



## 高精度時間プロトコル（PTP）の設定

- [PTP の制約事項と制限（1 ページ）](#)
- [PTP に関する情報（2 ページ）](#)
- [PTP の設定方法（10 ページ）](#)
- [例：レイヤ 2 およびレイヤ 3 PTP の設定（17 ページ）](#)
- [PTP の機能の履歴（21 ページ）](#)

### PTP の制約事項と制限

- 高精度時間プロトコル（PTP）は、C9300-48UXM スイッチモデルの最初の 16 個のみのダウンリンクポートと、すべてのアップリンクポートでサポートされます。
- デバイスの **show clock** コマンドの出力と、**show platform software fed switch active ptp domain** コマンドの出力に表示される PTP サーボクロックが同期していません。これらの出力は、スイッチで使用される 2 つの異なるクロックです。
- PTP トランスペアレントクロックモードでは、VLAN 間はサポートされません。
- PTP は、スタックされたデバイスではサポートされていません。
- スイッチは、IEEE 802.1AS および IEEE 1588 デフォルトプロファイルをサポートしており、どちらも相互に排他的です。スイッチで一度に有効化できるプロファイルは 1 つだけです。
- Cisco PTP の実装では、2 ステップクロックのみがサポートされ、1 ステップクロックはサポートされません。スイッチがグランドマスタークロックから 1 ステップメッセージを受信すると、メッセージはドロップされます。
- クロック同期の精度が低下するため、PTP が有効化されていないデバイスを PTP ネットワークに配置することは推奨されません。
- シグナリングメッセージは、Cisco IOS XE Gibraltar 16.12.1 ではサポートされていません。これらのメッセージは、処理されずにスイッチでドロップされます。
- 境界クロックモードが有効化されている場合、ブロードキャストターゲット ID を持つ管理メッセージはホップカウントを減らして転送されます。トランスペアレントクロック

モードが有効化されている場合、管理メッセージは境界ホップカウントを減らすことなく転送されます。

- ある PTP モードから別の PTP モードに直接移行することは推奨されません。 **no PTP mode** を使用して既存のモードをクリアし、新しいモードを設定します。
- IPv6 および VRF は PTP をサポートしません。
- トランスペアレント クロック モードは、ネイティブレイヤ 3 ポートおよび EtherChannel インターフェイスではサポートされません (境界クロックモードはネイティブレイヤ 3 ポートでサポートされます)。
- PTP はスーパーバイザモジュールのどのポートにも設定できません。
- ステートフルスイッチオーバー (SSO) は、PTP をサポートしていません。 PTP プロトコルは、スイッチオーバー後に再起動します。
- Precision Time Protocol (PTP) を備えた MACsec はサポートされません。

## PTP に関する情報

Precision Time Protocol (PTP) は、IEEE 1588 で、ネットワーク化された測定および制御システムのための高精度クロック同期として定義されており、さまざまな精度と安定性の分散デバイス クロックを含むパケットベース ネットワークでクロックを同期させるために開発されました。 PTP は、産業用のネットワーク化された測定および制御システム向けに特別に設計されており、最小限の帯域幅とわずかな処理オーバーヘッドしか必要としないため、分散システムでの使用に最適です。

### PTP を使用する理由

ピーク時課金、仮想発電機、停電の監視/管理などのスマート グリッド電力自動化アプリケーションは、非常に正確な時刻精度と安定性を必要とします。 タイミングの精度は、ネットワーク監視の精度とトラブルシューティング能力を向上させます。

時刻精度および同期の提供に加えて、PTP メッセージベースプロトコルは、イーサネットネットワークなどのパケットベース ネットワークに実装することもできます。 イーサネット ネットワークで PTP を使用する利点は次のとおりです。

- 既存のイーサネット ネットワークでコストを削減でき、セットアップも容易
- PTP データパケットは限られた帯域幅しか必要としない

## イーサネット スイッチと遅延

Precision Time Protocol (PTP) は、IEEE 1588 で、ネットワーク化された測定および制御システムのための高精度クロック同期として定義されており、さまざまな精度と安定性の分散デバイス クロックを含むパケットベース ネットワークでクロックを同期させるために開発されまし

た。PTPは、産業用のネットワーク化された測定および制御システム向けに特別に設計されており、最小限の帯域幅とわずかな処理オーバーヘッドしか必要としないため、分散システムでの使用に最適です。

### PTP を使用する理由

イーサネット ネットワークでは、スイッチは、ネットワーク デバイス間の全二重通信パスを提供します。スイッチは、パケットに含まれるアドレス情報を使用して、データパケットをパケット宛先に送信します。スイッチは、複数のパケットを同時に送信しようとする場合、送信前に失われないようにパケットの一部をバッファします。バッファがいっぱいになると、スイッチはパケットの送信を遅延させます。この遅延により、ネットワーク上のデバイスクロックが相互に同期しなくなる可能性があります。

スイッチがMACアドレステーブルを検索してパケットCRCフィールドを確認している間に、スイッチに入るパケットがローカルメモリに保存されると、追加の遅延が発生する可能性があります。このプロセスによりパケット転送時間のレイテンシが変動し、これらの変動によってパケット遅延時間が非対称になる場合があります。

PTPをネットワークに追加することで、デバイスクロックを正しく調整し、相互の同期を維持することにより、これらのレイテンシおよび遅延の問題を補うことができます。PTPにより、ネットワークスイッチは、境界クロック (BC) やトランスペアレントクロック (TC) などのPTP デバイスとして機能することが可能になります。

## メッセージベースの同期

クロック同期を確保するために、PTPでは、時刻源 (マスター) と受信者 (スレーブ) の間の通信パス遅延を正確に測定する必要があります。PTP は、マスタ デバイスとスレーブ デバイスの間でメッセージを送信して、遅延測定を決定します。メッセージの種類については、[PTPv2 メッセージタイプ \(4 ページ\)](#) で詳しく説明しています。次に、PTP は正確なメッセージ送受信時間を測定し、これらの時間を使用して通信パス遅延を計算します。その後、PTP は、計算された遅延に対してネットワーク データに含まれる現在の時刻情報を調整し、より正確な時刻情報を生成します。

この遅延測定原理によってネットワーク上のデバイス間のパス遅延が決定され、マスターとスレーブの間で送信される一連のメッセージを使用して、この遅延に関してローカルクロックが調整されます。一方向の遅延時間は、送信メッセージと受信メッセージのパス遅延を平均化することによって計算されます。この計算は対称的な通信パスを前提としていますが、スイッチドネットワークは、バッファリングプロセスのために必ずしも対称的な通信パスを持つとはかぎりません。

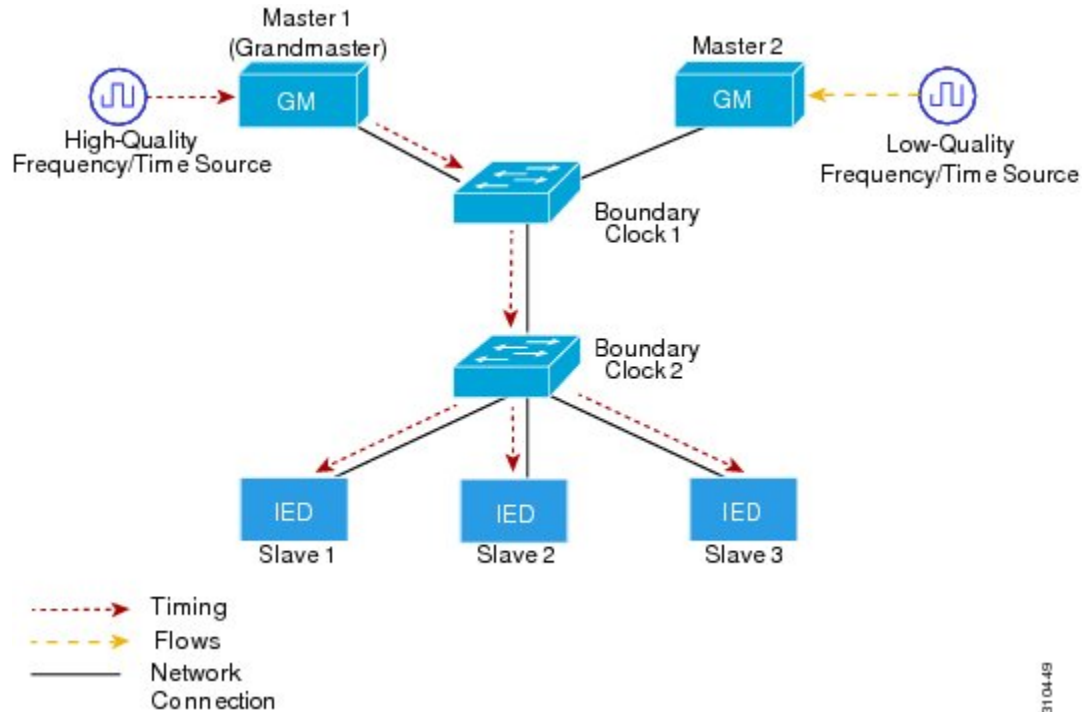
PTP は、トランスペアレントクロックを使用し、スイッチをネットワーク上のマスター ノードとスレーブ ノードに対して一時的に透過的にして、ネットワーク タイミング パケットの時間間隔フィールドの遅延を測定し、割り出す方法を提供します。エンドツーエンド透過クロックは、スイッチと同じ方法で、ネットワーク上のすべてのメッセージを転送します。



(注) Cisco PTP は、マルチキャスト PTP メッセージのみをサポートしています。

次の図に、グランドマスタークロック、境界クロックモードのスイッチ、およびデジタルリレーや保護デバイスなどのインテリジェントな電子機器 (IED) を含む標準的な 1588 PTP ネットワークを示します。この図では、Master 1 がグランドマスタークロックです。Master 1 が使用不能になると、同期のために境界クロックスレーブが Master 2 に切り替わります。

図 1: PTP ネットワーク



310449

## PTPv2 メッセージタイプ

PTP メッセージは、次の 2 つのタイプに分類されます。

- イベントメッセージ: データパケットがポートに到達するとき、またはポートから出るときにタイムスタンプでタグ付けされ、タイムスタンプに基づいてリンク遅延を計算するために使用されます。イベントメッセージは次のとおりです。

- Sync
- Delay\_Req
- Pdelay\_Req
- Pdelay\_Resp

•

アナウンスメッセージは、同期階層を確立するために使用されます。

一般的なメッセージ: タイムスタンプでタグ付けされず、マスター/スレーブ階層を確立するために使用されます。一般的なメッセージは次のとおりです。

- アナウンス
- Follow\_Up
- Delay\_Resp
- Pdelay\_Resp\_Follow\_Up

Sync、Delay\_Req、Follow\_Up、および Delay\_Resp メッセージは、通常のクロックと境界クロックを同期するために使用されます。

Pdelay\_Req、Pdelay\_Resp、および Pdelay\_Resp\_Follow\_Up メッセージは、トランスペアレントクロックのリンク遅延を測定するために使用されます。

**ベストマスタークロックアルゴリズム** は、グランドマスタークロックを選択し、ポートをマスターまたはスレーブとして割り当てます。これに続いて、すべてのマスターポートが、Sync メッセージとフォローアップメッセージを使用して、ダウンストリームスレーブへのクロックの供給を開始します。ダウンストリームスレーブはクロックを受信し、リンクの遅延、時間オフセット、周波数オフセット、および誤差パラメータを計算した後にクロックを更新します。

ダウンストリームスレーブは、以下のいずれかのメカニズムを使用してリンク遅延を計算します。

- [エンドツーエンドの遅延メカニズム \(5 ページ\)](#)
- [ピアツーピアの遅延メカニズム \(6 ページ\)](#)

## PTP イベントメッセージシーケンス

ここでは、同期中に発生する PTP イベントメッセージシーケンスについて説明します。

### エンドツーエンドの遅延メカニズム

遅延要求/応答メカニズム用に設定された通常クロックと境界クロックは、次のイベントメッセージを使用してタイミング情報を生成し、伝えます。

- Sync
- Delay\_Req
- Follow\_Up
- Delay\_Resp

これらのメッセージは、次のシーケンスで送信されます。

1. マスターが、スレーブに Sync メッセージを送信し、それが送信された時刻 (t1) を記録します。
2. スレーブが、Sync メッセージを受信し、受信した時刻 (t2) を記録します。
3. マスターが、Follow\_Up メッセージにタイムスタンプ t1 を組み込むことによって、タイムスタンプ t1 をスレーブに伝えます。

4. スレーブが、マスターに Delay\_Req メッセージを送信し、それが送信された時刻 (t3) を記録します。
5. マスターが、Delay\_Req メッセージを受信し、受信した時刻 (t4) を記録します。
6. マスターが、Delay\_Resp メッセージにタイムスタンプ t4 を組み込むことによって、タイムスタンプ t4 をスレーブに伝えます。

このシーケンスの後、スレーブは4つのタイムスタンプをすべて保有します。これらのタイムスタンプを使用して、マスターに対するスレーブクロックのオフセットと、2つのクロック間のメッセージの平均伝達時間を計算できます。

オフセット計算は、メッセージがマスターからスレーブに伝達される時間がスレーブからマスターに伝達されるために必要な時間と同じであるという前提に基づいています。この前提は、非対称的なパケット遅延時間のためにイーサネットネットワーク上では必ずしも妥当ではありません。

図 2: 手順の詳細: エンドツーエンドの遅延メカニズム



## ピアツーピアの遅延メカニズム

ネットワークの階層内に複数のレベルの境界クロックが含まれており、それらの間に非PTP対応デバイスがある場合は、同期の精度が低下します。

ラウンドトリップ時間は  $\text{mean\_path\_delay}/2$  と等しいことが前提となっていますが、この前提はイーサネットネットワークでは必ずしも妥当ではありません。精度を向上させるために、各中間クロックの滞留時間がエンドツーエンド透過クロックのオフセットに追加されます。ただし、滞留時間にはピア間のリンク遅延が考慮されていません。ピア間のリンク遅延はピアツーピア透過クロックによって処理されます。

ピアツーピア透過クロックは、ピア遅延メカニズムを実装する2つのクロックポート間のリンク遅延を測定します。リンク遅延は、Sync メッセージと Follow\_Up メッセージのタイミング情報を補正するために使用されます。

ピアツーピア透過クロックは、次のイベント メッセージを使用します。

- Pdelay\_Req
- Pdelay\_Resp
- Pdelay\_Resp\_Follow\_Up

これらのメッセージは、次のシーケンスで送信されます。

1. ポート 1 が、Pdelay\_Req メッセージのタイムスタンプ t1 を生成します。
  2. ポート 2 が、このメッセージを受信してタイムスタンプ t2 を生成します。
  3. ポート 2 が、Pdelay\_Resp メッセージを返してタイムスタンプ t3 を生成します。
- 2つのポート間の周波数オフセットによるエラーを最小限に抑えるために、ポート 2 は、Pdelay\_Req メッセージを受信した後に、できるかぎり迅速に Pdelay\_Resp メッセージを返します。

4. ポート 2 が、Pdelay\_Resp メッセージと Pdelay\_Resp\_Follow\_Up メッセージでそれぞれタイムスタンプ  $t_2$  とタイムスタンプ  $t_3$  を返します。
5. ポート 1 が、Pdelay\_Resp メッセージを受信した後に、タイムスタンプ  $t_4$  を生成します。その後、ポート 1 が、4 つのタイムスタンプ ( $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$ ) を使用して平均リンク遅延を計算します。

図 3: 手順の詳細: ピアツーピアの遅延メカニズム



## ローカルクロックの同期

理想的な PTP ネットワークでは、マスタークロックとスレーブクロックは同じ周波数で動作します。ただし、このネットワークでは「ばらつき」が発生する可能性があります。このばらつきは、マスタークロックとスレーブクロックの間の周波数の差です。デバイスハードウェアのタイムスタンプ情報とフォローアップメッセージ（スイッチで代行受信）を使用してローカルクロックの周波数を調整し、マスタークロックの周波数と一致させることによって、ばらつきを補うことができます。

## ベストマスタークロックアルゴリズム

ベストマスタークロックアルゴリズム (BMCA) は PTP 機能の基盤です。BMCA は、ネットワーク上の各クロックが、そのサブドメイン内で認識できるすべてのクロック（そのクロック自体を含む）のうちのベストマスタークロックを決定する方法を指定します。BMCA は、アナウンス間隔ごとにネットワーク内の各ポート上でローカルかつ継続的に動作し、ネットワーク構成における変更を迅速に調整します。IEEE 1588-2008 に基づく BMCA は、クロックプロパティのアドバタイジングに対するアナウンスメッセージを使用します。

BMCA は、次の基準を使用して、サブドメイン内のベストマスタークロックを決定します。

- クロック品質（たとえば、GPS は最高品質とみなされます）
- クロックの時刻基準のクロック精度
- 局部発振器の安定性
- グランドマスターに最も近いクロック

IEEE 1588-2008 に基づく BMCA は、受信したデータセットとともに独自のデータセットを使用し、次のプロパティを持つ属性に基づいて、指定された順序で最適なクロックを決定します。

- 優先順位 1：各クロックにユーザーが割り当てた優先順位。有効な範囲は 0 ~ 255 です。デフォルト値は 128 です。
- クラス：クロックが属するクラス。各クラスには独自の優先順位があります。
- 精度：クロックと UTC 間の精度（ナノ秒）
- バリエーション：クロックの変動
- 優先順位 2：最終的な優先順位。有効な範囲は 0 ~ 255 です。デフォルト値は 128 です。



- 固有識別子：64 ビット拡張固有識別子 (EUI)

BMCA は、ベスト マスター クロックを特定するだけでなく、次のことを保証して、PTP ネットワーク上でのクロック競合の発生を確実に防止します。

- クロックが相互にネゴシエートする必要がない。
- マスター クロック特定プロセスの結果として、マスター クロックが2つ存在する、またはマスター クロックが存在しないといった不適切な設定になっていない。

## PTP クロック

PTP ネットワークは、PTP 対応デバイスで構成されます。PTP 対応デバイスは、通常、次のクロック タイプで構成されます。

### グランドマスター クロック

PTP ドメイン内では、グランドマスター クロックが、PTP によるクロック同期の主時刻源です。グランドマスター クロックは、通常、GPS や原子時計などの非常に正確な時刻源を持っています。ネットワークが外部時刻リファレンスを必要とせず、内部で同期する必要のみがある場合、グランドマスター クロックはフリー ランできます。



- (注) クロック精度の低下が懸念されるため、ネットワークでスイッチを GM クロックとして使用することは推奨されません。

### オーディナリ クロック

オーディナリ クロックは、1 つの PTP ポートを持つ PTP クロックです。このクロックは PTP ネットワークでノードとして機能し、BMCA がサブドメイン内のマスターまたはスレーブとして選択できます。オーディナリ クロックは、同期が必要なデバイスに接続されているネットワーク上のエンドノードとして使用されるため、PTP ネットワーク上で最も一般的なクロックタイプです。オーディナリ クロックには、外部デバイスに対するさまざまなインタフェースがあります。

### 境界クロック

PTP ネットワークにおける境界クロックは、標準のネットワークにおけるスイッチやルータに代わる動作をします。境界クロックには複数の PTP ポートがあり、各ポートは個別の PTP 通信パスへのアクセスを提供します。境界クロックは、PTP ドメイン間のインタフェースを提供します。このクロックは、すべての PTP メッセージを代行受信して処理し、他のすべてのネットワーク トラフィックを通過させます。また、境界クロックは、BMCA を使用して、任意のポートから見えるクロックから最善のものを選択します。選択されたポートは、スレーブとして設定されます。マスターポートはダウンストリームに接続されたクロックを同期させ、スレーブポートはアップストリーム マスター クロックと同期します。



## 透過クロック

PTP ネットワークの透過クロックの役割は、PTP イベントメッセージの一部である時間間隔フィールドを更新することです。この更新により、スイッチの遅延が補われ、1 ピコ秒未満の精度が実現されます。

次の 2 種類の透過クロックがあります。

**エンドツーエンド (E2E) トランスペアレントクロック**は、SYNC メッセージと DELAY\_REQUEST メッセージに関して PTP イベントメッセージ中継時間（「常駐時間」とも呼ばれる）を測定します。この測定された中継時間は、対応するメッセージのデータフィールド（補正フィールド）に追加されます。

- SYNC メッセージの測定された中継時間は、対応する SYNC メッセージまたは FOLLOW\_UP メッセージの補正フィールドに追加されます。
- DELAY\_REQUEST メッセージの測定された中継時間は、対応する DELAY\_RESPONSE メッセージの補正フィールドに追加されます。

スレーブは、スレーブの時刻とマスターの時刻の間のオフセットを決定するときにこの情報を使用します。E2E 透過クロックは、リンク自体の伝播遅延は補正しません。

**ピアツーピア (P2P) 透過クロック**は、前述のように、E2E 透過クロックと同じ方法で PTP イベントメッセージ中継時間を測定します。さらに、P2P 透過クロックは上流リンク遅延を測定します。上流リンク遅延は、上流の隣接する P2P 透過クロックと考慮対象の P2P 透過クロックの間の推定パケット伝搬遅延です。

これらの 2 つの時間（メッセージ中継時間とアップストリームリンク遅延時間）は両方とも PTP イベントメッセージの修正フィールドに追加され、スレーブによって受信されるメッセージの修正フィールドにはすべてのリンク遅延の合計が含まれます。理論的には、これは、SYNC パケットのエンドツーエンドの遅延の合計（マスターからスレーブまで）です。

次の図に、PTP ネットワーク内のマスター/スレーブ階層に含まれる PTP クロックを示します。

図 4: PTP クロック階層



## PTP プロファイル

PTP プロファイルの IEEE 1588 定義は、「デバイスに適用可能な、許容される一連の PTP 機能」です。PTP プロファイルは、通常、特定のタイプのアプリケーションまたは環境に固有のものであり、次の値を定義します。

- ベスト マスター クロック アルゴリズム オプション
- 設定管理オプション
- パス遅延メカニズム（ピア遅延または遅延要求/応答）
- すべての PTP 設定可能属性およびデータ セット メンバーの範囲とデフォルト値
- グランドマスターに最も近いクロック
- 必要な、許可される、または禁止されるトランスポート メカニズム

- 必要な、許可される、または禁止されるノードタイプ
- 必要な、許可される、または禁止されるオプション

## Default Profile

スイッチのデフォルトの PTP プロファイルモードは、デフォルトプロファイルモードです。トランスポートの PTP モードはレイヤ 2 およびレイヤ 3 です。

デフォルトでは、PTP デフォルトプロファイルはこれらのプラットフォームでグローバルに無効化されています。

# PTP の設定方法

## PTP Default プロファイルの設定

レイヤ 2 PTP をグローバルに設定するには、次の手順を実行します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例 : Device> <b>enable</b>	特権 EXEC モードを有効にします。 パスワードを入力します (要求された場合)。
ステップ 2	<b>configure terminal</b> 例 : Device <b>configure terminal</b>	グローバル コンフィギュレーションモードを開始します。
ステップ 3	<b>ptp mode {boundary {delay-req             pdelay-req}              e2transparent p2pttransparent }</b> 例 : Device(config)# <b>ptp mode boundary            delay-req</b> Device(config)# <b>ptp mode boundary            pdelay-req</b> Device(config)# <b>ptp mode e2transparent</b> Device(config)# <b>ptp mode p2pttransparent</b>	同期クロックモードを指定します。 <ul style="list-style-type: none"> <li>• <b>boundary</b> : スイッチが最良のマスタークロックを選択する作業に参加できるようにするモード。自らのクロックよりも優れたクロックが検出されない場合、スイッチはネットワークのグランドマスタークロックになり、接続しているすべての装置の親クロックになります。最良のマスターがスイッチに接続されたクロックであると判断された場合、スイッチはそのクロックにクロックの子として同期し、他のポートに接続された装置の親クロックとして機能します。最初の同期のあと、スイッ</li> </ul>

	コマンドまたはアクション	目的
		<p>チと接続済み装置は、タイミングメッセージを交換して、クロックのオフセットとネットワークの遅延による時間の歪みを修正します。このモードは、過負荷または重負荷の状態により大きな遅延ジッタが生じるときに使用します。</p> <ul style="list-style-type: none"> <li>• <b>e2transparent</b> : すべてのスイッチポートをスイッチに接続されたグランドマスタークロックと同期させるためのスイッチのモード。これがデフォルトのクロックモードです。スイッチは、スイッチを通過するすべてのパケットが被る遅延（「滞留時間」といいます）を修正します。このモードでは、境界モードよりもジッタとエラーの蓄積が少なくなります。</li> <li>• <b>p2transparent</b> : スイッチが自身のクロックをマスタークロックと同期させないようにするモード。このモードのスイッチは、マスタークロックの選択に参加せず、すべてのポートでデフォルト PTP クロックモードを使用します。</li> </ul> <p>(注) PTP デフォルトプロファイルがデバイスでグローバルに有効になると、PTP はすべてのインターフェイスで有効になります。個別のインターフェイスで PTP を選択的に無効化するには、インターフェイスの設定で <b>no ptp enable</b> コマンドを使用します。</p>
ステップ 4	<p><b>[no]ptp domain value</b></p> <p>例 :</p> <pre>Device(config)# ptp domain 8</pre>	<p>PTP のドメイン値を設定します。</p> <ul style="list-style-type: none"> <li>• 単一のドメイン値を設定できます。指定できる範囲は4～127です。デフォルト値は0です ptp domain コマ</li> </ul>

	コマンドまたはアクション	目的
		ンドによって値がデフォルトに設定されることはありません。

## レイヤ2インターフェイス上の Precision Time Protocol の設定

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>interface interface-id</b> 例： Device(config)# <b>interface TenGigabitEthernet1/0/1</b>	設定する物理インターフェイスを指定し、インターフェイスコンフィギュレーション モードを開始します。指定するインターフェイスは、EtherChannel の一部にすることができます。
ステップ 4	<b>[no ] ptp enable</b>	
ステップ 5	<b>ptp vlan vlan-id</b> 例： Device(config-if)# <b>ptp vlan 5</b>	トランク ポートで PTP VLAN を設定します。デフォルトは、トランク ポートのネイティブ VLAN です。境界モードでは、PTP VLAN 内の PTP パケットのみが処理され、他の VLAN からの PTP パケットは破棄されます。インターフェイスで PTP VLAN を設定する前に、PTP VLAN を作成し、トランクポートで許可する必要があります。
ステップ 6	<b>end</b> 例： Device(config-if)# <b>end</b>	特権 EXEC モードに戻ります。

## SVI またはレイヤ 3 インターフェイスでの PTP の設定

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>ptp transport ipv4 udp</b> 例： Device(config)# <b>ptp transport ipv4 udp</b>	PTP 転送モードとして IPv4 を設定します。 (注) レイヤ 3 PTP の PTP 転送方式としてサポートされるのは IPv4 だけです。

## PTP での送信元 IP の設定

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>[no]ptp source {source-address   loopback   vlan}</b> 例： Device(config)# <b>ptp source source address</b> Device(config)# <b>ptp source loopback</b> Device(config)# <b>ptp source vlan</b>	同期クロックモードを指定します。  <ul style="list-style-type: none"> <li>• <b>source address</b> : 設定すると、すべてのインターフェイスの PTP メッセージでこの送信元 IP が伝送されます。</li> <li>• <b>loopback</b> : すべてのインターフェイスの PTP メッセージで、ループバック</li> </ul>

	コマンドまたはアクション	目的
		<p>ク インターフェイスで設定された IP が伝送されます。</p> <ul style="list-style-type: none"> <li>• <b>vlan</b> : PTP メッセージで、ポートに対応する SVI インターフェイスで設定された IP が伝送されます。</li> </ul> <p>(注) デフォルトとして <code>no ptp source</code> コマンドを使用できます。</p>
ステップ 4	<b>end</b> 例 : Device(config-if)# <b>end</b>	特権 EXEC モードに戻ります。

## PTP タイマーの設定

PTP タイマー値をデフォルト値から必要な値に変更するには、次の手順を実行します。

### 始める前に

タイマー入力は、ログ平均メッセージ間隔値の単位で測定されます。**interval** キーワードの *value* (秒単位) を決定するには、対数目盛を使用します。次の表に、対数目盛で秒数に変換された *value* キーワードの例を示します。

入力される値	対数計算	秒単位の値
-1	$2^{-1}$	1/2
0	$2^0$	1

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例 : Device> <b>enable</b>	<p>特権 EXEC モードを有効にします。</p> <p>パスワードを入力します (要求された場合)。</p>
ステップ 2	<b>configure terminal</b> 例 : Device# <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	<b>interface</b> <i>interface-id</i> 例 : Device(config)# <b>interface</b> <b>gigabitethernet2/0/1</b>	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	<b>ptp announce</b> { <i>interval value</i>   <i>timeout count</i> } 例 : Device(config-if) # <b>ptp announce</b> <b>interval 1</b>	(任意) インターフェイス上の PTP アナウンス メッセージ間の間隔またはタイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 <ul style="list-style-type: none"> <li>• <b>interval value</b> : アナウンスメッセージを送信する対数平均間隔を設定します。範囲は0～4です。デフォルト値は0 (1秒) です。</li> <li>• <b>timeout count</b> : タイムアウトメッセージをアナウンスする対数平均間隔を秒単位で設定します。範囲は2～10です。デフォルトは3 (8秒) です。</li> </ul>
ステップ 5	<b>ptp sync</b> { <i>interval value</i>   <i>limit offset-value</i> } 例 : Device(config-if) # <b>ptp sync interval</b> <b>1</b>	(任意) インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 <ul style="list-style-type: none"> <li>• <b>interval value</b> : 同期メッセージを送信する対数平均間隔を設定します。範囲は、-3～1です。デフォルト値は0 (1秒) です。</li> <li>• <b>limit offset-value</b> : PTP が再同期を試みるまでの、最大クロックオフセット値を設定します。範囲は50～500000000 ナノ秒です。デフォルトは500000000 ナノ秒です。</li> </ul>
ステップ 6	<b>ptp delay-req interval value</b> 例 : Device(config-if) # <b>ptp delay-req</b> <b>interval 1</b>	(任意) ポートがマスターステートの場合に PTP 遅延要求メッセージ間で許可される対数平均間隔を設定します。指定できる範囲は0～5です。デフォルト値は0 (1秒) です。
ステップ 7	<b>ptp pdelay-req interval value</b> 例 : Device(config-if) # <b>ptp pdelay-req</b> <b>interval 1</b>	(任意) ポートがマスターステートの場合に遅延要求メッセージ間で許可される対数平均間隔を設定します。指定できる範囲は0～5です。デフォルト値は0 (1秒) です。



	コマンドまたはアクション	目的
ステップ 8	<b>end</b> 例： Device(config-if)# <b>end</b>	特権 EXEC モードに戻ります。

## Precision Time Protocol のクロック値の設定

PTP クロックの値（優先順位 1 および優先順位 2）を設定するには、次の手順を実行します。

### 手順

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> <b>enable</b>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<b>configure terminal</b> 例： Device# <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>ptp priority1 value</b> 例： Device(config)# <b>ptp priority1 120</b>	PTP クロックの優先順位 1 の値を設定します。有効な範囲は 0 ~ 255 です。デフォルト値は 128 です。  (注) 優先順位 1 の値が 255 に設定されると、クロックはグランドマスターとは見なされません。
ステップ 4	<b>ptp priority2 value</b> 例： Device(config)# <b>ptp priority2 120</b>	PTP クロックの優先順位 2 の値を設定します。有効な範囲は 0 ~ 255 です。デフォルト値は 128 です。
ステップ 5	<b>exit</b> 例： Device(config)# <b>exit</b>	グローバル コンフィギュレーション モードに戻ります。

## 例：レイヤ2およびレイヤ3 PTP の設定

例

**show ptp port interface** *interface-name*

PTP ポートの状態を確認するには、**show ptp port interface** *interface-name* コマンドを使用します。

すべてのインターフェイスの PTP ポートの状態を確認するには、**show ptp brief** コマンドを使用します。

次に、遅延要求メカニズムを使用した境界モード設定の出力例を示します。

```
Device# show ptp port GigabitEthernet1/0/45
PTP PORT DATASET: GigabitEthernet1/0/45
  Port identity: clock identity: 0xCC:46:D6:FF:FE:C5:24:0
  Port identity: port number: 45
  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: 500000000
```

次に、遅延要求メカニズムを使用した境界モード設定の出力例を示します。

```
Device# show ptp port GigabitEthernet1/0/45
PTP PORT DATASET: GigabitEthernet1/0/45
  Port identity: clock identity: 0xCC:46:D6:FF:FE:C5:24:0
  Port identity: port number: 45
  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: Peer to Peer
  Peer delay request interval(log mean): 0
  Sync fault limit: 500000000
```

**show ptp brief**

すべてのインターフェイスの PTP ポートの状態を確認するには、**show ptp brief** コマンドを使用します。

次に、**show ptp brief** コマンドの出力例を示します。

```
Device# show ptp brief
Interface                               Domain      PTP State
```

```
TenGigabitEthernet1/0/1      0      MASTER
TenGigabitEthernet1/0/2      0      SLAVE
TenGigabitEthernet1/0/3      0      FAULTY
```

### show ptp clock

PTP クロックアイデンティティの詳細およびプライオリティ 1 とプライオリティ 2 の設定値を確認するには、**show ptp clock** コマンドを使用します。

次に、**show ptp clock** コマンドの出力例を示します。

```
Device# show ptp clock
PTP CLOCK INFO
  PTP Device Type: Boundary clock
  PTP Device Profile: Default Profile
  Clock Identity: 0xCC:46:D6:FF:FE:C5:24:0 <<clock identity of this
switch>>
  Clock Domain: 0
  Number of PTP ports: 52
  Priority1: 128
  Priority2: 128
  Clock Quality:
    Class: 248
    Accuracy: Unknown
    Offset (log variance): 16640
  Offset From Master(ns): 0
  Mean Path Delay(ns): 0
  Steps Removed: 1
```

### show ptp parent

境界モードでデバイスが同期されているグランドマスタークロック ID を特定するには、**show ptp parent** コマンドを使用します。




---

(注) **show ptp parent** は、デバイスがトランスペアレントクロック モードに設定されている場合、出力を表示しません。

---

次に、**show ptp parent** コマンドの出力例を示します。

```
Device# show ptp parent
PTP PARENT PROPERTIES
  Parent Clock:
  Parent Clock Identity: 0x0:11:1:FF:FE:0:0:1
  Parent Port Number: 1
  Observed Parent Offset (log variance): 16640
  Observed Parent Clock Phase Change Rate: N/A

  Grandmaster Clock:
  Grandmaster Clock Identity: 0x0:11:1:FF:FE:0:0:1 <<Grandmaster
clock identity to which the device is synced to>>
  Grandmaster Clock Quality:
    Class: 6
```

```
Accuracy: Within 25ns
Offset (log variance): 0
Priority1: 128
Priority2: 128
```

### show platform software fed switch active ptp domain 0

遅延要求メカニズムを使用して境界モードで設定されたデバイスのグランドマスタークロックに対するローカルサーボ PTP クロックの同期を確認するには、**show platform software fed switch active ptp domain 0** コマンドを使用します。

```
Device# show platform software fed switch active ptp domain 0
```

```
Displaying data for domain number 0
=====
```

```
Profile Type : DEFAULT
Profile State: enabled
Clock Mode : BOUNDARY CLOCK
Delay mechanism: End-to-End
PTP clock : 2017-6-28 5:58:59
Transport Method: L2 Ethernet
```

デフォルトでは、デバイスが PTP グランドマスタークロックに同期されていない場合、ローカルの PTP クロックは EPOCH 時間 (1970 年 1 月 1 日) を表示します。

### 例

#### show ptp port interface *interface-name*

PTP ポートの状態を確認するには、**show ptp port interface *interface-name*** コマンドを使用します。

すべてのインターフェイスの PTP ポートの状態を確認するには、**show ptp brief** コマンドを使用します。

次に、遅延要求メカニズムを使用した境界モード設定の出力例を示します。

```
Device# show ptp port FortyGigabitEthernet1/0/10
PTP PORT DATASET: FortyGigabitEthernet1/0/10
Port identity: clock identity: 0x0:A3:D1:FF:FE:5A:12:0
Port identity: port number: 10
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
    << PTP mode delay >>
Peer delay request interval(log mean): 0
Sync fault limit: 500000000
```

#### show ptp parent

境界モードでデバイスが同期されているグランドマスタークロック ID を特定するには、**show ptp parent** コマンドを使用します。



(注) **show ptp parent** は、デバイスがトランスペアレントクロック モードに設定されている場合、出力を表示しません。

次に、**show ptp parent** コマンドの出力例を示します。

```
Device# show ptp parent
PTP PARENT PROPERTIES
  Parent Clock:
    Parent Clock Identity: 0x38:E:4D:FF:FE:81:FE:29
  << Immediate next Master >>
  Parent Port Number: 196
  Observed Parent Offset (log variance): 17258
  Observed Parent Clock Phase Change Rate: N/A

  Grandmaster Clock:
    Grandmaster Clock Identity: 0x0:0:0:5:0:0:0:1
  << GM: External Clock Source acting Grand Master >>
  Grandmaster Clock Quality:
    Class: 6
    Accuracy: Within 1us
    Offset (log variance): 0
    Priority1: 128
    Priority2: 128
```

#### **show platform software fed switch active ptp domain 0**

遅延要求メカニズムを使用して境界モードで設定されたデバイスのグランドマスタークロックに対するローカルサーボ PTP クロックの同期を確認するには、**show platform software fed switch active ptp domain 0** コマンドを使用します。

```
Device# show platform software fed switch active ptp domain 0
Displaying data for domain number 0
=====

Profile Type : DEFAULT
Profile State: enabled
Clock Mode : BOUNDARY CLOCK
Delay Mechanism: : END-TO-END
PTP clock : 2017-12-15 15:27:27
mean_path_delay 214 nanoseconds
Transport Method : udp-ipv4                                << PTP Transport Method
>>
```

表 1: debug コマンド

コマンド	目的
<b>debug ptp messages</b>	PTP メッセージのデバッグをイネーブルにします。
<b>debug ptp error</b>	PTP エラーのデバッグをイネーブルにします。
<b>debug ptp bmc</b>	PTP ベストマスタークロックアルゴリズムのデバッグをイネーブルにします。
<b>debug ptp event</b>	PTP ステート イベントのデバッグをイネーブルにします。

## PTP の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	IEEE 1588v2 高精度時間プロトコル (PTP) のサポート	PTP は、精度と安定性が異なる分散デバイスクロックを含むパケットベースのネットワークでクロックを同期させるために開発されました。  レイヤ 2 ポートでの PTP のサポートが導入されました。  この機能のサポートは、Cisco Catalyst 9300 シリーズスイッチの 9300 スイッチモデルでのみサポートされるようになりました。
Cisco IOS XE Gibraltar 16.12.1	ネイティブレイヤ 3 ポートの PTP	ネイティブレイヤ 3 ポートでの PTP のサポートが導入されました。
Cisco IOS XE Amsterdam 17.2.1	EtherChannel インターフェイス上の IEEE 1588v2 PTP	EtherChannel での PTP のサポートが導入されました。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> [英語] からアクセスします。





## 翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。