



Implementing NTP on Cisco IOS XR Software

Network Time Protocol (NTP) is a protocol designed to time-synchronize devices within a network. The 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 new and revised tasks you need to implement NTP on your Cisco IOS XR network.



Note

For more information about NTP on the Cisco IOS XR software and complete descriptions of the NTP commands listed in this module, you can refer to the [“Related Documents”](#) section of this module. To locate documentation for other commands that might appear in the course of running a configuration task, search online in the Cisco IOS XR software master command index.

Feature History for Implementing NTP on Cisco IOS XR Software Contents

Release	Modification
Release 2.0	This feature was introduced on the Cisco CRS-1.
Release 3.0	No modification.
Release 3.2	Support was added for the Cisco XR 12000 Series Router.
Release 3.3.0	No modification.
Release 3.4.0	No modification.
Release 3.5.0	No modification.

Contents

- [Prerequisites for Implementing NTP on Cisco IOS XR Software, page 102](#)
- [Information About Implementing NTP on Cisco IOS XR Software, page 102](#)
- [How to Implement NTP on Cisco IOS XR Software, page 103](#)
- [Configuration Examples for Implementing NTP on Cisco IOS XR Software, page 118](#)
- [Additional References, page 122](#)

Prerequisites for Implementing NTP on Cisco IOS XR Software

The following prerequisites are required to implement NTP in your network operating center (NOC):

- You must be in a user group associated with a task group that includes the proper task IDs for CDP commands. For detailed information about user groups and task IDs, see the *Configuring AAA Services on Cisco IOS XR Software* module of the *Cisco IOS XR System Security Configuration Guide*.
- You must have connectivity with at least one server that is running NTP.

Information About Implementing NTP on Cisco IOS XR Software

To implement NTP, you need to understand the following concept:

- [“NTP Functional Overview” section on page 102](#)

NTP Functional Overview

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 will never synchronize to a machine that is not in turn 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.

A number of 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.

However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity, because each machine can simply be configured to send or receive broadcast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

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.

How to Implement NTP on Cisco IOS XR Software

This section contains the following procedures:

- [Configuring Poll-Based Associations, page 103](#) (optional)
- [Configuring Broadcast-Based NTP Associations, page 105](#) (optional)
- [Configuring NTP Access Groups, page 107](#) (optional)
- [Configuring NTP Authentication, page 109](#) (optional)
- [Disabling NTP Services on a Specific Interface, page 111](#) (optional)
- [Configuring the Source IP Address for NTP Packets, page 113](#) (optional)
- [Configuring the System as an Authoritative NTP Server, page 114](#) (optional)
- [Updating the Hardware Clock, page 116](#) (optional)
- [Verifying the Status of the External Reference Clock, page 117](#) (optional)

Configuring Poll-Based Associations

This task explains how to configure poll-based NTP associations.

**Note**

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

Poll-Based Associations

Networking devices running NTP can be configured to operate in variety of association modes when synchronizing time with reference time sources. There are two ways that a networking device can obtain time information on a network: by polling host servers and by listening to NTP broadcasts. In this task, we will focus on the poll-based association modes. Broadcast-based NTP associations will be discussed in the next task, “Configuring Broadcast-Based NTP Associations.”

The following are two most commonly used, poll-based association modes:

- 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 is a number of 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*.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **server** *ip-address* [**version number**] [**key key-id**] [**minpoll interval**] [**maxpoll interval**] [**source interface-type interface-instance**] [**prefer**]
4. **peer** *ip-address* [**version number**] [**key key-id**] [**minpoll interval**] [**maxpoll interval**] [**source interface-type interface-instance**] [**prefer**]
5. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	<pre>server ip-address [version number] [key key-id] [minpoll interval] [maxpoll interval] [source interface-type interface-instance] [prefer]</pre> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# server 172.16.22.44 minpoll 8 maxpoll 12</pre>	Forms a server association with another system.
Step 4	<pre>peer ip-address [version number] [key key-id] [minpoll interval] [maxpoll interval] [source interface-type interface-instance] [prefer]</pre> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# peer 192.168.22.33 minpoll 8 maxpoll 12 source pos 0/0/0/1</pre>	Forms a peer association with another system.
Step 5	<pre>end OR commit</pre> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config)# end OR RP/0/RP0/CPU0:router(config)# 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 Associations

This task explains how to configure broadcast-based NTP associations.



Note

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

Broadcast-Based NTP Associations

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.

When a networking device is operating in the *broadcastclient mode*, it does not engage in any polling. Instead, it listens for NTP broadcast packets transmitted by broadcast time servers. Consequently, time accuracy can be marginally reduced, because time 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 *broadcastclient 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.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **broadcastdelay** *microseconds*
4. **interface** *type instance*
5. **broadcast client**
6. **broadcast** [*destination ip-address*] [**key** *key-id*] [**version** *number*]
7. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	broadcastdelay <i>microseconds</i> Example: RP/0/RP0/CPU0:router(config-ntp)# broadcastdelay 5000	Adjusts the estimated round-trip delay for NTP broadcasts.
Step 4	interface <i>type instance</i> Example: RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/1/0/0	Enters NTP interface configuration mode.
Step 5	broadcast client Example: RP/0/RP0/CPU0:(config-ntp-int)# broadcast client	Configures the specified interface to receive NTP broadcast packets.

	Command or Action	Purpose
Step 6	<p>broadcast [destination <i>ip-address</i>] [key <i>key-id</i>] [version <i>number</i>]</p> <p>Example: RP/0/RP0/CPU0:(config-ntp-int)# broadcast destination 10.50.32.149</p>	Configures the specified interface to send NTP broadcast packets.
Step 7	<p>end OR commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end OR RP/0/RP0/CPU0:router(config)# commit</p>	<p>Saves configuration changes.</p> <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 Access Groups

This task explains how to configure NTP access groups.



Note

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

NTP Access Groups

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:

- 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.
- 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.
- serve-only**—Allows only time requests from a system whose address passes the access list criteria.
- 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. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	<pre>access-group {peer query-only serve serve-only} access-list-name</pre> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# access-group peer access1</p>	Creates an access group and applies a basic IP access list to it.
Step 4	<pre>end or commit</pre> <p>Example: RP/0/RP0/CPU0:router(config)# end or RP/0/RP0/CPU0:router(config)# commit </p>	<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 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.

NTP Authentication

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. Cryptographic checksum keys are generated using the MD5 Message Digest Algorithm and are embedded into the NTP synchronization packet that is sent to a receiving client. When a packet is received by a client, its cryptographic checksum key is decrypted and checked against a list of trusted keys. If authentication is enabled and a key is trusted, the system is allowed to sync to the server that uses this key in its packets.

It is important to note that the encryption and decryption processes used in NTP authentication can be very CPU-intensive and can seriously degrade the accuracy of the time that is propagated within a network. If your network setup permits a more comprehensive model of access control, you should consider the use of the access-list-based form of control instead.

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. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	authenticate Example: RP/0/RP0/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/RP0/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. Currently the only key type supported is md5.

	Command or Action	Purpose
Step 5	<p>trusted-key <i>key-number</i></p> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# trusted-key 42</p>	<p>Defines trusted authentication keys.</p> <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	<p>end or commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end or RP/0/RP0/CPU0:router(config)# commit</p>	<p>Saves configuration changes.</p> <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

This task explains how to disable 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

- configure**
- ntp**
- no interface** *type instance*
or
interface *type instance* **disable**
- end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure</p> <p>Example: RP/0/RP0/CPU0:router# configure</p>	Enters global configuration mode.
Step 2	<p>ntp</p> <p>Example: RP/0/RP0/CPU0:router(config)# ntp</p>	Enters NTP configuration mode.
Step 3	<p>no interface <i>type instance</i> OR interface <i>type instance</i> disable</p> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# no interface pos 0/0/0/1 OR RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/0/0/1 disable</p>	Disables NTP services on the specified interface.
Step 4	<p>end OR commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end OR RP/0/RP0/CPU0:router(config)# commit</p>	<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 Source IP Address for NTP Packets

This task explains how to configure the source IP address for NTP packets.

When the system sends an NTP packet, the source IP address is normally set to the address of the interface through which the NTP packet is sent.



Note

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

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **source** *interface-type interface-instance*
4. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	<p>source <i>interface-type interface-instance</i></p> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# source POS 0/0/0/1</p>	<p>Configures an interface from which the IP source address will be taken.</p> <p>Note This interface will be 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 parameter on the peer or server command shown in the “Configuring Poll-Based Associations” task.</p>
Step 4	<p>end or commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end or RP/0/RP0/CPU0:router(config)# commit</p>	<p>Saves configuration changes.</p> <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 the System as an Authoritative NTP Server

This task explains how to configure the router 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

- configure**
- ntp**
- master** *stratum*
- end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure</p> <p>Example: RP/0/RP0/CPU0:router# configure</p>	Enters global configuration mode.
Step 2	<p>ntp</p> <p>Example: RP/0/RP0/CPU0:router(config)# ntp</p>	Enters NTP configuration mode.
Step 3	<p>master stratum</p> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# master 9</p>	<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 timekeeping if the machines do not agree on the time.</p>
Step 4	<p>end OR commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end OR RP/0/RP0/CPU0:router(config)# commit</p>	<p>Saves configuration changes.</p> <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.

Updating the Hardware Clock

This task explains how to configure the hardware clock to be periodically updated from the software clock running NTP.

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 any device using NTP, because the time and date on the software clock (set using NTP) is more accurate than the hardware clock, because 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. **end**
or
commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.

	Command or Action	Purpose
Step 3	<p>update-calendar</p> <p>Example: RP/0/RP0/CPU0:router(config-ntp)# update-calendar</p>	Configures the system to update its hardware clock from the software clock at periodic intervals.
Step 4	<p>end or commit</p> <p>Example: RP/0/RP0/CPU0:router(config)# end or RP/0/RP0/CPU0:router(config)# commit</p>	<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.

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

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	show ntp associations [detail] [location node-id] Example: RP/0/RP0/CPU0:router# show ntp associations	Displays the status of NTP associations.
Step 2	show ntp status [location node-id] Example: RP/0/RP0/CPU0:router# show ntp status	Displays the status of NTP.

Examples

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

```
RP/0/0/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 (synced), # master (unsynced), + selected, - candidate, ~ configured
```

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

```
RP/0/0/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 22 2004)
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 on Cisco IOS XR Software

This section contains the following examples:

- [Configuring Poll-Based Associations: Example, page 119](#)
- [Configuring Broadcast-Based Associations: Example, page 119](#)
- [Configuring NTP Access Groups: Example, page 119](#)
- [Configuring NTP Authentication: Example, page 120](#)
- [Disabling NTP on an Interface: Example, page 120](#)
- [Configuring the Source IP Address for NTP Packets: Example, page 121](#)
- [Configuring the System as an Authoritative NTP Server: Example, page 121](#)
- [Updating the Hardware Clock: Example, page 121](#)

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:

```
RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# server 10.0.2.1 minpoll 5 maxpoll 7
RP/0/RP0/CPU0:router(config-ntp)# peer 192.168.22.33

RP/0/RP0/CPU0:router(config-ntp)# server 172.19.69.1
```

Configuring Broadcast-Based Associations: Example

The following example shows an NTP client configuration in which Gigabit Ethernet 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:

```
RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# interface GigabitEthernet 0/2/0/0
RP/0/RP0/CPU0:router(config-ntp-int)# broadcast client
RP/0/RP0/CPU0:router(config-ntp-int)# exit
RP/0/RP0/CPU0:router(config-ntp)# broadcastdelay 2
```

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

```
RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# interface GigabitEthernet 0/2/0/2
RP/0/RP0/CPU0:router(config-ntp-int)# 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.

```
RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# peer 10.1.1.1
RP/0/RP0/CPU0:router(config-ntp)# peer 10.1.1.1
RP/0/RP0/CPU0:router(config-ntp)# peer 10.2.2.2
RP/0/RP0/CPU0:router(config-ntp)# peer 10.3.3.3
RP/0/RP0/CPU0:router(config-ntp)# peer 10.4.4.4
RP/0/RP0/CPU0:router(config-ntp)# peer 10.5.5.5
RP/0/RP0/CPU0:router(config-ntp)# peer 10.6.6.6
RP/0/RP0/CPU0:router(config-ntp)# peer 10.7.7.7
RP/0/RP0/CPU0:router(config-ntp)# peer 10.8.8.8
RP/0/RP0/CPU0:router(config-ntp)# access-group peer peer-acl
```

```

RP/0/RP0/CPU0:router(config-ntp)# access-group serve serve-acl
RP/0/RP0/CPU0:router(config-ntp)# access-group serve-only serve-only-acl
RP/0/RP0/CPU0:router(config-ntp)# access-group query-only query-only-acl
RP/0/RP0/CPU0:router(config-ntp)# exit
RP/0/RP0/CPU0:router(config)# ipv4 access-list peer-acl
RP/0/RP0/CPU0:router(config-if)# 10 permit ip host 10.1.1.1 any
RP/0/RP0/CPU0:router(config-if)# 20 permit ip host 10.8.8.8 any
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# ipv4 access-list serve-acl
RP/0/RP0/CPU0:router(config-if)# 10 permit ip host 10.4.4.4 any
RP/0/RP0/CPU0:router(config-if)# 20 permit ip host 10.5.5.5 any
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# ipv4 access-list query-only-acl
RP/0/RP0/CPU0:router(config-if)# 10 permit ip host 10.2.2.2 any
RP/0/RP0/CPU0:router(config-if)# 20 permit ip host 10.3.3.3 any
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# ipv4 access-list serve-only-acl
RP/0/RP0/CPU0:router(config-if)# 10 permit ip host 10.6.6.6 any
RP/0/RP0/CPU0:router(config-if)# 20 permit ip host 10.7.7.7 any
RP/0/RP0/CPU0:router(config-if)# 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.

```

RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# authenticate
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 2 md5 encrypted 06120A2D40031D1008124
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 3 md5 encrypted 1311121E074110232621
RP/0/RP0/CPU0:router(config-ntp)# trusted-key 3
RP/0/RP0/CPU0:router(config-ntp)# server 10.3.32.154 key 3
RP/0/RP0/CPU0:router(config-ntp)# peer 10.32.154.145 key 2

```

Disabling NTP on an Interface: Example

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

```

!
RP/0/RP0/CPU0:router(config)# ntp
RP/0/RP0/CPU0:router(config-ntp)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-ntp-int)# disable
RP/0/RP0/CPU0:router(config-ntp-int)# exit
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 2 md5 encrypted 06120A2D40031D1008124
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 3 md5 encrypted 1311121E074110232621

```

```
RP/0/RP0/CPU0:router(config-ntp)# authenticate  
RP/0/RP0/CPU0:router(config-ntp)# trusted-key 3  
RP/0/RP0/CPU0:router(config-ntp)# server 10.3.32.154 key 3  
RP/0/RP0/CPU0:router(config-ntp)# 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:

```
RP/0/RP0/CPU0:router(config)# ntp  
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 2 md5 encrypted 06120A2D40031D1008124  
RP/0/RP0/CPU0:router(config-ntp)# authentication-key 3 md5 encrypted 1311121E074110232621  
RP/0/RP0/CPU0:router(config-ntp)# authenticate  
RP/0/RP0/CPU0:router(config-ntp)# trusted-key 3  
RP/0/RP0/CPU0:router(config-ntp)# server 10.3.32.154 key 3  
RP/0/RP0/CPU0:router(config-ntp)# peer 10.32.154.145 key 2  
RP/0/RP0/CPU0:router(config-ntp)# 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:

```
RP/0/RP0/CPU0:router(config)# ntp  
RP/0/RP0/CPU0:router(config-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:

```
RP/0/RP0/CPU0:router(config)# ntp  
RP/0/RP0/CPU0:router(config-ntp)# server 10.3.32.154  
RP/0/RP0/CPU0:router(config-ntp)# master 6  
RP/0/RP0/CPU0:router(config-ntp)# update-calendar
```

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 Cisco IOS XR Software</i> module of <i>Cisco IOS XR System Management Command Reference</i> , Release 3.5
Cisco IOS XR NTP commands	<i>NTP Commands on Cisco IOS XR Software</i> module of <i>Cisco IOS XR System Management Command Reference</i> , Release 3.5
Cisco IOS XR getting started material	<i>Cisco IOS XR Getting Started Guide</i> , Release 3.5
Cisco IOS XR master command index	<i>Cisco IOS XR Commands Master List</i> , Release 3.5
Information about user groups and task IDs	<i>Configuring AAA Services on Cisco IOS XR Software</i> module of <i>Cisco IOS XR System Security Configuration Guide</i> , Release 3.5

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

Technical Assistance

Description	Link
The Cisco Technical Support web site 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/techsupport

