Precision Time Protocol

Precision Time Protocol

Precision Time Protocol (PTP) is defined in IEEE 1588 as Precision Clock Synchronization for Networked Measurements and Control Systems, and was developed to synchronize the clocks in packet-based networks that include distributed device clocks of varying precision and stability. PTP is designed specifically for industrial, networked measurement and control systems, and is optimal for use in distributed systems because it requires minimal bandwidth and little processing overhead.

Benefits of PTP

Smart grid power automation applications such as peak-hour billing, virtual power generators, and outage monitoring and management, require precise time accuracy and stability. Timing precision improves network monitoring accuracy and troubleshooting ability.

In addition to providing time accuracy and synchronization, the PTP message-based protocol can be implemented on packet-based networks, such as Ethernet networks. The benefits of using PTP in an Ethernet network include:

- Low cost and easy setup in existing Ethernet networks

- Limited bandwidth is required for PTP data packets

Message-Based Synchronization

To ensure clock synchronization, PTP requires an accurate measurement of the communication path delay between time source (grandmaster clock) and the time recipient. PTP sends messages between the time source and time recipient to determine the delay measurement. Then, PTP measures the exact message transmit and receive times and uses these times to calculate the communication path delay. PTP then adjusts current time information contained in network data for the calculated delay, resulting in more accurate time information.

This delay measurement principle determines path delay between devices on the network. The local clocks are adjusted for this delay using a series of messages sent between time source and time recipient devices. The one-way delay time is calculated by averaging the path delay of the transmit and receive messages. This calculation assumes a symmetrical communication path; however, switched networks do not necessarily have symmetrical communication paths, due to the buffering process.

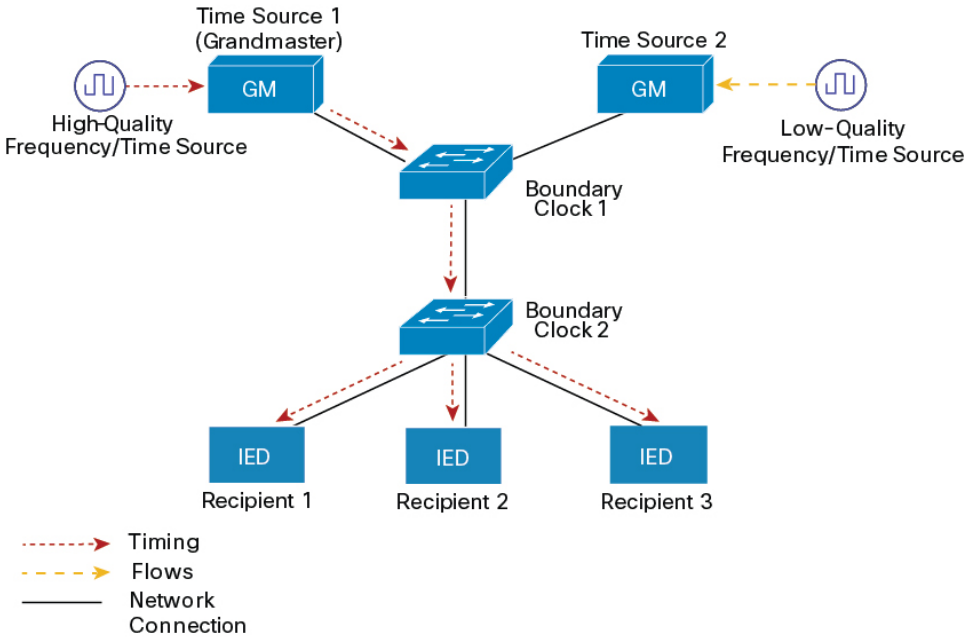
PTP provides a method, using transparent clocks, to measure and account for the delay in a time-interval field in network timing packets. Doing so makes the switches temporarily transparent to the time source and time recipient nodes on the network. An end-to-end transparent clock forwards all messages on the network in the same way that a switch does.

Note Cisco PTP supports multicast PTP messages only.

To read a detailed description of synchronization messages, refer to PTP Event Message Sequences, on page 3. To learn more about how transparent clocks calculate network delays, refer to Transparent Clock, on page 7.

The following figure shows a typical 1588 PTP network that includes grandmaster clocks, switches in boundary clock mode, and Intelligent Electronic Device (IEDs) such as a digital relays or protection devices. In this diagram, Time Source 1 is the grandmaster clock. If Time Source 1 becomes unavailable, the time recipient boundary clocks switch to Time Source 2 for synchronization.

Figure 1: PTP Network



PTP Event Message Sequences

This section describes the PTP event message sequences that occur during synchronization.

Synchronizing with Boundary Clocks

The ordinary and boundary clocks configured for the delay request-response mechanism use the following event messages to generate and communicate timing information:

- Sync
- Delay_Req
- Follow_Up
- Delay_Resp

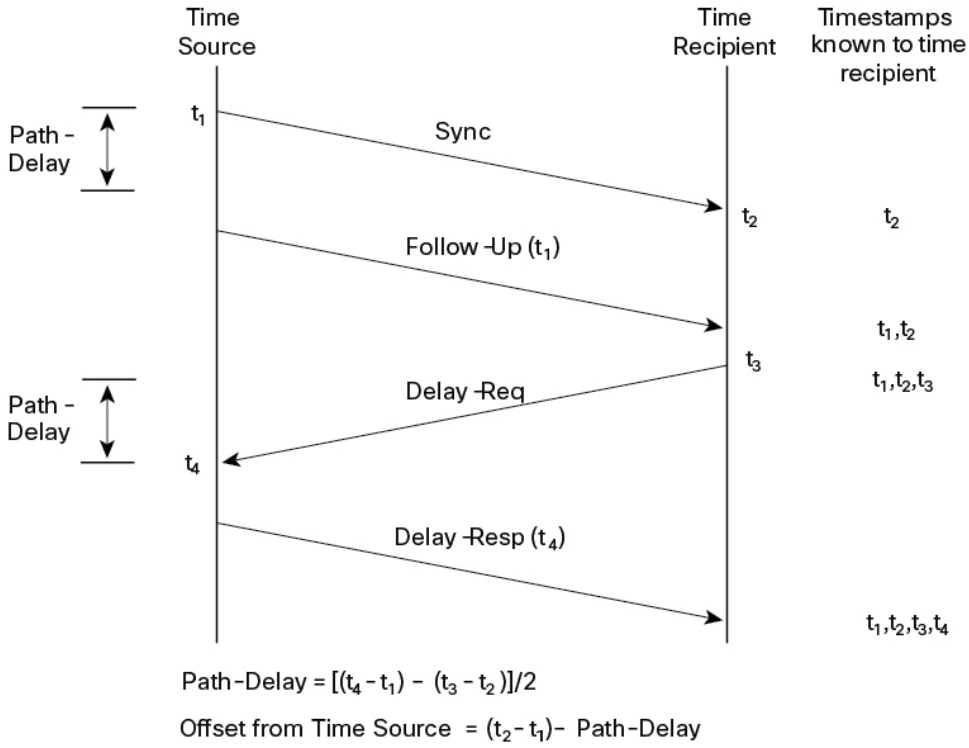
These messages are sent in the following sequence:

1. The time source sends a Sync message to the time recipient and notes the time (t1) at which it was sent.
2. The time recipient receives the Sync message and notes the time of reception (t2).
3. The time source conveys to the time recipient the timestamp t1 by embedding the timestamp t1 in a Follow_Up message.
4. The time recipient sends a Delay_Req message to the time source and notes the time (t3) at which it was sent.
5. The time source receives the Delay_Req message and notes the time of reception (t4).
6. The time source conveys to the time recipient the timestamp t4 by embedding it in a Delay_Resp message.

After this sequence, the time recipient possesses all four timestamps. These timestamps can be used to compute the offset of the time recipient clock relative to the time source, and the mean propagation time of messages between the two clocks.

The offset calculation is based on the assumption that the time for the message to propagate from time source to time recipient is the same as the time required from time recipient to time source. This assumption is not always valid on an Ethernet network due to asymmetrical packet delay times.

Figure 2: Detailed Steps—Boundary Clock Synchronization



Synchronizing with Peer-to-Peer Transparent Clocks

When the network includes multiple levels of boundary clocks in the hierarchy, with non-PTP enabled devices between them, synchronization accuracy decreases.

The round-trip time is assumed to be equal to mean_path_delay/2, however this is not always valid for Ethernet networks. To improve accuracy, the resident time of each intermediary clock is added to the offset in the end-to-end transparent clock. Resident time, however, does not consider the link delay between peers, which is handled by peer-to-peer transparent clocks.

Peer-to-peer transparent clocks measure the link delay between two clock ports implementing the peer delay mechanism. The link delay is used to correct timing information in Sync and Follow_Up messages.

Peer-to-peer transparent clocks use the following event messages:

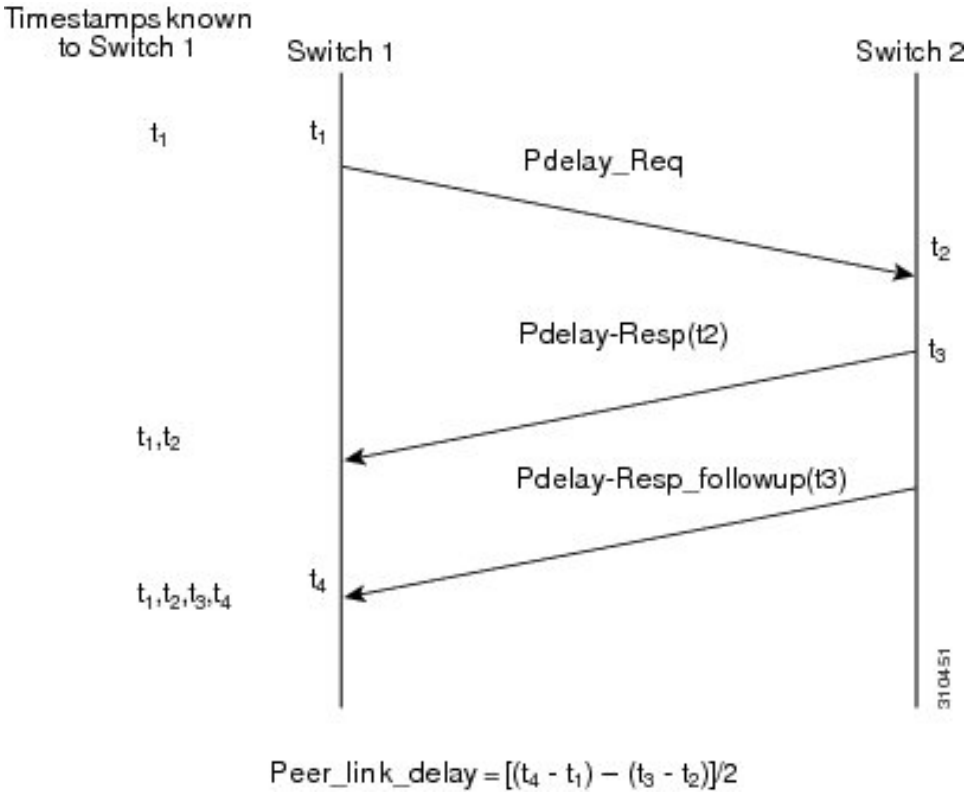
- Pdelay_Req
- Pdelay_Resp
- Pdelay_Resp_Follow_Up

These messages are sent in the following sequence:

1. Port 1 generates timestamp t1 for a Pdelay_Req message.
2. Port 2 receives and generates timestamp t2 for this message.
3. Port 2 returns and generates timestamp t3 for a Pdelay_Resp message.

   To minimize errors due to any frequency offset between the two ports, Port 2 returns the Pdelay_Resp message as quickly as possible after the receipt of the Pdelay_Req message.
4. Port 2 returns timestamps t2 and t3 in the Pdelay_Resp and Pdelay_Resp_Follow_Up messages respectively.
5. Port 1 generates timestamp t4 after receiving the Pdelay_Resp message. Port 1 then uses the four timestamps (t1, t2, t3, and t4) to calculate the mean link delay.

Figure 3: Detailed Steps—Peer-to-Peer Transparent Clock Synchronization



Synchronizing the Local Clock

In an ideal PTP network, the time source and time recipient clocks operate at the same frequency. However, *drift* can occur on the network. Drift is the frequency difference between the time source and time recipient clocks. You can compensate for drift by using the time stamp information in the device hardware and follow-up messages (intercepted by the switch) to adjust the frequency of the local clock to match the frequency of the time source clock.

Best Master Clock Algorithm

The Best Master Clock Algorithm (BMCA) is the basis of PTP functionality. The BMCA specifies how each clock on the network determines the best time source clock in its subdomain of all the clocks it can see, including itself. The BMCA runs on the network continuously and quickly adjusts for changes in network configuration.

The BMCA uses the following criteria to determine the best time source clock in the subdomain:

- Clock quality (for example, GPS is considered the highest quality)
- Clock accuracy of the clock's time base.
- Stability of the local oscillator
- Closest clock to the grandmaster

In addition to identifying the best time source clock, the BMCA also ensures that clock conflicts do not occur on the PTP network by ensuring that:

- Clocks do not have to negotiate with one another.
- There is no misconfiguration, such as two time source clocks or no time source clocks, as a result of the time source clock identification process.

PTP Clocks

A PTP network is made up of PTP-enabled devices and devices that are not using PTP. The PTP-enabled devices typically consist of the following clock types.

Note Transparent Clock mode is the only clock mode supported in Power Profile 2017. See PTP Profiles, on page 8 in this document.

Grandmaster Clock

The grandmaster clock is a network device physically attached to the server time source. All clocks are synchronized to the grandmaster clock.

Within a PTP domain, the grandmaster clock is the primary source of time for clock synchronization using PTP. The grandmaster clock usually has a precise time source, such as a GPS or atomic clock. When the network does not require any external time reference and only needs to be synchronized internally, the grandmaster clock can free run.

Boundary Clock

A boundary clock in a PTP network operates in place of a standard network switch or router. Boundary clocks have more than one PTP port, and each port provides access to a separate PTP communication path. They intercept and process all PTP messages, and pass all other network traffic. The boundary clock uses the BMCA to select the best clock seen by any port. The selected port is then set to nonmaster mode. The master port synchronizes the clocks connected downstream, while the nonmaster port synchronizes with the upstream master clock.

Transparent Clock

The role of transparent clocks in a PTP network is to update the time-interval field that is part of the PTP event message. This update compensates for switch delay and has an accuracy of within one picosecond.

There are two types of transparent clocks:

End-to-end (E2E) transparent clocks measure the PTP event message transit time (also known as *resident time)* for SYNC and DELAY_REQUEST messages. This measured transit time is added to a data field (correction field) in the corresponding messages:

- The measured transit time of a SYNC message is added to the correction field of the corresponding SYNC or the FOLLOW_UP message.
- The measured transit time of a DELAY_REQUEST message is added to the correction field of the corresponding DELAY_RESPONSE message.

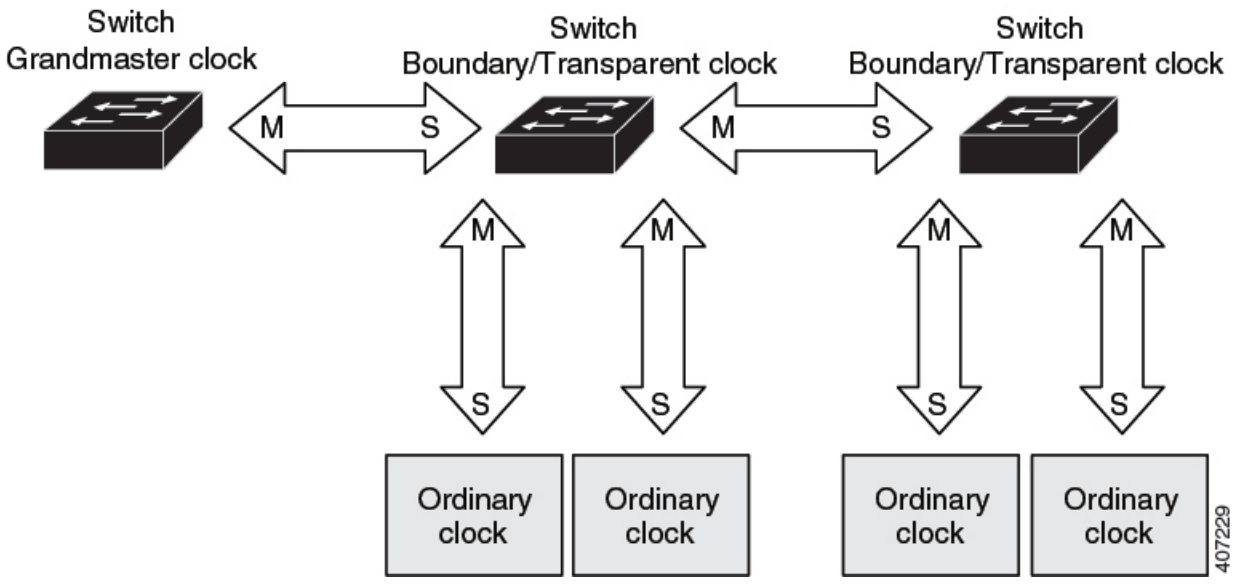
The time recipient uses this information when determining the offset between the time recipient's and the time source's time. E2E transparent clocks do not provide correction for the propagation delay of the link itself.

Peer-to-peer (P2P) transparent clocks measure PTP event message transit time in the same way E2E transparent clocks do, as described above. In addition, P2P transparent clocks measure the upstream link delay. The upstream link delay is the estimated packet propagation delay between the upstream neighbor P2P transparent clock and the P2P transparent clock under consideration.

These two times (message transit time and upstream link delay time) are both added to the correction field of the PTP event message, and the correction field of the message received by the time recipient contains the sum of all link delays. In theory, this is the total end-to-end delay (from time source to time recipient) of the SYNC packet.

The following figure illustrates PTP clocks in a time source-time recipient hierarchy within a PTP network.

Figure 4: PTP Clock Hierarchy



Note In the preceding illustration, *M* signifies master port, and *S* signifies nonmaster, or subordinate port.

Clock Configuration

- All PHY PTP clocks are synchronized to the grandmaster clock. The switch system clock is not synchronized as part of PTP configuration and processes.
- When VLAN is enabled on the grandmaster clock, it must be in the same VLAN as the native VLAN of the PTP port on the switch.
- Grandmaster clocks can drop untagged PTP messages when a VLAN is configured on the grandmaster clock. To force the switch to send tagged packets to the grandmaster clock, enter the global **vlan dot1q tag native** command.

PTP Profiles

This section describes the following PTP profiles available on the switch:

- Default Profile
- Power Profile

Power Profile-2011 is supported as defined in PC37.238-2011 - IEEE Draft Standard Profile for Use of IEEE 1588 Precision Time Protocol in Power System Applications. This documentation uses the terms Power Profile mode and Default Profile mode when referring to this IEEE 1588 profile and its associated configuration values.

Two Power Profiles are supported: Power Profile-2011 and Power Profile-2017. Power Profile-2017 is defined in IEEE Standard C37.238™-2017 (Revision of IEEE Std C37.238-2011) for use of IEEE 1588 Precision Time Protocol in Power System Applications.

This documentation uses the terms Power Profile mode and Default Profile mode when referring to this IEEE 1588 profile and its associated configuration values. The IEEE 1588 definition of a PTP profile is *the set of allowed PTP features applicable to a device*. A PTP profile is usually specific to a particular type of application or environment and defines the following values:

- Best master clock algorithm options
- Configuration management options
- Path delay mechanisms (peer delay or delay request-response)
- Range and default values of all PTP configurable attributes and data set members
- Transport mechanisms that are required, permitted, or prohibited
- Node types that are required, permitted, or prohibited
- Options that are required, permitted, or prohibited

Default Profile Mode

The default PTP profile mode on the switch is Default Profile mode. In this mode:

- Supports transparent clock, boundary clock, grandmaster boundary clock, and PTP forward mode (PTP passthrough) on the default profile.
- Ordinary clocks are not supported.

Power Profile Mode

The IEEE Power Profile defines specific or allowed values for PTP networks used in power substations. The defined values include the optimum physical layer, the higher-level protocol for PTP messages, and the preferred best master clock algorithm. The Power Profile values ensure consistent and reliable network time distribution within substations, between substations, and across wide geographic areas.

The following table lists the configuration values defined by the IEEE 1588 Power Profile and the values that the switch uses for each PTP profile mode.

Table 1: Configuration Values for the IEEE PTP Power Profile and Switch Modes

PTP Field	**Switch Configuration Value**	
	Power Profile Mode	**Default Profile Mode**
Message transmission	**Access ports**: Untagged Layer 2 packets. **Trunk ports**: PTP packets are tagged with the PTP VLAN. If the PTP VLAN is not configured, packets go untagged over the native VLAN.	Layer 3 packets. By default, 802.1q tagging is disabled.
MAC address– Nonpeer delay messages	01-00-5e-00-01-81.	Default profile uses L3 transport multicast address 224.0.1.129 for all PTP messages. Equivalent mac address is 01-00-5e-00-01-81.
MAC address– Peer delay messages	01-80-C2-00-00-0E.	Not applicable to this mode.
Domain number	0.	0.
Path delay calculation	Peer-to-peer transparent clocks using the peer_delay mechanism.	End-to-end transparent clocks using the delay_request mechanism.
BMCA	Enabled.	Enabled.
Clock type	Two-step.	Two-step.
Time scale	Epoch.	Epoch.
Grandmaster ID and local time determination	PTP-specific TLV to indicate Grandmaster ID.	PTP-specific type, length, and value to indicate Grandmaster ID.
Time accuracy over network hops	Over 16 hops, end device synchronization accuracy is within 1 usec (1 microsecond).	Not applicable in this mode.

PTP Profile Comparison

Table 2: Comparison of PTP Profiles on IE Switches

Profile	Default (*)		Power Profile-2011		Power Profile-2017
Standard	IEEE1588 v2 (J.3)		IEEE C37.238-2011		IEEE C37.238-2017
Mode	Boundary	End-to-End transparent	Boundary	Peer-to-Peer transparent	Peer-to-Peer transparent
Path Delay	Delay req/res	Delay req/res	Peer delay req/res	Peer delay req/res	Peer delay req/res
Non-PTP device allowed in PTP domain	Yes	Yes	No	No	No
Transport	UDP over IP (multicast)		L2 Multicast		L2 Multicast

* Delay Request-Response Default PTP profile (as defined in IEEE1588 J.3).

Tagging Behavior of PTP Packets

The following table describes the switch tagging behavior in Power Profile and Default Profile modes.

Table 3: Tagging Behavior for PTP Packets

Switch Port Mode	Configuration	Power Profile Mode		Default Profile Mode	
		Behavior	Priority	Behavior	Priority
Trunk Port	**vlan dot1q tag native** enabled	Switch tags packets	7	Switch tags packets	7
Trunk Port	**vlan dot1q tag native** disabled	PTP software tags packets	4	Untagged	None
Access Port	N/A	Untagged	None	Untagged	None

Configurable Boundary Clock Synchronization Algorithm

You can configure the BC synchronization algorithm to accommodate various PTP use cases, depending on whether you need to prioritize filtering of input time errors or faster convergence. A PTP algorithm that filters packet delay variation (PDV) converges more slowly than a PTP algorithm that does not.

By default, the BC uses a linear feedback controller (that is, a servo) to set the BC's time output to the next clock. The linear servo provides a small amount of PDV filtering and converges in an average amount of time. For improved convergence time, BCs can use the TC feedforward algorithm to measure the delay added by the network elements forwarding plane (the disturbance) and use that measured delay to control the time output.

While the feedforward BC dramatically speeds up the boundary clock, the feedforward BC does not filter any PDV. The adaptive PDV filter provides high-quality time synchronization in the presence of PDV over wireless access points (APs) and enterprise switches that do not support PTP and that add significant PDV.

Three options are available for BC synchronization (all are compliant with IEEE 1588-2008):

- Feedforward: For very fast and accurate convergence; no PDV filtering.
- Adaptive: Filters as much PDV as possible, given a set of assumptions about the PDV characteristics, the hardware configuration, and the environmental conditions.

Note With the adaptive filter, the switch does not meet the time performance requirements specified in ITU-T G.8261.

- Linear: Provides simple linear filtering (the default).

Adaptive mode (**ptp transfer filter adaptive**) is not available in Power Profile mode.

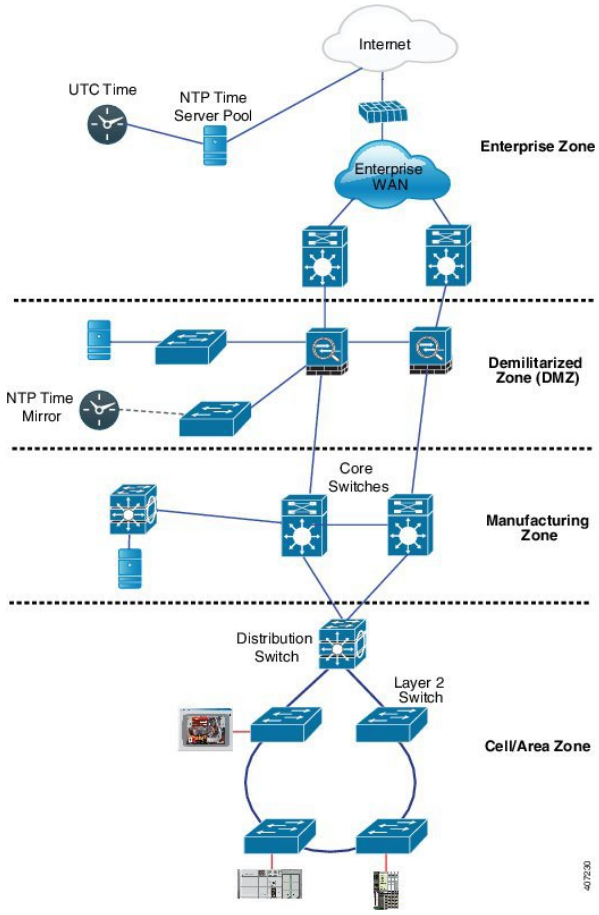
NTP to PTP Time Conversion

NTP to PTP Time Conversion allows you to use Network Time Protocol (NTP) as a time source for PTP. Customers who use PTP for precise synchronization within a site can use NTP across sites, where precise synchronization is not required.

NTP is the traditional method of synchronizing clocks across packet-based networks. NTP uses a two-way time transfer mechanism, between a time source and an end device. NTP is capable of synchronizing a device within a few 100 milliseconds across the Internet, and within a few milliseconds in a tightly controlled LAN. The ability to use NTP as a time source for PTP allows customers to correlate data generated in their PTP network with data in their enterprise data centers running NTP.

The following figure shows an example of an industrial network based on the Industrial Automation and Control System Reference Model. The enterprise zone and demilitarized zone run NTP, and the manufacturing zone and cell/area zone run PTP with NTP as the time source. The switch with the NTP to PTP conversion feature can be either the Layer 2 Switch or the Distribution Switch in the Cell/Area Zone.

Figure 5: Industrial Network with NTP and PTP



Note The NTP to PTP feature supports the Default E2E Profile and Power Profile.

Clock Manager

The clock manager is the component in the Cisco NTP to PTP software architecture that tracks the various time services and selects the clock that actively provides time. The clock manager notifies the time services of important changes, such as state changes, leap seconds, or daylight saving time.

The clock manager selects the NTP or manually set clock first, followed by PTP and the real-time clock if NTP is not active. The following table shows the results of the clock selection process.

Table 4: Time Service Selection

NTP (Active) or Manually Set	PTP (Active)	Real-Time Clock	Selected Output
True	Don’t care	Don’t care	NTP or Manually Set

NTP (Active) or Manually Set	PTP (Active)	Real-Time Clock	Selected Output
False	True	Don’t care	PTP
False	False	True	Real-Time Clock

In general, the clock manager ensures that the time displayed in the Cisco IOS commands **show ptp clock** and **show clock** match. The **show clock** command always follows this priority, but there are two corner cases where the **show ptp clock** time may differ:

- The switch is either a TC or a BC, and there is no other active reference on the network. To preserve backwards compatibility, the TC and BC never take their time from the clock manager, only from the network PTP GMC. If there is no active PTP GMC, then the time displayed in the **show clock** and the **show ptp clock** command output may differ.
- The switch is a synchronizing TC, a BC with a subordinate port, or a GMC-BC with subordinate port, and the time provided by the PTP GMC does not match the time provided by NTP or the user (that is, manually set). In this case, the PTP clock must forward the time from the PTP GMC. If the PTP clock does not follow the PTP GMC, then the PTP network ends up with two different time bases, which would break any control loops or sequence of event applications using PTP.

The following table shows how the Cisco IOS and PTP clocks behave given the various configurations. Most of the time, the two clocks match. Occasionally, the two clocks are different; those configurations are highlighted in the table.

Table 5: Expected Time Flow

IOS Clock Configuration	PTP Clock Configuration	IOS Clock Source	PTP Clock Source
Calendar	PTP BC, E2E TC, or GMC-BC in BC Mode	PTP	PTP
Manual	PTP BC, E2E TC, or GMC-BC in BC Mode	Manual	PTP
NTP	PTP BC, E2E TC, or GMC-BC in BC Mode	NTP	PTP
Calendar	GMC-BC in GM Mode	Calendar	Calendar
Manual	GMC-BC in GM Mode	Manual	Manual
NTP	GMC-BC in GM Mode	NTP	NTP

GMC Block

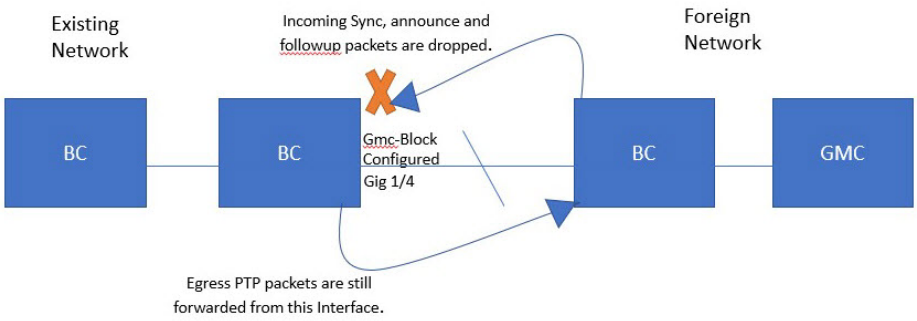
GMC Block protects an existing network from any rogue GMC that might try to synchronize with the devices inside the network. This feature is supported for all PTP clock modes except Forward mode. After the feature is enabled on an interface, only the egress Announce, Sync, and Followup PTP packets are allowed and all ingress Announce, Sync, and Followup packets are dropped on this interface. This prevents the port state transition to time recipient.

Information about a rogue GMC is retrieved from the packets before dropping them. However, egress PTP packets are still allowed from this interface, so it can act as a GMC. To identify the rogue device, details such as IP address and clock ID are stored and displayed for the interface. Two Syslog messages are also generated to notify the presence and clearance of rogue devices.

You can configure PTP gmc-block on multiple ports, if you suspect multiple foreign networks are connected to your existing system. Per-port Syslog messages are displayed after an interval of 30 seconds of receiving rogue packets and after 180 -240 seconds when packets stop coming. Relay minor alarms and SNMP traps are also generated to notify of the presence of foreign rogue devices.

Packet Flow with GMC Block

The following figure shows an example of a PTP network topology with the GMC Block feature configured on an interface.



PTP packets originate in the GMC of the foreign network in an attempt to sync with the existing network. When the PTP packets reach the port configured with GMC Block, the packets are dropped after the system retrieves the required information from them.

Because packets from the foreign network are restricted, the system syncs with the local GMC present in the existing system. PTP packets originated on the port configured with GMC Block are still allowed to egress from this interface, which allows devices in the existing network to be GMC.

Guidelines and Limitations

This section lists the guidelines and limitations when using PTP.

General PTP Guidelines

- The Cisco PTP implementation supports only two-step clock operation in BC mode; one-step clock in not supported.
- In transparent clock mode, the switch converts any one-step Sync message received from the grandmaster clock into a two-step message.
- Cisco PTP supports multicast PTP messages only.
- Cisco PTP supports only PTP version2.
- Power Profile-2017 supports only transparent clock mode.

PTP Mode and Profile

- The switch and the grandmaster clock must be in the same PTP domain.

- When Power Profile mode is enabled, the switch drops the PTP announce messages that do not include these two Types, Length, Value (TLV) message extensions: *Organization_extension* and *Alternate_timescale*.

  If the grandmaster clock is not compliant with PTP and sends announce messages without these TLVs, configure the switch to process the announce message by entering the following command:

  ```
  ptp clock boundary domain 1 profile power
  allow-without-tlv
  ```

- When the switch is in Power Profile mode, only the peer_delay mechanism is supported.

  To enable power profile boundary mode and associate interfaces using the clock-port suboption, enter the following command:

  ```
  ptp clock boundary domain 1 profile power
  clock-port 1
  transport ethernet multicast interface gi1/1
  ```

- To disable power profile transparent mode, enter the following command, which returns the switch to forward mode.

  ```
  no ptp clock transparent domain x profile power
  ```

- To enable the E2E transparent clock, use the following command:

  ```
  ptp clock transparent domain x profile default
  ```

- In Default Profile mode, only the delay_request mechanism is supported.

  To enable default profile boundary clock mode and interfaces associated with clock-port suboption, enter the following command:

  ```
  ptp clock boundary domain 1 profile default
  clock-port 1
  transport ipv4 multicast interface gi1/1
  ```

Packet Format

- The packet format for PTP messages can be 802.1q tagged packets or untagged packets.
- The switch does not support 802.1q QinQ tunneling of PTP packets.
- In Power Profile mode:
  - When the PTP interface is configured as an access port, PTP messages are sent as untagged, Layer 2 packets.
  - When the PTP interface is configured as a trunk port, two cases are possible:
    - When native VLAN is enabled on the interface, PTP packets go untagged over the native VLAN.
    - When PTP VLAN is configured under the clock-port, PTP packets are tagged with the PTP VLAN configured.
- Time recipient IEDs must support tagged and untagged packets.
- When PTP packets are sent on the native VLAN in E2E Transparent Clock Mode, they are sent as untagged packets. To configure the switch to send them as tagged packets, enter the global **vlan dot1q tag native** command.

NTP to PTP Conversion

The NTP to PTP feature supports the Default E2E Profile and Power Profile.

PTP Interaction with Other Features

- PTP over Media Redundancy Protocol (MRP) is not supported.
- PTP over Port Channels is not supported.
- PTP over Cisco Resilient Ethernet Protocol (REP) is not supported.
- The following PTP clock modes only operate on a single VLAN:
  - e2etransparent
  - p2ptransparent

Default Settings

- PTP is enabled on the switch by default.
- By default, the switch uses configuration values defined in the Default Profile (Default Profile mode is enabled).
- The switch default PTP clock mode is E2E Transparent Clock Mode.
- The default BC synchronization algorithm is linear filter.

VLAN Configuration

This section contains information about VLAN configuration.

- Sets the PTP VLAN on a trunk port. The range is from 1 to 4094. The default is the native VLAN of the trunk port.
- In boundary mode, only PTP packets in PTP VLAN are processed; PTP packets from other VLANs are dropped.
- Before configuring the PTP VLAN on an interface, the PTP VLAN must be created and allowed on the trunk port.
- Most grandmaster clocks use the default VLAN 0. In Power Profile mode, the switch default VLAN is VLAN 1 and VLAN 0 is reserved. When you change the default grandmaster clock VLAN, it must be changed to a VLAN other than 0.
- When VLAN is disabled on the grandmaster clock, the PTP interface must be configured as an access port.

Configuring GMC Mode

The following sections provide steps for configuring GMC mode for default and power profiles:

- Configuring GMC Mode for a Default Profile, on page 17
- Configure GMC Mode for a Power Profile, on page 17

Configuring GMC Mode for a Default Profile

Complete the steps in this section to configure GMC mode for a default profile.

Procedure

	Command or Action	Purpose
Step 1	**ptp clock boundary domain** *domain number* **profile default** **Example:** `switch(config)# ptp clock boundary domain 0 profile default`	Enable the default profile boundary mode.
Step 2	**gmc default** **Example:** `switch(config-ptp-clk)# gmc default`	Enable the GMC boundary clock.
Step 3	**clock-port** *port name* **Example:** `switch(config-ptp-clk)# clock-port port1`	Define a new clock port.
Step 4	**transport ipv4 multicast** *interface type interface number* **Example:** `switch(config-ptp-port)# transport ipv4 multicast interface Gi1/1`	Specify the transport mechanism for clocking traffic.

Configure GMC Mode for a Power Profile

Complete the steps in this section to configure GMC mode for a power profile.

Procedure

	Command or Action	Purpose
Step 1	**ptp clock boundary domain** *domain number* **profile power**	Enable the power profile boundary mode.

	Command or Action	Purpose
	Example: `switch(config)# ptp clock boundary domain 0 profile power`	
Step 2	**gmc default** **Example:** `switch(config-ptp-clk)# gmc default`	Enable the GMC boundary clock.
Step 3	**clock-port** *port name* **Example:** `switchswitch(config-ptp-clk)# clock-port port1`	Defines a new clock port.
Step 4	**transport ethernet multicast** *interface type interface number* **Example:** `switch(config-ptp-port)# transport ethernet multicast interface gi1/1`	Specifies the transport mechanism for clocking traffic.

Configuring PTP Default Profile

This section describes how to configure the switch to operate in Default Profile mode.

Configure a Boundary Clock

If an interface is not added as part of BC clock, it will be in forward mode exchanging PTP packets, which will cause instability in PTP operation. To avoid this, it is recommended to disable PTP on all such interfaces using the **no ptp enable** command.

Follow these steps to configure the switch as a boundary clock:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `switch# configure terminal`	Enters configuration mode.
Step 3	**ptp clock boundary domain** *domain-number* **profile default**	This step configures the boundary type PTP clock, which terminates the PTP session from

	Command or Action	Purpose
	Example: `switch(config)# ptp clock boundary domain 0 profile default`	the grandmaster clock and acts as a PTP server or client clock downstream.
Step 4	**clock-port** *port-name* **Example:** `switch(config-ptp-clk)# clock-port dyn1`	Defines a new clock port.
Step 5	**transport ipv4 multicast interface** *interface-type interface-number* **Example:** `switch(config-ptp-port)# transport ipv4 multicast interface Gi1/1`	Specifies the transport mechanism for clocking traffic.
Step 6	(Optional) **vlan** *vlan-id* **Example:** `config-ptp-port)# vlan 100`	Configure VLAN for tagged packets.

Example

Example of Untagged

```
ptp clock boundary domain 0  profile default
 clock-port dyn1
  transport ipv4 multicast interface Gi1/1
 clock-port dyn2
  transport ipv4 multicast interface Gi1/1
```

Example of Tagged

```
ptp clock boundary domain 0  profile default
 clock-port dyn1
  transport ipv4 multicast interface Gi1/1
  vlan 100
 clock-port dyn2
  transport ipv4 multicast interface Gi1/1
  vlan 200
```

Configure a Transparent Clock

All interfaces will be part of TC mode once configured.

Follow these steps to configure the switch as a transparent clock.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`switch> enable`	
Step 2	**configure terminal** **Example:** `switch# configure terminal`	Enters configuration mode.
Step 3	**ptp clock transparent domain** *domain-number* **profile default** **Example:** `switch(config)# ptp clock transparent domain 0  profile default`	This step configures the transparent type PTP clock, which updates the PTP time correction field to account for the delay in forwarding the traffic.
Step 4	(Optional) **vlan** *vlan-id* **Example:** `(config-ptp-clk)# vlan 100`	Configure VLAN for tagged packets.

Example

Example of Untagged

```
ptp clock transparent domain 0  profile default
```

Example of Tagged

```
ptp clock transparent domain 0  profile default
vlan 100
```

Configuring a PTP Power Profile

This section describes how to configure the switch to use the PTP Power Profile.

The Power Profile defines a subset of PTP which is intended to run over layer 2 networks, that is, Ethernet, but no Internet Protocol.

Note Power Profile-2017 is supported only in Transparent Clock mode.

Configure a Boundary Clock

If an interface is not added as part of BC clock, it is in forward mode exchanging PTP packets, which causes instability in PTP operation. To avoid this, disable PTP on all such interfaces using the **no ptp enable** command.

Follow these steps to configure the switch as a boundary clock:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `switch# configure terminal`	Enters configuration mode.
Step 3	**ptp clock boundary domain** *domain-number* **profile power** **Example:** `switch(config)# ptp clock boundary domain 0 profile default`	This step configures the boundary type PTP clock, which stops the PTP session from the grandmaster clock and acts as a PTP server or client clock downstream.
Step 4	**clock-port** *port-name* **Example:** `switch(config-ptp-clk)# clock-port dyn1`	Defines a new clock port.
Step 5	**transport ethernet multicast interface** *interface-type interface-number* **Example:** `switch(config-ptp-port)# transport ethernet multicast interface Gi1/1`	Specifies the transport mechanism for clocking traffic.
Step 6	(Optional) **vlan** *vlan-id* **Example:** `(config-ptp-port)# vlan 100`	Configure VLAN for tagged packets.

Example

Example of Untagged

```
ptp clock boundary domain 0  profile power
 clock-port dyn1
  transport ethernet multicast interface Gi1/1
 clock-port dyn2
  transport ethernet multicast interface Gi1/2
```

Example of Tagged

```
ptp clock boundary domain 0  profile power
 clock-port dyn1
  transport ethernet multicast interface Gi1/1
  vlan 100
 clock-port dyn2
  transport ethernet multicast interface Gi1/2
  vlan 100
```

Example of not Including TLV Extensions

```
ptp clock boundary domain 0 profile power
allow-without-tlv
```

Configure a Transparent Clock

Follow these steps to configure the switch as a transparent clock.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `switch# configure terminal`	Enters configuration mode.
Step 3	**ptp clock transparent domain** *domain-number* **profile power** **Example:** `switch(config)# ptp clock transparent domain 0  profile power`	This step configures the transparent type PTP clock, which updates the PTP time correction field to account for the delay in forwarding the traffic.

Step 4	• (Power Profile-2011): **ptp clock transparent domain** *domain-number* **profile power**
	• (Power Profile-2017): **ptp clock transparent domain** *domain-number* **profile power-2017**
	Example:
	`switch(config)# ptp clock transparent domain 0  profile power`
	Example:
	`switch(config)# ptp clock transparent domain 254  profile power-2017`
		This step configures the transparent type PTP clock, which updates the PTP time correction field to account for the delay in forwarding the traffic. The update helps improve the accuracy of the 1588 clock at the client.

Step 5	(Optional) **vlan** *vlan-id* **Example:** `(config-ptp-clk)# vlan 100`	Configure VLAN for tagged packets.

Example

Example of Untagged

```
ptp clock transparent domain 0 profile power
```

Example of Tagged

```
ptp clock transparent domain 0 profile power
vlan 100
```

Example of Tagged Power Profile-2017

```
ptp clock transparent domain 254 profile power-2017
vlan 100
```

Example of not Including TLV Extensions: Power Profile-2011

```
ptp clock transparent domain 0 profile power
allow-without-tlv
```

Example of not Including TLV Extensions: Power Profile-2017

```
ptp clock transparent domain 0 profile power-2017
allow-without-tlv
```

Enable PTP Forward Mode

Complete the steps in this section to enable PTP forward mode.

To enable PTP forward mode, and remove existing PTP clock configurations, you remove the existing PTP clock. When you do so, all interfaces automatically become part of forward mode.

Note Forward mode supports only the default profile.

Procedure

	Command or Action	Purpose
Step 1	**ptp clock boundary domain** *domain-number* **profile default** **Example:** `switch(config)# ptp clock boundary domain 0 profile default`	Configure the boundary type PTP clock. Doing so terminates the PTP session from the grandmaster clock and acts as a PTP server or client clock downstream.
Step 2	**clock-port** *port-name* **Example:** `switch(config)# clock-port 1`	Define a new clock port.
Step 3	**transport ipv4 multicast interface** *interface-type interface-number* **Example:** `switch(config-ptp-port)# transport ipv4 multicast interface Gi1/1`	Specifies the transport mechanism for clocking traffic.
Step 4	**exit**	Enters global configuration mode.

	Command or Action	Purpose
Step 5	**no ptp clock boundary domain** *domain-number* **profile default** **Example:** `switch(config)# no ptp clock boundary domain 0 profile default`	Remove the PTP clock configuration.
Step 6	**end**	Exit global configuration mode and returns to privileged EXEC mode.

Remove PTP Forward Mode

Complete the steps in this section to remove PTP forward mode.

To remove forward PTP forward mode configuration, you enable a PTP clock.

Note Forward mode supports only the default profile.

Procedure

Step 1 **no ptp clock**

Configure a clock to get out of forward mode.

Step 2 **ptp clock boundary domain** *domain-number* **profile default**

Example:

```
switch(config)# ptp clock boundary domain 0 profile default
```

Configure the boundary type PTP clock. Doing so terminates the PTP session from the grandmaster clock and acts as a PTP server or client clock downstream.

Step 3 **end**

Exit global configuration mode and returns to privileged EXEC mode.

Disable PTP

Complete the steps in this section to disable PTP on an interface.

Note The following procedure applies to both default and power modes.

Procedure

	Command or Action	Purpose
Step 1	**interface** *interface-id* **Example:** `switch(config)# interface gi1/1`	Enter interface configuration mode.
Step 2	**no ptp enable**	Disable PTP on the interface.

Enable GMC Block in Boundary Mode

Complete the steps in this section to enable GMC Block in boundary mode.

Procedure

	Command or Action	Purpose
Step 1	**ptp clock boundary domain** *domain number* **profile default** **Example:** `switch(config)# ptp clock boundary domain 0 profile default`	Configure the boundary type PTP clock, which terminates the PTP session from the grandmaster clock and acts as a PTP server or client clock downstream.
Step 2	**clock-port** *port-name* **Example:** `switch(config-ptp-clk)# clock-port 1`	Define a new clock port.
Step 3	**transport ipv4 multicast interface** *interface type interface number* **Example:** `switch(config-ptp-port)# transport ipv4 multicast interface Gi1/1`	
Step 4	**gmc-block** **Example:** `switch(config-ptp-port)# gmc-block`	Enable GMC Block.

Enable GMC Block in Transparent Mode

Complete the steps in this section to enable GMC Block in transparent mode.

Procedure

	Command or Action	Purpose
Step 1	**ptp clock transparent domain** *domain number* **profile power** **Example:** `switch(config)# ptp clock transparent domain 0 profile power`	This step configures the transparent type PTP clock, which updates the PTP time correction field to account for the delay in forwarding the traffic. The transparent clock can update some fields in the PTP packets to ensure that the client has greater time accuracy.
Step 2	**gmc-block** *interface* **Example:** `switch(config-ptp-clk)# gmc-block gi1/1`	Enable GMC Block.

PTP Alarms

PTP alarms can help you manage and monitor PTP on the switch. You can configure the PTP alarms to trigger the external alarm relay output and send system messages to a syslog server. The PTP alarms are raised only once for the first 5-minute interval and subsequently once every 30 minutes. PTP alarms are disabled by default.

The following sequence describes how PTP alarm timing works:

1. PTP alarm monitoring starts 5 minutes after bootup.
2. The PTP alarm is raised only once for the first 5-minute interval and subsequently once for an interval of 30 minutes.
3. The alarms are damped when there is continuous state change, for example, PTP port state flapping or PTP parent flapping.

The following table describes the types of PTP alarms:

Table 6: PTP Alarms

Alarm	Alarm Type	Clock Mode Supported	Description
PTP SLAVE port state change	Minor	Boundary and transparent clock modes	This alarm is raised when the PTP port state changes from "SLAVE" to any of the following PTP port states: Initializing, Faulty, Disabled, Listening, Pre_Master, Master, Passive, or Uncalibrated. A system message is generated when the PTP port state transitions between Slave and Passive Slave. This alarm remains raised until you clear the alarm.
PTP PASSIVE_SLAVE port state change	Minor	Boundary and transparent clock modes	This alarm is raised when the PTP port state changes from "PASSIVE-SLAVE" to any of the following PTP port states: Initializing, Faulty, Disabled, Listening, Pre_Master, Master, Passive, or Uncalibrated. A system message is generated when the PTP port state transitions between Slave and Passive Slave.
PTP Parent change	Minor	Boundary clock mode	This alarm raised when there is a change in PTP parent. This alarm remains raised until you clear the alarm.
PTP Time Property Clock Synchronized	Minor	Transparent clock mode	This alarm is raised when the PTP Clock Time Property "Clock Syntonized" field changes from TRUE to FALSE. This alarm is cleared when the "Clock Syntonized" field changes from FALSE to TRUE.

Configuring PTP Alarms

To enable and configure the global PTP alarms:

Procedure

Step 1 Enter global configuration mode:

configure terminal

Step 2 Enable PTP alarms:

alarm facility ptp enable

Step 3 Enable notifications to be sent to an SNMP server:

alarm facility ptp notifies

Step 4 Associate the PTP alarms to a relay.

alarm facility ptp relay major

Step 5 Send PTP alarm traps to a syslog server.

alarm facility ptp syslog

Example

```
Switch#configure terminal
Switch(config)#alarm facility ptp enable
Switch(config)#alarm facility ptp syslog
Switch(config)#end
Switch#show alarm settings
…..
…..
…..
PTP
     Alarm     Enabled
     Relay     MIN
     Notifies  Enabled
     Syslog    Enabled
Switch#show facility-alarm status
Source                   Severity Description                          Relay    Time
Switch                   MINOR    32 PTP Clock Parent change          NONE     Mar 09 2022
 01:23:45
GigabitEthernet1/1    MINOR    5 PTP SLAVE port state changed      NONE     Mar 09 2022
01:23:45
```

PTP over MACsec

Precision Time Protocol (PTP) over MACsec is a network timing feature that

- allows PTP messages to be transmitted and received over encrypted MACsec links,

- ensures accurate time synchronization even when traffic is secured by MACsec encryption, and
- provides robust, standards-based timing for industrial and enterprise networks with high-security requirements.

This capability allows devices to maintain precise timing across secure Ethernet connections, supporting applications where both security and clock synchronization are critical.

Table 7: Feature History Table

Feature Name	Release Information	Description
PTP over MACsec	26.1.1	This feature allows highly accurate time synchronization between devices, even when MACsec encrypts Ethernet traffic for security. This ensures industrial, utility, or automation networks can maintain precise timing and robust data protection on the same infrastructure.

MACsec encrypts the network traffic. When the network receives PTP packets, timestamp cannot be identified until after they are decrypted. This decryption process introduces variable delays that change timing precision. To solve this, packet timestamps are handled directly in the hardware before MACsec encrypts for both ingress and egress traffic. MACsec encrypts the entire packet while essential timing metadata is preserved.

Ingress timestamp

When the network receives a PTP packet, a precise hardware timestamp is captured at the physical port, before any other processing. The device securely carries this timestamp with the PTP packet and attaches it after decryption to ensure that encryption does not affect timing accuracy.

Egress timestamp

Outgoing PTP packets are flagged with a special signature that instructs the hardware to add a timestamp. The hardware then applies a precise departure timestamp as the packet leaves the port. This guarantees timing accuracy.

Limitations of PTP over MACsec

PTP over MACsec supports:

- IEEE 1588v2 default, C37.238-2011, and C37.238-2017 power profiles.
- Default PTP profile for boundary clock (BC) mode, End-to-End (E2E), GMC-BC mode, and Transparent Clock (TC)
- Power profile for boundary clock (BC) mode, peer-to-peer (P2P) mode, GMC-BC mode with forward mode, and transparent clock (TC) with less than 50 ns time inaccuracy in transparent clock mode.
- All types of MACsec (pre-shared key or certificate-based).
- 100 Mbps, 1 Gbps, and 10 Gbps interface speeds, but not 2.5 Gbps (mGig) interface speed.
- Simple Network Management Protocol (SNMP).

PTP over MACsec configuration on interfaces does not support Parallel Redundancy Protocol (PRP), High-availability Seamless Redundancy (HSR), Resilient Ethernet Protocol (REP), Device Level Ring (DLR), Media Redundancy Protocol (MRP), or port-channels.

Ensure that both PTP and MACsec are enabled and operational on the same interface.

Configure PTP over MACsec

Enable and verify PTP operation over encrypted MACsec links to maintain accurate time synchronization with secure traffic.

Use this task when you require both PTP timing and MACsec encryption on the same interface, for example, in secure industrial or utility networks.

Before you begin

- Ensure your MACsec and PTP licenses (if required) are active. For information on licenses, see Data Sheet

Perform these steps to configure PTP over MACsec.

Procedure

Step 1 Enable MACsec. To enable MACsec on the desired interfaces refer to How to Configure MACsec Encryption section of chapter Configuring MACsec Encryption.

Step 2 Configure PTP. To configure PTP globally and on the same interface(s), see the *Precision Time Protocol* chapter from the *Cisco IE3500 Series Switch Software Configuration Guide*.

Step 3 Verify PTP configuration. see the *Verifying the Configuration* chapter from the*Cisco IE3500 Series Switch Software Configuration Guide*.

Step 4 (Optional) Use the **show macsec interface GigabitEthernet***1/1* command to verify that encryption is active.

Example:

```
Switch# show macsec interface GigabitEthernet 1/1

MACsec is enabled
  Replay protect : enabled
  Replay window : 0
  Include SCI : yes
  Use ES Enable : no
  Use SCB Enable : no
  Admin Pt2Pt MAC : forceTrue(1)
  Pt2Pt MAC Operational : no
  Cipher : GCM-AES-128
  Confidentiality Offset : 0

 Capabilities
  ICV length : 16
  Data length change supported: yes
  Max. Rx SA : 16
  Max. Tx SA : 16
  Max. Rx SC : 8
  Max. Tx SC : 8
  Validate Frames : strict
  PN threshold notification support : Yes
  Ciphers supported : GCM-AES-128
```

```
                    GCM-AES-256
                    GCM-AES-XPN-128
                    GCM-AES-XPN-256

Access control : must secure

Transmit Secure Channels
 SCI : 3C5731BBB5850475
 SC state : inUse(1)
  Elapsed time : 7w0d
  Start time : 7w0d
  Current AN: 0
  Previous AN: -
  Next PN: 149757
  SA State: inUse(1)
  Confidentiality : yes
  SAK Unchanged : yes
  SA Create time : 00:04:41
  SA Start time : 7w0d
  SC Statistics
   Auth-only Pkts : 0
   Auth-only Bytes : 0
   Encrypted Pkts : 0
   Encrypted Bytes : 0
  SA Statistics
   Auth-only Pkts : 0
   Auth-only Bytes : 0
   Encrypted Pkts : 149756
   Encrypted Bytes : 16595088

 Port Statistics
  Egress untag pkts  0
  Egress long pkts  0

Receive Secure Channels
 SCI : 3C5731BBB5C504DF
 SC state : inUse(1)
  Elapsed time : 7w0d
  Start time : 7w0d
  Current AN: 0
  Previous AN: -
  Next PN: 149786
  RX SA Count: 0
  SA State: inUse(1)
  SAK Unchanged : yes
  SA Create time : 00:04:39
  SA Start time : 7w0d
  SC Statistics
   Notvalid pkts 0
   Invalid pkts 0
   Valid pkts 0
   Late pkts 0
   Uncheck pkts 0
   Delay pkts 0
   UnusedSA pkts 0
   NousingSA pkts 0
   Validated Bytes 0
   Decrypted Bytes 0
  SA Statistics
   Notvalid pkts 0
   Invalid pkts 0
   Valid pkts 149784
   Late pkts 0
   Uncheck pkts 0
```

```
   Delay pkts 0
   UnusedSA pkts 0
   NousingSA pkts 0
   Validated Bytes 0
   Decrypted Bytes 16654544

 Port Statistics
  Ingress untag pkts  0
  Ingress notag pkts  631726
  Ingress badtag pkts  0
  Ingress unknownSCI pkts  0
  Ingress noSCI pkts  0
  Ingress overrun pkts  0
```

SNMP Support for PTP MIBs

SNMP management information bases (MIBs) for Precision Time Protocol (PTP) is supported. These include CISCO-PTP-MIB. The feature enables you to get PTP-related information from a switch remotely.

The MIB is supported with boundary clock and transparent clock modes. It is supported in both the default and power profiles.

SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for monitoring and managing devices in a network. An SNMP networks includes the following components:

- **SNMP Manager:** A system that controls and monitors the activities of network hosts using SNMP. The most common managing system is a network management system (NMS). The term An can be a dedicated device used for network management or the applications used on such a device.
- **SNMP Agent:** The software component within a managed device that maintains the data for the device and reports this data, as needed, to managing systems. The agent resides on the switch. To enable an SNMP agent on a Cisco switch, you must define the relationship between the manager and the agent.
- **SNMP MIB**: An SNMP agent contains MIB variables. The SNMP manager can request information from an agent to store information in the agent. The agent gathers data from the SNMP MIB, the repository for information about device parameters and network data. The agent can also respond to manager requests to get or set data.

Note

- PTP over REP is not supported.

SNMP MIBs Supported with PTP Modes

This section lists the SNMP MIBs supported in different PTP modes.

The following MIBs are supported when the switch is configured with PTP boundary clock mode:

MIB	OID
cPtpClockNodeTable	1.3.6.1.4.1.9.9.760.1.1.3

MIB	OID
cPtpClockCurrentDSTable	1.3.6.1.4.1.9.9.760.1.2.1
cPtpClockParentDSTable	1.3.6.1.4.1.9.9.760.1.2.2
cPtpClockDefaultDSTable	1.3.6.1.4.1.9.9.760.1.2.3
cPtpClockTimePropertiesDSTable	1.3.6.1.4.1.9.9.760.1.2.5
cPtpClockPortTable	1.3.6.1.4.1.9.9.760.1.2.7
cPtpClockPortRunningTable	1.3.6.1.4.1.9.9.760.1.2.9

The following MIBs are supported when the switch is configured with PTP transparent clock mode:

MIB	OID
cPtpClockNodeTable	1.3.6.1.4.1.9.9.760.1.1.3
cPtpClockParentDSTable	1.3.6.1.4.1.9.9.760.1.2.2
cPtpClockDefaultDSTable	1.3.6.1.4.1.9.9.760.1.2.3
cPtpClockPortTable	1.3.6.1.4.1.9.9.760.1.2.7
cPtpClockPortRunningTable	1.3.6.1.4.1.9.9.760.1.2.9
cPtpClockSystemTimePropertiesTable	1.3.6.1.4.1.9.9.760.1.2.12

Prerequisites for Configuring SNMP PTP MIBs

Before you configure SNMP PTP MIBs, you should be familiar with the PTP protocol and configurations.

You should also be familiar with the Cisco SNMP Object Navigator, which translates an object identifier (OID) into object name or an object name into OID, enabling you to receive PTP object details. OIDs identify manged objects in an MIB.

Verifying the Configuration

PTP Configuration

You can use the following commands to verify the PTP configuration:

- show ptp clock dataset parent
- show ptp clock dataset current
- show ptp clock dataset time-properties
- show ptp clock dataset default
- show ptp clock running

- show ptp port dataset port
- show ptp lan clock
- show ptp lan port counters messages
- show ptp lan port counters errors
- show ptp lan foreign-master-record
- show ptp lan rogue-master-record
- show ptp lan histogram ?

  delay—Show PTP histogram of mean path delay

  offset—Show PTP histogram of offset

  time-error—Show PTP history of time error (last 15 days)
- show ptp lan history ?

  delay—Show PTP history of mean path delay (last 15 days)

  offset—Show PTP history of offset (last 15 days)

  time-error—Show PTP history of time error (last 15 days)

Default Profile Configuration

The following example shows the Default profile configuration:

```
Default profile MASTER
————————————
```

```
Switch# show run | sec ptp
ptp clock boundary domain 0  profile default
 clock-port 1
  transport ipv4 multicast interface Gi1/1
Switch#
Switch#sh ptp clock run

                      PTP Boundary Clock [Domain 0]  [Profile: default]

         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         FREERUN        1              140            30             Hot standby

                                PORT SUMMARY
                                                                        PTP Master
Name  Tx Mode      Role         Transport    State         Sessions     Port Addr

1     mcast        negotiated   Gi1/1        Master        1            UNKNOWN
Switch#
Switch#sh ptp lan port
  PTP PORT DATASET: GigabitEthernet1/1
    Port identity: clock identity: 0x84:eb:ef:ff:fe:d2:e0:3f
    Port identity: port number: 1
    PTP version: 2
    Port state: MASTER
    Delay request interval(log mean): 0
    Announce receipt time out: 3
    Announce interval(log mean): 1
    Sync interval(log mean): 0
```

```
    Delay Mechanism: End to End
    Peer delay request interval(log mean): 0
    Sync fault limit: 500000
    Rogue master block: FALSE
    Ingress phy latency: 590
    Egress phy latency: 0

Switch#

Default profile SLAVE
_____________
Switch#sh run | sec ptp
ptp clock boundary domain 0  profile default
 clock-port 1
  transport ipv4 multicast interface Gi1/1
Switch#
Switch#sh ptp clock run

                     PTP Boundary Clock [Domain 0]  [Profile: default]

         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         PHASE_ALIGNED  1              72             272            Hot standby

                               PORT SUMMARY
                                                                        PTP Master
Name  Tx Mode      Role         Transport    State         Sessions     Port Addr

1     mcast        negotiated   Gi1/1        Slave         1            UNKNOWN
Switch#
Switch#sh ptp lan po
Switch#sh ptp lan port
  PTP PORT DATASET: GigabitEthernet1/1
    Port identity: clock identity: 0x84:eb:ef:ff:fe:d2:e5:3f
    Port identity: port number: 0
    PTP version: 2
    Port state: SLAVE
    Delay request interval(log mean): 0
    Announce receipt time out: 3
    Announce interval(log mean): 1
    Sync interval(log mean): 0
    Delay Mechanism: End to End
    Peer delay request interval(log mean): 0
    Sync fault limit: 500000
    Rogue master block: FALSE
    Ingress phy latency: 590
    Egress phy latency: 0

Switch#
```

Power Profile Configuration

The following example shows the Power profile configuration:

```
Power profile MASTER
 ____________

Switch#show run | sec ptp
ptp clock boundary domain 0  profile power
 clock-port 1
  transport ethernet multicast interface Gi1/1
Switch#
```

```
Switch#
Switch# sh ptp clock running


                    PTP Boundary Clock [Domain 0]  [Profile: power]

         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         FREERUN        1              875328         1578627        Hot standby

                              PORT SUMMARY
                                                                      PTP Master
Name  Tx Mode      Role         Transport    State         Sessions     Port Addr

1     mcast        negotiated   Ethernet     Master        1            UNKNOWN
Switch#
Switch#
Switch#
Switch#
Switch#sh ptp lan port
  PTP PORT DATASET: GigabitEthernet1/1
    Port identity: clock identity: 0x84:eb:ef:ff:fe:d2:e0:3f
    Port identity: port number: 1
    PTP version: 2
    Port state: MASTER
    Delay request interval(log mean): 0
    Announce receipt time out: 3
    Peer mean path delay(ns): 35
    Announce interval(log mean): 0
    Sync interval(log mean): 0
    Delay Mechanism: Peer to Peer
    Peer delay request interval(log mean): 0
    Sync fault limit: 10000
    Rogue master block: FALSE
    Ingress phy latency: 590
    Egress phy latency: 0

Switch#
Switch#
Switch#

Power profile SLAVE
_________

Switch#show run | sec ptp
ptp clock boundary domain 0  profile power
 clock-port 1
  transport ethernet multicast interface Gi1/1
Switch#
Switch#
Switch#show ptp clock run


                    PTP Boundary Clock [Domain 0]  [Profile: power]

         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         PHASE_ALIGNED  1              57056          113937         Hot standby

                              PORT SUMMARY
                                                                      PTP Master
Name  Tx Mode      Role         Transport    State         Sessions     Port Addr

1     mcast        negotiated   Ethernet     Slave         1            UNKNOWN
```

```
Switch#
Switch#
Switch#
Switch#show ptp lan port
  PTP PORT DATASET: GigabitEthernet1/1
    Port identity: clock identity: 0x84:eb:ef:ff:fe:d2:e5:3f
    Port identity: port number: 0
    PTP version: 2
    Port state: SLAVE
    Delay request interval(log mean): 0
    Announce receipt time out: 3
    Peer mean path delay(ns): 35
    Announce interval(log mean): 0
    Sync interval(log mean): 0
    Delay Mechanism: Peer to Peer
    Peer delay request interval(log mean): 0
    Sync fault limit: 10000
    Rogue master block: FALSE
    Ingress phy latency: 590
    Egress phy latency: 0

Switch#
```

PTP Alarm Configuration

Use the following `show` commands to verify the PTP alarm configuration.

- `show facility-alarm status`

```
Switch#show facility-alarm status
Source                   Severity   Description                          Relay    Time
GigabitEthernet1/1    MINOR      5 PTP SLAVE port state changed        MIN      Jan 01
 1970 21:17:59
```

- `show ptp clock running`

```
Switch#show ptp clock running
 PTP Boundary Clock [Domain 10]  [Profile: power]
         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode
         PHASE_ALIGNED  2              1806           2615           Hot standby
                                PORT SUMMARY
                                                                        PTP Master
Name  Tx Mode      Role         Transport    State         Sessions     Port Addr
21    mcast       negotiated   Ethernet     Slave         1            UNKNOWN
Switch#
```

- `show ptp clock running`

```
Switch#show ptp clock running
 PTP Boundary Clock [Domain 10]  [Profile: power]
         State          Ports          Pkts sent      Pkts rcvd      Redundancy Mode
         PHASE_ALIGNED  2              1806           2615           Hot standby
                                PORT SUMMARY
                                                                        PTP Master
Name  Tx Mode      Role         Transport    State          Sessions     Port Addr
21    mcast       negotiated   Ethernet     Slave          1            UNKNOWN
```

Troubleshooting PTP

This section contains instructions for troubleshooting PTP by checking if the Transparent Clock is receiving messages from the Grandmaster Clock, verifying packet message and error counters, and running debug commands.

Verify that the Transparent Clock is Syntonized

You might want to verify that the Transparent Clock is syntonized to the Grand Master Clock—that is, that the Transparent Clock is logged to the Grand Master Clock. You might want to verify syntonization because the `show ptp clock running` command does not appy to the Transparent Clock. Subordinate clocks in the PTP network do not synchronize with the Grand Master Clock if the Transparent Clock is not syntonized.

Procedure

Verify that the Transparent Clock is syntonized.

Example:

```
switch(config-ptp-port)# sh ptp clock dataset time-properties
Clock Syntonized: TRUE
```

The command output is `TRUE` if the Transparent Clock is syntonized and `FALSE` if it is not. You also can check counters to see if PTP messages are being received.

Verify PTP Messages

You can verify whether messages are being received from the Grandmaster Clock.

Procedure

Verify PTP LAN port packet message.

Example:

```
switch# show ptp lan port counters messages

GigabitEthernet1/1

    Transmit                          Receive
    250    Announce                   0    Announce
    248    Sync                       0    Sync
    248    Follow_Up                  0    Follow_Up
     0     Delay_Req                  0    Delay_Req
     0     Delay_Resp                 0    Delay_Resp
    251    Pdelay_Req                251   Pdelay_Req
    251    Pdelay_Resp               251   Pdelay_Resp
    251    Pdelay_Resp_Follow_Up     251   Pdelay_Resp_Follow_Up
```

```
     0     Signaling                   0     Signaling
     0     Management                  0     Management
```

The preceding example shows that all the packets are being received.

The output of the command would vary, depending on which packets are not received. The following example shows output if follow-ups are not received.

```
GigabitEthernet1/1

Transmit Receive
0 Announce 1359 Announce
0 Sync 1359 Sync
0 Follow_Up 0 Follow_Up <<<
0 Delay_Req 0 Delay_Req
0 Delay_Resp 0 Delay_Resp
1362 Pdelay_Req 1359 Pdelay_Req
1359 Pdelay_Resp 1360 Pdelay_Resp
1359 Pdelay_Resp_Follow_Up 1360 Pdelay_Resp_Follow_Up
0 Signaling 0 Signaling
0 Management 0 Management
```

Note
You can use the following command to reset the counters: **clear ptp all all-clocks**

Verify PTP Error Counters

You can verify whether the error counters are continously incrementing, indicating that messages from the Grandmaster Clock aren't being received.

Procedure

Verify PTP LAN port

Example:

```
switch# show ptp lan port counters errors

GigabitEthernet1/1

     0   Sanity check failed      0   Blocked port
     0   Timestamp get failed     0   ParentId invalid
     0   Vlan mismatch            0   Gmcld invalid
     0   Domain mismatch          0   SequenceId invalid
     0   Sync fault               0   Unmatched Follow_Up
     0   Duplicate Sync           0   Unmatched Delay_Resp
     0   Duplicate Announce       0   Unmatched Pdelay_Resp
     0   Send error               0   Unmatched Pdelay_Resp_Follow_Up
     0   Misc error               0   Rogue master Sync
     0   Rogue master Follow-Up   0   Rogue master Announce
```

The preceding example shows that no error counters are being incremented.

The following example shows how errors increment when the VLAN in the ingress PTP message is different from the PTP VLAN used on the port.

```
switch# sh ptp lan port counters errors | beg 1/1
GigabitEthernet1/1

0 Sanity check failed 0 Blocked port
0 Timestamp get failed 0 ParentId invalid
1482 Vlan mismatch 0 GmcId invalid
0 Domain mismatch 0 SequenceId invalid
0 Sync fault 0 Unmatched Follow_Up
0 Duplicate Sync 0 Unmatched Delay_Resp
0 Duplicate Announce 0 Unmatched Pdelay_Resp
0 Send error 0 Unmatched Pdelay_Resp_Follow_Up
0 Misc error 0 Rogue master Sync
0 Rogue master Follow_Up 0 Ro
```

Note
You can use the following command to reset the counters: **clear ptp all all-clocks**

Debugging Commands

The debugging feature collects logs that can be analyzed to resolve any issues on the switch. You can enable debugging on the switch, which logs debugging lists to a file on the switch or to a boot device.

Note

- We recommend that you save the debugging information to a boot device rather than to an internal file.

  Make sure that you have enough space on the boot device for the debugging logs.

- Enable debugging only when you are troubleshooting and disable debugging when you finish. Disabling debugging when not troubleshooting reduces CPU overhead.

Enabling Debugging

Enter both of the following commands to enable debugging on the switch:

- `switch# set platform software trace timingd switch active R0 iot-ptp debug`
- `switch# set platform software trace timingd switch active R0 timingd debug`

Note When you use the preceding commands, debugging information is not printed on the screen and will be logged to an internal file. You cannot access the file directly, but you can store the debugging information to a boot device, which you can access.

Storing Debugging Information on a Boot Device

Use the following command to store the debugging information in the internal file to a boot device:

Note You can give the debug file any name you choose. The following example uses `timing-logs` as the filename.

```
Switch# show log process timingd internal to-file bootflash:timing-logs
```

When you use the preceding command, the debugging information is printed on the screen in addition to being saved to the boot device.

Checking Debugging

Enter both of the following commands to see if debugging information is being collected:

```
switch#sh platform software trace level timingd switch active R0  | inc iot-ptp
iot-ptp                         Debug
```

```
switch#sh platform software trace level timingd switch active R0  | inc timingd
timingd                         Debug
```

Disabling Debugging

Enter both of the following commands to disable debugging on the switch:

- `switch# set platform software trace timingd switch active R0 iot-ptp notice`
- `switch# set platform software trace timingd switch active R0 timingd notice`