Configuring NTP

The Network Time Protocol (NTP) synchronizes the time of day among a set of distributed time servers and clients so that you can correlate events when you receive system logs and other time-specific events from multiple network devices. NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communications use Coordinated Universal Time (UTC).

Prerequisites for Configuring NTP

- The NID must have an IP address.

Restrictions for Configuring NTP

- Maximum number of servers supported is 5.

Information About NTP

Network Time Protocol

Network Time Protocol (NTP) is a protocol designed to time-synchronize a network of machines. NTP runs on UDP, which in turn runs on IP. NTP Version 3 (NTPv3) is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to the accuracy of within a millisecond of one another.

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 Global Positioning System [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 has two ways to avoid synchronizing to a machine whose time may not be accurate. NTP does not synchronize to a machine that is not in turn synchronized with the NTP. NTP compares the time reported by several machines and does not synchronize to a machine whose time is significantly different from others, even if its stratum is lower. This strategy effectively builds a self-organizing tree of NTP servers.

Our implementation of NTP does not support stratum 1 service; that is, you cannot connect to a radio or atomic clock (for some specific platforms, however, you can connect to a GPS time-source device). We recommend that the time service you derive for your network from the public NTP servers that are available in the IP Internet.

If the network is isolated from the Internet, our implementation of NTP allows a machine to be configured so that it acts as though it is synchronized via NTP, when in fact the network has determined the time by 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. 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 devices.

The communication 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 through exchange of 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 be configured to send or receive broadcast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is only one way.

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 security 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.

NTP services are disabled on all interfaces by default.

For more information about NTP, see the following sections:

How to Configure NTP

Provisioning the Cisco ME 1200 NID to Configure NTP

DETAILED STEPS

	Command or Action	Purpose
Step 1	**NtpPortType** **Example:** `Switch# NtpPortType`	Enters NTP provisioning mode.

Step 2	**NtpPortType** {**default** | **deleteNtpConfig** | **exit** | **getNtpConfig** | **no** | **setNtpConfig**}
	
	Example:
	`Switch(NtpPortType)# ?`
	`NtpPortType sub-mode commands:`
	`  default  Set a command to its defaults`
	`  deleteNtpConfig  delete NTP config request`
	`  exit  Exit from NtpPortType sub configuration mode`
	`  getNtpConfig  get ntp properties request`
	`  no  Negate a command or set its defaults`
	`  setNtpConfig  Set Ntp Server Details`
		Displays the supported configurations for NTP.

Step 3	**exit** **Example:** `Switch(NtpPortType)# exit`	Exits the NTP mode.

Configuration Example

The following example shows the supported NTP configuration:

```
Switch(NtpPortType)# ?
NtpPortType sub-mode commands:
  default          Set a command to its defaults
  deleteNtpConfig  delete NTP config request
  exit             Exit from NtpPortType sub configuration mode
  getNtpConfig     get ntp properties request
  no               Negate a command or set its defaults
  setNtpConfig     Set Ntp Server Details
```

Configuring NTP on the Cisco ME 1200 NID

Before You Begin

- Ensure that the NID is reachable for the provided NTP server.

- Set the time zone for synchronization with the NTP server. See Configuring the System Clock.
- Perform the steps to provision NTP on the Cisco ME 1200 NID . See Provisioning the Cisco ME 1200 NID to Configure NTP, on page 3

DETAILED STEPS

	Command or Action	Purpose

Step 1	**setNtpConfig** {**commit** | **flush** | **ntpConfig** | **review**}
	
	Example:
	`Switch(NtpPortType)# setNtpConfig ?`
	`  commit  commit deleteNtpConfig`
	`  flush  flush all deleteNtpConfig commands from queue`
	`  ntpConfig  Set Ntp Server Details`
	`  review  review deleteNtpConfig commands`
		Sets NTP configuration
		• **commit—**Sends the NTP configuration to NID.
		• **flush—**Flushes all NTP configuration from the queue.
		• **ntpConfig—**Sets the NTP server configuration on the Cisco ME 1200 NID .
		• **review—**Displays the configuration on the Cisco ME 1200 NID .

Step 2	**setNtpConfig ntpConfig** {**hostinfo** {**hostname** *host-name*} | **ipv4address** *IPv4-address* | **ipv6address** *IPv6-address*} | **ntpmode** {**enable** | **number** *server-number*}}
	
	Example:
	`Switch(NtpPortType)# setNtpConfig hostinfo hostname host1`
	`Switch(NtpPortType)# setNtpConfig ipv4address 192.34.7.8`
	`Switch(NtpPortType)# setNtpConfig ipv6address 2001:DB8:0:ABCD::1`
	`Switch(NtpPortType)# setNtpConfig ntpmode enable`
	`Switch(NtpPortType)# setNtpConfig ntpmode number 5`
		Configures NTP.
		• **hostinfo—**Sets the host information such as host name, IPv4 address and IPv6 address on the Cisco ME 1200 NID .
		• **ntpmode—**Enables or disables the NTP mode on the Cisco ME 1200 NID .
		• **number** *server-number*—Sets the NTP server details. The valid range is from 1 to 5.

Step 3	**setNtpconfig review**
	
	Example:
	`Switch(NtpPortType)# setNtpconfig review`
	
	`Commands in queue:`
	`  setNtpConfig ntpConfig hostInfo hostName host1`
	`  setNtpConfig ntpConfig hostInfo ipv4Address 192.34.7.8`
	`  setNtpConfig ntpConfig ntpMode enable`
	`  setNtpConfig ntpConfig number 5`
	`  setNtpConfig ntpConfig ntpMode enable`
		Displays the NTP configuration on the Cisco ME 1200 NID .

Step 4	**setNtpconfigcommit** **Example:** `Switch(NtpPortType)# setNtpconfig commit`	Sends the NTP configuration to the NID.
Step 5	**exit** **Example:** `Switch(NtpPortType)# exit`	Exits the NTP mode.

Configuration Example

The example shows how to configure NTP on the Cisco ME 1200 NID :

```
Switch(NtpPortType)# setNtpConfig hostinfo hostname host1
Switch(NtpPortType)# setNtpConfig ipv4address 192.34.7.8
Switch(NtpPortType)# setNtpConfig ipv6address 2001:DB8:0:ABCD::1
Switch(NtpPortType)# setNtpConfig ntpmode enable
Switch(NtpPortType)# setNtpConfig ntpmode number 5
Switch(NtpPortType)# setNtpconfig review

Commands in queue:
        setNtpConfig ntpConfig hostInfo hostName host1
        setNtpConfig ntpConfig hostInfo ipv4Address 192.34.7.8
        setNtpConfig ntpConfig ntpMode enable
        setNtpConfig ntpConfig number 5
        setNtpConfig ntpConfig ntpMode enable

Switch(NtpPortType)# setNtpconfig commit
Switch(NtpPortType)# exit
```

Configuring NTP with Default Configuration

You can set the default NTP configuration on the Cisco ME 1200 NID.

Before You Begin

- Perform the steps to provision NTP on the Cisco ME 1200 NID.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**default** {**getNtpConfig** | **setNtpConfig** | **deleteNtpConfig** | **exit** }
	
	Example:
	`Switch(NtpPortType)# default ?`
	
	`deleteNtpConfig  delete NTP config request`
	`exit  Exit from NtpPortType sub configuration mode`
	`getNtpConfig  get ntp properties request`
	`setNtpConfig  Set Ntp Server Details`
		Sets the default NTP configuration.
		• **getNtpConfig—**View the configuration on the ME 1200 NID.
		• **setNtpConfig—**Sets the configuration on theME 1200 NID.
		• **deleteNtpConfig—**Deletes the configuration from the ME 1200 NID.
		• **exit—**Exits from NtpPortType configuration mode.

Step 2	**exit** **Example:** `Switch(NtpPortType)# exit`	Exits the NTP mode.

Viewing the NTP Configuration

Before You Begin

- Perform the steps to provision NTP on the Cisco ME 1200 NID.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**getNtpConfig** {**commit** | **flush** | **ntpStatusRequest** *ntp-status*| **review**}
	
	Example:
	`Switch(NtpPortType)#getNtpConfig ntpStatusRequest 1`
	`Switch(NtpPortType)#getNtpConfig review`
	`Switch(NtpPortType)#getNtpConfig commit`
		• **ntpStatusRequest—**Request NTP configuration properties.
		• **commit—**Sends the NTP configuration to NID.
		• **flush—**Flushes all NTP configuration from the queue.
		• **review—**Displays the configuration.

Step 2	**exit** **Example:** `Switch(NtpPortType)# exit`	Exits the NTP mode.

Configuration Example

The example shows how to view the configuration:

```
Switch(NtpPortType)# getNtpConfig ntpStatusRequest 1
Switch(NtpPortType)# getNtpConfig review

Commands in queue:
        getNtpConfig ntpStatusRequest 1
        getNtpConfig ntpStatusRequest 2
        getNtpConfig ntpStatusRequest 3

Switch(NtpPortType)# getNtpConfig commmit
Switch(NtpPortType)# end
```

Deleting the NTP Configuration

Before You Begin

- Perform the steps to provision NTP on the Cisco ME 1200 NID.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**deleteNtpConfig** {**commit** | **flush** | **ntpDeleteConfig** | **review**}
	
	Example:
	`Switch(NtpPortType)# deleteNtpConfig ?`
	`  commit  commit deleteNtpConfig`
	`  flush  flush all deleteNtpConfig commands from queue`
	`  ntpDeleteConfig  delete NTP config request`
	`  review  review deleteNtpConfig commands`
		Removes the NTP configuration.
		• **commit—**Sends the NTP configuration to NID.
		• **flush—**Flushes all NTP configuration from the queue.
		• **ntpDeleteConfig—**Deletes the NTP configuration request on the Cisco ME 1200 NID.
		• **review—**Displays the configuration on the Cisco ME 1200 NID .

Step 2	**ntpDeleteConfig** {**ntpEnable** | **ntpServerNo***server-num*}
	
	Example:
	`Switch(NtpPortType)# deleteNtpConfig ntpDeleteConfig ntpEnable`
	`Switch(NtpPortType)# deleteNtpConfig ntpDeleteConfig ntpServer 1`
		Removes NTP configuration.
		• **ntpEnable—**Disables the NTP configuration.
		• **ntpServerNo—**Disables the NTP server.
		• *server-num***—**Specifies the NTP server. The valid range is from 1 to 5.

Step 3	**ntpDeleteConfig review** **Example:** `Switch(NtpPortType)# deleteNtpConfig review`	Displays the NTP configuration.
Step 4	**ntpDeleteConfig commit** **Example:** `Switch(NtpPortType)# deleteNtpConfig commit`	Sends the NTP configuration to the NID.
Step 5	**exit** **Example:** `Switch(NtpPortType)# exit`	Exits the NTP mode.

Configuration Example

The following example shows how to delete the NTP configuration:

```
Switch(NtpPortType)# deleteNtpConfig ntpDeleteConfig ntpEnable
Switch(NtpPortType)# deleteNtpConfig ntpDeleteConfig ntpServer 1
Switch(NtpPortType)# deleteNtpConfig review
Commands in queue:
        deleteNtpConfig ntpDeleteConfig ntpEnable
        deleteNtpConfig ntpDeleteConfig ntpServerNo 2
Switch(NtpPortType)# deleteNtpConfig commit
DeleteNtpConfig Commit Success!!!
Switch(NtpPortType)# deleteNtpConfig exit
```