-vm:0_RSP0:Router# show install repository all | i "host|sysadmin"
asr9k-sysadmin-6.2.3
  asr9k-sysadmin-6.5.2.13I
  host-6.2.3
  host-6.5.2.13I
sysadmin-vm:0_RSP0:Router#
```

!# Performing ISSU installation

```
sysadmin-vm:0_RSP0:Router# install activate issu asr9k-sysadmin-6.5.2.13I host-6.5.2.13I
This install operation will result in admin VMs reload
Do you want to proceed [yes/no]: yes
Proceeding with operation
result Wed Oct 31 21:12:21 2018 Install operation 2 (install prepare and activate issu)
started by user 'root' will continue asynchronously.
sysadmin-vm:0_RSP0:Router#
```

!# Monitoring the progress of the installation.
!# The installation may take up to 30 minutes.

```
sysadmin-vm:0_RSP0:Router# show install request
```

User root, Op Id 2

```
install prepare issu
```

```
host-6.5.2.13I
```

This operation is 40% complete

Waiting for agents to complete host prepare ..

```
sysadmin-vm:0_RSP0:#
```

!# Verifying the installation status again after few minutes

```

sysadmin-vm:0_RSP0:Router# show install request
User root, Op Id 2
install activate issu
ISSU stage Phase1
asr9k-sysadmin-6.5.2.13I
Node 0/RSP0 [RP] : 90% of current state is completed
Node 0/RSP1 [RP] : 90% of current state is completed
Node 0/1 [LC] : 90% of current state is completed
Node 0/3 [LC] : 90% of current state is completed
Node 0/4 [LC] : 90% of current state is completed
Node 0/7 [LC] : 90% of current state is completed

sysadmin-vm:0_RSP0:MYISSU#

!# Message after successful completion. Admin VM will reload after this message. . There
should be no packet drop.

0/RSP0/ADMIN0:Oct 31 21:27:53.260 : inst_mgr[5019]: %INFRA-INSTMGR-2-OPERATION_SUCCESS :
Install operation 2 completed successfully

!# Verifying the active package

sysadmin-vm:0_RSP1# show install active summary
Active Packages: 1
asr9k-sysadmin-6.5.2.13I version=6.5.2.13I [Boot image]

!# Verifies the image previously committed

sysadmin-vm:0_RSP1# show install commit summary
Committed Packages: 1
asr9k-sysadmin-6.2.3 version=6.2.3 [Boot image]

!# Commits the latest image

sysadmin-vm:0_RSP1# install commit
result Wed Oct 31 21:32:58 2018 Install operation 3 (install commit) started by user 'root'
will continue asynchronously.
sysadmin-vm:0_RSP1# 0/RSP1/ADMIN0:Oct 31 21:33:02.061 : inst_mgr[6913]:
%INFRA-INSTMGR-2-OPERATION_SUCCESS : Install operation 3 completed successfully
Wed Oct 31 21:33:02 2018 Install operation 3 completed successfully.
sysadmin-vm:0_RSP1#

```

This example shows performing XR VM upgrade using ISSU and verifying the installation using show commands.

```

!# Verify the active packages

RP/0/RSP0/CPU0:Router# show install active summary
Active Packages: 8
asr9k-xr-6.2.3 version=6.2.3 [Boot image]
asr9k-isis-x64-1.3.0.0-r623
asr9k-mpls-x64-2.0.0.0-r623
asr9k-mpls-te-rsvp-x64-1.3.0.0-r623
asr9k-ospf-x64-1.0.0.0-r623
asr9k-mgbl-x64-2.0.0.0-r623
asr9k-mcast-x64-2.2.0.0-r623
asr9k-k9sec-x64-3.2.0.0-r623

!# Performing ISSU Installation

RP/0/RSP0/CPU0:Router# install activate issu asr9k-xr-6.5.2.13I *r65213I
Oct 31 21:48:14 Install operation 10 started by root:

```

```

install activate issu asr9k-xr-6.5.2.13I *r65213I
Oct 31 21:48:14 Package list:
Oct 31 21:48:14   asr9k-isis-x64-1.1.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-ospf-x64-1.0.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-mcast-x64-2.0.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-mppls-te-rsvp-x64-2.1.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-mgbl-x64-2.0.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-k9sec-x64-2.1.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-mppls-x64-2.0.0.0-r65213I.x86_64
Oct 31 21:48:14   asr9k-xr-6.5.2.13I
This install operation will start the issu, continue?
[yes/no]:[yes] yes
Oct 31 21:49:13 Install operation will continue in the background
RP/0/RSP0/CPU0:Router#

!# Monitoring the progress of the installation.
!# The installation may take up to 30 minutes.

RP/0/RSP1/CPU0:Router# show issu
INSTALL Operation ID   : Operation 11 Started at Wed Oct 31 22:23:30 2018
ISSU Progress          : 100.0%
Total ISSU Time        : 00:25:07
ISSU Type              : SMU

Phase                  Start-Time      End-Time      State
-----
Prepare                22:23:30      22:34:21      Completed
Load                   22:34:28      22:47:01      Completed
Run                    22:47:01      22:47:44      Completed
Cleanup                22:47:44      22:48:44      Completed
-----
Current Status         : ISSU Orchestration Successfully Completed

Setup Information       : Single Chassis
ISSU Ready/Not Ready   : 0 / 0

Node ISSU readiness per rack per slot
Key: Ready - 'Y', Not ready - 'N', Primary node - '*', Complete - '-'

Rack 0  RP0  RP1  LC1  LC3  LC4  LC7
      -   -   -   -   -   -   -

!# Verifying the VM status after the installation

RP/0/RSP1/CPU0:Router# admin show sdr

SDR: default-sdr
Location      IP Address      Status      Boot Count  Time Started
-----
0/1/VM2       192.0.4.4        RUNNING      1           10/31/2018 22:34:55
0/3/VM2       192.0.12.4       RUNNING      1           10/31/2018 22:34:54
0/RSP0/VM2    192.0.16.6       RUNNING      1           10/31/2018 22:49:58
0/RSP1/VM2    192.0.20.6       RUNNING      1           10/31/2018 22:35:39
0/4/VM2       192.0.24.4       RUNNING      1           10/31/2018 22:34:55
0/7/VM2       192.0.36.4       RUNNING      1           10/31/2018 22:34:55

!# Verify the active packages

RP/0/RSP1/CPU0:Router# show install active sum
Active Packages: 8
  asr9k-xr-6.5.2.13I version=6.5.2.13I [Boot image]
  asr9k-isis-x64-1.1.0.0-r65213I
  asr9k-ospf-x64-1.0.0.0-r65213I
  asr9k-mcast-x64-2.0.0.0-r65213I

```

```

asr9k-mpls-te-rsvp-x64-2.1.0.0-r65213I
asr9k-mgbl-x64-2.0.0.0-r65213I
asr9k-k9sec-x64-2.1.0.0-r65213I
asr9k-mpls-x64-2.0.0.0-r65213I

```

```

!# You can either perform install commit and stay with the latest image or reload the router
to continue using the old image.

```

```

!# Commits the latest image after the necessary checks
RP/0/RSP0/CPU0:Router# install commit

```

ISSU on ASR 9000 with IOS XR 64 Bit: Multi Step Installation

This section shows how to perform ISSU on ASR 9000 with IOS XR 64 Bit in multiple steps.

You should perform the following steps before performing the steps in this task:

- Copy the package to be installed either on the router's hard disk or on a network server to which the router has access.
- Ensure that dual route processor (RP) system with standby is in "is ready" state.

Perform the following steps to upgrade the system or install a patch in multiple phases.



Note

Depending on whether you are installing a System Admin package or a XR package, execute these commands in the System Admin EXEC mode or XR EXEC mode respectively.



Note

You should update the System Admin VM first and then update the XR VM. IOS XR 64 bit ISSU will fail if the System Admin VM is not updated first.

1. (Optional) Prepare the installable files by using the **install prepare issu** *package_name* command. During the prepare phase, pre-activation checks are performed and the components of the installable files are loaded on to the router setup.

For System Admin VM :

```
sysadmin-vm:0_RSP0# install prepare issu asr9k-xr-<release-version>
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install prepare issu asr9k-xr-<release-version>
```

2. Start the load phase by issuing the **install activate issu load** command.

For System Admin VM :

```
sysadmin-vm:0_RSP0# install activate issu load asr9k-xr-<release-version>
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install activate issu load asr9k-xr-<release-version>
```

This step downloads the new image (V2) to all nodes in the router. The new image is checked for compatibility to ensure that the router can be upgraded. At the start of the *Load* phase, the router

configuration mode is locked, and you cannot perform any configuration on the router until ISSU completes the phase. At the end of this stage, all standby nodes run V2 and all active nodes (including all line cards) still run the original software images (V1).

- Starts the run phase by issuing the **install activate issu run** command.

For System Admin VM :

```
sysadmin-vm:0_RSP0# install activate issu run
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install activate issu run
```

This phase starts version switch from V1 to V2. All the packages that have been prepared are activated to make the package configurations active on the router.

- Starts the cleanup phase by issuing the **install activate issu cleanup** command.

For System Admin VM :

```
sysadmin-vm:0_RSP0# install activate issu cleanup
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install activate issu cleanup
```

This phase Initiates shutdown of VMs with previous versions after running the activation. The cleanup phase concludes the ISSU process and the new software runs on all nodes in the system.

- Commit the newly active software by using the **install commit** command.

For System Admin VM:

```
sysadmin-vm:0_RSP0# install commit
```

For XR VM:

```
RP/0/RSP0/CPU0:router# install commit
```

Installing Packages Using ISSU : Related Commands

Related Commands	Purpose
show install active	Displays the active packages on the system.
show install request	Displays the progress of the ISSU installation.
show issu	Displays the state or status of the ISSU operation. Effective with Cisco IOS XR version 6.5.1, this command is also supported for System Admin VM ISSU.
install prepare clean	Clears the existing prepared image. If there is a failure in the prepare phase, you can run this command to clear the prepared image.

Related Commands	Purpose
install activate ISSU abort	Initiates ISSU abort in XR VM. ISSU aborts if the command is run before ISSU Run phase starts. All the changes due to the install activity are reset.



CHAPTER 5

Configuring Network Time Protocol

Network Time Protocol (NTP) is a protocol designed to time-synchronize devices within a network. Cisco IOS XR software implements NTPv4. NTPv4 retains backwards compatibility with the older versions of NTP, including NTPv3 and NTPv2 but excluding NTPv1, which has been discontinued due to security vulnerabilities.

This module describes the tasks you need to implement NTP on the Cisco IOS XR software.

For more information about NTP on the Cisco IOS XR software and complete descriptions of the NTP commands listed in this module, see [Related Documents, on page 61](#). To locate documentation for other commands that might appear in the course of running a configuration task, search online in *Cisco ASR 9000 Series Aggregation Services Router Commands Master List*.

Table 3: Feature History for Implementing NTP on Cisco IOS XR Software

Release	Modification
Release 3.7.2	This feature was introduced.
Release 3.9.0	Support was added for IPv6 addresses, VRFs, multicast-based associations, and burst and iburst modes for poll-based associations.
Release 4.3.0	Support was added for NTP-PTP interworking.
Release 4.3.1	Support was added for NTP server inside VRF interface

This module contains the following topics:

- [Prerequisites for Implementing NTP on Cisco IOS XR Software, on page 36](#)
- [Information About Implementing NTP, on page 36](#)
- [How to Implement NTP, on page 38](#)
- [Configuration Examples for Implementing NTP, on page 55](#)
- [FQDN for NTP Server, on page 58](#)
- [Configuring NTP server inside VRF interface, on page 60](#)
- [Additional References, on page 61](#)

Prerequisites for Implementing NTP on Cisco IOS XR Software

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Implementing NTP

NTP synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time-specific events occur.

NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communication uses Coordinated Universal Time (UTC). An NTP network usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each other.

NTP uses the concept of a “stratum” to describe how many NTP “hops” away a machine is from an authoritative time source. A “stratum 1” time server typically has an authoritative time source (such as a radio or atomic clock, or a GPS time source) directly attached, a “stratum 2” time server receives its time via NTP from a “stratum 1” time server, and so on.

NTP avoids synchronizing to a machine whose time may not be accurate, in two ways. First, NTP never synchronizes to a machine that is not synchronized itself. Second, NTP compares the time reported by several machines and does not synchronize to a machine whose time is significantly different than the others, even if its stratum is lower. This strategy effectively builds a self-organizing tree of NTP servers.

The Cisco implementation of NTP does not support stratum 1 service; in other words, it is not possible to connect to a radio or atomic clock (for some specific platforms, however, you can connect a GPS time-source device). We recommend that time service for your network be derived from the public NTP servers available in the IP Internet.

If the network is isolated from the Internet, the Cisco implementation of NTP allows a machine to be configured so that it acts as though it is synchronized via NTP, when in fact it has determined the time using other means. Other machines can then synchronize to that machine via NTP.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software also allows UNIX-derivative servers to acquire the time directly from an atomic clock, which would subsequently propagate time information along to Cisco routers.

The communications between machines running NTP (known as *associations*) are usually statically configured; each machine is given the IP address of all machines with which it should form associations. Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines with an association.

The Cisco implementation of NTP supports two ways that a networking device can obtain NTP time information on a network:

- By polling host servers
- By listening to NTP broadcasts

In a LAN environment, NTP can be configured to use IP broadcast messages. As compared to polling, IP broadcast messages reduce configuration complexity, because each machine can simply be configured to send or receive broadcast or multicast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

An NTP broadcast client listens for broadcast messages sent by an NTP broadcast server at a designated IPv4 address. The client synchronizes the local clock using the first received broadcast message.

The time kept on a machine is a critical resource, so we strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

When multiple sources of time (VINES, hardware clock, manual configuration) are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.



Note NTP associations will not be formed if the packets received are from a VRF which is different from the VRF that is configured for the NTP server or peer.

Preventing Issues due to GPS Week Number Rollover (WNRO)

- If there are no GPS sources in the NTP source chain or server chain, there is no impact of GPS Week Number Rollover (WNRO).
- GPS WNRO affects only the system clock and not user traffic.
- Contact your GPS manufacturer to fix the GPS source for this condition.

To mitigate impact of GPS sources that are subject to GPS WNRO perform the following optional workarounds:

- If the GPS source has been identified to be a cause of potential disruption on April 6, 2019 (or after), configure `ntp master` in the Cisco that is device connected to this source, and its clock on the Stratum 1 device to preventively isolate it. This configuration enables the device to present its own clock for synchronization to downstream NTP clients.



Note The usage of `ntp master` command as mentioned above is only a workaround to this condition. Use this command until the GPS source-related conditions are resolved, and to prevent the distribution of incorrect clock values throughout the network.

- Configure multiple NTP servers (ideally 4, but more than 3) at Stratum 2 level of the network, to enable NTP clients at Stratum 2 level to get clock from more than one Stratum 1 server. This way, WNRO affected Stratum 1 servers are staged to be marked as ‘false ticker’ or ‘outlier’ clock sources as compared to other non-WNRO affected Stratum 1 servers.

NTP-PTP Interworking

NTP-PTP interworking provides the ability to use PTP, as well as other valid time of day (TOD) sources such as Data over Cable Service Interface Specification (DOCSIS) Timing Interface (DTI) and global positioning

system (GPS), as the time source for the operating system. Prior to the support of NTP-PTP interworking, only backplane time was supported for the operating system time.

NTP-PTP interworking also provides the means to communicate status changes between PTP and NTP processes. It also supports the unambiguous control of the operating system time and backplane time in the event of bootstrap, switchovers or card and process failures.

Related Topics

[Configuring NTP-PTP Interworking](#), on page 51

How to Implement NTP

Configuring Poll-Based Associations



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

You can configure the following types of poll-based associations between the router and other devices (which may also be routers):

- Client mode
- Symmetric active mode

The client and the symmetric active modes should be used when NTP is required to provide a high level of time accuracy and reliability.

When a networking device is operating in the client mode, it polls its assigned time serving hosts for the current time. The networking device then picks a host from all the polled time servers to synchronize with. Because the relationship that is established in this case is a client-host relationship, the host does not capture or use any time information sent by the local client device. This mode is most suited for file-server and workstation clients that are not required to provide any form of time synchronization to other local clients. Use the **server** command to individually specify the time-serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the client mode.

When a networking device is operating in the symmetric active mode, it polls its assigned time-serving hosts for the current time and it responds to polls by its hosts. Because this is a peer-to-peer relationship, the host also retains time-related information about the local networking device that it is communicating with. This mode should be used when there are several mutually redundant servers that are interconnected via diverse network paths. Most stratum 1 and stratum 2 servers on the Internet today adopt this form of network setup. Use the **peer** command to individually specify the time-serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the symmetric active mode.

When the router polls several other devices for the time, the router selects one device with which to synchronize.



Note To configure a peer-to-peer association between the router and another device, you must also configure the router as a peer on the other device.

You can configure multiple peers and servers, but you cannot configure a single IP address as both a peer and a server at the same time.

To change the configuration of a specific IP address from peer to server or from server to peer, use the **no** form of the **peer** or **server** command to remove the current configuration before you perform the new configuration. If you do not remove the old configuration before performing the new configuration, the new configuration does not overwrite the old configuration.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **server** *ip-address* [**version number**] [**key key-id**] [**minpoll interval**] [**maxpoll interval**] [**source type interface-path-id**] [**prefer**] [**burst**] [**iburst**]
4. **peer** *ip-address* [**version number**] [**key key-id**] [**minpoll interval**] [**maxpoll interval**] [**source type interface-path-id**] [**prefer**]
5. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	server <i>ip-address</i> [version number] [key key-id] [minpoll interval] [maxpoll interval] [source type interface-path-id] [prefer] [burst] [iburst] Example: RP/0/RSP0/CPU0:router(config-ntp)# server 172.16.22.44 minpoll 8 maxpoll 12	Forms a server association with another system. This step can be repeated as necessary to form associations with multiple devices.

	Command or Action	Purpose
Step 4	<p>peer <i>ip-address</i> [version number] [key key-id] [minpoll interval] [maxpoll interval] [source type interface-path-id] [prefer]</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# peer 192.168.22.33 minpoll 8 maxpoll 12 source tengige 0/0/0/1</pre>	<p>Forms a peer association with another system. This step can be repeated as necessary to form associations with multiple systems.</p> <p>Note To complete the configuration of a peer-to-peer association between the router and the remote device, the router must also be configured as a peer on the remote device.</p>
Step 5	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Broadcast-Based NTP Associates

In a broadcast-based NTP association, an NTP server propagates NTP broadcast packets throughout a network. Broadcast clients listen for the NTP broadcast packets propagated by the NTP server and do not engage in any polling.

Broadcast-based NTP associations should be used when time accuracy and reliability requirements are modest and if your network is localized and has a large number of clients (more than 20). Broadcast-based NTP associations also are recommended for use on networks that have limited bandwidth, system memory, or CPU resources. Time accuracy is marginally reduced in broadcast-based NTP associations because information flows only one way.

Use the **broadcast client** command to set your networking device to listen for NTP broadcast packets propagated through a network. For broadcast client mode to work, the broadcast server and its clients must be located on the same subnet. The time server that is transmitting NTP broadcast packets must be enabled on the interface of the given device using the **broadcast** command.

Use the **broadcast** command to set your networking device to send NTP broadcast packets.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.



Note If you enable NTP broadcast on the physical interface, subinterface or bundle interface, then it breaks the inter-VRF Poll-Based association between client and server. As these interfaces also handle NTP unicast traffic, the interface designated as broadcast, rejects service unicast clients on it. So, NTP broadcast and NTP unicast are not allowed on the same interface.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. (Optional) **broadcastdelay** *microseconds*
4. **interface** *type interface-path-id*
5. **broadcast client**
6. **broadcast** [**destination** *ip-address*] [**key** *key-id*] [**version** *number*]
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	(Optional) broadcastdelay <i>microseconds</i> Example: RP/0/RSP0/CPU0:router(config-ntp)# broadcastdelay 5000	Adjusts the estimated round-trip delay for NTP broadcasts.

	Command or Action	Purpose
Step 4	interface <i>type interface-path-id</i> Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# interface POS 0/1/0/0</pre>	Enters NTP interface configuration mode.
Step 5	broadcast client Example: <pre>RP/0/RSP0/CPU0:router(config-ntp-int)# broadcast client</pre>	Configures the specified interface to receive NTP broadcast packets. Note Go to next step to configure the interface to send NTP broadcast packets.
Step 6	broadcast [<i>destination ip-address</i>] [<i>key key-id</i>] [<i>version number</i>] Example: <pre>RP/0/RSP0/CPU0:router(config-ntp-int)# broadcast destination 10.50.32.149</pre>	Configures the specified interface to send NTP broadcast packets. Note Go to previous step to configure the interface to receive NTP broadcast packets.
Step 7	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RSP0/CPU0:router(config-ntp-int)# end</pre> or <pre>RP/0/RSP0/CPU0:router(config-ntp-int)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP Access Groups



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

The access list-based restriction scheme allows you to grant or deny certain access privileges to an entire network, a subnet within a network, or a host within a subnet.

The access group options are scanned in the following order, from least restrictive to most restrictive:

1. **peer**—Allows time requests and NTP control queries and allows the system to synchronize itself to a system whose address passes the access list criteria.
2. **serve**—Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.
3. **serve-only**—Allows only time requests from a system whose address passes the access list criteria.
4. **query-only**—Allows only NTP control queries from a system whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted. If no access groups are specified, all access types are granted to all systems. If any access groups are specified, only the specified access types are granted.

For details on NTP control queries, see RFC 1305 (NTP version 3).

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **access-group** {**peer** | **query-only** | **serve** | **serve-only**} *access-list-name*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# <code>configure</code>	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# <code>ntp</code>	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	access-group { peer query-only serve serve-only } <i>access-list-name</i> Example: RP/0/RSP0/CPU0:router(config-ntp)# access-group peer access1	Creates an access group and applies a basic IPv4 or IPv6 access list to it.
Step 4	Use one of the following commands: <ul style="list-style-type: none">• end• commit Example: RP/0/RSP0/CPU0:router(config-ntp)# end or RP/0/RSP0/CPU0:router(config-ntp)# commit	Saves configuration changes. <ul style="list-style-type: none">• When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: <ul style="list-style-type: none">• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP Authentication

This task explains how to configure NTP authentication.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

The encrypted NTP authentication scheme should be used when a reliable form of access control is required. Unlike the access-list-based restriction scheme that is based on IP addresses, the encrypted authentication scheme uses authentication keys and an authentication process to determine if NTP synchronization packets sent by designated peers or servers on a local network are deemed as trusted, before the time information that it carries along is accepted.

The authentication process begins from the moment an NTP packet is created. A message authentication code (MAC) is computed using the MD5 Message Digest Algorithm and the MAC is embedded into an NTP synchronization packet. The NTP synchronization packet together with the embedded MAC and key number are transmitted to the receiving client. If authentication is enabled and the key is trusted, the receiving client

computes the MAC in the same way. If the computed MAC matches the embedded MAC, the system is allowed to sync to the server that uses this key in its packets.

After NTP authentication is properly configured, your networking device only synchronizes with and provides synchronization to trusted time sources.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **authenticate**
4. **authentication-key** *key-number* **md5** [**clear** | **encrypted**] *key-name*
5. **trusted-key** *key-number*
6. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	authenticate Example: RP/0/RSP0/CPU0:router(config-ntp)# authenticate	Enables the NTP authentication feature.
Step 4	authentication-key <i>key-number</i> md5 [clear encrypted] <i>key-name</i> Example: RP/0/RSP0/CPU0:router(config-ntp)# authentication-key 42 md5 clear key1	Defines the authentication keys. <ul style="list-style-type: none"> • Each key has a key number, a type, a value, and, optionally, a name. Starting from Cisco IOS-XR Release 7.5.1 the following authentication types are supported: <ul style="list-style-type: none"> • cmac CMAC authentication • md5 MD5 authentication • sha1 SHA1 authentication

	Command or Action	Purpose
		<ul style="list-style-type: none"> • sha2 SHA2 authentication
Step 5	trusted-key <i>key-number</i> Example: RP/0/RSP0/CPU0:router(config-ntp)# trusted-key 42	Defines trusted authentication keys. <ul style="list-style-type: none"> • If a key is trusted, this router only synchronizes to a system that uses this key in its NTP packets.
Step 6	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: RP/0/RSP0/CPU0:router(config-ntp)# end or RP/0/RSP0/CPU0:router(config-ntp)# commit	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Disabling NTP Services on a Specific Interface

NTP services are disabled on all interfaces by default.

NTP is enabled globally when any NTP commands are entered. You can selectively prevent NTP packets from being received through a specific interface by turning off NTP on a given interface.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. Use one of the following commands:
 - **no interface** *type interface-path-id*
 - **interface** *type interface-path-id* **disable**
4. Use one of the following commands:

- **end**
- **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RSP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	ntp Example: <pre>RP/0/RSP0/CPU0:router(config)# ntp</pre>	Enters NTP configuration mode.
Step 3	Use one of the following commands: <ul style="list-style-type: none"> • no interface <i>type interface-path-id</i> • interface <i>type interface-path-id</i> disable Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# no interface pos 0/0/0/1</pre> or <pre>RP/0/RSP0/CPU0:router(config-ntp)# interface POS 0/0/0/1 disable</pre>	Disables NTP services on the specified interface.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> or <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Source IP Address for NTP Packets

By default, the source IP address of an NTP packet sent by the router is the address of the interface through which the NTP packet is sent. Use this procedure to set a different source address.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **source** *type interface-path-id*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# <code>configure</code>	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# <code>ntp</code>	Enters NTP configuration mode.
Step 3	source <i>type interface-path-id</i> Example:	Configures an interface from which the IP source address is taken. Note

	Command or Action	Purpose
	<pre>RP/0/RSP0/CPU0:router(config-ntp)# source POS 0/0/0/1</pre>	This interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the source keyword in the peer or server command shown in Configuring Poll-Based Associations, on page 38 .
Step 4	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the System as an Authoritative NTP Server

You can configure the router to act as an authoritative NTP server, even if the system is not synchronized to an outside time source.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **master** *stratum*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	master stratum Example: RP/0/RSP0/CPU0:router(config-ntp)# master 9	<p>Makes the router an authoritative NTP server.</p> <p>Note Use the master command with caution. It is very easy to override valid time sources using this command, especially if a low stratum number is configured. Configuring multiple machines in the same network with the master command can cause instability in time keeping if the machines do not agree on the time.</p>
Step 4	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> RP/0/RSP0/CPU0:router(config-ntp)# end or RP/0/RSP0/CPU0:router(config-ntp)# commit	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP-PTP Interworking

Use this task to configure NTP to use PTP as the time source.

Before you begin

PTP must be supported and enabled on the router before NTP-PTP interworking can be configured. If PTP is not enabled, you receive an error message similar to the following when you try to commit the configuration:

```
RP/0/RSP0/CPU0:router(config)# ntp master primary-reference-clock
RP/0/RSP0/CPU0:router(config)# commit

% Failed to commit one or more configuration items. Please issue
'show configuration failed' from this session to view the errors

RP/0/RSP0/CPU0:router(config)# show configuration failed
[:::]
ntp
 master primary-reference-clock
!!% 'ip-ntp' detected the 'fatal' condition 'PTP is not supported on this platform'
!
end
```

Refer to the [Configuring PTP, on page 136](#) module for more information.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **master primary-reference-clock**
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	master primary-reference-clock Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# master primary-reference-clock</pre>	Specifies PTP to be the NTP time source.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> or <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Updating the Hardware Clock

On devices that have hardware clocks (system calendars), you can configure the hardware clock to be periodically updated from the software clock. This is advisable for devices using NTP, because the time and date on the software clock (set using NTP) is more accurate than the hardware clock. The time setting on the hardware clock has the potential to drift slightly over time.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **update-calendar**
4. Use one of the following commands:

- **end**
- **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RSP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	ntp Example: <pre>RP/0/RSP0/CPU0:router(config)# ntp</pre>	Enters NTP configuration mode.
Step 3	update-calendar Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# update-calendar</pre>	Configures the router to update its system calendar from the software clock at periodic intervals.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> or <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Verifying the Status of the External Reference Clock

This task explains how to verify the status of NTP components.



Note The commands can be entered in any order.

SUMMARY STEPS

1. **show ntp associations [detail] [location node-id]**
2. **show ntp status [location node-id]**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	show ntp associations [detail] [location node-id] Example: RP/0/RSP0/CPU0:router# show ntp associations	Displays the status of NTP associations.
Step 2	show ntp status [location node-id] Example: RP/0/RSP0/CPU0:router# show ntp status	Displays the status of NTP.

Examples

The following is sample output from the **show ntp associations** command:

```
RP/0/RSP0/CPU0:router# show ntp associations

      address      ref clock      st  when  poll reach  delay  offset  disp
+~127.127.1.1      127.127.1.1      5    5  1024   37    0.0    0.00  438.3
*~172.19.69.1      172.24.114.33     3   13  1024    1    2.0   67.16   0.0
 * master (syncd), # master (unsyncd), + selected, - candidate, ~ configured
```

The following is sample output from the **show ntp status** command:

```
RP/0/RSP0/CPU0:router# show ntp status

Clock is synchronized, stratum 4, reference is 172.19.69.1
nominal freq is 1000.0000 Hz, actual freq is 999.9988 Hz, precision is 2**26
reference time is C54C131B.9EECF6CA (07:26:19.620 UTC Mon Nov 24 2008)
clock offset is 66.3685 msec, root delay is 7.80 msec
root dispersion is 950.04 msec, peer dispersion is 3.38 msec
```

Configuration Examples for Implementing NTP

Configuring Poll-Based Associations: Example

The following example shows an NTP configuration in which the router's system clock is configured to form a peer association with the time server host at IP address 192.168.22.33, and to allow the system clock to be synchronized by time server hosts at IP address 10.0.2.1 and 172.19.69.1:

```
ntp
 server 10.0.2.1 minpoll 5 maxpoll 7
 peer 192.168.22.33

 server 172.19.69.1
```

Configuring Broadcast-Based Associations: Example

The following example shows an NTP client configuration in which interface 0/2/0/0 is configured to receive NTP broadcast packets, and the estimated round-trip delay between an NTP client and an NTP broadcast server is set to 2 microseconds:

```
ntp
 interface tengige 0/2/0/0
   broadcast client
 exit
 broadcastdelay 2
```

The following example shows an NTP server configuration where interface 0/2/0/2 is configured to be a broadcast server:

```
ntp
 interface tengige 0/2/0/2
   broadcast
```

Configuring NTP Access Groups: Example

The following example shows a NTP access group configuration where the following access group restrictions are applied:

- Peer restrictions are applied to IP addresses that pass the criteria of the access list named peer-acl.
- Serve restrictions are applied to IP addresses that pass the criteria of access list named serve-acl.
- Serve-only restrictions are applied to IP addresses that pass the criteria of the access list named serve-only-acl.
- Query-only restrictions are applied to IP addresses that pass the criteria of the access list named query-only-acl.

```
ntp
 peer 10.1.1.1
```

```

peer 10.1.1.1
peer 10.2.2.2
peer 10.3.3.3
peer 10.4.4.4
peer 10.5.5.5
peer 10.6.6.6
peer 10.7.7.7
peer 10.8.8.8
access-group peer peer-acl
access-group serve serve-acl
access-group serve-only serve-only-acl
access-group query-only query-only-acl
exit
ipv4 access-list peer-acl
 10 permit ip host 10.1.1.1 any
 20 permit ip host 10.8.8.8 any
exit
ipv4 access-list serve-acl
 10 permit ip host 10.4.4.4 any
 20 permit ip host 10.5.5.5 any
exit
ipv4 access-list query-only-acl
 10 permit ip host 10.2.2.2 any
 20 permit ip host 10.3.3.3 any
exit
ipv4 access-list serve-only-acl
 10 permit ip host 10.6.6.6 any
 20 permit ip host 10.7.7.7 any
exit

```

Configuring NTP Authentication: Example

The following example shows an NTP authentication configuration. In this example, the following is configured:

- NTP authentication is enabled.
- Two authentication keys are configured (key 2 and key 3).
- The router is configured to allow its software clock to be synchronized with the clock of the peer (or vice versa) at IP address 10.3.32.154 using authentication key 2.
- The router is configured to allow its software clock to be synchronized with the clock by the device at IP address 10.32.154.145 using authentication key 3.
- The router is configured to synchronize only to systems providing authentication key 3 in their NTP packets.

```

ntp
authenticate
authentication-key 2 md5 encrypted 06120A2D40031D1008124
authentication-key 3 md5 encrypted 1311121E074110232621
trusted-key 3
server 10.3.32.154 key 3
peer 10.32.154.145 key 2

```

Disabling NTP on an Interface: Example

The following example shows an NTP configuration in which 0/2/0/0 interface is disabled:

```
ntp
 interface tengige 0/2/0/0
   disable
   exit
 authentication-key 2 md5 encrypted 06120A2D40031D1008124
 authentication-key 3 md5 encrypted 1311121E074110232621
 authenticate
 trusted-key 3
 server 10.3.32.154 key 3
 peer 10.32.154.145 key 2
```

Configuring the Source IP Address for NTP Packets: Example

The following example shows an NTP configuration in which Ethernet management interface 0/0/CPU0/0 is configured as the source address for NTP packets:

```
ntp
 authentication-key 2 md5 encrypted 06120A2D40031D1008124
 authentication-key 3 md5 encrypted 1311121E074110232621
 authenticate
 trusted-key 3
 server 10.3.32.154 key 3
 peer 10.32.154.145 key 2
 source MgmtEth0/0/CPU0/0
```

Configuring the System as an Authoritative NTP Server: Example

The following example shows a NTP configuration in which the router is configured to use its own NTP master clock to synchronize with peers when an external NTP source becomes unavailable:

```
ntp
 master 6
```

Updating the Hardware Clock: Example

The following example shows an NTP configuration in which the router is configured to update its hardware clock from the software clock at periodic intervals:

```
ntp
 server 10.3.32.154
 update-calendar
```

FQDN for NTP Server

Table 4: Feature History Table

Feature Name	Release Information	Feature Description
FQDN for NTP Server on Non-default VRF	Release 7.9.1	<p>You can now specify a Fully Qualified Domain Name (FQDN) as the hostname for NTP server configuration over nondefault VRFs.</p> <p>FQDNs are easy to remember compared to numeric IP addresses. Service migration from one host to another can cause a change in IP address leading to outages.</p> <p>Prior releases allowed FQDN handling in only default VRFs.</p>

NTP on Cisco IOS XR Software supports configuration of servers and peers using their Fully Qualified Domain Names (FQDN). While configuring, the FQDN is resolved via DNS into its corresponding IPv4 or IPv6 address and is stored in the running-configuration of the system. NTP supports FQDN for both IPv4 and IPv6 protocols. You can configure FQDN on default vrf.

Starting Cisco IOS XR Software Release 7.9.1 you can configure FQDN in nondefault vrf also.

Configure FQDN for NTP server

Prerequisites for configuring FQDN in a nondefault VRF

- Configuration must exist for DNS resolution over that specific VRF.
- The server must be reachable.

Configuration Example for FQDN on NTP Server on Default VRF

Use the **ntp server** command with the FQDN name to configure FQDN on default VRF. You dont need to specify VRF name. In the following example, time.cisco.com is the FQDN.

```
Router#configure
Router(config)#ntp server time.cisco.com
Router(config)#commit
```



Note When you are configuring FQDN over default VRF, you don't need to specify VRF name.

Running Configuration

Use the **show running-config ntp** command to see the ntp running configuration.

```
Router#show running-config ntp
ntp
server 10.48.59.212
!
```

Verification

Use the **show ntp associations** command to verify that an NTP association has come up.

```
Router#show ntp associations

address          ref clock      st  when  poll reach  delay  offset  disp
~10.48.59.212     173.38.201.67  2   42   128    3  196.06 -14.25 3949.4
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

Configuration Example for FQDN on NTP Server on Nondefault VRF

FQDN must be reachable from the router to configure it as an NTP server or peer. You can use the **ping** command and verify that FQDN is reachable. In the following example, time.cisco.com is the FQDN and vrf_1 is the VRF over which it is reachable.

```
Router#ping time.cisco.com vrf vrf_1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.1 timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 171/171/172 ms
```

When you have confirmed that FQDN is pingable, you can configure FQDN to be used as an NTP server/peer. The following example shows how to configure an NTP server using its FQDN over a nondefault vrf.

```
Router#configure
Router(config)#ntp server vrf vrf_1 time.cisco.com minpoll 4 maxpoll 4 iburst
Router(config)#commit
```



Note If the FQDN you're trying to configure isn't reachable, the CLI treats it as invalid input.

Running Configuration

Use the **show running-config ntp** command to see the ntp running configuration.

```
Router#show running-config ntp
ntp
server vrf vrf_1 192.0.2.1 minpoll 4 maxpoll 4 iburst
!
```

Verification

Use the **show ntp associations** command to verify that an NTP association has come up.

```
Router#show ntp associations

address          ref clock      st  when  poll reach  delay  offset  disp
~192.0.2.1 vrf vrf_1
                    173.38.201.115  2   14   16   37  179.10 13.492 16.680
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

Configuring NTP server inside VRF interface

This task explains how to configure NTP server inside VRF interface.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **vrf vrf-name**
4. **source interface-type interface-instance**
5. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RSP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	vrf vrf-name Example: RP/0/RSP0/CPU0:router(config)# ntp vrf Customer_A	Specify name of a VRF (VPN- routing and forwarding) instance to configure.
Step 4	source interface-type interface-instance Example: RP/0/RSP0/CPU0:router(config)# ntp vrf Customer_A source bvi 70	Configures an interface from which the IP source address is taken. This allows IOS-XR to respond to NTP queries on VRF interfaces, in this case the source is BVI. Note This interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the source keyword in the

	Command or Action	Purpose
		peer or server command shown in Configuring Poll-Based Associations, on page 38 .
Step 5	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Additional References

The following sections provide references related to implementing NTP on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR clock commands	<i>Clock Commands on the Cisco ASR 9000 Series Router module of System Management Command Reference for Cisco ASR 9000 Series Routers</i>
Cisco IOS XR NTP commands	<i>NTP Commands on module of System Management Command Reference for Cisco ASR 9000 Series Routers</i>
Information about getting started with Cisco IOS XR Software	<i>Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide</i>
Cisco IOS XR master command index	<i>Cisco ASR 9000 Series Aggregation Services Router Commands Master List</i>

Related Topic	Document Title
Information about user groups and task IDs	<i>Configuring AAA Services on the Cisco ASR 9000 Series Router</i> module of <i>System Security Configuration Guide for Cisco ASR 9000 Series Routers</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC 1059	<i>Network Time Protocol, Version 1: Specification and Implementation</i>
RFC 1119	<i>Network Time Protocol, Version 2: Specification and Implementation</i>
RFC 1305	<i>Network Time Protocol, Version 3: Specification, Implementation, and Analysis</i>

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 6

Configuring Open Flow Agent

OpenFlow is a specification from the Open Networking Foundation (ONF) that defines a flowbased forwarding infrastructure (L2-L4 Ethernet switch model) and a standardized application programmatic interface (protocol definition) to learn capabilities, add and remove flow control entries and request statistics. OpenFlow allows a controller to direct the forwarding functions of a switch through a secure channel.

This module has details about the Open Flow Agent, relevant concepts and configurations.

Table 5: Feature History for Implementing OFA Cisco IOS XR Software

Release	Modification
Release 5.1.2	This feature was introduced.
Release 5.3.4	OnePK support was discontinued.

- [OpenFlow](#), on page 64
- [OpenFlow Agent Packet In and Out Feature](#), on page 66
- [OpenFlow Agent with NetFlow Collection and Analytics](#), on page 67
- [OFA on Cisco Routers and Switches](#), on page 68
- [Functional Components](#), on page 68
- [OFA on ASR 9000 series routers](#), on page 68
- [OpenFlow Matches](#), on page 68
- [OpenFlow Actions](#), on page 71
- [Cisco Extension Actions](#), on page 72
- [Set Field Actions](#), on page 73
- [Configuring OneP for Openflow](#), on page 75
- [Configuring a Layer 2 Logical Switch for the OpenFlow Agent](#), on page 76
- [Configuring a Layer 2_Layer 3 Logical Switch for the OpenFlow Agent](#), on page 78
- [Configuring a Layer 3_VRF Logical Switch for the OpenFlow Agent](#), on page 80
- [Configuring a Layer 3_Dual-stack Logical Switch for the OpenFlow Agent](#), on page 81
- [Enabling TLS](#) , on page 83
- [Configuring NetFlow for the OpenFlow Agent](#), on page 84
- [Configuration Examples: Openflow](#), on page 87
- [Usecase for Layer2](#), on page 89
- [Usecase for Layer3](#), on page 90

OpenFlow

Openflow is an open standard to communicate between controllers, which are running applications and network elements (such as, routers and switches).

For details regarding OpenFlow, please refer the OpenFlow chapter in the *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

An overview of OFA

OpenFlow is a specification from the Open Networking Foundation (ONF) that defines a flowbased forwarding infrastructure (L2-L4 Ethernet switch model) and a standardized application programmatic interface (protocol definition) to learn capabilities, add and remove flow control entries and request statistics. OpenFlow allows a controller to direct the forwarding functions of a switch through a secure channel. Local device configuration is out of scope of the OpenFlow protocol. OpenFlow essentially provides a forwarding instruction set, allowing applications to directly program any-to-any routing and switching, with header field rewrite. New matches and actions can be applied to packets in arbitrary unconstrained fashion, allowing routing and switching on the new criteria. Routers and switches embed the fast packet forwarding and the high level routing decisions together into their software on the same device. With only a few exceptions based on user configuration, all routing and switching decisions are made by the built-in protocols and control plane logic that reside on the switch.

Prerequisites for OpenFlow Agent

The following prerequisites are required to use the OpenFlow agent on the platforms supporting IOS-XR:

- Special build of the Release 5.1.x software that has the OpenFlow functionality is required.
- The Enhanced Ethernet line card for the Cisco ASR 9000 Series Router is required for the OpenFlow agent feature.
- Any controller with version 1.1 or 1.3 is required (example, POX, ODL).
- The asr9k-k9sec Package Installation Envelope (PIE) must be present. The asr9k-mpls PIE is required for support on MPLS core (such as, PWHE).

Restrictions for OpenFlow Agent

- Same interface cannot be added to more than one logical open flow switch.
- No support for output as an action for layer3 openflow logical switch (such as pipeline131, 132).
- Only layer 3 interface support for netflow sampling statistics.

Advantages

The advantages with Open Flow Agent are:

- increases network scalability
- reduces network complexity
- allows greater application control

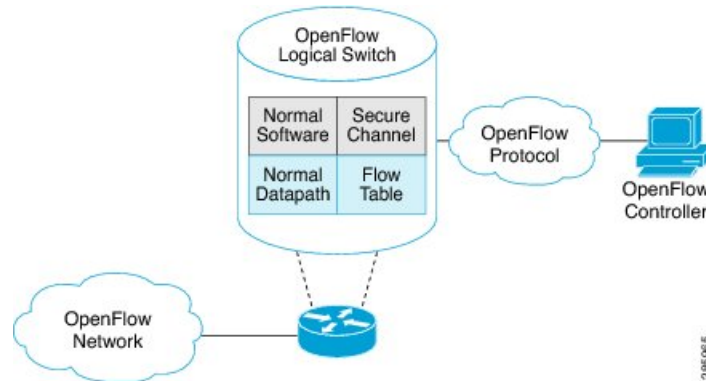
- enables customer-feature-independence

About OpenFlow

The OpenFlow protocol is based on the concept of an Ethernet switch, with an internal flow-table and standardized interface to allow traffic flows on a switch to be added or removed. The OpenFlow protocol defines the communications channel between the OpenFlow agent and the OpenFlow controller. In an OpenFlow network, the OpenFlow Agent exists on the switch and the OpenFlow controller exists on a server, which is external to the switch. Any network management is either part of the controller or accomplished through the controller.

In the Cisco OpenFlow scheme, the physical switch is divided into multiple logical switches by using the CLI to configure the connection to the controller for each logical switch and enable interfaces for each logical switch. The Openflow Agent software manages these logical switches.

The following figure shows the Cisco implementation of the OpenFlow network.



Openflow Mode for ASR9000

Openflow for the Cisco ASR 9000 Series router functions in the Integrated Hybrid mode. In this mode, both Openflow and normal switching and routing (for layer 3) operations such as L2 ethernet switching, L3 routing, etc are supported. Packets processed as the Openflow forwarding path can be processed as a normal forwarding path.

OpenFlow Table Types

An OpenFlow flow table consists of a set of flows. Each flow contains a set of matches and actions. A table has a set of capabilities in terms of supported matches and actions. Just like a policy-map, a table can be applied to a set of targets but only in the ingress direction. Hence, OpenFlow matches and actions are applied to the incoming traffic only.



Note

A set of ordered tables is referred to as a pipeline. A pipeline may contain one or more ordered tables. An OpenFlow pipeline of an OpenFlow switch on ASR9K supports only one flow table.

Table 6: OpenFlow Table Types

Table Type	Pipeline	Supported Interfaces	Description
L2	129	Bridge-domain, Gigabit ethernet, Bundle, Bundle-subinterfaces, PWHE-subinterfaces	<ul style="list-style-type: none"> • Supports L2 header matches. • Supports L2 actions. • Can be applied to the ingress L2 interfaces.
L2_L3	130	Bridge-domain, Gigabit ethernet, Bundle, Bundle-subinterfaces, PWHE-subinterfaces	<ul style="list-style-type: none"> • Supports L2 and L3 (IPv4/IPv6) header matches. • Supports L2 actions. • Can be applied to the ingress L2 interfaces.
L3_V4	131	VRF and global interfaces, BVI (ipv4 only), Bridge-domain, Gigabit ethernet, Bundle, Bundle-subinterfaces	<ul style="list-style-type: none"> • Supports L3 (IPv4) header matches. • Supports L3 (IPv4) actions. • Can be applied to the ingress L3 interfaces.
L3_DS	132	VRF and global interfaces, BVI, Bridge-domain, Gigabit ethernet, Bundle, Bundle-subinterfaces	<ul style="list-style-type: none"> • Supports L2 and L3 (IPv4/IPv6) header matches. • Supports L3 (IPv4/IPv6) actions. • Can be applied to the ingress L3 interfaces.

- L2 Table--Supports L2 header matches and has L2 actions only. This table type can be applied to the ingress of an L2 interface.
- L2_L3 Table--Supports L2 and L3 header matches and has L2 actions only. Match parameters can be IPv4 or IPv6 type. This table type can be applied to the ingress of an L2 interface.
- L3_V4 Table--Supports L3 IPv4 header matches and has L3 actions only. This table type can be applied to the ingress of L3 interfaces.
- L3_DS(Dual Stack) Table--Supports L2 and L3 IPv4 and IPv6 (Dual Stack) matches and has L3 actions only. This table type can be applied to the ingress of L3 interfaces.

OpenFlow Agent Packet In and Out Feature

The Packet In and Out feature allows a flow to be programmed by the OpenFlow Agent logical switch so that packets are sent to the Controller. The special output port: **OFP_CONTROLLER** is specified for the flow action.

The Packet In and Out feature enables support for the OpenFlow output-to-port action. The output action tells the OpenFlow Agent to send all packets matching the flow to a specific port.

OpenFlow Agent with NetFlow Collection and Analytics

Applications can be provided with on-demand analytics by using the OpenFlow protocol with NetFlow. NetFlow provides statistics on packets flowing through the router, and is the standard for acquiring IP operational data from IP networks.

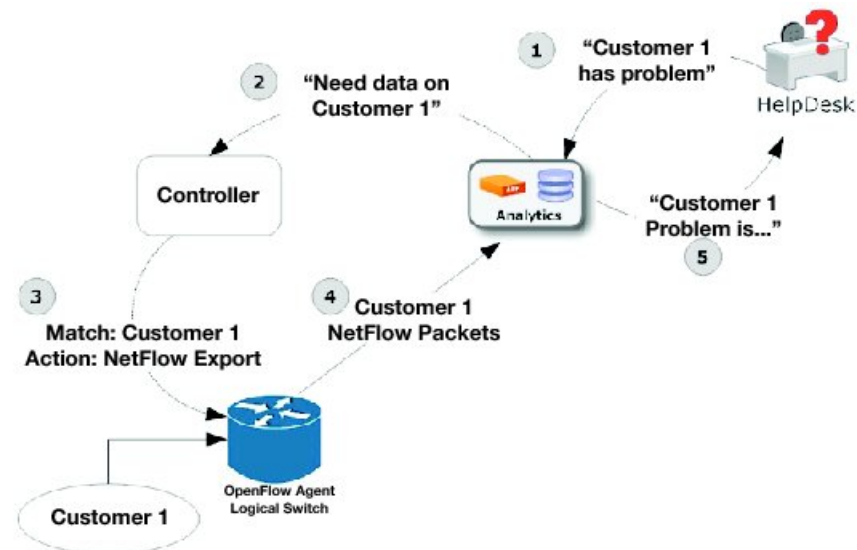
The following NetFlow maps must be configured:

- Flow Exporter Map—Specifies the destination IP address of the NetFlow collector where the NetFlow Version 9 packets are sent.
- Flow Monitor Map—Specifies the profile of the NetFlow producer, including the timeout values of active and inactive timers, size of the NetFlow cache and the exporter to be used.
- Sampler Map—Specifies how often Network Processor (NPU) needs to sample incoming and outgoing packets and create flow-packets to punt to the Line Card (LC) Central Processing Unit (CPU).

The following parameters must be specified on the OpenFlow Agent logical switch:

- Interface associated with the OpenFlow Agent logical switch that is enabled for NetFlow.
- Flow Monitor Map
- Sampler Map
- Controller IP address

Figure 1: OpenFlow Agent and NetFlow collection and analytics workflow



1. The help desk application tells the analytics application that Customer 1 has a problem.
2. The analytics application determines that it requires more information and requests more network data about Customer 1 from the Controller.
3. The Controller instructs the OpenFlow logical switch on the router to look for Customer 1 packets and generate and export NetFlow data based on Customer 1 packet flows.

4. The OpenFlow Agent logical switch exports NetFlow packets to the analytics application where they are processed.
5. The analytics application informs the help desk application of the problem.

OFA on Cisco Routers and Switches

OpenFlow SDN Applications expect network elements to speak standard OpenFlow protocol and to implement standard OpenFlow switch model. The OpenFlow Agent as a local process provides:

- OF protocol stack
- OF switch model derived from disparate Cisco software and hardware
- Version, model and feature negotiation
- Local aggregation of state and statistics
- Native dedicated CLI and troubleshooting
- High Availability

Functional Components

OpenFlow supports the configuration of multiple controllers for a logical switch. The Openflow agent can connect to a single controller or up to 8 controllers. It creates connections to all configured controllers to provide the controllers access to the OpenFlow logical switch flow tables and interfaces. It will receive flow entries from the controllers and report interface and flow status and statistics to the controllers.

The set nexthop action for layer 3 matches is implemented through a Cisco extension to the OpenFlow (1.0 and 1.3) protocol.

OFA on ASR 9000 series routers

The OpenFlow Agent supports multiple logical switch instances on ASR9K platform, with each logical switch managing a set of physical/logical interfaces, an L2 bridge domain or a VRF. Each logical switch may have one openflow connection to a single controller, or multiple connects for reliability, each to a different controller. The openflow connection to the controller uses standard TLS or plain TCP.

When the logical switch initialises a connection to the configured controller, the signaling version for the agent-controller connection is negotiated based on the bitmap version supported on both- agent and controller sides. When a logical switch starts up for the first time or at the time a logical switch loses contact with all controllers, it operates in either fail-secure mode (with default-set rule) or fail-standalone mode depending on the CLI of fail-standalone (on or off). The default for configuration is in the fail-secure mode.

OpenFlow Matches

Matches are supported on ingress port and various packet headers depending upon the packet type. Flows can have priorities. Hence, the highest priority flow entry that matches the packet gets selected.

Following table shows the list of matches supported for various table types on Cisco ASR 9000 series routers

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OXM Flow match field type for OpenFlow basic class	Description	L2 only	L2_L3	L3_V4	L3_DS
OFPXMT_OFB_IN_PORT	Switch input port	Yes	Yes	Yes	Yes
OFPXMT_OFB_IN_PHY_PORT	Switch physical port	No	No	No	No
OFPXMT_OFB_METADATA	Metadata passed between tables	No	No	No	No
OFPXMT_OFB_ETH_DST	Ethernet destination address	Yes	Yes	No	Yes
OFPXMT_OFB_ETH_SRC	Ethernet source address	Yes	Yes	No	Yes
OFPXMT_OFB_ETH_TYPE	Ethernet frame type	Yes	Yes	No	Yes
OFPXMT_OFB_VLAN_VID	VLAN ID	Yes	Yes	No	Yes
OFPXMT_OFB_VLAN_PCP	VLAN priority	Yes	Yes	No	Yes
OFPXMT_OFB_IP_DSCP	IP DSCP (6 bits in ToS field)	No	Yes	Yes	Yes
OFPXMT_OFB_IP_ECN	IP ECN (2 bits in ToS field)	No	No	No	No
OFPXMT_OFB_IP_PROTO	IP protocol	No	Yes	Yes	Yes
OFPXMT_OFB_IPV4_SRC	IPv4 source address	No	Yes	Yes	Yes
OFPXMT_OFB_IPV4_DST	IPv4 destination address	No	Yes	Yes	Yes
OFPXMT_OFB_TCP_SRC	TCP source port	No	Yes	Yes	Yes
OFPXMT_OFB_TCP_DST	TCP destination port	No	Yes	Yes	Yes

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OFPXMT_OFB_UDP_SRC	UDP source port	No	Yes	Yes	Yes
OFPXMT_OFB_UDP_DST	UDP destination port	No	Yes	Yes	Yes
OFPXMT_OFB_SCTP_SRC	SCTP source port	No	Yes	Yes	Yes
OFPXMT_OFB_SCTP_DST	SCTP destination port	No	No	No	No
OFPXMT_OFB_ICMPV4_TYPE	ICMP type	No	No	No	No
OFPXMT_OFB_ICMPV4_CODE	ICMP code	No	No	No	No
OFPXMT_OFB_ARP_OP	ARP opcode	No	No	No	No
OFPXMT_OFB_ARP_SPA	ARP source IPv4 address	No	No	No	No
OFPXMT_OFB_ARP_TPA	ARP target IPv4 address	No	No	No	No
OFPXMT_OFB_ARP_SHA	ARP source hardware address	No	No	No	No
OFPXMT_OFB_ARP_THA	ARP target hardware address	No	No	No	No
OFPXMT_OFB_IPV6_SRC	IPv6 source address	No	Yes	No	Yes
OFPXMT_OFB_IPV6_DST	IPv6 destination address	No	Yes	No	Yes
OFPXMT_OFB_IPV6_LABEL	IPv6 Flow Label	No	No	No	No
OFPXMT_OFB_ICMPV6_TYPE	ICMPv6 type	No	No	No	No
OFPXMT_OFB_ICMPV6_CODE	ICMPv6 code	No	No	No	No
OFPXMT_OFB_IPV6_ND_TARGET	Target address for ND	No	No	No	No
OFPXMT_OFB_IPV6_ND_SLL	Source link-layer for ND	No	No	No	No

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OFPXMT_OFB_IPV6_ND_TLL	Target link-layer for ND	No	No	No	No
OFPXMT_OFB_MPLS_LABEL	MPLS label	No	No	No	Yes
OFPXMT_OFB_MPLS_TC	MPLS TC	No	No	No	No
OFPXMT_OFB_MPLS_BOS	MPLS BoS bit	No	No	No	Yes
OFPXMT_OFB_PBB_ISID	PBB I-SID	No	No	No	No
OFPXMT_OFB_TUNNEL_ID	Logical Port Metadata	No	No	No	No
OFPXMT_OFB_IPV6_EXTHDR	IPv6 Extension Header pseudo-field	No	No	No	No

OpenFlow Actions

Packet forwarding and packet modification types of actions are supported. The lists of actions are always immediately applied to the packet.



Note

- Only “Apply-actions” instruction (OFPIT_APPLY_ACTIONS) of OpenFlow 1.3 is supported.
- Pipeline processing instructions that allow packets to be sent to subsequent tables for further processing are not supported in this release.
- Group tables and Meter tables are not supported.

Following table shows the list of action types supported on ASR9K for various table types.

OpenFlow Actions		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OXM Flow action field type for OpenFlow basic class	Description	L2 only	L2_L3	L3_V4	L3_DS
OFPAT_OUTPUT	Output to switch port.	Yes	Yes	No	No
OFPAT_COPY_TTL_OUT	Copy TTL "outwards"	No	No	No	No

OpenFlow Actions		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OFPAT_COPY_TTL_IN	Copy TTL "inwards"	No	No	No	No
OFPAT_SET_MPLS_TTL	MPLS TTL	No	No	No	No
OFPAT_DEC_MPLS_TTL	Decrement MPLS TTL	No	No	No	No
OFPAT_PUSH_VLAN	Push a new VLAN tag	Yes	Yes	No	No
OFPAT_POP_VLAN	Pop the outer VLAN tag	Yes	Yes	No	No
OFPAT_PUSH_MPLS	Push a new MPLS tag	No	No	No	No
OFPAT_POP_MPLS	Pop the outer MPLS tag	No	No	No	No
OFPAT_SET_QUEUE	Set queue id when outputting to a port	No	No	No	No
OFPAT_GROUP	Apply group	No	No	No	No
OFPAT_SET_NW_TTL	IP TTL	No	No	No	No
OFPAT_DEC_NW_TTL	Decrement IP TTL	No	No	No	No
OFPAT_SET_FIELD	Set a header field using OXM TLV format	Yes	Yes	Yes	Yes
OFPAT_PUSH_PBB	Push a new PBB service tag (I-TAG)	No	No	No	No
OFPAT_POP_PBB	Pop the outer PBB service tag	No	No	No	No

Cisco Extension Actions

The set ipv4 or set ipv6 nexthop actions are used to redirect an ipv4 or ipv6 packet to the specified nexthop address, instead of using the destination address in the packet. This provides ABF (ACL Based Forwarding) kind of functionality using OpenFlow. However, VRF support and nexthop tracking as supported by CLI based ABF feature is not supported in this release.

The set fcid (Forward Class ID) action can be used to support PBTS (Policy Based Tunnel Selection) functionality using OpenFlow.

Following table shows the list of actions added by Cisco to support some extra features on ASR9K.

Cisco proprietary actions		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OXM Flow match field type for OpenFlow basic class	Description	L2 only	L2_L3	L3_V4	L3_DS
Set Ipv4 Nexthop	Set ipv4 nexthop address	No	No	Yes	Yes
Set Ipv6 Nexthop	Set ipv6 nexthop address	No	No	No	Yes
Set Forward Class ID	Set forward class ID	No	No	Yes	Yes
Set VRF	Set forward ipv4/ipv6 packet based on VRF	No	No	Yes	Yes

Set Field Actions

This table lists the set field actions supported by the Cisco ASR 9000 series router:

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OXM Flow match field type for OpenFlow basic class	Description	L2 only	L2_L3	L3_V4	L3_DS
OFPXMT_OFB_ETH_DST	Ethernet destination address	Yes	Yes	No	No
OFPXMT_OFB_ETH_SRC	Ethernet source address	Yes	Yes	No	No
OFPXMT_OFB_ETH_TYPE	Ethernet frame type	No	No	No	No
OFPXMT_OFB_VLAN_VID	VLAN ID	Yes	Yes	No	No
OFPXMT_OFB_VLAN_PCP	VLAN priority	Yes	Yes	No	No

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OFPXMT_OFB_IP_DSCP	IP DSCP (6 bits in ToS field)	No	No	Yes	Yes
OFPXMT_OFB_IP_ECN	IP ECN (2 bits in ToS field)	No	No	No	No
OFPXMT_OFB_IP_PROTO	IP protocol	No	No	No	No
OFPXMT_OFB_IPV4_SRC	IPv4 source address	No	No	Yes	Yes
OFPXMT_OFB_IPV4_DST	IPv4 destination address	No	No	Yes	Yes
OFPXMT_OFB_TCP_SRC	TCP source port	No	No	Yes	Yes
OFPXMT_OFB_TCP_DST	TCP destination port	No	No	Yes	Yes
OFPXMT_OFB_UDP_SRC	UDP source port	No	No	Yes	Yes
OFPXMT_OFB_UDP_DST	UDP destination port	No	No	Yes	Yes
OFPXMT_OFB_SCTP_SRC	SCTP source port	No	No	No	No
OFPXMT_OFB_SCTP_DST	SCTP destination port	No	No	No	No
OFPXMT_OFB_ICMPV4_TYPE	ICMP type	No	No	No	No
OFPXMT_OFB_ICMPV4_CODE	ICMP code	No	No	No	No
OFPXMT_OFB_ARP_OP	ARP opcode	No	No	No	No
OFPXMT_OFB_ARP_SPA	ARP source IPv4 address	No	No	No	No
OFPXMT_OFB_ARP_TPA	ARP target IPv4 address	No	No	No	No
OFPXMT_OFB_ARP_SHA	ARP source hardware address	No	No	No	No

OpenFlow Matches		OpenFlow Switch Types Supported on ASR9K			
		Applied to L2 Bridge domain		Applied to L3 or L3 VRF interface	
OFPXMT_OFB_ARP_THA	ARP target hardware address	No	No	No	No
OFPXMT_OFB_IPV6_SRC	IPv6 source address	No	No	No	No
OFPXMT_OFB_IPV6_DST	IPv6 destination address	No	No	No	No
OFPXMT_OFB_IPV6_LABEL	IPv6 Flow Label	No	No	No	No
OFPXMT_OFB_ICMPV6_TYPE	ICMPv6 type	No	No	No	No
OFPXMT_OFB_ICMPV6_CODE	ICMPv6 code	No	No	No	No
OFPXMT_OFB_IPV6_ND_TARGET	Target address for ND	No	No	No	No
OFPXMT_OFB_IPV6_ND_SLL	Source link-layer for ND	No	No	No	No
OFPXMT_OFB_IPV6_ND_TLL	Target link-layer for ND	No	No	No	No
OFPXMT_OFB_MPLS_LABEL	MPLS label	No	No	No	No
OFPXMT_OFB_MPLS_TC	MPLS TC	No	No	No	No
OFPXMT_OFB_MPLS_BOS	MPLS BoS bit	No	No	No	No
OFPXMT_OFB_PBB_ISID	PBB I-SID	No	No	No	No
OFPXMT_OFB_TUNNEL_ID	Logical Port Metadata	No	No	No	No
OFPXMT_OFB_IPV6_EXTHDR	IPv6 Extension Header pseudo-field	No	No	No	No

Configuring OneP for Openflow

SUMMARY STEPS

1. configure
2. onep

3. **datapath transport vpathudp sender-id** *number*
4. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	onep Example: RP/0/RSP0/CPU0:router (config) # onep	Enters the OneP configuration mode.
Step 3	datapath transport vpathudp sender-id <i>number</i> Example: RP/0/RSP0/CPU0:router (config) # datapath transport vpathudp sender-id 1	Configures the virtual-path udp transport datapath for the specified sender-id.
Step 4	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Configuring a Layer 2 Logical Switch for the OpenFlow Agent

SUMMARY STEPS

1. **configure**
2. **openflow**
3. **switch** *switch -id* **pipeline** *pipeline-number*
4. **tls trust-point local** *local-tp-name* **remote** *remote-tp-name*
5. **bridge-group** *SDN-id* **bridge-domain** *switch-id*
6. **controller ipv4** *ip-address* **security** [tls | none]
7. **commit**

8. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	openflow Example: RP/0/RSP0/CPU0:router(config)# openflow	Enters the openflow configuration mode.
Step 3	switch switch-id pipeline pipeline-number Example: RP/0/RSP0/CPU0:router(config-openflow)# switch 1 pipeline 129	Enters the logical switch configuration mode. For L2-only switch, the pipeline number is 129.
Step 4	tls trust-point local local-tp-name remote remote-tp-name Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# tls trust-point local tp1 remote tp2	Enters the TLS configuration mode. Configures the local and remote trustpoints.
Step 5	bridge-group SDN-id bridge-domain switch-id Example: RP/0/RSP0/CPU0:router (config-openflow) # bridge-group SDN-1 bridge-domain of2	Configures the bridge-domain for the openflow switch. For layer2, the bridge-domain can be configured in the openflow switch and the interfaces of the bridge-domain will be learnt by the openflow switch.
Step 6	controller ipv4 ip-address security [tls none] Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# controller ipv4 5.0.1.1 port 6633 security tls	Configures the Openflow controller for the logical switch. Configures the Openflow controller for the logical switch. Once the controller command is entered, a connection to the OpenFlow controller is started for the logical switch. The tls keyword enables the TLS connection, whereas the none keyword enables the TCP connection. Note The OpenFlow Agent can connect to a single Controller or up to 8 Controllers. Repeat this step if you need to configure additional Controllers. An openflow switch can communicate to multiple controllers (the support for high-availability is a controller functionality).
Step 7	commit Example:	Adds the Layer 2 logical switch configuration for the OpenFlow agent to the running configuration.

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(logical-switch)# commit	
Step 8	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

What to do next

Repeat these steps to configure another logical switch for the OpenFlow Agent.

Configuring a Layer 2_Layer 3 Logical Switch for the OpenFlow Agent

SUMMARY STEPS

1. **configure**
2. **openflow**
3. **switch** *switch-id* **pipeline** *pipeline-number*
4. **tls trust-point local** *local-tp-name* **remote** *remote-tp-name*
5. **bridge-group** *SDN-id* **bridge-domain** *switch-id*
6. **controller ipv4** *ip-address* **security** [**tls** | **none**]
7. **commit**
8. Use the **commit** or **end** command.

DETAILED STEPS**Procedure**

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.

	Command or Action	Purpose
Step 2	openflow Example: RP/0/RSP0/CPU0:router(config)# openflow	Enters the openflow configuration mode.
Step 3	switch switch-id pipeline pipeline-number Example: RP/0/RSP0/CPU0:router(config-openflow)# switch 1 pipeline 130	Enters the logical switch configuration mode. For L2_L3 switch, the pipeline number is 130.
Step 4	tls trust-point local local-tp-name remote remote-tp-name Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# tls trust-point local tp1 remote tp2	Enters the TLS configuration mode. Configures the local and remote trustpoints.
Step 5	bridge-group SDN-id bridge-domain switch-id Example: RP/0/RSP0/CPU0:router (config-openflow) # bridge-group SDN-1 bridge-domain of2	Configures a bridge-domain for the openflow switch.
Step 6	controller ipv4 ip-address security [tls none] Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# controller ipv4 5.0.1.1 port 6633 security tls	<p>Configures the Openflow controller for the logical switch.</p> <p>Configures the Openflow controller for the logical switch. Once the controller command is entered, a connection to the OpenFlow controller is started for the logical switch. The tls keyword enables the TLS connection, whereas the none keyword enables the TCP connection.</p> <p>Note The OpenFlow Agent can connect to a single Controller or up to 8 Controllers. Repeat this step if you need to configure additional Controllers. An openflow switch can communicate to multiple controllers (the support for high-availability is a controller functionality).</p>
Step 7	commit Example: RP/0/RSP0/CPU0:router(logical-switch)# commit	Adds the Layer 2 logical switch configuration for the OpenFlow agent to the running configuration.
Step 8	Use the commit or end command.	<p>commit —Saves the configuration changes and remains within the configuration session.</p> <p>end —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Cancel —Remains in the configuration session, without committing the configuration changes.

What to do next

Repeat these steps to configure another logical switch for the OpenFlow Agent.

Configuring a Layer 3_VRF Logical Switch for the OpenFlow Agent

SUMMARY STEPS

1. **configure**
2. **openflow**
3. **switch** *switch -id pipeline pipeline-number*
4. **vrf IPv4**
5. **tls trust-point local** *local-tp-name* **remote** *remote-tp-name*
6. **controller ipv4** *ip-address* **security** [**tls** | **none**]
7. **commit**
8. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	openflow Example: RP/0/RSP0/CPU0:router(config)# openflow	Enters the openflow configuration mode.
Step 3	switch <i>switch -id pipeline pipeline-number</i> Example: RP/0/RSP0/CPU0:router(config-openflow)# switch 1 pipeline 131	Enters the logical switch configuration mode. For L3_V4(VRF) switch, the pipeline number is 131.
Step 4	vrf IPv4 Example:	VRF configuration. All the interfaces belonging to IPv4 VRF will be learnt by the openflow switch.

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config)# vrf IPv4	
Step 5	tls trust-point local <i>local-tp-name</i> remote <i>remote-tp-name</i> Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# tls trust-point local tp1 remote tp2	Enters the TLS configuration mode. Configures the local and remote trustpoints.
Step 6	controller ipv4 <i>ip-address</i> security [tls none] Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# controller ipv4 5.0.1.1 port 6633 security tls	Configures the Openflow controller for the logical switch. Configures the Openflow controller for the logical switch. Once the controller command is entered, a connection to the OpenFlow controller is started for the logical switch. Note The OpenFlow Agent can connect to a single Controller or up to 8 Controllers. Repeat this step if you need to configure additional Controllers.
Step 7	commit Example: RP/0/RSP0/CPU0:router(logical-switch)# commit	Adds the Layer 2 logical switch configuration for the OpenFlow agent to the running configuration.
Step 8	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

What to do next

Repeat these steps to configure another logical switch for the OpenFlow Agent.

Configuring a Layer 3_Dual-stack Logical Switch for the OpenFlow Agent

SUMMARY STEPS

1. **configure**
2. **openflow**
3. **switch *switch-id* pipeline *pipeline-number***

4. **interface** *type interface-path-id*
5. **tls trust-point local** *local-tp-name* **remote** *remote-tp-name*
6. **bridge-group** *SDN-id* **bridge-domain** *switch-id*
7. **controller ipv4** *ip-address* **security** [**tls** | **none**]
8. **commit**
9. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	openflow Example: RP/0/RSP0/CPU0:router(config)# openflow	Enters the openflow configuration mode.
Step 3	switch <i>switch-id</i> pipeline <i>pipeline-number</i> Example: RP/0/RSP0/CPU0:router(config-openflow)# switch 1 pipeline 132	Enters the logical switch configuration mode. For L3_DS switch, the pipeline number is 132.
Step 4	interface <i>type interface-path-id</i> Example: RP/0/RSP0/CPU0:router(config-openflow)# interface Bundle-Ether2.1	Interface configuration. Note VRFs can be configured here. Both IPv4 and IPv6 VRFs are supported.
Step 5	tls trust-point local <i>local-tp-name</i> remote <i>remote-tp-name</i> Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# tls trust-point local tp1 remote tp2	Enters the TLS configuration mode. Configures the local and remote trustpoints.
Step 6	bridge-group <i>SDN-id</i> bridge-domain <i>switch-id</i> Example: RP/0/RSP0/CPU0:router (config-openflow) # bridge-group SDN-1 bridge-domain of2	
Step 7	controller ipv4 <i>ip-address</i> security [tls none] Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# controller ipv4 5.0.1.1 port 6633 security tls	Configures the Openflow controller for the logical switch. Configures the Openflow controller for the logical switch. Once the controller command is entered, a connection to the OpenFlow controller is started for the logical switch. Note

	Command or Action	Purpose
		The OpenFlow Agent can connect to a single Controller or up to 8 Controllers. Repeat this step if you need to configure additional Controllers.
Step 8	commit Example: RP/0/RSP0/CPU0:router(logical-switch)# commit	Adds the Layer 2 logical switch configuration for the OpenFlow agent to the running configuration.
Step 9	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

What to do next

Repeat these steps to configure another logical switch for the OpenFlow Agent.

Enabling TLS

SUMMARY STEPS

1. **configure**
2. **openflow switch** *logical-switch-id*
3. **tls trust-point local** *local-tp-name* **remote** *remote-tp-name*
4. **commit**
5. **end**

DETAILED STEPS**Procedure**

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.

	Command or Action	Purpose
Step 2	openflow switch <i>logical-switch-id</i> Example: RP/0/RSP0/CPU0:router(config)# openflow switch 100	Enters the OpenFlow logical switch configuration mode.
Step 3	tls trust-point local <i>local-tp-name</i> remote <i>remote-tp-name</i> Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# tls trust-point local tp1 remote tp2	Enters the TLS configuration mode. Configures the local and remote trustpoints.
Step 4	commit Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# commit	Adds the logical switch configuration for the OpenFlow agent to the running configuration.
Step 5	end Example: RP/0/RSP0/CPU0:router(config-openflow-switch)# end	Exits logical switch configuration mode and enters EXEC mode.

Configuring NetFlow for the OpenFlow Agent

SUMMARY STEPS

1. **configure**
2. **flow exporter-map** *fem-name*
3. **destination** *location*
4. **version** *v9*
5. **commit**
6. **exit**
7. **flow monitor-map** *map-name*
8. **record** *ipv4*
9. **exporter** *map-name*
10. **cache entries** *number*
11. **cache timeout** {**active** *timeout-value* | **inactive** *timeout-value* | **update** *timeout-value*}
12. **commit**
13. **exit**
14. **sampler-map** *map-name*
15. **random 1 out-of** *sampling-interval*
16. **commit**
17. **exit**
18. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	flow exporter-map <i>fem-name</i> Example: RP/0/RSP0/CPU0:router(config)# flow exporter-map fem	Enters flow exporter map configuration mode. Note A single flow monitor map can support up to eight exporters.
Step 3	destination <i>location</i> Example: RP/0/RSP0/CPU0:router(config-fem)# destination 10.0.1.2	Configures the export destination for the flow exporter map. The destination location argument can be a hostname or an IP address.
Step 4	version <i>v9</i> Example: RP/0/RSP0/CPU0:router(config-fem)# version v9	Specifies export version parameters and enters the flow exporter map version configuration mode.
Step 5	commit Example: RP/0/RSP0/CPU0:router(config-fem-ver)# commit	Commits the configuration changes to running to the running configuration.
Step 6	exit Example: RP/0/RSP0/CPU0:router(config-fem-ver)# exit	Exits flow exporter map version configuration mode and enters global configuration mode.
Step 7	flow monitor-map <i>map-name</i> Example: RP/0/RSP0/CPU0:router(config)# flow monitor-map mmap	Creates a monitor map and configures a monitor map name and enters flow monitor map configuration mode
Step 8	record <i>ipv4</i> Example: RP/0/RSP0/CPU0:router(config-fmm)# record ipv4	Configures the flow record map name for IPv4. By default, the originating autonomous system (AS) numbers are collected and exported.
Step 9	exporter <i>map-name</i> Example: RP/0/RSP0/CPU0:router(config-fmm)# exporter fmap	Associates an exporter map with a monitor map. Note A single flow monitor map can support up to eight exporters.

	Command or Action	Purpose
Step 10	cache entries <i>number</i> Example: RP/0/RSP0/CPU0:router(config-fmm)# cache entries 4096	(Optional) Configures the number of entries in the flow cache. Replace the number argument with the number of flow entries allowed in the flow cache, in the range from 4096 through 1000000. The default number of cache entries is 65535.
Step 11	cache timeout { active <i>timeout-value</i> inactive <i>timeout-value</i> update <i>timeout-value</i> } Example: RP/0/RSP0/CPU0:router(config-fmm)# cache timeout active 10	(Optional) Configures the active, inactive, or update flow cache timeout value. <ul style="list-style-type: none"> • The default timeout value for the inactive flow cache is 15 seconds. • The default timeout value for the active flow cache is 1800 seconds. • The default timeout value for the update flow cache is 1800 seconds. Note The update keyword and <i>timeout-value</i> argument are used for permanent caches only. It specifies the timeout value that is used to export entries from permanent caches. In this case, the entries are exported but remain the cache.
Step 12	commit Example: RP/0/RSP0/CPU0:router(config-fmm)# commit	Commits the configuration changes to running to the running configuration.
Step 13	exit Example: RP/0/RSP0/CPU0:router(config-fmm)# exit	Exits flow monitor map version configuration mode and enters global configuration mode.
Step 14	sampler-map <i>map-name</i> Example: RP/0/RSP0/CPU0:router(config)# sampler-map	Creates a sampler map and enters sampler map configuration mode. Note When configuring a sampler map, be aware that NetFlow supports policing at a rate of 35,000 packets per second per direction for each individual line card.
Step 15	random 1 out-of <i>sampling-interval</i> Example: RP/0/RSP0/CPU0:router(config-sm)# random 1 out-of 65535	Configures the sampling interval to use random mode for sampling packets. For the <i>sampling-interval</i> argument, specify a number from 1 to 65535.
Step 16	commit Example: RP/0/RSP0/CPU0:router(config-sm)# commit	Commits the configuration changes to running to the running configuration.

	Command or Action	Purpose
Step 17	exit Example: RP/0/RSP0/CPU0:router(config-sm)# exit	Exits sampler map version configuration mode and enters global configuration mode.
Step 18	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

What to do next

Go to the “Associating the OpenFlow Agent Logical Switch with NetFlow” section to complete the second part of this configuration.

Configuration Examples: Openflow

Attaching a bridge domain to an Openflow Switch: Examples

- Attaching a L2-only Openflow switch

```
openflow
switch 1 pipeline 129
  tls trust-point local tp1 remote tp1
  bridge-group SDN-2 bridge-domain OF-2
  controller ipv4 5.0.1.200 port 6653 security tls
```

- Attaching a L2_L3 Openflow switch

```
openflow
switch 1 pipeline 130
  tls trust-point local tp1 remote tp1
  bridge-group SDN-2 bridge-domain OF-2
  controller ipv4 5.0.1.200 port 6653 security tls
```

- L3_V4 switch can be attached either to a VRF or directly to layer 3 interfaces under global VRF. In case of VRF, all the interfaces in that VRF become part of the OpenFlow switch.

```
openflow
switch 11 pipeline 131
  vrf IPv4
```

```
controller ipv4 5.0.1.200 port 6653 security none
!
```

- L3_DS switch can be attached either to a VRF or directly to layer 3 interfaces under global VRF.

```
openflow
switch l2 pipeline 132
vrf IPv4
controller ipv4 5.0.1.200 port 6653 security none
!
```

OpenFlow Agent with NetFlow Collection and Analytics Configuration: Example

The following example describes the NetFlow exporter map configuration for the OpenFlow logical switch.

```
Device> enable
Device# configure terminal
Device(config)# flow exporter-map fem
Device(config-fem)# destination 10.0.1.2
Device(config-fem)# version v9
Device(config-fem-ver)# commit
Device(config-fem-ver)# exit
```

The following example describes the NetFlow monitor map configuration for the OpenFlow logical switch.

```
Device(config)# flow monitor-map mmap
Device(config-fmm)# record ipv4
Device(config-fmm)# exporter fmap
Device(config-fmm)# cache entries 4096
Device(config-fmm)# commit
Device(config-fmm)# exit
```

The following example describes the NetFlow sampler map configuration for the OpenFlow logical switch.

```
Device(config)# sampler-map
Device(config-sm)# random 1 out-of 65535
Device(config-sm)# commit
Device(config-sm)# exit
```

The following example describes how the OpenFlow Agent logical switch is configured so that the NetFlow collection and analytics are associated with it.

```
Device(config)# openflow switch 100 netflow
Device(logical-switch)# flow monitor mmap sampler smap
Device(logical-switch)# interface GigabitEthernet0/1/0/6
Router(logical-switch)# controller 10.0.1.2 port 6633
Device(logical-switch)# commit
Device(logical-switch)# end
```

The following example describes **show** command output for an OpenFlow Agent logical switch that is configured with NetFlow collection and analytics.

```

Device# show openflow switch 100
Fri Jan 25 14:29:21.078 UTC

Logical Switch Context
  Id: 100
  Switch type: Netflow
  Layer: NONE
  Signal version: Openflow 1.0
  Data plane: secure
  Fallback: normal
  Config state: no-shutdown
  Working state: enabled
  TLS version: NONE
  TLS private key: none:none
  TLS private key file: NONE
  TLS certificate file: NONE
  Controller: 10.0.1.2:6633, last alive ping: 2013-01-25 14:29:20
  Netflow Monitor: mmap
  Netflow Sampler: smap
  Loopback i/f: <none>
  Loopback addr: <none>
  Interfaces:
    GigabitEthernet0/1/0/6

Device# show openflow switch 100 flows
Fri Jan 25 14:29:24.787 UTC

Logical Openflow Switch [100]:
NXST_FLOW reply (xid=0x0):
cookie=0x0, duration=204.729s, table=0, n_packets=0, n_bytes=0, priority=500 actions=netflow

Switch flow count: 1

Device# show openflow switch 100 controllers
Fri Jan 25 14:29:28.660 UTC

Logical Openflow Switch [100]:
Controller [tcp:10.0.1.2:6633]
  role : Other
  connected : Yes
  state : ACTIVE
  sec_since_connect : 487

```

Usecase for Layer2

The Scenario: Enterprise Data Center needs to perform data backup to multiple other backup sites based on the Traffic flow. The Main DC is in Vlan 100 and Backup sites are at VLAN 1000,1001,1002. These Sites are interconnected through L2VPN.

The Solution: Openflow, we can match any Layer 2 header field (in this example we have taken priority bits) and steer the traffic to go on any L2 interconnect and also rewrite the VLANs appropriately.

Usecase for Layer3

The Scenario: Three different flows from 3 different sites connected to PE1 are trying to send 350 mbps of traffic each to PE2. The bandwidth of the shortest link, Path-2 (between PE1 and PE2) is only 1 Gigabit. Hence Path-2 gets congested as soon as the third site begins to send traffic.

The Solution: Openflow controller can be used to install rules on PE1:

- Match on Flow 1 (destined to Video server) and redirect traffic to Path-2
- Match on Flow 2 (destined to Web server) and redirect traffic to Path-1
- Match on Flow 3 (destined to File transfer server) and redirect traffic to Path-3

The Inference: Effectively utilizing the network bandwidth by redirecting destination specific traffic using OpenFlow rules.



CHAPTER 7

Configuring Frequency Synchronization

Frequency Synchronization is used to distribute precision frequency around a network. Frequency is synchronized accurately using Synchronized Ethernet (SyncE) in devices connected by Ethernet in a network.

This module describes the concepts around this and details the various configurations involved. For information on SyncE commands, see *System Management Command Reference for Cisco ASR 9000 Series Routers*.

This module contains the following topics:

- [Overview, on page 91](#)
- [Clocking Support for nV Cluster , on page 95](#)
- [Configuring Frequency Synchronization, on page 97](#)

Overview

Frequency or timing synchronization is the ability to distribute precision frequency around a network. In this context, timing refers to precision frequency, not an accurate time of day. Precision frequency is required in next generation networks for applications such as circuit emulation.

To achieve compliance to ITU specifications for TDM, differential method circuit emulation must be used, which requires a known, common precision frequency reference at each end of the emulated circuit. The incumbent example of frequency synchronization is provided by SDH equipment. This is used in conjunction with an external timing technology to provide synchronization of precision timing across the network.

SDH equipments are widely replaced by Ethernet equipments and synchronized frequency is required over such Ethernet ports. Synchronous Ethernet (SyncE) is used to accurately synchronize frequency in devices connected by Ethernet in a network. SyncE provides level frequency distribution of known common precision frequency references to a physical layer Ethernet network.

To maintain SyncE links, a set of operational messages are required. These messages ensure that a node is always deriving timing information from the most reliable source and then transfers the timing source quality information to clock the SyncE link. In SDH networks, these are known as Synchronization Status Messages (SSMs). SyncE uses Ethernet Synchronization Message Channel (ESMC) to provide transport for SSMs.

Source and Selection Points

Frequency Synchronization implementation involves Sources and Selection Points.

A Source inputs frequency signals into a system or transmits them out of a system. There are four types of sources:

- Line interfaces. This includes SyncE interfaces and SONET interfaces.
- Clock interfaces. These are external connectors for connecting other timing signals, such as BITS, UTI and GPS.
- PTP clock. If IEEE 1588 version 2 is configured on the router, a PTP clock may be available to frequency synchronization as a source of the time-of-day and frequency.
- Internal oscillator. This is a free-running internal oscillator chip.

Each source has a Quality Level (QL) associated with it which gives the accuracy of the clock. This QL information is transmitted across the network using ESMC or SSMS contained in the SDH frames. This provides information about the best available source the devices in the system can synchronize to. To define a predefined network synchronization flow and prevent timing loops, you can assign priority values to the sources on each router. The combination of QL information and user-assigned priority levels allow each router to choose a source to synchronize its SyncE or SDH interfaces, as described in the ITU standard G.781.

A Selection Point is any point where a choice is made between several frequency signals and possibly one or many of them are selected. Selection points form a graph representing the flow of timing signals between different cards in a router running Cisco IOS XR software. For example, there can be one or many selection points between different Synchronous Ethernet inputs available on a single line card. This information is forwarded to a selection point on the RSP, to choose between the selected source from each card.

The input signals to the selection points can be:

- Received directly from a source.
- Received as the output from another selection point on the same card.
- Received as the output from a selection point on a different card.

The output of a selection point can be used in a number of ways, like:

- To drive the signals sent out of a set of interfaces.
- As input into another selection point on a card.
- As input into a selection point on an another card.

Use **show frequency synchronization selection** command to see a detailed view of the different selection points within the system.



Note

- We recommend you to configure, and enable Frequency Synchronization selection input on two interfaces per line card.
- For link aggregation, you must configure and enable Frequency Synchronization selection input on a single bundle member.

SyncE Hardware Support Matrix

This table provides details on the hardware that supports SyncE:



Note The table also contains support details of upcoming releases. You can read this table in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

Table 7: Feature History Table

Feature Name	Release Information	Feature Description
SyncE Support on 5th Generation 10-Port 400 Gigabit Ethernet Line Cards: <ul style="list-style-type: none"> • A99-10X400GE-X-SE • A99-10X400GE-X-TR 	Release 7.3.2	Frequency Synchronization is used to distribute precision frequency around a network. Frequency is synchronized accurately using Synchronized Ethernet (SyncE) in devices connected by Ethernet in a network. SyncE is now supported on the line cards: <ul style="list-style-type: none"> • A99-10X400GE-X-SE • A99-10X400GE-X-TR

Hardware Variant	Cisco IOS XR	Cisco IOS XR 64 bit
A9K-8X100GE-L-SE/TR (10GE and 100GE)	5.3.0	6.1.1
A9K-RSP880-SE/TR	5.3.0	6.1.1
A9K-8X100GE-L-SE/TR (40-GE)	6.0.1	6.1.1
A9K-4X100GE-SE/TR	5.3.2 (100G LAN only)	6.1.1
A9K-8X100GE-SE/TR	6.0.1	
A9K-MOD400-SE/TR	6.0.1	6.2.2
A9K-MOD200-SE/TR with MPA 20x10GE and Legacy MPAs		
A9K-MOD400-SE/TR	6.1.3	6.2.2
A9K-MOD200-SE/TR with MPAs 2x100 and 1x100		
A9K-400G-DWDM-TR	5.3.3 6.0.1	
A9K-24X10GE-1G-SE/TR	6.2.1	6.3.2
A9K-48X10GE-1G-SE/TR		

Hardware Variant	Cisco IOS XR	Cisco IOS XR 64 bit
A99-RSP-SE/TR (Cisco ASR 9910 Series Routers)	6.1.4	6.3.2
RSP880-LT-SE/TR	6.2.2	6.4.1
A9K-RSP440-TR/SE Enhanced Ethernet Linecards A99-RP-SE	4.3.4	
A99-RP2-TR/SE	5.3.0	6.3.2 6.4.1
Cisco ASR 9001 Series Routers	4.3.4	
Cisco ASR 9901 Series Routers	NA	6.4.1
A99-RSP-SE/TR (Cisco ASR 9906 Series Routers)	6.3.1	6.3.2
A9K-RSP5-SE/TR	NA	6.5.15
A99-RP3-SE/TR	NA	6.5.15
A9K-8X100GE-X-TR	NA	6.5.15
A9K-16X100GE-TR	NA	6.5.15
A9K-32X100GE-TR	NA	6.5.15
A99-32X100GE-X-TR	NA	7.1.15
A9K-8HG-FLEX-SE/TR	NA	7.1.15
A9K-20HG-FLEX-SE/TR	NA	7.1.15
ASR-9903	NA	7.1.3
A9903-20HG-PEC	NA	7.1.3
A99-10X400GE-X-SE/TR	NA	7.3.2
A99-12X100GE	NA	7.4.1
A9K-4X100GE	NA	7.4.1
ASR-9902	NA	7.4.1
A9K-4HG-FLEX-SE/TR	NA	7.4.1
A99-4HG-FLEX-SE/TR	NA	7.4.1

SyncE Restrictions

This section lists a few restrictions in configuring frequency synchronization. They are:

- On SyncE line interfaces, you can configure multiple interfaces for SyncE input. However, only one interface from each PHY gets selected as best source and programmed as SyncE input (there is no restriction on SyncE output) on the A9K-24X10GE-1G-SE/TR and A9K-48X10GE-1G-SE/TR line cards.

Clocking Support for nV Cluster

ASR9K cluster consists of two chassis connected together to provide redundancy and to meet higher bandwidth requirements. RSP440 provides two ICS (Inter-Chassis Synchronization) interfaces on the front plate. Clocking functionality support is added to the ICS interfaces. The ICS interfaces could be used for clocking, in the absence of other methods to synchronize frequency and Time-of-day information between the two cluster racks

nV Cluster Limitations

The limitations for the frequency synchronization support for cluster are:

- This feature is supported only on RSP440.
- The two chassis of the cluster have to be co-located. The length of the cable used for the ICS link should be less than 10 meters. This is needed to ensure the phase delay added due the length of the cable is within limits.
- SSM and QL is not supported on ICS links. SSM messages are not exchanged over the ICS interface. Hence, QL value needs to be configured under ICS clock interface configuration.
- The selection of an input clock source is based on the configuration of priority, QL as well as the clock quality. For SyncE, the ICS interfaces are similar to the SyncE line interfaces as far as input clock selection is concerned.
- All Input clock sources to cluster setup has to be redundant.
- No support for 1588 BC on LAG interfaces with member links across racks.

Inter-Chassis Synchronization (ICS)

ICS-Frequency Synchronization

Frequency synchronization is provided using Inter-Chassis Synchronization links (ICS). These are dedicated interfaces on the RSP used to synchronize the time and frequency.

The ICS link between the Primary DSC and Backup DSC carries the clock. There is no transfer of QL information from Primary DSC to Backup DSC. The clock direction is always from Primary DSC to Backup DSC. The Primary DSC transmits the clock and Backup DSC receives the clock.

The ICS clock interface (sync 2 or sync 3) is a clock input on the Backup DSC. The clock selection algorithm for SyncE is independent on each RSP. So, output clock from the rack which has Primary DSC is the outcome

of the clock selection on the Primary DSC. The output clock from the rack which has Backup DSC is the outcome of the clock selection on the Backup DSC. If the ICS clock interface configuration is such that it is the selected clock on the Backup DSC, then the output clocks from the Primary rack and Backup rack are synchronised.

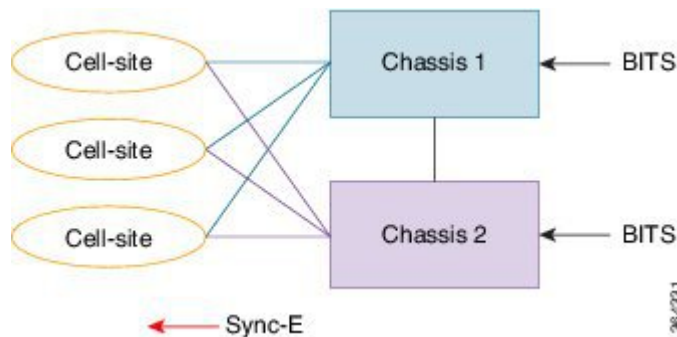
ICS-Time-of-Day

The ICS links also carry Time of Day (ToD) information when the ICS clock interfaces are configured for the same. Only the Backup DSC can synchronise with ToD from the Primary DSC and not vice versa. The 1588 clock information transmitted on all 1588 interfaces in the cluster (including interfaces on Backup rack) is of the clock selected at the Primary DSC. Thus, it is important that ICS clock interface on Backup DSC is configured such that it is the clock which is selected for ToD on the Backup DSC.

Recommended ICS Interface Connections

No inter-chassis frequency or time synchronization support:

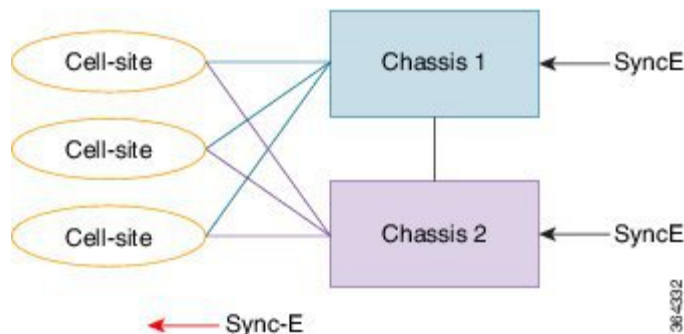
Figure 2: No inter-chassis frequency support



SyncE is used from the ASR9K cluster to provide precision frequency to mobile cell sites. A BITS clock is connected to each chassis of the cluster, meaning that the frequencies of both chassis are synchronized and the cell sites will all be synchronized, regardless of which chassis they synchronize to. In most deployments redundant BITS connections would be made to each chassis, to prevent against failure of any single BITS link.

With inter-chassis synchronization support:

Figure 3: With inter-chassis synchronization support



SyncE is used to synchronize the frequency of an ASR9k cluster to an upstream device. To provide redundancy in the case of one of the external SyncE inputs going down, the frequencies of the different cluster chassis

must somehow be synchronized; else cell sites which select links from different chassis to synchronize may be out of sync if one of the SyncE links goes down.

Configuring Frequency Synchronization

Enabling Frequency Synchronization on the Router

This task describes the router-level configuration required to enable frequency synchronization.



Important **Frequency synchronization** configuration is mandatory to enable any type of frequency or time source on the router. This includes SyncE, clock-interfaces, Precision Time Protocol (PTP) or internal oscillator.



Note If timing mode system is not configured, the major alarm `T4 PLL is in FREERUN mode` is raised. This alarm has no functional impact to the system behavior.

SUMMARY STEPS

1. **configure**
2. **frequency synchronization**
3. **clock-interface timing-mode {independent | system}**
4. **quality itu-t option {1 | 2 generation {1 | 2}}**
5. **log selection {changes | errors}**
6. Use one of these commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: <code>RP/0/RSP0/CPU0:router# configure</code>	Enters global configuration mode.
Step 2	frequency synchronization Example: <code>RP/0/RSP0/CPU0:router(config)# frequency synchronization</code>	Enables frequency synchronization on the router.

	Command or Action	Purpose
Step 3	clock-interface timing-mode {independent system} Example: <pre>RP/0/RSP0/CPU0:router(config-freqsync)# clock-interface timing-mode system</pre>	<p>Configures the type of timing sources that can be used to drive the output from a clock interface. If this command is not used, the default quality mode is used. In the default mode, the clock interface output is driven only by input from line interfaces and the internal oscillator; it is never driven by input from another clock interface. In addition, some heuristic tests are run to detect if the signal being sent out of one clock interface can be looped back by some external box and sent back in via the same, or another clock interface.</p> <ul style="list-style-type: none"> • independent—Specifies that the output of clock interfaces is driven only by the line interfaces (SyncE and SONET/SDH), as in the default mode. Loopback detection is disabled. • system—Specifies that the output of a clock interface is driven by the system-selected timing source (the source used to drive all SyncE and SONET/SDH interfaces), including clock interfaces. Loopback detection is disabled.
Step 4	quality itu-t option {1 2 generation {1 2}} Example: <pre>RP/0/RSP0/CPU0:router(config-freqsync)# quality itu-t option 2 generation 1</pre>	<p>(Optional) Specifies the quality level for the router. The default is option 1.</p> <ul style="list-style-type: none"> • option 1—Includes PRC, SSU-A, SSU-B, SEC and DNU. • option 2 generation 1—Includes PRS, STU, ST2, ST3, SMC, ST4, RES and DUS. • option 2 generation 2—Includes PRS, STU, ST2, ST3, TNC, ST3E, SMC, ST4, PROV and DUS. <p>Note The quality option configured here must match the quality option specified in the quality receive and quality transmit commands in interface frequency synchronization configuration mode.</p>
Step 5	log selection {changes errors} Example: <pre>RP/0/RSP0/CPU0:router(config-freqsync)# log selection changes</pre>	<p>Enables logging to frequency synchronization.</p> <ul style="list-style-type: none"> • changes—Logs every time there is a change to the selected source, in addition to errors. • errors—Logs only when there are no available frequency sources, or when the only available frequency source is the internal oscillator.
Step 6	Use one of these commands:	Saves configuration changes.

Command or Action	Purpose
<ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-freqsync)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-freqsync)# commit</pre>	<ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file, and remain within the configuration session.

What to do next

Configure frequency synchronization on any interfaces that should participate in frequency synchronization.

Configuring Frequency Synchronization on an Interface

By default, there is no frequency synchronization on line interfaces. Use this task to configure an interface to participate in frequency synchronization.

Before you begin

You must enable frequency synchronization globally on the router.

SUMMARY STEPS

1. **configure**
2. **interface** *type interface-path-id*
3. **frequency synchronization**
4. **selection input**
5. **priority** *priority-value*
6. **wait-to-restore** *minutes*
7. **ssm disable**
8. **time-of-day-priority** *priority*
9. **quality transmit** {**exact** | **highest** | **lowest**} **itu-t option** *ql-option*
10. **quality receive** {**exact** | **highest** | **lowest**} **itu-t option** *ql-option*
11. Use one of these commands:

- end
- commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	interface type interface-path-id Example: RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/1/1/0	Enters interface configuration mode.
Step 3	frequency synchronization Example: RP/0/RSP0/CPU0:router(config-if)# frequency synchronization	Enables frequency synchronization on the interface and enters interface frequency synchronization mode to configure the various options. By default, this causes the system selected frequency signal to be used for clocking transmission, but does not enable the use of the interface as an input.
Step 4	selection input Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# selection input	(Optional) Specifies the interface as a timing source to be passed to the selection algorithm.
Step 5	priority priority-value Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# priority 100	(Optional) Configures the priority of the frequency source on a controller or an interface. Values can range from 1 (highest priority) to 254 (lowest priority). The default value is 100. This command is used to set the priority for an interface or clock interface. The priority is used in the clock-selection algorithm to choose between two sources that have the same quality level (QL). Lower priority values are preferred.
Step 6	wait-to-restore minutes Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# wait-to-restore 300	(Optional) Configures the wait-to-restore time, in minutes, for frequency synchronization on an interface. This is the amount of time after the interface comes up before it is used for synchronization. Values can range from 0 to 12. The default value is 5.

	Command or Action	Purpose
Step 7	ssm disable Example: <pre>RP/0/RSP0/CPU0:router(config-if-freqsync)# ssm disable</pre>	(Optional) Disables Synchronization Status Messages (SSMs) on the interface. <ul style="list-style-type: none"> • For SyncE interfaces, this disables sending ESMC packets, and ignores any received ESMC packets. • For SONET and clock interfaces, this causes DNUs to be sent, and ignores any received QL value.
Step 8	time-of-day-priority priority Example: <pre>RP/0/RSP0/CPU0:router(config-if-freqsync)# time-of-day-priority 50</pre>	(Optional) Specifies the priority of this time source as the time-of-day (ToD) source. The priority is used as the first criterion when selecting between sources for a time-of-day selection point. Values can range from 1 (highest priority) to 254 (lowest priority); the default value is 100.
Step 9	quality transmit {exact highest lowest} itu-t option ql-option Example: <pre>RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality transmit highest itu-t option 1 prc</pre>	(Optional) Adjusts the QL that is transmitted in SSMs. <ul style="list-style-type: none"> • exact ql—Specifies the exact QL to send, unless DNU would otherwise be sent. • highest ql—Specifies an upper limit on the QL to be sent. If the selected source has a higher QL than the QL specified here, this QL is sent instead. • lowest ql—Specifies a lower limit on the QL to be sent. If the selected source has a lower QL than the QL specified here, DNU is sent instead. <p>The quality option specified in this command must match the globally-configured quality option in the quality itu-t option command.</p> <p>Note For clock interfaces that do not support SSM, only the lowest QL can be specified. In this case, rather than sending DNU, the output is squelched, and no signal is sent.</p>
Step 10	quality receive {exact highest lowest} itu-t option ql-option Example: <pre>RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality receive highest itu-t option 1 prc</pre>	(Optional) Adjusts the QL value that is received in SSMs, before it is used in the selection algorithm. <ul style="list-style-type: none"> • exact ql—Specifies the exact QL regardless of the value received, unless the received value is DNU. • highest ql—Specifies an upper limit on the received QL. If the received value is higher than this specified QL, this QL is used instead. • lowest ql—Specifies a lower limit on the received QL. If the received value is lower than this specified QL, DNU is used instead.

	Command or Action	Purpose
		<p>The quality option specified in this command must match the globally-configured quality option in the quality itu-t option command.</p> <p>Note For clock interfaces that do not support SSM, only the exact QL can be specified.</p>
Step 11	<p>Use one of these commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-if-freqsync)# end</pre> <p>or</p> <pre>RP/0/RSP0/CPU0:router(config-if-freqsync)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file, and remain within the configuration session.

Configuring Frequency Synchronization on a Clock Interface

To enable a clock interface to be used as frequency input or output, you must configure the port parameters and frequency synchronization, as described in this task.



Note The configuration on clock interfaces must be the same for corresponding clock interfaces across all RSPs to avoid changes in frequency synchronization behavior in the event of an RSP switchover.

SUMMARY STEPS

1. **configure**
2. **clock-interface sync** *port-no* **location** *node-id*
3. **port-parameters** {**bits-input** *mode* | **bits-output** *mode* | **dti**}
4. **ics**

5. **frequency synchronization**
6. **selection input**
7. **priority** *priority-value*
8. **wait-to-restore** *minutes*
9. **ssm disable**
10. **time-of-day-priority** *priority*
11. **quality transmit** {**exact** | **highest** | **lowest**} **itu-t option** *ql-option*
12. **quality receive** {**exact** | **highest** | **lowest**} **itu-t option** *ql-option*
13. Use one of these commands:
 - **end**
 - **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	clock-interface sync <i>port-no</i> location <i>node-id</i> Example: RP/0/RSP0/CPU0:router(config)# clock-interface sync 2 location 0/2/0	Enters clock interface configuration mode to configure the clock interface.
Step 3	port-parameters { bits-input <i>mode</i> bits-output <i>mode</i> dti } Example: RP/0/RSP0/CPU0:router(config-clock-if)# port-parameters dti	Specifies the type of external clock source for the clock interface. Options are BITS RX, BITS TX or DTI. The possible <i>mode</i> values for BITS interfaces are 2m , 6m-output-only , e1 or t1 .
Step 4	ics Example: RP/0/RSP0/CPU0:router(config)# ics	Enables chassis synchronization.
Step 5	frequency synchronization Example: RP/0/RSP0/CPU0:router(config-clock-if)# frequency synchronization RP/0/RSP0/CPU0:router(config-clk-freqsync)#	Enters clock interface frequency synchronization mode to configure frequency synchronization parameters. Note The remaining steps in this task are the same as those used to configure the interface frequency synchronization.
Step 6	selection input Example:	(Optional) Specifies the interface as a timing source to be passed to the selection algorithm.

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-if-freqsync)# selection input	
Step 7	priority <i>priority-value</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# priority 100	<p>(Optional) Configures the priority of the frequency source on a controller or an interface. Values can range from 1 (highest priority) to 254 (lowest priority). The default value is 100.</p> <p>This command is used to set the priority for an interface or clock interface. The priority is used in the clock-selection algorithm to choose between two sources that have the same quality level (QL). Lower priority values are preferred.</p>
Step 8	wait-to-restore <i>minutes</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# wait-to-restore 300	<p>(Optional) Configures the wait-to-restore time, in minutes, for frequency synchronization on an interface. This is the amount of time after the interface comes up before it is used for synchronization. Values can range from 0 to 12. The default value is 5.</p>
Step 9	ssm disable Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# ssm disable	<p>(Optional) Disables Synchronization Status Messages (SSMs) on the interface.</p> <ul style="list-style-type: none"> For SyncE interfaces, this disables sending ESMC packets, and ignores any received ESMC packets. For SONET and clock interfaces, this causes DNUs to be sent, and ignores any received QL value.
Step 10	time-of-day-priority <i>priority</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# time-of-day-priority 50	<p>(Optional) Specifies the priority of this time source as the time-of-day (ToD) source. The priority is used as the first criterion when selecting between sources for a time-of-day selection point. Values can range from 1 (highest priority) to 254 (lowest priority); the default value is 100.</p>
Step 11	quality transmit {exact highest lowest} itu-t option <i>ql-option</i> Example: RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality transmit highest itu-t option 1 prc	<p>(Optional) Adjusts the QL that is transmitted in SSMs.</p> <ul style="list-style-type: none"> exact ql—Specifies the exact QL to send, unless DNU would otherwise be sent. highest ql—Specifies an upper limit on the QL to be sent. If the selected source has a higher QL than the QL specified here, this QL is sent instead. lowest ql—Specifies a lower limit on the QL to be sent. If the selected source has a lower QL than the QL specified here, DNU is sent instead. <p>The quality option specified in this command must match the globally-configured quality option in the quality itu-t option command.</p> <p>Note For clock interfaces that do not support SSM, only the lowest QL can be specified. In this case, rather than</p>

	Command or Action	Purpose
		sending DNU, the output is squelched, and no signal is sent.
Step 12	<p>quality receive {exact highest lowest} itu-t option ql-option</p> <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality receive highest itu-t option 1 prc</pre>	<p>(Optional) Adjusts the QL value that is received in SSMs, before it is used in the selection algorithm.</p> <ul style="list-style-type: none"> • exact ql—Specifies the exact QL regardless of the value received, unless the received value is DNU. • highest ql—Specifies an upper limit on the received QL. If the received value is higher than this specified QL, this QL is used instead. • lowest ql—Specifies a lower limit on the received QL. If the received value is lower than this specified QL, DNU is used instead. <p>The quality option specified in this command must match the globally-configured quality option in the quality itu-t option command.</p> <p>Note For clock interfaces that do not support SSM, only the exact QL can be specified.</p>
Step 13	<p>Use one of these commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RSP0/CPU0:router(config-if-freqsync)# end or RP/0/RSP0/CPU0:router(config-if-freqsync)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file, and remain within the configuration session.

Configuring Clock Interface with DTI input

This procedure describes the steps involved to configure a Clock interface with DTI input.

1. To configure a clock interface, use **clock-interface sync value location node** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# clock-interface sync 1 location 0/RSP0/CPU0
```

2. To configure port parameters for the given clock interface, use **port-parameters dti** command in the clock-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-clock-if)# port-parameters dti
```

3. To enable frequency synchronization, use **frequency synchronization** command in the clock-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-clock-if)# frequency synchronization
```

4. To configure selection input for the given clock interface, use **selection input** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# selection input
```

5. To configure priority for the clock interface, use **priority number** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# priority 1
```

6. To configure wait-to-restore time for the clock interface, use **wait-to-restore number** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# wait-to-restore 0
```

7. To disable SSM packets for the clock interface, use **ssm disable** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# ssm disable
```

8. To configure quality settings for the clock interface, use **quality receive exact itu-t option number generation number PRS** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality receive exact itu-t option 2
generation 2 PRS
```

Verification

To display the current running configuration of an interface, use **show run clock-interface** command.

```
RP/0/RSP0/CPU0:router# show run clock-interface sync 1 location 0/RSP0/CPU0

clock-interface sync 1 location 0/RSP0/CPU0
port-parameters
dti
```

```

!
frequency synchronization
  selection input
  priority 1
  wait-to-restore 0
  ssm disable
  quality receive exact itu-t option 2 generation 2 PRC
!
!RP/0/RSP0/CPU0:router#

```

Configuring GPS Settings for a sync2 interface

This procedure describes the steps involved to configure GPS settings for a sync2 interface.

1. To configure a clock interface, use **clock-interface sync port-number location interface-location** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# clock-interface sync 2 location 0/RSP0/CPU0
```

2. To configure port parameters for the given clock interface, use **port-parameters** command in the clock-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-clock-if)# port-parameters
```

3. To configure GPS input parameters, use **gps-input tod-format gprmc pps-input ttl** command.

```
RP/0/RSP0/CPU0:router(config-clk-parms)# gps-input tod-format
gprmc pps-input ttl
```

4. To return to the clock-interface configuration mode, use **exit** command.

```
RP/0/RSP0/CPU0:router(config-clk-parms)# exit
```

5. To enable frequency synchronization, use **frequency synchronization** command in the clock-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-clock-if)# frequency synchronization
```

6. To configure selection input for the given clock interface, use **selection input** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# selection input
```

7. To configure priority for the clock interface, use **priority number** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# priority 10
```

8. To configure wait-to-restore time for the clock interface, use **wait-to-restore number** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# wait-to-restore 0
```

9. To disable SSM packets for the clock interface, use **ssm disable** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# ssm disable
```

10. To configure quality settings for the clock interface, use **quality receive exact itu-t option number generation number PRS** command in the frequency-synchronization clock-configuration mode.

```
RP/0/RSP0/CPU0:router(config-clk-freqsync)# quality receive exact itu-t option 2
generation 2 PRS
```

Verification

To verify the configured GPS parameters, use **show run clock-interface** command.

```
RP/0/RSP0/CPU0:router# show run clock-interface sync 2 location 0/RSP0/CPU0

clock-interface sync 2 location 0/RSP0/CPU0
port-parameters
gps-input tod-format gprmc pps-input ttl
!
```

GPS ToD Support for NMEA

National Marine Electronics Associations (NMEA) 0183 is a standard protocol used by GPS receivers to transmit data and is responsible for creating a standard uniform interface for digital data exchange between different marine electronic products. NMEA provides protocol strings to send out GPS updates. GPRMC is one such NMEA string that provides exact data and time (Greenwich time), latitude, longitude, heading, and speed. Router receives GPS ToD messages in serial ASCII stream through the RS422 interface in three formats - NTP Type 4, Cisco, and GPRMC. The timing data is extracted from this stream.



Note Cisco ASR 9000 Series Routers can support ToD in NMEA or GPRMC format. Currently, this is supported only on RS422.



Note You can refer to the below support information in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

Supported hardware are:

- A9K-RSP440-SE/TR
- A9K-RSP880-SE/TR
- A99-RP2-SE/TR
- A9K-RSP880-LT-SE/TR
- A99-RSP-SE/TR

Configuring ICS

This task enables inter-chassis synchronization for interfaces.

SUMMARY STEPS

1. **configure**
2. **clock-interface sync** *port-no location node-id*
3. **port-parameters ics**
4. **frequency synchronization**
5. **selection input**
6. **priority** *priority-value*
7. **wait-to-restore** *minutes*
8. **time-of-day-priority** *priority*
9. **quality receive** { **exact** | **highest** | **lowest** } **itu-t option** *option*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RSP0/CPU0:router# configure	Enters global configuration mode.
Step 2	clock-interface sync <i>port-no location node-id</i> Example: RP/0/RSP0/CPU0:router(config)# clock-interface sync 2 location 1/RSP0/CPU0	Enters clock interface configuration mode to configure the clock interface.
Step 3	port-parameters ics Example: RP/0/RSP0/CPU0:router(config-clock-if)# port-parameters ics	Enables inter-chassis synchronization.
Step 4	frequency synchronization Example: RP/0/RSP0/CPU0:router(config-clock-if)# frequency synchronization RP/0/RSP0/CPU0:router(config-clk-freqsync)#	Enters clock interface frequency synchronization mode to configure frequency synchronization parameters. Note The remaining steps in this task are the same as those used to configure the interface frequency synchronization.
Step 5	selection input Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# selection input	(Optional) Specifies the interface as a timing source to be passed to the selection algorithm.

	Command or Action	Purpose
Step 6	priority <i>priority-value</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# priority 100	(Optional) Configures the priority of the frequency source on a controller or an interface. Values can range from 1 (highest priority) to 254 (lowest priority). The default value is 100. This command is used to set the priority for an interface or clock interface. The priority is used in the clock-selection algorithm to choose between two sources that have the same quality level (QL). Lower priority values are preferred.
Step 7	wait-to-restore <i>minutes</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# wait-to-restore 300	(Optional) Configures the wait-to-restore time, in minutes, for frequency synchronization on an interface. This is the amount of time after the interface comes up before it is used for synchronization. Values can range from 0 to 12. The default value is 5.
Step 8	time-of-day-priority <i>priority</i> Example: RP/0/RSP0/CPU0:router(config-if-freqsync)# time-of-day-priority 50	(Optional) Specifies the priority of this time source as the time-of-day (ToD) source. The priority is used as the first criterion when selecting between sources for a time-of-day selection point. Values can range from 1 (highest priority) to 254 (lowest priority); the default value is 100.
Step 9	quality receive { exact highest lowest } itu-t option <i>option</i> Example: RP/0/RSP0/CPU0:router (config-clk-freqsync) # quality receive exact itu-t option 1 PRC	

Verifying the Frequency Synchronization Configuration

After performing the frequency synchronization configuration tasks, use this task to check for configuration errors and verify the configuration.

SUMMARY STEPS

1. **show frequency synchronization configuration-errors**
2. **show frequency synchronization interfaces brief**
3. **show frequency synchronization interfaces** *node-id*
4. **show processes fsyncmgr location** *node-id*

DETAILED STEPS

Procedure

Step 1 **show frequency synchronization configuration-errors**

Example:

```
RP/0/RSP0/CPU0:router# show frequency synchronization configuration-errors

Node 0/2/CPU0:
=====
interface GigabitEthernet0/2/0/0 frequency synchronization
  * Frequency synchronization is enabled on this interface, but isn't enabled globally.

interface GigabitEthernet0/2/0/0 frequency synchronization quality transmit exact itu-t option 2
generation 1 PRS
  * The QL that is configured is from a different QL option set than is configured globally.
```

Displays any errors that are caused by inconsistencies between shared-plane (global) and local-plane (interface) configurations. There are two possible errors that can be displayed:

- Frequency Synchronization is configured on an interface (line interface or clock-interface), but is not configured globally. Refer to [Enabling Frequency Synchronization on the Router, on page 97](#)
- The QL option configured on some interface does not match the global QL option. Under an interface (line interface or clock interface), the QL option is specified using the **quality transmit** and **quality receive** commands. The value specified must match the value configured in the global **quality itu-t option** command, or match the default (option 1) if the global **quality itu-t option** command is not configured.

Once all the errors have been resolved, meaning there is no output from the command, continue to the next step.

Step 2 show frequency synchronization interfaces brief

Example:

```
RP/0/RSP0/CPU0:router# show frequency synchronization interfaces brief

Flags:  > - Up                D - Down                S - Assigned for selection
        d - SSM Disabled      x - Peer timed out    i - Init state

Fl  Interface                QLrcv  QLuse  Pri  QLsnt  Source
===  =====
>Sx GigabitEthernet0/2/0/0    Fail   Fail   100  DNU    None
Dd  GigabitEthernet0/2/0/1    n/a    Fail   100  n/a    None

RP/0/RSP0/CPU0:router# show frequency synchronization clock-interfaces brief

Flags:  > - Up                D - Down                S - Assigned for selection
        d - SSM Disabled      s - Output squelched  L - Looped back

Node 0/0/CPU0:
=====
Fl  Clock Interface          QLrcv  QLuse  Pri  QLsnd  Source
===  =====
>S  Sync0                    PRC    Fail   100  SSU-B  Internal0 [0/0/CPU0]
>  Sync1                    SSU-A  Fail   100  SSU-B  Internal0 [0/0/CPU0]
>S  Internal0                n/a    SSU-B  255  n/a    None

Node 0/1/CPU0:
=====
Fl  Clock Interface          QLrcv  QLuse  Pri  QLsnd  Source
===  =====
D   Sync0                    None    Fail   100  SSU-B  Internal0 [0/1/CPU0]
D   Sync1                    None    Fail   100  SSU-B  Internal0 [0/1/CPU0]
>S  Internal0                n/a    SSU-B  255  n/a    None
```

Verifies the configuration. Note the following points:

- All line interface that have frequency synchronization configured are displayed.
- All clock interfaces and internal oscillators are displayed.
- Sources that have been nominated as inputs (in other words, have **selection input** configured) have 'S' in the Flags column; sources that have not been nominated as inputs do not have 'S' displayed.

Note

Internal oscillators are always eligible as inputs.

- '>' or 'D' is displayed in the flags field as appropriate.

If any of these items are not true, continue to the next step.

Step 3 **show frequency synchronization interfaces** *node-id*

Example:

```
RP/0/RSP0/CPU0:router# show frequency synchronization interfaces GigabitEthernet0/2/0/2
```

```
Interface GigabitEthernet0/2/0/2 (shutdown)
  Assigned as input for selection
  SSM Enabled
  Input:
    Down
    Last received QL: Failed
    Effective QL:      Failed, Priority: 100
  Output:
    Selected source:    Sync0 [0/0/CPU0]
    Selected source QL: Opt-I/PRC
    Effective QL:       Opt-I/PRC
  Next selection points: LC_INGRESS
```

```
RP/0/RSP0/CPU0:router# show frequency synchronization clock-interfaces location 0/1/CPU0
```

```
Node 0/1/CPU0:
```

```
=====
```

```
Clock interface Sync0 (Down: mode not configured)
  SSM supported and enabled
  Input:
    Down
    Last received QL: Opt-I/PRC
    Effective QL:      Failed, Priority: 100
  Output:
    Selected source:    Internal0 [0/1/CPU0]
    Selected source QL: Opt-I/SSU-B
    Effective QL:       Opt-I/SSU-B
  Next selection points: RP_SYSTEM
```

```
Clock interface Sync1 (Down: mode not configured)
  SSM supported and enabled
  Input:
    Down
    Last received QL: Opt-I/PRC
    Effective QL:      Failed, Priority: 100
  Output:
    Selected source:    Internal0 [0/1/CPU0]
    Selected source QL: Opt-I/SSU-B
    Effective QL:       Opt-I/SSU-B
  Next selection points: RP_SYSTEM
```

```

Clock interface Internal0 (Up)
  Assigned as input for selection
  Input:
    Default QL:  Opt-I/SSU-B
    Effective QL: Opt-I/SSU-B, Priority: 255
  Next selection points: RP_SYSTEM RP_CLOCK_INTF

```

Investigates issues within individual interfaces. If the clock interface is down, a reason is displayed. This may be because there is missing or conflicting platform configuration on the clock interface.

Step 4 **show processes fsyncmgr location *node-id***

Example:

```

RP/0/RSP0/CPU0:router# show processes fsyncmgr location 0/0/CPU0

      Job Id: 134
      PID: 30202
      Executable path: /pkg/bin/fsyncmgr
      Instance #: 1
      Version ID: 00.00.0000
      Respawn: ON
      Respawn count: 1
Max. spawns per minute: 12
      Last started: Mon Mar  9 16:30:43 2009
      Process state: Run
      Package state: Normal
      Started on config: cfg/gl/freqsync/g/a/enable
      core: MAINMEM
      Max. core: 0
      Placement: None
      startup_path: /pkg/startup/fsyncmgr.startup
      Ready: 0.133s
      Process cpu time: 1730768.741 user, -133848.-361 kernel, 1596920.380 total
-----

```

Verifies that the fsyncmgr process is running on the appropriate nodes.



CHAPTER 8

Configuring Precision Time Protocol

Precision Time Protocol (PTP) is a protocol that defines a method to distribute time around a network. PTP support is based on the IEEE 1588-2008 standard.

This module describes the concepts around this protocol and details the various configurations involved. For information on PTP commands, see *System Management Command Reference for Cisco ASR 9000 Series Routers*.

This module contains the following topics:

- [Overview, on page 115](#)
- [ITU-T Telecom Profiles for PTP, on page 131](#)
- [Configuring PTP, on page 136](#)
- [Configuring PTP Delay Asymmetry, on page 152](#)
- [Double Failure Clock Class Over-ride, on page 154](#)
- [PTP Holdover Traceability suppression, on page 155](#)
- [Configuration Examples, on page 156](#)

Overview

The Precision Time Protocol (PTP), as defined in the IEEE 1588 standard, synchronizes with nanosecond accuracy the real-time clocks of the devices in a network. The clocks are organized into a server-client hierarchy. PTP identifies the port that is connected to a device with the most precise clock. This clock is referred to as the server clock. All the other devices on the network synchronize their clocks with the server and are referred to as members. Constantly-exchanged timing messages ensure continued synchronization. PTP ensures that the best available clock is selected as the source of time (the grandmaster clock) for the network and that other clocks in the network are synchronized to the grandmaster.

Table 8: PTP Clocks

Network Element	Description
Grandmaster (GM)	A network device physically attached to the primary time source. All clocks are synchronized to the grandmaster clock.

Network Element	Description
Ordinary Clock (OC)	<p>An ordinary clock is a 1588 clock with a single PTP port that can operate in one of the following modes:</p> <ul style="list-style-type: none"> • server mode—Distributes timing information over the network to one or more client clocks, thus allowing the client to synchronize its clock to the server. • client mode—Synchronizes its clock to a server clock. You can enable the client mode on up to two interfaces simultaneously in order to connect to two different server clocks.
Boundary Clock (BC)	<p>The device participates in selecting the best server clock and can act as the server clock if no better clocks are detected.</p> <p>Boundary clock starts its own PTP session with a number of downstream clients. The boundary clock mitigates the number of network hops and results in packet delay variations in the packet network between the Grandmaster and client.</p>
Transparent Clock (TC)	<p>A transparent clock is a device or a switch that calculates the time it requires to forward traffic and updates the PTP time correction field to account for the delay, making the device transparent in terms of time calculations.</p>

PTP consists of two parts:

- The port State machine and Best Master Clock Algorithm: This provides a method to determine the ports in the network that will remain passive (neither server nor client), run as a server (providing time to other clocks in the network), or run as clients (receiving time from other clocks in the network).
- Delay-Request/Response mechanism and a Peer-delay mechanism: This provides a mechanisms for client ports to calculate the difference between the time of their own clocks and the time of their server clock.



Note Cisco ASR 9000 Series routers do not support Peer-delay mechanism.

The implementation of PTP on Cisco IOS XR software is designed to operate effectively in Telecommunication networks, which are different from the networks for which PTP was originally designed.

PTP is supported on Gigabit Ethernet interfaces and subinterfaces (1G, 10G, 40G, and 100G), and Bundle Ethernet main interfaces. PTP is not supported on Bundle Ethernet subinterfaces.

Frequency and Time Selection

The selection of the source to synchronize the backplane clock frequency is made by frequency synchronization, and is outside of the scope of PTP. The Announce, Sync, and Delay-request frequencies must be the same on the server and client.

Delay-Response Mechanism

The Delay Request-response mechanism (defined in section 11.3 of IEEE Std 1588-2008) lets a client port estimate the difference between its own clock-time and the clock-time of its server. The following options are supported:

- One-step mechanism - The timestamp for a Sync message is sent in the Sync message itself.
- Two-step mechanism - The timestamp for a Sync message is sent later in a Follow-up message.

When running a port in client state, a router can send Delay-request messages and handle incoming Sync, Follow-up, and Delay-response messages. The timeout periods for both Sync and Delay-response messages are individually configurable.

Hybrid Mode

Your router allows the ability to select separate sources for frequency and time-of-day (ToD). Frequency selection can be between any source of frequency available to the router, such as: BITS, GPS, SyncE or IEEE 1588 PTP. The ToD selection is between the source selected for frequency and PTP, if available (ToD selection is from GPS, DTI or PTP). This is known as hybrid mode, where a physical frequency source (BITS or SyncE) is used to provide frequency synchronization, while PTP is used to provide ToD synchronization.

Frequency selection uses the algorithm described in ITU-T recommendation G.871, and is described in the *Configuring Frequency Synchronization* module in this document. The ToD selection is controlled using the time-of-day priority configuration. This configuration is found under the source interface frequency synchronization configuration mode and under the global PTP configuration mode. It controls the order for which sources are selected for ToD. Values in the range of 1 to 254 are allowed, with lower numbers indicating higher priority.

Port States

State machine indicates the behavior of each port. The possible states are:

State	Description
INIT	Port is not ready to participate in PTP.
LISTENING	First state when a port becomes ready to participate in PTP: In this state, the port listens to PTP servers for a (configurable) period of time.
PRE-MASTER	Port is ready to enter the Server state.
MASTER	Port provides timestamps for any client or boundary clocks that are listening.
UNCALIBRATED	Port receives timestamps from a server clock but, the router's clock is not yet synchronized to the server.

State	Description
SLAVE	Port receives timestamps from a server clock and the router's clock is synchronized to the server.
PASSIVE	Port is aware of a better clock than the one it would advertise if it was in server state and is not a client clock to that server clock.

Leap Seconds

In prior releases, IOS-XR only offered a static and time-consuming solution to manage leap seconds. For every upcoming leap second inclusion, the number of leap seconds had to be hard-coded into a Software Maintenance Update (SMU) and also installed on the router for the same. It is a prolonged and tedious process to provide and install a SMU each time a new leap second is announced.

From Release 6.4.1 onward, Cisco IOS-XR supports leap-second configuration instead of SMU installations or reloads.

Time is measured using a common timescale. Leap second factor is used to adjust the current time to compensate for any drift from the common timescale. Leap seconds are introduced to dynamically adjust the UTC offset in response to leap second events. The two most relevant timescales are:

- **TAI - International Atomic Time** : This is a notional passage of time determined by weighted average of readings across a large number of atomic clocks.
- **UTC - Universal Coordinated Time** : This differs from TAI by an integer number of seconds to remain in synchronization with mean solar time. UTC is related to a notion of time called **UT1**, which represents the mean solar time at 0° longitude. Leap seconds are periodically inserted to ensure UTC and UT1 are never more than 0.9 seconds apart.

PTP uses TAI timescale. UTC time is derived using UTC offset. UTC offset and the number of seconds in the last minute of the current UTC day are sent in the PTP header of Announce messages.

UTC is calculated as: **UTC = TAI - offset**.

IOS-XR PTP implementation uses the following sources (in order of decreasing precedence) to determine the current UTC offset value:

- The current grandmaster clock, if present.
- UTC offset configuration, if present.
- The previous grandmaster clock, if one exists.
- The hardware (e.g. a locally connected GPS receiver), if available.
- Zero, indicating that no UTC offset information is available.

If any upcoming leap second (being advertised at the time synchronization with a grandmaster) is lost, that too will be applied at the appropriate time while in holdover

**Note**

- Leap seconds are generally added by including an extra second (23:59:60), either on June 30th or on December 31st.
- UTC offset is + 37 seconds, as of 01 Jan 2017.

Multiple PTP Profile Interoperability

Communication between two different profiles was not possible previously due to various factors like, incompatible domain numbers, BMCA, or clock-class leading to drop in packets. Also, you cannot compare devices running different profiles in such configurations. For example, the domain number for G.8275.1 profile (24) is incompatible with the domain number for G.8275.2 profile (44).

Multiple PTP Profile Interoperability feature lets you develop a configuration to communicate with a peer device running a different PTP profile than the profile that is configured on the source router. This means that multiple profiles can interoperate on a single device in this implementation.

Interoperation is achieved by converting packets on ingress/egress so that it is acceptable to the profile configured on the receiving device. This prevents packet loss and allows comparison of different profiles. You can configure the interoperation using the **interop** command. Configuration details are described in a later section in this chapter. For command details, refer to Precision Time Protocol (PTP) Commands chapter in the *System Management Command Reference for Cisco ASR 9000 Series Routers* guide.

**Note**

- Multiple ingress conversions are performed for interfaces configured with multiple servers.
- Only G.8275.1 and G.8275.2 profiles can be configured to interoperate.

PTP Support Information

This table lists different types of support information related to PTP:

Transport Media	<ul style="list-style-type: none">• UDP over IPv4• Ethernet• IPv6
-----------------	---

Messages	<ul style="list-style-type: none"> • Signaling • Announce • Sync • Follow-up • Delay-request • Delay-response • Management
Transport Modes	<ul style="list-style-type: none"> • Unicast: This is the default mode. All packets are sent as unicast messages. • Mixed: Announce and Sync messages are sent as multicast messages. Signaling, Delay-request, and Delay-response messages are sent as unicast messages. • Multicast: All packets are sent as multicast messages.

PTP hardware support matrix

Table 9: Feature History Table

Feature Name	Release Information	Feature Description
Precision time protocol on 12-port 100 Gigabit Ethernet line cards, ASR 9000 5th generation 400G line cards, ASR 9902 Series Routers, and 0.8T PEC	Release 7.4.1	<p>Support for IEEE-1588 PTP is extended to the following routers and line cards:</p> <ul style="list-style-type: none"> • A99-12X100GE • A9K-4X100GE • ASR-9902 • A9K-8HG-FLEX-SE/TR • A9K-4HG-FLEX-SE • A9K-4HG-FLEX-TR • A99-4HG-FLEX-SE • A99-4HG-FLEX-TR

Feature Name	Release Information	Feature Description
PTP support on 5th generation 10-Port 400 Gigabit Ethernet line cards: <ul style="list-style-type: none"> A99-10X400GE-X-SE A99-10X400GE-X-TR 	Release 7.3.2	Support for IEEE-1588 PTP is extended to the following line cards: <ul style="list-style-type: none"> A99-10X400GE-X-SE A99-10X400GE-X-TR



Note The table also contains support details of upcoming releases. You can read this table in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

This table provides a detailed information on the supported hardware:

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-8X100GE-L-SE/TR (10GE and 100GE)	Default & G.8265.1	5.3.3	6.3.2 6.4.1	PTP over Ethernet does not work on 100G ports on Cisco IOS XR until 6.4.1. Support was introduced in 6.4.1.
	G.8275.1 & G.8275.2	6.2.1	6.3.2 6.4.1	
	G.8273.2	6.2.1	6.3.2	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-RSP880-SE/TR	1588/PTP	5.3.3	6.3.2	-
	Default & G.8265.1		6.4.1	
	1588/PTP G.8275.1 & G.8275.2	6.2.1	6.3.2 6.4.1	
	1588/PTP G.8273.2	6.2.1	6.3.2 6.4.1	
	PTP Multiprofile	6.5.1	6.5.1	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-8X100GE-L-SE/TR (40-GE)	1588/PTP Default & G.8265.1	6.0.1	6.3.2 6.4.1	-
	1588/PTP G.8275.1 & G.8275.2	6.2.1	6.3.2 6.4.1	
	1588/PTP G.8273.2	NA	NA	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-4X100GE-SE/TR A9K-8X100GE-SE/TR	1588/PTP Default & G.8265.1	6.2.1	6.4.1	PTP over Ethernet does not work on 100G ports on Cisco IOS XR until 6.4.1. Support was introduced in 6.4.1. In 6.2.1, only G.8275.1 PTP profile is supported on the cards; No support for G.8273.2 PTP profile.
	1588/PTP G.8275.1 & G.8275.2	6.2.1	6.4.1	
	1588/PTP G.8273.2	6.4.1	6.4.1	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-MOD400-SE/TR & A9K-MOD200-SE/TR with Legacy MPAs	1588/PTP Default & G.8265.1	6.1.3	6.4.1	-
	1588/PTP G.8275.1 & G.8275.2	6.2.2	6.4.1	-
	1588/PTP G.8273.2	-	-	-
	PTP Multiprofile	6.5.1	6.5.1	-

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-MOD400-SE/TR & A9K-MOD200-SE/TR with MPA 20x10GE , A9K-MPA-1X100GE and A9K-MPA-2X100GE	1588/PTP Default & G.8265.1	6.1.3	6.4.1	PTP over Ethernet does not work on 100G ports on Cisco IOS XR until 6.4.1. Support was introduced in 6.4.1. In 6.2.2, only G.8275.1 PTP profile is supported on the cards. No support for G.8273.2 PTP profile until 6.5.1.
	1588/PTP G.8275.1 & G.8275.2	6.2.2	6.4.1	
	1588/PTP G.8273.2	6.5.1	6.5.1	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-24X10GE-IG-SE/TR A9K-48X10GE-IG-SE/TR	1588/PTP Default & G.8265.1	6.2.2 6.3.1	6.3.2	-
	1588/PTP G.8275.1 & G.8275.2	6.2.2 6.3.1	6.3.2	
	1588/PTP G.8273.2	6.3.1	6.3.2	
	PTP Multiprofile	6.5.1	6.5.1	
A99-RSP-SE/TR (Cisco ASR 9910 Series Routers)	1588/PTP Default & G.8265.1	6.3.1	6.3.2	-
	1588/PTP G.8275.1 & G.8275.2	6.3.1	6.3.2	
	1588/PTP G.8273.2	6.4.1	6.3.2	
	PTP Multiprofile	6.5.1	6.5.1	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-RSP880-LT-SE/TR	1588/PTP Default & G.8265.1	6.2.2	6.4.1	-
	1588/PTP G.8275.1 & G.8275.2	6.2.2	6.4.1	
	1588/PTP G.8273.2	6.4.1	6.4.1	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-RSP440-TR/SE A99-RP-SE Enhanced Ethernet Linecards	1588/PTP Default & G.8265.1	4.3.4	NA	Enhanced Ethernet linecards do not support G.8273.2 with G.8275.1 PTP profile. .
	1588/PTP G.8275.1 & G.8275.2	NA	NA	
	1588/PTP G.8273.2	NA	NA	
A99-RP2-TR/SE	1588/PTP Default & G.8265.1	5.3.3	6.3.2 6.4.1	-
	1588/PTP G.8275.1 & G.8275.2	6.2.1	6.3.2 6.4.1	
	1588/PTP G.8273.2	NA	NA	
Cisco ASR 9001 Series Routers	1588/PTP Default & G.8265.1	4.3.4	NA	Enhanced Ethernet based hardware does not support G.8273.2 with G.8275.1 PTP profile.
	1588/PTP G.8275.1 & G.8275.2	NA	NA	
	1588/PTP G.8273.2	NA	NA	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
Cisco ASR 9901 Series Routers	1588/PTP Default & G.8265.1	NA	6.4.1	-
	1588/PTP G.8275.1 & G.8275.2	NA	6.4.1	
	1588/PTP G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.1	
A99-RSP-SE/TR (Cisco ASR 9906 Series Routers)	1588/PTP Default & G.8265.1	6.3.1	6.3.2	-
	1588/PTP G.8275.1 & G.8275.2	6.3.1	6.3.2	
	1588/PTP G.8273.2	6.4.1	6.3.2	
	PTP Multiprofile	6.5.1	6.5.1	
A9K-RSP5-SE	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
A9K-RSP5-TR	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A99-RP3-SE	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
A99-RP3-TR	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
A9K-8X100GE-X-TR	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
A9K-16X100GE-TR	1588/PTP Default & G.8265.1	NA	6.5.15	NA
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
A99-16X100GE-X-SE A99-32X100GE-TR/CM	1588/PTP Default & G.8265.1	NA	6.6.1	NA
	1588/PTP G.8275.2	NA	6.6.1	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.6.1	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-32X100GE-TR	1588/PTP Default & G.8265.1	NA	6.5.15	-
	1588/PTP G.8275.2	NA	6.5.15	
	1588/PTP G.8275.1 & G.8273.2	NA	6.6.1	
	PTP Multiprofile	NA	6.5.15	
Cisco ASR 9903 Series Routers	1588/PTP Default & G.8265.1	NA	7.1.3	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.1.3	
	1588/PTP G.8275.1 & G.8273.2	NA	7.1.3	
	PTP Multiprofile	NA	7.1.3	
A9903-20HG-PEC	1588/PTP Default & G.8265.1	NA	7.1.3	
	1588/PTP G.8275.2	NA	7.1.3	
	1588/PTP G.8275.1 & G.8273.2	NA	7.1.3	
	PTP Multiprofile	NA	7.1.3	
A9932X100GE-X-SE/TR	1588/PTP Default & G.8265.1	NA	7.1.15	
	1588/PTP G.8275.2	NA	7.1.15	
	1588/PTP G.8275.1 & G.8273.2	NA	7.1.15	
	PTP Multiprofile	NA	7.1.15	
A9K-8HG-FLEX-SE/TR	1588/PTP Default & G.8265.1	NA	7.1.15	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.1.15	
	1588/PTP G.8275.1 & G.8273.2	NA	7.1.15	
	PTP Multiprofile	NA	7.1.15	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-20HG-FLEX-SE/TR	1588/PTP Default & G.8265.1	NA	7.1.15	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.1.15	
	1588/PTP G.8275.1 & G.8273.2	NA	7.1.15	
	PTP Multiprofile	NA	7.1.15	
A99-10X400GE-X-SE/TR	1588/PTP Default & G.8265.1	NA	7.3.2	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate. Class B Performance (Applicable to 1588/PTP G.8275.1 & G.8273.2)
	1588/PTP G.8275.2	NA	7.3.2	
	1588/PTP G.8275.1 & G.8273.2	NA	7.3.2	
	PTP Multiprofile	NA	7.3.2	
A99-12x100GE A99-12X100GE-CM	1588/PTP Default & G.8265.1	NA	7.4.1	
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	Class B Performance
	PTP Multiprofile	NA	7.4.1	
A99-8X100GE-SE/TRCM	1588/PTP Default & G.8265.1	6.2.2	6.2.2	
A9K-8X100GE-CM				
A9K-8X100G-LB-SE/TR	1588/PTP G.8275.2	6.2.2	6.2.2	
A9K-400G-DWDM-TR	1588/PTP G.8275.1 & G.8273.2	6.2.2	6.2.2	
A99-48X10GE-1G-SE/TR				
	PTP Multiprofile	6.2.2	6.2.2	
A9K-4X100GE	1588/PTP Default & G.8265.1	NA	7.4.1	
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	Class B Performance
	PTP Multiprofile	NA	7.4.1	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-400GE-SE/TR	1588/PTP Default & G.8265.1	NA	7.4.1	
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	Class B Performance
	PTP Multiprofile	NA	7.4.1	
A99-400GE-SE/TR	1588/PTP Default & G.8265.1	NA	7.4.1	
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	Class B Performance
	PTP Multiprofile	NA	7.4.1	
ASR 9902	1588/PTP Default & G.8265.1	NA	7.4.1	Port 12 to Port 35 provides Class B Performance and requires two-step clock operation on PTP master interface Port 0 to port 11 and port 36 to port 47 provide Class C performance and requires one-step clock operation on PTP master interface.
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	
	PTP Multiprofile	NA	7.4.1	
ASR-9903	1588/PTP Default & G.8265.1	NA	7.4.1	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	
	PTP Multiprofile	NA	7.4.1	

Hardware Variant	1588/PTP	Cisco IOS XR	Cisco IOS XR 64 bit	Comments
A9K-4HG-FLEX-SE/TR	1588/PTP Default & G.8265.1	NA	7.4.1	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	
	PTP Multiprofile	NA	7.4.1	
A99-4HG-FLEX-SE/TR	1588/PTP Default & G.8265.1	NA	7.4.1	Card supports 'one-step' clock operation only and must have one-step configuration on PTP master interface to operate.
	1588/PTP G.8275.2	NA	7.4.1	
	1588/PTP G.8275.1 & G.8273.2	NA	7.4.1	
	PTP Multiprofile	NA	7.4.1	



Note The following 2nd generation line cards support all IEEE-1588 PTP telecom profiles (Default, G.8265.1, G.8275.2, G.8275.1, G.8273.2, and PTP Multiprofile) in Cisco IOS XR 32 bit:

Table 10: 2nd Generation Line Cards Supporting IEEE-1588 PTP

Hardware Variant	Hardware Variant	Hardware Variant
A9K-2X100GE-SE/TR	A9K-40GE-SE/TR	A9K-40GE-SE/TR
A9K-1X100GE-SE/TR	A9K-MOD160-SE/TR	A9K-VSM-500
A9K-36X10GE-SE/TR	A9K-MOD80-SE/TR	A9K-SIP-700
A9K-4T16GE-SE/TR	A9K-4T16GE-SE/TR	

Restrictions

- PTP Grandmaster (GM) is not supported with all the PTP profiles.
- RSP IEEE 1588 port on RSP/RP is not supported.
- If PTP clock operation CLI is not configured, the default clock operation is two-step on all ASR9000 hardware variants.
- Due to the difference in PTP timestamp unit, which involves the PHY injecting the timestamp instead of the NPU, you must configure PTP clock operation **one-step** on the PTP master interface of the line cards which are explicitly specified in the [PTP hardware support matrix, on page 120](#). Rest of the line cards only support PTP clock operation **two-step** on the PTP master interface.

- PTP clock operation one-step or two-step restriction is only for PTP master interface. PTP slave interface can operate in either one-step or two-step.
- Cisco ASR 9000 Series Routers do not support Class B 1 Pulse Per Second (PPS) performance with Forward Error Correction (FEC) enabled optics.
- The upgrade of TimingIC-X firmware impacts timing functionality on the ASR 9902 and ASR 9903 Routers, and 5th Generation Line Cards. You must reload the linecard after upgrade of timing firmware for proper functioning of SyncE and PTP features. Interface flapping and traffic drops are expected during this process.
- G.8275.1 and G.8275.2 profiles are not supported on Cisco ASR 9001 chassis, Cisco ASR 9000 Ethernet line cards, Cisco ASR 9000 Enhanced Ethernet line cards, and A9K-400G-DWDM-SE/TR line cards.
- As recommended in Appendix VI of ITU-T G.8275.1 document, G.8275.1 profile is supported only on Bundle Link Aggregation (LAG) member links and not supported on a bundle interface.
- G.8273.2 Telecom Boundary Clock (T-BC) performance is not supported on 40G interfaces.
- The G.8273.2 Class B performance is observed when the same type of line card is used for both PTP server and PTP client ports. Class A performance is observed when different types of line cards are used for PTP server and PTP client on T-BC.
- G.8275.2 profile is supported on Cisco ASR 9000 Series Routers. However, the performance standards of this profile are not aligned with any of the ITU-T standards because performance specifications for G.8275.2 profile has not yet been made available by ITU-T.
- Transparent Clock (TC) is not supported.
- PTP Multiprofile is not supported for G.8273.2 Class B performance.
- Platform Fault Manager (PFM) alarms for the 10MHz port are not supported on A9K-RSP5-SE, A9K-RSP5-TR, A99-RP3-SE, and A99-RP3-TR.
- Select 5th generation line cards (A9K-20HG-FLEX-xx and A9K-8HG-FLEX-xx) will support PTP Telecom Profile G.8275.2 in combination with transit G.8265.1/G.8275.2 packets, in a future version of these cards.

**Note**

Forwarding PTP packets as IP or MPLS isn't possible without the redirecting device not being PTP-aware. If each node across the PTP path isn't performing the T-BC function, timing accuracy can't be maintained.

ITU-T Telecom Profiles for PTP

Cisco IOS XR software supports ITU-T Telecom Profiles for PTP as defined in the ITU-T recommendation. A profile consists of PTP configuration options applicable only to a specific application.

Separate profiles can be defined to incorporate PTP in different scenarios based on the IEEE 1588-2008 standard. A telecom profile differs in several ways from the default behavior defined in the IEEE 1588-2008 standard and the key differences are mentioned in the subsequent sections.

The following sections describe the ITU-T Telecom Profiles that are supported for PTP.

G.8265.1 Profile

G.8265.1 profile fulfills specific frequency-distribution requirements in telecom networks. Features of G.8265.1 profile are:

- *Clock advertisement*: G.8265.1 profile specifies changes to values used in Announce messages for advertising PTP clocks. The clock class value is used to advertise the quality level of the clock, while the other values are not used.
- *Clock Selection*: G.8265.1 profile also defines an alternate Best Master Clock Algorithm (BMCA) to select port states and clocks is defined for the profile. This profile also requires to receive Sync messages (and optionally, Delay-Response messages) to qualify a clock for selection.
- *Port State Decision*: The ports are statically configured to be Master or Slave instead of using FSM to dynamically set port states.
- *Packet Rates*: The packet rates higher than rates specified in the IEEE 1588-2008 standard are used. They are:
 - Sync/Follow-Up Packets: Rates from 128 packets-per-second to 16 seconds-per-packet.
 - Delay-Request/Delay-Response Packets: Rates from 128 packets-per-second to 16 seconds-per-packet.
 - Announce Packets: Rates from 8 packets-per-second to 64 packets-per-second.
- *Transport Mechanism*: G.8265.1 profile only supports IPv4 PTP transport mechanism.
- *Mode*: G.8265.1 profile supports transport of data packets only in unicast mode.
- *Clock Type*: G.8265.1 profile only supports Ordinary Clock-type (a clock with only one PTP port).
- *Domain Numbers*: The domain numbers that can be used in a G.8265.1 profile network ranges from 4 to 23. The default domain number is 4.
- *Port Numbers*: All PTP port numbers can only be 1 because all clocks in a this profile network are Ordinary Clocks.

G.8265.1 profile defines an alternate algorithm to select between different master clocks based on the local priority given to each master clock and their quality levels (QL). This profile also defines Packet Timing Signal Fail (PTSF) conditions to identify the master clocks that do not qualify for selection. They are:

- PTSF-lossSync condition: Raised for master clocks that do not receive a reliable stream of Sync and Delay-Resp messages. Cisco IOS XR software requests Sync and Delay-Resp grants for each configured master clock to track the master clock with this condition.
- PTSF-lossAnnounce condition: Raised for master clocks that do not receive a reliable stream of Announce messages.
- PTSF-unusable condition: Raised for master clocks that receives a reliable stream of Announce, Sync, and Delay-Resp messages, but not usable by slave clocks. Cisco IOS XR software does not use this condition.

Hardware variant-specific behavior

The profile G8265.1 displays the following behavior on these hardware variants A9K-RSP5-SE, A9K-RSP5-TR, A99-RP3-SE, and A99-RP3-TR:

- Configuring either a master or slave clock type is mandatory.
- G.8265.1 is only a frequency synchronization profile and the servo state is displayed as `FREQ_LOCKED` and the PTP slave interface remains as slave. Phase synchronization is not supported.
- G.8265.1 profile supports only PTP pure mode and not PTP hybrid mode.

G.8273.2 Profile

The G.8273.2 profile allows distribution of time and phase synchronization across packet-based networks. Cisco's implementation supports the enhanced Class C timing mode.

Class C mode enables highly accurate clock synchronization crucial for telecom networks with stringent timing requirements, including 5G networks. This mode significantly reduces the Maximum Absolute Time Error ($\text{Max}|TE|$) and improves the synchronization of Telecom Boundary Clocks (T-BC) and Telecom Time Secondary Clocks (T-TSC).

Class C timing support is available for both PTP and Frequency Synchronization, ensuring comprehensive synchronization capabilities for your network.

For information on how to configure PTP, see [Configuring PTP](#).

G.8275.1 Profile

G.8275.1 profile fulfills the time-of-day and phase synchronization requirements in telecom networks with all network devices participating in the PTP protocol. G.8275.1 profile with SyncE provides better frequency stability for the time-of-day and phase synchronization.

Features of G.8275.1 profile are:

- *Synchronization Model*: G.8275.1 profile adopts hop-by-hop synchronization model. Each network device in the path from master to slave synchronizes its local clock to upstream devices and provides synchronization to downstream devices.
- *Clock Selection*: G.8275.1 profile also defines an alternate BMCA that selects a clock for synchronization and port state for the local ports of all devices in the network is defined for the profile. The parameters defined as a part of the BMCA are:
 - Clock Class
 - Clock Accuracy
 - Offset Scaled Log Variance
 - Priority 2
 - Clock Identity
 - Steps Removed
 - Port Identity
 - notSlave flag
 - Local Priority

- *Port State Decision*: The port states are selected based on the alternate BMCA algorithm. A port is configured to a **master-only** port state to enforce the port to be a master for multicast transport mode.
- *Packet Rates*: The nominal packet rate for Announce packets is 8 packets-per-second and 16 packets-per-second for Sync/Follow-Up and Delay-Request/Delay-Response packets.
- *Transport Mechanism*: G.8275.1 profile only supports Ethernet PTP transport mechanism.
- *Mode*: G.8275.1 profile supports transport of data packets only in multicast mode. The forwarding is done based on forwardable or non-forwardable multicast MAC address.
- *Clock Type*: G.8275.1 profile supports the following clock types:
 - *Telecom Grandmaster (T-GM)*: Provides timing for other network devices and does not synchronize its local clock to other network devices.
 - *Telecom Time Slave Clock (T-TSC)*: A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other network devices.
 - *Telecom Boundary Clock (T-BC)*: Synchronizes its local clock to a T-GM or an upstream T-BC clock and provides timing information to downstream T-BC or T-TSC clocks.
- *Domain Numbers*: The domain numbers that can be used in a G.8275.1 profile network ranges from 24 to 43. The default domain number is 24.

Hardware variant-specific behavior

The profile G8275.1 displays the following behavior on these hardware variants A9K-RSP5-SE, A9K-RSP5-TR, A99-RP3-SE, and A99-RP3-TR:

- SyncE input is mandatory as only PTP hybrid mode is supported.
- The frequency is derived from the SyncE interface and phase adjustments are based on PTP.
- If you configure SyncE before you configure PTP, the Servo state is set to `FREQ_LOCKED` by default.
- After the Servo is in `PHASE_LOCKED` state, if the SyncE input is lost or removed, the Servo transitions to `HOLDOVER` state.
- After the Servo is in `PHASE_LOCKED` state, if the PTP input is lost or removed, the Servo transitions to `FREQ_LOCKED` state.



Note For the hardware variants A9K-8X100GE-X-TR, A9K-16X100GE-TR and A9K-32X100GE-TR you are not required to shut the 100 GE link to configure this profile.

G.8275.2 Profile

G.8275.2 profile fulfills the time-of-day and phase synchronization requirements in telecom networks with partial timing support from the network. Features of G.8275.2 profile are:

- *Clock Selection*: G.8275.2 profile also defines an alternate BMCA that selects a clock for synchronization and port state for the local ports of all devices in the network is defined for the profile. The parameters defined as a part of the BMCA are:

- Clock Class
- Clock Accuracy
- Offset Scaled Log Variance
- Priority 2
- Clock Identity
- Steps Removed
- Port Identity
- notSlave flag
- Local Priority



Note See ITU-T G.8275.2 document to determine the valid values for Clock Class parameter.

- *Port State Decision*: The port states are selected based on the alternate BMCA algorithm. A port is configured to a **master-only** port state to enforce the port to be a master for unicast transport mode.
- *Packet Rates*:
 - Synchronization/Follow-Up—minimum is one packet-per-second and maximum of 128 packets-per-second.
 - Packet rate for Announce packets—minimum of one packet-per-second and maximum of eight packets-per-second.
 - Delay-Request/Delay-Response packets—minimum is one packet-per-second and maximum of 128 packets-per-second
- *Transport Mechanism*: G.8275.2 profile supports only IPv4 and IPv6 PTP transport mechanism.
- *Mode*: G.8275.2 profile supports transport of data packets only in unicast mode.
- *Clock Type*: G.8275.2 profile supports the following clock types:
 - *Telecom Grandmaster (T-GM)*: Provides timing for other network devices and does not synchronize its local clock to other network devices.
 - *Telecom Time Slave Clock (T-TSC)*: A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other network devices.
 - *Telecom Boundary Clock (T-BC)*: Synchronizes its local clock to a T-GM or an upstream T-BC clock and provides timing information to downstream T-BC or T-TSC clocks.
- *Domain Numbers*: The domain numbers that can be used in a G.8275.2 profile network ranges from 44 to 63. The default domain number is 44.

Hardware variant-specific behavior

The profile G8275.2 displays the following behavior on these hardware variants A9K-RSP5-SE, A9K-RSP5-TR, A99-RP3-SE, and A99-RP3-TR:

- Hybrid PTP and pure PTP are supported on this profile.
- The physical-layer-frequency command must be used to configure Hybrid PTP.
- To switch from Hybrid PTP to Pure PTP, you must remove the physical-layer-frequency configuration and frequency synchronization configuration to remove SyncE inputs from line card interfaces and RSP clock-interfaces.

Configuring PTP

Prerequisite

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

PTP Interface and Profile Configuration

When a global PTP profile is attached to an interface, its values are used as default settings for that interface. When additional settings are configured under an interface itself, these settings override the defaults in that profile. When no profile is attached to an interface, the configuration on the interface is used to determine the PTP settings for that interface.

When configuring PTP, use one of the following approaches:

- Create a profile (or multiple profiles) containing all the default settings to use on all PTP interfaces. Override any settings that differ for particular interfaces by using the interface configuration under the interfaces themselves.
- Configure all settings separately for each interface, without using any global profiles. Use this approach if the interfaces do not have consistent settings, or if you are configuring only a small number of PTP interfaces.

Configuring Frequency Synchronization and Quality Settings for PTP

This procedure describes the steps involved to configure frequency and quality settings for PTP on a router.

1. To enable frequency synchronization on the router, use **frequency synchronization** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# frequency synchronization
```

2. To configure ITU-T quality parameters, use **quality itu-t option *option* generation *number*** command in the frequency synchronization configuration mode.
 - **option 1**: Includes PRC, SSU-A, SSU-B, SEC, and DNU. This is the default option.
 - **option 2 generation 1**: Includes PRS, STU, ST2, ST3, SMC, and DUS.
 - **option 2 generation 2**: Includes PRS, STU, ST2, ST3, TNC, ST3E, SMC, and DUS.



Note The **quality** option configured here must match the **quality** option specified in the **quality receive** and **quality transmit** commands.

```
RP/0/RSP0/CPU0:router(config-freqsync)# quality itu-t
option 2 generation 2
```

Verification

To display the frequency synchronization selection, use **show frequency synchronization selection** command.

```
RP/0/RSP0/CPU0:router# show frequency synchronization selection
Node 0/RSP1/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
  Last programmed 06:49:27 ago, and selection made 06:49:15 ago
  Next selection points
    SPA scoped      : None
    Node scoped     : T4-SEL-C CHASSIS-TOD-SEL
    Chassis scoped: LC_TX_SELECT
    Router scoped  : None
  Uses frequency selection
  Used for local line interface output
  S  Input                Last Selection Point          QL  Pri  Status
  == =====
  1  Sync1 [0/RSP1/CPU0]   n/a                      PRC   1   Locked
    HundredGigE0/5/0/2    0/5/CPU0 ETH_RXMUX 1      PRC   1   Available
    Internal0 [0/RSP1/CPU0] n/a                      SEC  255  Available

Selection point: T4-SEL-A (1 inputs, 1 selected)
  Last programmed 06:49:27 ago, and selection made 06:49:15 ago
  Next selection points
    SPA scoped      : None
    Node scoped     : T4-SEL-C
    Chassis scoped: None
    Router scoped  : None
  Uses frequency selection
  S  Input                Last Selection Point          QL  Pri  Status
  == =====
  1  HundredGigE0/5/0/2    0/5/CPU0 ETH_RXMUX 1      PRC   1   Available

Selection point: T4-SEL-C (2 inputs, 1 selected)
  Last programmed 06:49:15 ago, and selection made 06:49:15 ago
  Next selection points
    SPA scoped      : None
    Node scoped     : None
    Chassis scoped: None
    Router scoped  : None
  Uses frequency selection
  Used for local clock interface output
  S  Input                Last Selection Point          QL  Pri  Status
  == =====
  1  Sync1 [0/RSP1/CPU0]   0/RSP1/CPU0 T0-SEL-B 1      PRC   1   Locked
    HundredGigE0/5/0/2    0/RSP1/CPU0 T4-SEL-A 1      PRC   1   Available

Selection point: CHASSIS-TOD-SEL (1 inputs, 1 selected)
  Last programmed 6d04h ago, and selection made 6d04h ago
  Next selection points
    SPA scoped      : None
```

```

Node scoped   : None
Chassis scoped: None
Router scoped : None
Uses time-of-day selection
S  Input                               Last Selection Point      Pri  Time  Status
== =====
1  Sync1 [0/RSP1/CPU0]                0/RSP1/CPU0 T0-SEL-B 1    100  Yes   Available

Node 0/3/CPU0:
=====
Selection point: ETH_RXMUX (0 inputs, 0 selected)
Last programmed 9w6d ago, and selection made 9w6d ago
Next selection points
  SPA scoped   : None
  Node scoped  : None
  Chassis scoped: T0-SEL-B T4-SEL-A
  Router scoped: None
Uses frequency selection

Selection point: LC_TX_SELECT (1 inputs, 1 selected)
Last programmed 9w6d ago, and selection made 9w6d ago
Next selection points
  SPA scoped   : None
  Node scoped  : None
  Chassis scoped: None
  Router scoped: None
Uses frequency selection
Used for local line interface output
S  Input                               Last Selection Point      QL  Pri  Status
== =====
24 Sync1 [0/RSP1/CPU0]              0/RSP1/CPU0 T0-SEL-B 1    PRC  1   Available

Node 0/5/CPU0:
=====
Selection point: ETH_RXMUX (1 inputs, 1 selected)
Last programmed 06:49:27 ago, and selection made 06:49:27 ago
Next selection points
  SPA scoped   : None
  Node scoped  : None
  Chassis scoped: T0-SEL-B T4-SEL-A
  Router scoped: None
Uses frequency selection
S  Input                               Last Selection Point      QL  Pri  Status
== =====
1  HundredGigE0/5/0/2                n/a                      PRC  1   Available

Selection point: LC_TX_SELECT (1 inputs, 1 selected)
Last programmed 6d04h ago, and selection made 6d04h ago
Next selection points
  SPA scoped   : None
  Node scoped  : None
  Chassis scoped: None
  Router scoped: None
Uses frequency selection
Used for local line interface output
S  Input                               Last Selection Point      QL  Pri  Status
== =====
24 Sync1 [0/RSP1/CPU0]              0/RSP1/CPU0 T0-SEL-B 1    PRC  1   Available

```

Configuring Global Profile

This procedure describes the steps involved to create a global configuration profile for a PTP interface that can then be assigned to any interface as required.



Note Prior to Cisco IOS XR Software Release 6.3.3, the default PTP timers for G2875.1 were not set to standard values. This could lead to interoperability issues with other routers running the timers with updated values. Hence, to prevent such issues arising due to difference in packet rates, you must explicitly configure the **announce interval** value to 8, **sync frequency** value to 16 and **delay-request frequency** value to 16 while configuring global g.2875.1 profile.

1. To enter the PTP configuration mode, use **ptp** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# ptp
```

2. To configure a PTP profile, use **profile** command in the ptp configuration mode.

```
RP/0/RSP0/CPU0:router(config-ptp)# profile tp64
```

3. To configure frequency for a Sync message for the given PTP profile, use **sync frequency rate** command in the ptp-profile configuration mode.

```
RP/0/RSP0/CPU0:router(config-ptp-profile)# sync frequency 16
```

4. To configure delay-request frequency for the given PTP profile, use **delay-request frequency rate** command in the ptp-profile configuration mode.

```
RP/0/RSP0/CPU0:router(config-ptp-profile)# delay-request frequency 16
```

Verification

To display the configured PTP profile details, use **show run ptp** command.

```
RP/0/RSP0/CPU0:router# show run ptp
```

```
Wed Feb 28 11:16:05.943 UTC
ptp
clock
  domain 24
  profile g.8275.1 clock-type T-BC
!
profile slave
  transport ethernet
  sync frequency 16
  announce interval 1
  delay-request frequency 16
!
profile master
  transport ethernet
  sync frequency 16
  announce interval 1
  delay-request frequency 16
!
```

```

profile slave1
  transport ethernet
  sync frequency 64
  announce interval 1
  delay-request frequency 64
!
```

Configuring PTP Slave Interface

This procedure describes the steps involved to configure a PTP interface to be a Slave.

1. To configure an interface, use **interface** *type interface-path-id* command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/5
```

2. To enter the PTP configuration mode for the given interface, use **ptp** command in the interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if)# ptp
```

3. To configure a PTP profile (or specify a previously defined profile), use **profile** *name* command in the ptp interface configuration mode.



Note Any additional commands entered in ptp-interface configuration mode overrides the global profile settings.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# profile tp64
```

4. To configure the transport mode for all PTP messages in the given PTP profile, use **transport** *mode_type* command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# transport ipv4
```

5. To configure timeout for PTP announce messages in the given PTP profile, use **announce** *interval interval-value* command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# announce interval 1
```

6. To configure the port state, use **port state** command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# port state slave-only
```

7. To configure IPv4 or IPv6 address for PTP master, use **master** *ipv4|ipv6 address* command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# master ipv4 192.168.2.1
```

```
RP/0/RSP0/CPU0:router(config-if-ptp)# master ipv6 2001:DB8::1
```

8. To return to the interface configuration mode, use **exit** command.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# exit
```


9. To configure a gateway for the given interface, use **ipv4 address** *address mask* command in the interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 1.7.1.2 255.255.255.0
```

Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
RP/0/RSP0/CPU0:router# show run interface TenGigE 0/1/0/5
```

```
Fri Aug 3 19:57:14.184 UTC
interface TenGigE 0/1/0/5
 ptp
  profile tp64
  transport ipv4
  port state slave-only
  master ipv4 192.168.2.1
  !
  announce interval 1
  !
  ipv4 address 1.7.1.1 255.255.255.0
  !
```

Configuring PTP Master Interface

This procedure describes the steps involved to configure a PTP interface to be a Master.

1. To configure an interface, use **interface** *type interface-path-id* command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/5
```

2. To enter the PTP configuration mode for the given interface, use **ptp** command in the interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if)# ptp
```

3. To configure a PTP profile (or specify a previously defined profile), use **profile** *name* command in the ptp interface configuration mode.



Note Any additional commands entered in PTP interface configuration mode override settings in this profile.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# profile tp64
```

4. To configure the transport mode for all PTP messages in the given PTP profile, use **transport** *mode_type* command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# transport ipv4
```

5. To configure timeout for PTP announce messages in the given PTP profile, use **announce interval** *interval-value* command in the ptp interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# announce interval 1
```

6. To return to the interface configuration mode, use **exit** command.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# exit
```

7. To configure a gateway for the given interface, use **ipv4 address address mask** command in the interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 1.7.1.2 255.255.255.0
```

Verification

To verify the port state details, use **show run interface interface-name** command.

```
RP/0/RSP0/CPU0:router# show run interface TenGigE 0/1/0/5
```

```
Fri Aug 3 13:57:44.366 PST
interface TenGigE 0/1/0/5
 ptp
  profile tp64
  transport ipv4
  !
  announce interval 1
  !
  ipv4 address 1.7.1.2 255.255.255.0
  !
```

Configuring PTP Hybrid Mode

This procedure describes the steps involved to configure router in a hybrid mode. You can do this by selecting PTP for Time-of-Day (ToD) and another source for frequency.



Note Configure appropriate clock priorities when using synchronous Ethernet for frequency synchronization. This ensures the system selects the frequency source from the same interface as the PTP source. Aligning both frequency and time synchronization to the same interface helps maintain consistency and reduces configuration complexity.

1. To enable frequency synchronization on the router, use **frequency synchronization** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# frequency synchronization
```

2. To configure a SyncE source, create an interface to be a SyncE input. This can be configured using **interface** command in the configuration mode.



Note The time-of-day-priority setting specifies that SyncE to be used as a ToD source if there is no source available with a lower priority.

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/0
RP/0/RSP0/CPU0:router(config-if)# frequency synchronization
RP/0/RSP0/CPU0:router(config-if-freqsync)# selection input
RP/0/RSP0/CPU0:router(config-if-freqsync)# time-of-day-priority 100
RP/0/RSP0/CPU0:router(config-if-freqsync)# commit
```

3. To configure PTP as the source for ToD, enable PTP on the router using **ptp** command in configuration mode. ToD priority values can range from 1 (highest priority) to 254 (lowest priority).

```
RP/0/RSP0/CPU0:router(config)# ptp
RP/0/RSP0/CPU0:router(config-ptp)# time-of-day-priority 1
RP/0/RSP0/CPU0:router(config)# commit
```

4. To configure a PTP interface, use **interface** command in configuration mode. To enable this interface as a PTP Master, use **master** command in ptp-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config)# interface gigabitEthernet 0/1/0/1
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.0.0.1/24
RP/0/RSP0/CPU0:router(config-if)# ptp
RP/0/RSP0/CPU0:router(config-if-ptp)# master ipv4 10.0.0.2
RP/0/RSP0/CPU0:router(config-if-ptp)# commit
```

Verification

To display the frequency synchronization selection, use **show frequency synchronization selection** command.

```
RP/0/RSP0/CPU0:router# show frequency synchronization selection
Node 0/RSP1/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 06:49:27 ago, and selection made 06:49:15 ago
Next selection points
  SPA scoped      : None
  Node scoped     : T4-SEL-C CHASSIS-TOD-SEL
  Chassis scoped: LC_TX_SELECT
  Router scoped  : None
Uses frequency selection
Used for local line interface output
S  Input                               Last Selection Point      QL  Pri  Status
== =====
1  Sync1 [0/RSP1/CPU0]                 n/a                        PRC   1  Locked
   HundredGigE0/5/0/2                 0/5/CPU0 ETH_RXMUX 1     PRC   1  Available
   Internal0 [0/RSP1/CPU0]             n/a                        SEC  255  Available

Selection point: T4-SEL-A (1 inputs, 1 selected)
Last programmed 06:49:27 ago, and selection made 06:49:15 ago
Next selection points
  SPA scoped      : None
  Node scoped     : T4-SEL-C
  Chassis scoped: None
  Router scoped  : None
Uses frequency selection
S  Input                               Last Selection Point      QL  Pri  Status
== =====
1  HundredGigE0/5/0/2                 0/5/CPU0 ETH_RXMUX 1     PRC   1  Available
```

```

Selection point: T4-SEL-C (2 inputs, 1 selected)
Last programmed 06:49:15 ago, and selection made 06:49:15 ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : None
  Router scoped   : None
Uses frequency selection
Used for local clock interface output
S  Input                      Last Selection Point      QL  Pri  Status
== =====
1  Sync1 [0/RSP1/CPU0]        0/RSP1/CPU0 T0-SEL-B 1      PRC  1  Locked
   HundredGigE0/5/0/2        0/RSP1/CPU0 T4-SEL-A 1      PRC  1  Available

Selection point: CHASSIS-TOD-SEL (1 inputs, 1 selected)
Last programmed 6d04h ago, and selection made 6d04h ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : None
  Router scoped   : None
Uses time-of-day selection
S  Input                      Last Selection Point      Pri  Time  Status
== =====
1  Sync1 [0/RSP1/CPU0]        0/RSP1/CPU0 T0-SEL-B 1      100  Yes  Available

Node 0/3/CPU0:
=====
Selection point: ETH_RXMUX (0 inputs, 0 selected)
Last programmed 9w6d ago, and selection made 9w6d ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : T0-SEL-B T4-SEL-A
  Router scoped   : None
Uses frequency selection

Selection point: LC_TX_SELECT (1 inputs, 1 selected)
Last programmed 9w6d ago, and selection made 9w6d ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : None
  Router scoped   : None
Uses frequency selection
Used for local line interface output
S  Input                      Last Selection Point      QL  Pri  Status
== =====
24 Sync1 [0/RSP1/CPU0]        0/RSP1/CPU0 T0-SEL-B 1      PRC  1  Available

Node 0/5/CPU0:
=====
Selection point: ETH_RXMUX (1 inputs, 1 selected)
Last programmed 06:49:27 ago, and selection made 06:49:27 ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : T0-SEL-B T4-SEL-A
  Router scoped   : None
Uses frequency selection
S  Input                      Last Selection Point      QL  Pri  Status
== =====
1  HundredGigE0/5/0/2        n/a                      PRC  1  Available

```

```

Selection point: LC_TX_SELECT (1 inputs, 1 selected)
Last programmed 6d04h ago, and selection made 6d04h ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped  : None
  Router scoped   : None
Uses frequency selection
Used for local line interface output
S  Input                      Last Selection Point          QL  Pri  Status
== =====
24 Sync1 [0/RSP1/CPU0]        0/RSP1/CPU0 T0-SEL-B 1      PRC   1  Available

```

Configuring Leap Seconds

This procedure describes the steps involved in leap second configuration. The configuration can be executed in two ways:

- By directly providing the **UTC offset value** in the command.
- By providing the path to a **file** in the command, where the UTC offset information is stored (or available).

1. To enter the PTP configuration mode, use **ptp** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# ptp
```

2. To configure the UTC offset information by providing the offset value directly, use **{ utc-offset {baseline | date } { offset-value } }** command in the ptp configuration mode.
 - Using the **baseline** keyword, enter a positive number for the *offset-value* (it is assumed that a negative UTC offset will not be required).
 - **OR** provide a date (in YYYY-MM-DD format) and the *offset-value*. UTC offset used by PTP will be updated on this date. If you do not specify a date, the configuration is applied for the current day, at midnight.



Note In both cases, providing the UTC *offset-value* directly in the command is mandatory.

```
RP/0/RSP0/CPU0:router(config-ptp)# utc-offset baseline 37
```

```
RP/0/RSP0/CPU0:router(config-ptp)# utc-offset 2018-07-01 38
```

3. To configure UTC offset information by providing the path to a file containing the UTC offset information, use **{ utc-offset leap-second-file {file-path} } [poll-frequency days]** command in the ptp configuration mode. Optionally, you can provide a polling frequency in days, at which to poll the file for changes. If a frequency for polling is not specified, the file will be polled on the day the file is set to expire.



Note The format of this file must be based on the canonical list present at <http://www.ietf.org/timezones/data/leap-seconds.list>.

```
RP/0/RSP0/CPU0:router(config-ptp)# utc-offset leap-second-file http://<remote-url>

RP/0/RSP0/CPU0:router(config-ptp)# utc-offset leap-second-file file://<local-path>
poll-frequency 7
```

Verification

To display the current UTC offset value, use **show ptp utc-offset** command.

```
RP/0/RSP0/CPU0:router# show ptp utc-offset

Current offset: +36 seconds (not valid)
Pending leap seconds:
  From 2017-01-01 offset will be +37 seconds
  From 2018-07-01 offset will be +38 second
  From 2019-07-01 offset will be +39 seconds
Source: User-configured
```

To display the current UTC offset value and related details, use **show ptp utc-offset detail** command.

```
RP/0/RSP0/CPU0:router# show ptp utc-offset detail

Current offset: +36 seconds (valid)
Known leap seconds:
  From 1996-01-01 offset was +30 seconds
  From 1997-07-01 offset was +31 seconds
  From 1999-01-01 offset was +32 seconds
  From 2006-01-01 offset was +33 seconds
  From 2009-01-01 offset was +34 seconds
  From 2012-07-01 offset was +35 seconds
  From 2015-07-01 offset was +36 seconds
  From 2017-01-01 offset will be +37 seconds
Source: file:///test/xxxuser/leapsec/test/list-leap-seconds.list
Expiry date: 2017-12-28
```

Configuring Multiple PTP Profile Interoperability

This procedure describes the steps involved in configuring interoperability for PTP profiles.

1. To configure an interface and then enter the PTP configuration mode, use **interface** and **ptp** commands respectively.

```
RP/0/RSP0/CPU0:router(config)# interface tenGigE 0/0/0/9

RP/0/RSP0/CPU0:router(config-if)# ptp
```

2. To configure PTP profile, use **profile** command in the interface-ptp configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# profile interop-slave
```

3. To configure interoperability, use **interop** command in the interface-ptp configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# interop
```

4. To configure the Telecom profile and domain number to interoperate with, use **profile {profile-type}** and **domain domain-number** commands in the interface-ptp-interop configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp-interop)# profile g.8275.2
```

```
RP/0/RSP0/CPU0:router(config-if-ptp-interop)# domain 44
```

5. To enable conversion of packets on ingress, use **ingress-conversion** command in the interface-ptp-interop configuration mode. The **ingress-conversion** command, converts the packets received from the incoming Announce messages.

```
RP/0/RSP0/CPU0:router(config-if-ptp-interop)#  
ingress-conversion
```

6. To explicitly configure the other related parameters, use the respective commands in the interop-ingress submode.



Note Default values are used for parameters that are not explicitly configured during ingress-conversion. For example, default values will be used for parameters like **ClockAccuracy** or **OffsetScaledLogVariance** if they are not explicitly configured.

```
RP/0/RSP0/CPU0:router(config-if-ptp-interop-ingress)#  
priority1 10  
priority2 10
```

7. To enable conversion of packets on egress, use **egress-conversion** command in the interface-ptp-interop configuration mode. The **egress-conversion** command converts the packets sent through the outgoing Announce messages. The configuration is the same as for ingress conversion.

```
RP/0/RSP0/CPU0:router(config-if-ptp-interop)#  
egress-conversion
```

Verification

To display the interop conversions, use **show ptp interop** command.

```
RP/0/RSP0/CPU0:router# show ptp interop tenGigE 0/0/0/9  
Egress Conversions:  
  Profile:                               Default -> G.8275.2  
  Domain:                               0 -> 10  
  Priority1:                             1 -> 128  
  Priority2:                             100 -> 100  
  ClockClass:                           52 -> 140  
  ClockAccuracy:                         0 -> 0x21  
  OffsetScaledLogVariance:               0 -> 0x4e5d  
  
Ingress Conversions:  
  Profile:                               G.8275.2 -> Default  
  Domain:                               10 -> 0  
  Master 51.51.51.51:  
    Priority1:                             1 -> 100  
    Priority2:                             2 -> 254  
    ClockClass:                           3 -> 13
```

```
ClockAccuracy:          0x20 -> 0x20
OffsetScaledLogVariance: 0x4e5d -> 0x4e5d
```

Configuring PTP Telecom Profile Interface

This procedure describes the steps involved to create an interface for PTP ITU-T Telecom Profiles.



Note It is also possible to make these definitions within a global PTP profile and attach them to the interface using the profile command in PTP interface configuration mode.

1. To configure an interface, use **interface** *type interface-path-id* command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/1/0/1
```

2. To enter the PTP configuration mode for the given interface, use **ptp** command in the interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if)# ptp
```

3. To configure a PTP profile (or specify a previously defined profile), use **profile** *name* command in the ptp-interface configuration mode.



Note Any additional commands entered in ptp-interface configuration mode overrides the global profile settings.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# profile tele64
```

4. To configure frequency for Sync or Delay-request messages for the given ptp interface, use **sync frequency** *rate* command or **delay-request frequency** *rate* command appropriately in the ptp-interface configuration mode. The valid configurable values are **2, 4, 8, 16, 32, 64 or 128**.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# sync frequency 128
```

```
RP/0/RSP0/CPU0:router(config-if-ptp)# delay-request frequency 128
```

5. To configure duration for different PTP messages, use one of the following commands in the ptp-interface configuration mode: **announce grant-duration** *duration*, **sync grant-duration** *duration*, or **delay-response grant-duration** *duration*. The duration value can be between **60 and 1000 seconds**.



Note This duration value represents the length of grant that is requested for a port in Slave state and represents the maximum grant-duration allowed when the port is in Master state.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# announce grant-duration 120
```

```
RP/0/RSP0/CPU0:router(config-if-ptp)# sync grant-duration 120
```



```
RP/0/RSP0/CPU0:router(config-if-ptp)# delay-response grant-duration 120
```

6. To configure a timeout value, length of time by when a PTP message must be received (before PTF-lossSync is raised), use one of the following commands in the ptp-interface configuration mode: **sync timeout *timeout*** or **delay-response timeout *timeout***. The timeout value can be between **100 to 10000 micro seconds**.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# sync timeout 120
```

```
RP/0/RSP0/CPU0:router(config-if-ptp)# delay-response timeout 120
```

7. To configure a response for unicast-grant invalid-request, use **unicast-grant invalid-request {reduce | deny}** command. The response for requests with unacceptable parameters would either be denied or granted with reduced parameters.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# unicast-grant
invalid-request reduce
```

8. To configure IPv4 or IPv6 address for a PTP master, use **master {ipv4 | ipv6} *ip-address*** command in the ptp-interface configuration mode.

```
RP/0/RSP0/CPU0:router(config-if-ptp)# master ipv4 192.168.2.1
```

```
RP/0/RSP0/CPU0:router(config-if-ptp)# master ipv6 2001:DB8::1
```

9. To override the clock-class received in Announce messages from the specified Master, use **clock-class *class*** command in the ptp-master-interface configuration mode. The class values can range from **0 to 255**.

```
RP/0/RSP0/CPU0:router(config-if-ptp-master)# clock-class 2
```

Verification

To display the PTP interface details, use **show ptp interfaces brief** command.

```
RP/0/RSP0/CPU0:router# show ptp interfaces brief
```

```
Fri Feb 9 11:16:45.248 UTC
```

Intf Name	Port Number	Port State	Encap	Line State	Mechanism
BE1	1	Slave	IPv4	up	2-step DRRM
Gi0/0/0/40	2	Master	IPv4	up	2-step DRRM

To verify the configured profile details, use **show run interface *interface-name*** command.

```
RP/0/RSP0/CPU0:router# show run interface Gi0/0/0/33
```

```
Wed Feb 28 11:49:16.940 UTC
```

```
interface GigabitEthernet0/0/0/33
```

```
ptp
```

```
profile slave
```

```
multicast target-address ethernet 01-1B-19-00-00-00
```

```
transport ethernet
```

```
port state slave-only
```

```
clock operation two-step
```

```
!
```

```
ipv4 address 21.1.1.2 255.255.255.0
```

```
frequency synchronization
```

```

selection input
priority 5
wait-to-restore 0
!
```

Configuring PTP Telecom Profile Clock

This procedure describes the steps involved to configure PTP clock and its settings to be consistent with ITU-T Telecom Profiles for Frequency.

1. To enter the PTP configuration mode, use **ptp** command in the configuration mode.

```
RP/0/RSP0/CPU0:router(config)# ptp
```

2. To enter the PTP-clock configuration mode, use **clock** command in the ptp-configuration mode.

```
RP/0/RSP0/CPU0:router(config-ptp)# clock
```

3. To configure the domain-number for a PTP profile, use **domain number** command in the ptp-configuration mode. The allowed domain number range for G.8265.1 profile is between **4 and 23** and the range for G.8275.1 profile is between **24 and 43**.

```
RP/0/RSP0/CPU0:router(config-ptp)# domain 24
```

4. To configure timescale, use **timescale source** command in the ptp-clock configuration mode.

```
RP/0/RSP0/CPU0:router(config-ptp-clock)# timescale PTP
```

5. To configure the time-source that will be advertised in Announce messages, use **time-source source** command in the ptp-clock configuration mode. The allowed options are: atomic-clock, GPS, hand-set, internal-oscillator, NTP, other, PTP, and terrestrial-radio.

```
RP/0/RSP0/CPU0:router(config-ptp-clock)# time-source GPS
```

6. To exit the ptp-clock configuration mode, use **exit** command.

```
RP/0/RSP0/CPU0:router(config-ptp-clock)# exit
```

7. To configure the desired telecom profile and the clock type for the profile, use **clock profile { g.8265.1 | g.8275.1 | g.8275.2 } clock-type {T-GM | T-BC | T-TSC}** command in the ptp configuration mode.



Note The **clock-selection telecom-profile** and **clock-advertisement telecom-profile** commands are deprecated from Release 6.1.2. They are replaced by the **clock profile** command.

```
RP/0/RSP0/CPU0:router(config-ptp)# clock profile g.8275.1 clock-type T-BC
```

Verification

To display the configured PTP clock profile details, use **show run ptp** command.

```
RP/0/RSP0/CPU0:router# show run ptp !
ptp
clock
    domain 24
    profile g.8275.1 clock-type T-BC
!
profile slave
    sync frequency 16
    announce frequency 8
    delay-request frequency 16
!
profile master
    sync frequency 16
    announce frequency 8
    delay-request frequency 16
!
log
    servo events
    best-master-clock changes
!
!
```

To verify that PTP has been enabled on the router and the device is in LOCKED Phase, use **show ptp platform servo** command.

```
RP/0/RSP0/CPU0:router # show ptp platform servo

Fri Feb  9 11:16:54.568 UTC
Servo status: Running
Servo stat_index: 2
Device status: PHASE_LOCKED
Servo log level: 0
Phase Alignment Accuracy: 1 ns
Sync timestamp updated: 111157
Sync timestamp discarded: 0
Delay timestamp updated: 111157
Delay timestamp discarded: 0
Previous Received Timestamp T1: 1518155252.263409770  T2: 1518155252.263410517  T3:
1518155252.287008362  T4: 1518155252.287009110
Last Received Timestamp T1: 1518155252.325429435  T2: 1518155252.325430194  T3:
1518155252.348938058  T4: 1518155252.348938796
Offset from master: 0 secs, 11 nsecs
Mean path delay    : 0 secs, 748 nsecs
setTime():2  stepTime():1  adjustFreq():10413  adjustFreqTime():0
Last setTime: 1.000000000 flag:1  Last stepTime:-736216, Last adjustFreq:465
```

Configuring PTP Delay Asymmetry

Table 11: Feature History Table

Feature Name	Release Information	Description
PTP Delay Asymmetry	Release 7.3.1	<p>Any delays on Precision Time Protocol (PTP) paths can impact PTP accuracy and in turn impact clock settings for all devices in a network. This feature allows you to configure the static asymmetry such that the delay is accounted for and the PTP synchronization remains accurate.</p> <p>The delay-symmetry command is introduced for this feature.</p>

Configure PTP delay asymmetry to offset the static delays on a PTP path that occur due to different route selection for forward and reverse PTP traffic. Delays can also be due to any node having different delay for ingress or egress path. These delays can impact PTP accuracy due to the asymmetry in PTP. With this feature, you can enable a higher degree of accuracy in the PTP server performance leading to better synchronization between real-time clocks of the devices in a network. Better synchronization between the clocks of different devices in a network leads to a network that performs efficiently.

Configuration of this delay asymmetry provides an option to configure static delays on a client clock for every server clock. You can configure this value in microseconds and nanoseconds. Configured PTP delay asymmetry is also synchronized with the Servo.



Note If you configure multiple PTP delay asymmetries for the same PTP profile, the latest PTP delay asymmetry that you configure is applied to the PTP profile.

A positive value indicates that the server-to-client propagation time is longer than the client-to-server propagation time, and conversely for negative values.

Supported PTP Profiles

The following PTP profiles support the configuration of PTP delay asymmetry:

- PTP over IP (G8275.2 or default profile)
- PTP over L2 (G8275.1)

Restrictions

- PTP delay asymmetry can be configured only on the PTP port of the grandmaster clock, which can either be a boundary clock or an ordinary clock.

- For G875.1 and G875.2 PTP profiles, PTP delay asymmetry is supported for both, client port and dynamic port that act as a client.
- PTP delay asymmetry is supported for fixed cable delay compensation and not for variable delay in the network.
- PTP delay asymmetry can be configured within the range of 3 microseconds and -3 microseconds or 3000 nanoseconds and -3000 nanoseconds.
- Fixed delay can be measured by using any test and measurement tool. Fixed delay can be compensated by using the positive or negative values. For example, if the fixed delay is +10 nanoseconds, configure -10 nanoseconds to compensate the fixed delay.

Configuration

To configure PTP delay asymmetry:

1. Configure an interface with PTP.
2. Configure PTP delay asymmetry on the client side.

Configuration Example

```
/* Configure an interface with PTP. */
Router# configure
Router(config)# interface HundredGigE 0/1/0/0
Router(config-if)# ptp

/* Configure PTP delay asymmetry on the client side. */
Router(config-if-ptp-master)# delay-asymmetry 3 microseconds
Router(config-if-ptp-master)# commit
```

Running Configuration

```
interface preconfigure HundredGigE 0/1/0/0
 ptp
 delay-asymmetry 3 microseconds
```

Verification

To verify if PTP delay asymmetry delay is applied, use the **show ptp foreign-masters** command:

```
Router# show ptp foreign-masters
Sun Nov 1 10:19:21.874 UTC
Interface HundredGigE0/1/0/0 (PTP port number 1)
IPv4, Address 209.165.200.225, Unicast
Configured priority: 1
Configured clock class: None
Configured delay asymmetry: 3 microseconds <----- configured variable delay asymmetry value
Announce granted: every 2 seconds, 300 seconds
Sync granted: 16 per-second, 300 seconds
Delay-resp granted: 16 per-second, 300 seconds
Qualified for 2 minutes, 45 seconds
Clock ID: 80e01dffffe8ab73f
Received clock properties:
Domain: 0, Priority1: 128, Priority2: 128, Class: 6
Accuracy: 0x22, Offset scaled log variance: 0xcd70
Steps-removed: 1, Time source: GPS, Timescale: PTP
Frequency-traceable, Time-traceable
```

```
Current UTC offset: 37 seconds (valid)
Parent properties:
Clock ID: 80e01dffffe8ab73f
Port number: 1
```

To validate the approximate compensated delay value, use the **show ptp platform servo** command:

```
Router# show ptp platform servo
Sat Nov 1 15:17:14.611 UTC
Servo status: Running
Servo stat_index: 2
Device status: PHASE_LOCKED
Servo Mode: Non Hybrid
Servo log level: 0
Phase Alignment Accuracy: -3 ns
Sync timestamp updated: 54754
Sync timestamp discarded: 0
Delay timestamp updated: 55196
Delay timestamp discarded: 0
Previous Received Timestamp T1: 1563984472.036333938 T2: 1563984472.036334935 T3:
1563984472.077066895 T4: 1563984472.077067478
Last Received Timestamp T1: 1563984472.100355188 T2: 1563984472.100356182 T3:
1563984472.139059682 T4: 1563984472.139060266
Offset from master: 0 secs, 6 microsecs <----- compensated value, showing 6
microseconds because actual fixed delay is 3 microseconds, we configured 3 microseconds and
router detected it as fixed delay and tries to compensate.
Mean path delay : 0 secs, 771 nsecs
setTime():1 stepTime():0 adjustFreq():4278
Last setTime: 1563981048.000000000 flag:0 Last stepTime:0 Last adjustFreq:51511
```

Double Failure Clock Class Over-ride

Table 12: Feature History Table

Feature Name	Release Information	Feature Description
PTP Double Failure Clock Class	Release 7.7.1	<p>This feature enables you to configure a clock class that will over-ride the existing class during a state of double-failure where PTP and SyncE are lost.</p> <p>This feature introduces the double-failure-clock-class command.</p>

PTP Holdover Traceability suppression

Table 13: Feature History Table

Feature Name	Release Information	Feature Description
PTP Holdover Traceability Suppression	Release 7.3.1	When a device which is configured as a Boundary clock (T-BC) loses synchronization with a quality Primary clock, to ensure that the downstream nodes continue to receive the configured clock class for a specified duration, and it's traceable you can configure this feature.

When the device loses synchronization with a quality Primary clock, to ensure the downstream nodes continue to receive the configured clock class, and it is traceable you can configure this feature.

This feature enables the device which is configured as a boundary clock (T-BC) with PTP Profiles G.8275.1 or G.8275.2 to send out the configured clock-class as holdover clock-class and the time traceability flag to be set as TRUE for the specified duration. This is to ensure the downstream nodes do not have an impact as this is a deviation from prescribed G.8275.1 ITU-T standards.



Note

- There will be PTP flaps during switchovers or ISSU as the PTP holdover timer is running on the active RSP.
- Once the configured holdover override duration has lapsed and the device is unable to receive quality Primary clock within this duration, the device will send the prescribed default clock class of 165 and the traceability flag will be set as FALSE to advertise loss of clock to downstream nodes.

Configuring PTP Holdover traceability suppression

This section describes how to configure the PTP holdover traceability suppression feature:

```
Router# config
Router(config)# ptp
Router(config-ptp)# holdover-spec-duration 1000
Router(config-ptp)# holdover-spec-clock-class 135
Router(config-ptp)# uncalibrated-traceable-override
Router(config-ptp)# holdover-spec-traceable-override
```

Configuration Examples

Slave Configuration Example

The following example shows a PTP slave configuration:

```
interface TenGigE 0/1/0/5
 ptp
  profile tp64
  transport ipv4
  port state slave-only
  master ipv4 1.7.1.2
  !
  announce interval 1
  !
  ipv4 address 1.7.1.1 255.255.255.0
  !
```

Master Configuration Example

This example shows a PTP master configuration:

```
ptp
 profile tp64
 transport ipv4
 announce interval 1
 !
 ipv4 address 1.7.1.2 255.255.255.0
 !
```

PTP Hybrid Mode Configuration Example

This example shows the configuration of PTP hybrid mode:

```
ptp
 time-of-day priority 10
 !
 interface GigabitEthernet0/1/1/0
  ptp
   transport ipv4
   port state slave-only
   master ipv4 192.168.52.38
   !
   sync frequency 64
   announce interval 1
   delay-request frequency 64
   !
 interface GigabitEthernet 0/1/0/1
  ipv4 address 192.168.52.41 255.255.255.0
  speed 100
```



```
frequency synchronization
selection input
priority 10
wait-to-restore 0
ssm disable
time-of-day-priority 100
!
```

ITU-T Telecom Profiles Configuration Examples

Master global configuration for the telecom profile:

-- For G.8265.1 profile --

```
ptp
clock
domain 4
profile g.8265.1
!
profile master
transport ipv4
sync frequency 16
announce interval 1
delay-request frequency 16
interface gi 0/2/0/4
ptp
profile master
transport ipv4
clock operation two-step
!
ipv4 address 17.1.1.1/24
```

-- For G.8275.1 profile --

```
ptp
clock
domain 24
profile g.8275.1
!
profile master
transport ethernet
sync frequency 16
announce interval 1
delay-request frequency 16
interface gi 0/2/0/4
ptp
profile master
transport ethernet
multicast target-address ethernet 01-1B-19-00-00-00
clock operation two-step
!
ipv4 address 17.1.1.1/24
```

Slave global configuration for the telecom profile:

-- For G.8265.1 profile --

```

ptp
 clock
  domain 4
  profile g.8265.1
  !
  profile slave
  transport ipv4
  sync frequency 16
  announce interval 1
  delay-request frequency 16
interface gi 0/1/0/0
 ptp
  profile slave
  transport ipv4
  Master ipv4 18.1.1.1
  port state slave-only
  !
  clock operation two-step
  !
  ipv4 address 18.1.1.2/24

```

-- For G.8275.1 profile --

```

ptp
 clock
  domain 24
  profile g.8275.1 clock-type T-TSC
  !
  profile slave
  transport ethernet
  sync frequency 16
  announce interval 1
  delay-request frequency 16
interface gi 0/1/0/0
 ptp
  profile slave
  transport ethernet
  multicast target-address ethernet 01-1B-19-00-00-00
  !
  clock operation two-step
  !
  ipv4 address 18.1.1.2/24

```

-*- For G.8275.2 profile -*-

```

ptp
 clock
  domain 44
  profile g.8275.2 clock-type T-TSC
  !
  profile slave
  transport ipv6
  port state slave-only
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !
  log
  servo events
  best-master-clock changes

```

```

!
!
interface GigabitEthernet0/2/0/12
 ptp
  profile slave
  master ipv6 30::2
!
!
ipv6 address 30::1/64
!

```

Global configuration with clock type as T-Boundary Clock (T-BC) for the telecom profile:

-- For G.8275.1 profile --

```

ptp
 clock
 domain 24
 profile g.8275.1 clock-type T-BC
!
 profile master
  transport ethernet
  sync frequency 16
  announce interval 1
  delay-request frequency 16
 exit
 profile slave
  transport ethernet
  sync frequency 16
  announce interval 1
  delay-request frequency 16
 exit
interface gi 0/2/0/4
 ptp
  profile slave
  transport ethernet
  multicast target-address ethernet 01-1B-19-00-00-00
  !
  clock operation two-step
  !
  ipv4 address 17.1.1.2/24
interface gi 0/2/0/0
 ptp
  profile master
  transport ethernet
  multicast target-address ethernet 01-1B-19-00-00-00
  clock operation two-step
  !
  ipv4 address 18.1.1.1/24

```



Note When G.8275.1 profile is configured on a 100G interface, keywords **commit replace** and **rollback config last 1** does not work and the router configuration rollback fails entirely. Use **rollback config last 1 best-effort** instead.

```

--*-- For G.8275.2 profile --*--
ptp

```

```
clock
  domain 44
  profile g.8275.2 clock-type T-BC
  !
profile slave
  transport ipv6
  port state slave-only
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !
profile master
  transport ipv6
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !
log
  servo events
  best-master-clock changes
  !
  !

interface GigabitEthernet0/2/0/11
  ptp
  profile master
  !
  ipv6 address 30::1/64
  !

interface GigabitEthernet0/2/0/12
  ptp
  profile slave
  master ipv6 40::2
  !
  !
  ipv6 address 40::1/64
  !
```



CHAPTER 9

Network Synchronization Design Best Practices

This chapter provides guidelines and best practices to follow when designing timing requirements for your network.

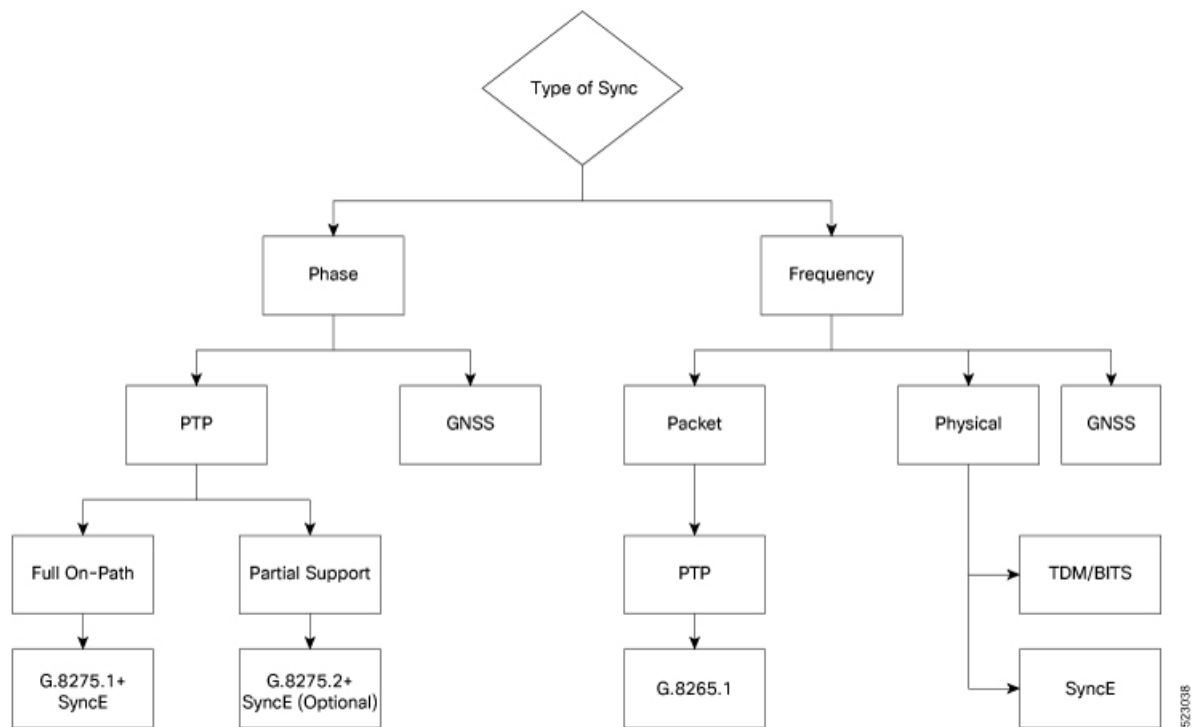
- [Network Synchronization Design Best Practices, on page 161](#)

Network Synchronization Design Best Practices

The synchronization of a network is essential for ensuring that all devices in a network run on the same clock time. It also ensures that the applications in the network function correctly. To design your network synchronization accurately, you must have a clear understanding of your network requirements, timing budget, application requirements, and the desired level of synchronization accuracy. This section describes some best practices to follow when designing your network synchronization.

Network Synchronization Decision Tree

Use the network synchronization decision tree for determining the appropriate synchronization solution for your network deployment. Network synchronization helps in ensuring that the network operates with accurate and synchronized time.



General Guidelines for Successful Synchronization Deployments

Network synchronization is crucial for maintaining reliable and efficient network operations, ensuring data integrity, complying with regulations, and facilitating troubleshooting and management tasks. The following guidelines help in deploying successful network synchronization for your network:

- Ensure that you use a standards-based solution designed for your need. For example, use the correct profile.
- Configure the appropriate clock source for your network. It can be Global Navigation Satellite System (GNSS) based such as a Global Positioning System (GPS) clock, or a Precision Time Protocol (PTP) grandmaster clock.
 - Frequency synchronization requires Building Integrated Timing Supply (BITS) or synchronous Ethernet, and Phase synchronization requires PTP and/or GNSS.
- Use a combination of GNSS over the air and/or PTP or synchronous Ethernet over transport.

For more information on [SyncE](#) and [PTP](#), refer to *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

- Set up the synchronization protocols that are required, which includes PTP, Network Time Protocol (NTP), or synchronous Ethernet.
 - NTP uses the system clock for logging events in the system, or to show clock output, whereas PTP and GNSS work on the IEEE 1588 hardware clock in the system.
 - The NTP clock of a node can't be used to synchronize the downstream network using PTP. However, a node can synchronize its NTP clock with the available PTP or GNSS clock.

**Note**

Most NTP implementations are software-based. Software-based time synchronization is less accurate than hardware-based synchronization, but it's still useful for applications where low levels of accuracy, such as 10's or 100's of milliseconds, are acceptable.

- Use PTP for phase synchronization in the absence of a GNSS.
- Synchronous Ethernet (SyncE) is a recommendation from ITU Telecommunication Standardization Sector (ITU-T) on how to deliver a frequency in a network. If you require a frequency-only synchronization solution, use SyncE instead of PTP.
- Configure the appropriate synchronization profiles and preferences for your network. It might include the accuracy, priority, and other parameters that determine how your network handles synchronization events.
- Design your network for phase synchronization with optimal time error budgets.
 - Use boundary clocks to reduce time error and to reset Packet Delay Variation (PDV).
 - Ensure that PTP awareness is implemented consistently throughout, including the transport system, and that boundary clocks accurately transmit time to minimize accumulated time error.

- For phase synchronization, use a hybrid clock that incorporates both SyncE and PTP.

For more information on [PTP Hybrid Mode](#), refer to *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

- Reduce the number of hops:
 - Distribute sources of time to meet the budget. If you have too many hops, install a GNSS receiver further out into the network.
 - Don't centralize two Primary Reference Time Clocks (PRTC) and Telecom Grandmasters (T-GM) in two different locations and try to run a synchronization signal accurately across the whole network.
- Minimize Packet Delay Variation (PDV) and jitter. Ensure that microwaves, Gigabit-capable Passive Optical Networks (GPON), Digital Subscriber Line (DSL), and Dense Wavelength Division Multiplexing (DWDM) are PTP aware.
- Monitor your synchronization deployment to ensure that it's functioning correctly and meeting your desired level of accuracy.

For more information, refer to [Verifying the Frequency Synchronization Configuration](#) in the *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

- Be aware of any relevant industry standards and practices when deploying synchronization.

Guidelines for Phase Synchronization Deployments

Follow these guidelines for phase synchronization deployments.

- Set up the necessary network infrastructure to support phase synchronization. It includes installing timing devices such as GPS receivers, synchronous Ethernet interfaces, and timing servers.

- Configure the phase synchronization protocols such as setting up PTP as appropriate.
- As best practice, use the G.8275.1 telecommunication profile standard with complete on-path support, including Layer-2 multicast in combination with SyncE.
- Minimize phase time error by performing the following tasks:
 - Remove asymmetric routing issues.
 - Reduce the number of hops, unless telecommunication grandmaster (T-GM) clocks are deployed in the preaggregation network.
 - Decrease PDV or packet jitter.
- If you use IP protocols for PTP, you can run into issues with rerouting, asymmetric routing, Equal Cost Multi-Path (ECMP), bundles, and so on.
- If you need tight timing budgets over many hops, ensure that your hardware supports the highest levels of clock accuracy.
- For GNSS deployments:
 - Meet all the requirements for cable and antenna installations.
 - Consult with a professional if you don't have experience with GNSS installation and calibration.
- Make sure that your deployment is working as intended. Monitor it regularly to identify any potential issues.
- Consult with Cisco technical support if you encounter any issues or have questions.

**Note**

When PTP is used with MACsec, achieving high accuracy can be challenging. PTP requires exact timestamping to maintain tight network synchronization. MACsec affixes and detaches a header that is between 24–32 bytes in size. This process can lead to significant inconsistencies in the time delays between where the link is connected and the location where the egress timestamps are applied.

PTP over IP Network Design

When using networks to carry frequency over Precision Time Protocol over Internet Protocol (PTPoIP), the goal is to minimize Packet Delay Variation (PDV) by reducing the number of hops. Use the following guidelines:

- The placement of the telecom grandmaster (T-GM) clock plays an important role in ensuring that the network operates within your timing budget. For example, place a pair of T-GM clocks in a centralized location only if the network has a small number of hops. In larger networks with multiple hops, it may be necessary to distribute T-GM clocks throughout the network to ensure proper timing management at each hop.
- Use a dedicated frequency synchronization protocol such as synchronous Ethernet or 1588v2, which is designed specifically to maintain precise frequency synchronization between devices.

- Use the G.8265.1 standard. Frequency synchronization using the G.8265.1 standard is a way to make sure multiple devices on a network are operating at the same frequency, allowing for more accurate and reliable communication.
- Configure Quality of Service (QoS) policies to prioritize network traffic and reduce delays. This can be done by using traffic shaping, traffic policing, and queue management.

Selecting the Correct Profile For Network Synchronization

G.8275.1 PTPoE

G.8275.1 is a technical specification standard for Precision Time Protocol over Ethernet (PTPoE). It defines how you can use the Precision Time Protocol (PTP) to synchronize clocks over Ethernet networks with layer 2 multicast. PTPoE is an extension of PTP that allows it to be used over Ethernet networks. It's used in applications where precise time synchronization is required.

For more information, refer to [G.8275.1](#) in the *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

G.8275.2 PTPoIP

G.8275.2 is a technical specification standard for Precision Time Protocol over Internet Protocol (PTPoIP). It defines the use of the Precision Time Protocol (PTP) over packet-based networks such as Internet Protocol (IP) networks, to provide precise time synchronization of network devices.

For more information, refer to [G.8275.2](#) in the *System Management Configuration Guide for Cisco ASR 9000 Series Routers*.

Feature Adaptability on Each Profile

The following table lists the adaptability of features on each profile:

Feature	G.8275.1 PTPoE	G.8275.2 PTPoIP
Network Model	Full on-path support	Partial on-path support
IP Routing	Not applicable	Can cause issues in rings and asymmetry from a number of causes
Transit Traffic	Not allowed	Can result in jitter and asymmetry
Performance	Optimal	Variable
Configuration Model	Physical port	L3 device
PTP over Bundles	No issues	Work in progress for Telecom Boundary Clocks (T-BC)
Asymmetry	Reduced due to T-BC on every node	Optimal when deployed as a Partial Support Telecom Boundary Clock (T-BC-P)

Feature	G.8275.1 PTPoE	G.8275.2 PTPoIP
PDV/Jitter	Reduced due to T-BC on every node	Optimal when deployed as a T-BC-P

Reducing Asymmetry

Asymmetry occurs in a PTP unaware network for the following scenarios:

- When routing large networks, complex topologies, rings, and Equal-cost multi-path (ECMP)
- When using PTP unaware transit nodes, especially with varying traffic patterns
- In the transport layer such as Passive Optical Network (PON), cable, DWDM, and complex optics



Note Every 2 seconds of asymmetry results in 1 microsecond of time error.

To reduce asymmetry in a PTP unaware network:

- Use QoS: QoS can help reduce asymmetry in an unaware network.
- Implement Telecom Boundary Clocks (T-BC): T-BCs can handle asymmetry in the nodes when implemented correctly.

Reducing Packet Delay Variation

To reduce the effects of Packet Delay Variation (PDV) on PTP clock recovery, you must have a steady layer of packets that arrive in minimum time.

- Implement Telecom Boundary Clocks (T-BC) in the PTP unaware node. T-BC introduces a time reference to the PTP unaware node, which then synchronizes its clock with the T-BC.
- Use a high-quality network connection between the T-BC and the PTP unaware node. A high-quality network connection, such as a dedicated fiber link, can help reduce PDV due to network impairments.

Remediating Transport Asymmetry

Transport asymmetry occurs when data is transported at varying rates in different directions over a communication link, leading to an imbalance in transport. To correct this issue:

- Ensure that your transport layer is PTP aware.
- In optical devices, use a wavelength division multiplexing (WDM) technology such as Optical Service Channel (OSC) for managing your fiber optic infrastructure effectively.

Synchronizing Across Networks

To avoid synchronization issues when connecting to other mobile networks:

- Make sure to align all mobile networks to a common source of time. For example, align mobile networks to the Coordinated Universal Time (UTC) from a Global Navigation Satellite System (GNSS) such as Global Positioning System (GPS).
- Monitor your clocks at the interconnect points.

**Note**

In 5G networks, using standalone GNSS receivers at every radio site may not provide the sub-100 nanosecond accuracy required for the timing requirements of Fronthaul radio systems.
