ネットワーク タイム プロトコル

ネットワーク タイム プロトコル

ネットワーク時間プロトコル（NTP）は、ネットワーク内のデバイスのクロックを協定世界時（UTC）に同期させるネットワークプロトコルです。システムが正確な同期された時間を維持するために重要な役割を果たします。これは、タイムスタンプ、ロギング、ネットワーク調整などのさまざまなアプリケーションに不可欠です。

NTP は UDP ポート 123 で動作し、正確な時間同期と、ローカルエリアネットワーク（LAN）内でのより高い精度を実現します。階層型ストラタムシステムを使用して、デバイス間で時間同期を分散できます。

ストラタムとは、時間同期プロセスの階層レベルを指し、プライマリ基準クロック（ストラタム 0）とのデバイスの距離を示します。

- **Stratum 0**：アトミッククロック、GPSクロック、無線クロックなど、非常に正確な基準クロックです。これらはネットワークデバイス自体ではありませんが、最も正確な時刻源を提供します。

- **ストラタム 1**：これらは、ストラタム 0 参照クロックに直接接続されているサーバーです。これらは、ネットワーク内の他のデバイスのプライマリ時間ソースとして機能します。
- **ストラタム 2 以降**：これらのデバイスは、ネットワークを介してストラタム 1 サーバー（またはそれ以上のストラタム サーバー）とクロックを同期します。ストラタム レベルが高くなと、デバイスは基準クロックから離れ、精度の低下や遅延増加の可能性が高まります。
- **ストラタム 15**：NTP 階層の最も低いレベル。参照クロックから最も距離が離れていることを示します。ストラタム 15 を超えるデバイスは、非同期と見なされます。

ストラタムシステムは、時間同期へのスケーラブルな分散アプローチを保証し、プライマリ時間ソースの過負荷を防止し、明確な階層を維持します。

NTP の関連 RFC

このトピックでは、NTP の最も重要な RFC の概要について説明し、その主な成果と推奨事項に焦点を当てます。

NTP の関連 RFC

RFC 1119：Network Time Protocol（バージョン 2）：この RFC は、NTP バージョン 2 について説明し、プロトコルの動作とアルゴリズムの詳細な仕様を示します。

主な貢献

- 時間同期、フィルタリング、および選択アルゴリズムを導入しました。
- エラー分析と高精度モデリングについて説明しました。
- ストラタムレベルの階層構造を定義しました。

RFC 5905：ネットワーク時間プロトコルバージョン 4：プロトコルおよびアルゴリズムの仕様：この RFC は、プロトコルの最新かつ最も広範に使用されているバージョンである NTP バージョン 4 を指定します。

主な貢献

- セキュリティ機能の強化。
- 同期の精度と拡張性が向上しました。
- IPv6 のサポート。
- 以前のバージョンとの後方互換性。

RFC 7384：パケット交換ネットワークにおける時間プロトコルのセキュリティ要件：この RFC では、NTP を含む時間同期プロトコルのセキュリティ要件について説明します。

主な貢献

- 時間プロトコルにおける潜在的な脆弱性を特定します。
- スプーフィングやパケット操作などの攻撃から NTP を保護するための推奨事項について説明します。

RFC 8633：Network Time Protocol Best Current Practices：NTP の展開と運用に関するガイダンスとベストプラクティスを提供します。

主な貢献

- NTP 展開を保護するための推奨事項。
- サーバー設定とクライアント動作に関するベストプラクティス。
- 攻撃を軽減し、堅牢な時刻同期を確保するためのガイダンス。

NTP のアーキテクチャとプロトコルの基礎

NTPアーキテクチャは、ネットワーク化されたデバイス間で時間を同期するサーバーとクライアントで構成されます。

NTP 設定セットアップは、ネットワークの一部としてサーバーとクライアントで構成されます。

NTP サーバー

- 他のデバイスに正確な時刻が提供されます。
- 同期の送信元に応じて、ストラタム 1、ストラタム 2、またはより高いレベルのサーバーとして動作できます。
- 上位ストラタムサーバーまたはローカル参照クロック（GPSなど）から時間を取得する場合があります。

NTP クライアント

- 内部クロックを同期するために、NTP サーバーから時間を要求します。
- ネットワーク遅延を考慮して、サーバーの時刻に基づいてクロックを調整します。
- 2 台のデバイスが相互に同期するピアモードでも動作します。

ピアリング関係

- デバイスをピアとして設定して、相互バックアップと冗長性を提供できます。
- いずれかのピアが上位階層サーバーへのアクセスを失った場合、もう一方のピアをフォールバック送信元として使用できます。

NTP の仕組み

時間同期プロセス：NTPは、クライアントのクロックとサーバーのクロックの違いを計算して最小限に抑えることによって、クロックを同期します。

- NTPは、アルゴリズムを使用してクライアントのクロックを調整し、オフセット、遅延、およびジッターをアカウンティングします。

process_summary

NTP には、時刻同期プロセスに複数の主要なアクターとコンポーネントが含まれます。

- クライアント：サーバーに時刻同期を要求し、受信データに基づいてクロックを調整します。
- サーバー：クライアントに正確な時間情報を提供し、ポーリング要求に応答します。
- NTPパケット：オフセット、遅延、およびジッターの測定値を含む同期データを転送します。

NTPは、オフセット、遅延、およびジッターを測定し、サーバーを動的にポーリングして正確な時間を維持します。

process_workflow

これらのステージでは、NTPがクライアントとサーバー間の時間を同期する方法について説明します。

1. **Clock Offset**：クライアントとサーバーのクロックの差を測定します。
   - オフセットは、クライアントとサーバーの両方のタイムスタンプを比較することによって計算されます。
2. **Delay**：パケットがクライアントとサーバーの間を移動するのにかかる時間を計算します。
   - 遅延は、NTP パケットのラウンドトリップ時間を測定することによって決定されます。
3. **Jitter**：ネットワークの変動による遅延の変動を推定します。
   - ジッターは、複数のNTP交換での遅延の変動を分析することによって計算されます。
4. NTPは、アルゴリズムを適用して、オフセット、遅延、およびジッターのアカウンティングを調整し、クライアントのクロックを調整します。
   - クライアントのクロックが、サーバーの時刻と可能な限り一致するように修正されます。

5. **Polling**：NTPは、同期を維持するため、設定可能な間隔でサーバーを定期的にポーリングします。
   - ポーリング間隔は、ネットワークの条件やクロックの安定性に基づいて動的に調整されます。

6. **NTP Packet Structure**：NTP パケットはコンパクトで効率的であり、オーバーヘッドが最小限に抑えられるように設計されています。
   - **Leap Indicator**（LI）：リープ秒の調整を示します。
   - **Version Number**（VN）：使用中の NTP バージョンを指定します。
   - **Mode**：パケットがクライアント、サーバー、またはピアから送信されたかどうかを決定します。
   - **Stratum**：サーバーのストラタムレベルを指定します。
   - **Transmit Timestamp**：サーバーによってパケットが送信された時刻が含まれます。
   - **Receive Timestamp**：パケットの受信時間を記録します。

NTPv4

高い精度と拡張性

精度

- インターネット経由でのミリ秒単位、または LAN 環境でのマイクロ秒単位の時間同期を実現します。
- 高度なアルゴリズムを実装して、精度を向上させ、ネットワーク遅延に適応します。

拡張性：

- 何千ものデバイスを含む大規模なネットワークをサポートします。
- 階層構造により、プライマリサーバーの負荷が最小限に抑えられます。

以前のバージョンとの後方互換性

- NTPv4 は、NTPv3 以前のバージョンと完全に互換性があります。
- 異なる NTP バージョンを使用するデバイス間のシームレスな通信を実現します。

NTP の高度な機能

孤立モード

場合によっては、NTPサブネットはローカル基準クロックやインターネットベースのクロックサーバーから切断され、隔離されて動作します。この分離期間中、サブネットサーバーとクライアントは共通の時間スケールに同期されます。従来、これはUTC送信元をシミュレートして共有時間参照を提供するローカル クロック ドライバを使用して実現されます。ドライバに直接または間接的に接続されたサーバーは、サブネット内の他のホストを同期します。ただし、ローカルクロックドライバに依存すると、サブネットに重大な回復不能な障害が発生する可能性があります。また、複数のサーバーを使用して冗長性を維持することは、常に実行可能とは限りません。これらの制限に対処するために、孤立モードはより堅牢なソリューションを提供し、ローカルクロックドライバの必要性を排除します。孤立モードでは、複数のサーバーが単一のUTC送信元をシミュレートできる一方で、サーバーが障害から回復する際のシームレスな切り替えが保証されます。

オーファンモードの仕組み

プライベートネットワークでは、通常、複数のコアサーバーがサブネット階層の最下位ストラタムで動作します。これらのサーバーは、対称モードまたはブロードキャストモードを使用する他のサーバーのバックアップとして設定されます。1つのコアサーバーがUTCソースに到達した場合でも、サブネット全体がそのサーバーに同期されます。どのサーバーもUTCソースにアクセスできない場合、孤立モードがアクティブになります。

孤立モード：

- 孤立した親と呼ばれる単一のサーバーが、サブネットのUTCソースをシミュレートするために選択されます。
- 孤立した子と呼ばれるサブネット内の他のすべてのホストは、この孤立した親と同期します。
- このメカニズムにより、外部時間ソースが存在しない場合でも、サブネットが一貫した時間スケールで動作し続けます。

孤立モードの設定

サーバーを孤立モード用に有効にするには、ntp orphan stratum コマンドを使用します。このコマンドでは、ストラタム値は次のとおりです。

- 16 未満であること
- 設定されたインターネット タイム サーバーのストラタム値よりも大きい値であること

他のサーバーまたは基準クロックのアソシエーションが設定されていない場合は、孤立ストラタム値を1に設定する必要があります。孤立した子に依存するすべてのサブネットホストのストラタム値が 16 未満であることを確認して、適切な階層を維持することが重要です。

孤立した親の動作

- 外部ソースのないストラタム 1 で動作している孤立した親には、参照 ID LOOP が表示されます。
- ストラタム 1 で動作していない孤立した親は、UNIX ループバックアドレス 127.0.0.1 を表示します。

孤立した親の選択

通常の NTP クライアントは遅延と分散メトリックに基づいてサーバーを選択しますが、孤立した子はサブネット内の各コアサーバーの IP アドレスに基づいて一意のメトリックを使用します。孤立した親は、メトリックが最も小さいサーバーとして選択されるため、サブネットに一貫したルートサーバーが確保されます。

冗長性および継続的な同期

サーバーがすべての時刻源を失った場合でも、サーバーはローカルクロックをサブネット内の他のサーバーと同期し続けます。この方法により、サーバーのバックアップを維持し、全体の時間スケールに合わせて調整できます。

次の図は、ピアネットワークにおける一般的な孤立モードの設定を示しています。

- 2 つのプライマリまたはセカンダリ（ストラタム 2）サーバーは、基準クロックまたはパブリック インターネット プライマリ サーバーを使用して設定されます。
- これらのサーバーは対称モードを使用して相互にバックアップし、サブネット内のシームレスな同期を確保します。

信頼できるキー設定の範囲

NTP（ネットワーク時間プロトコル）では、NTP クライアントとサーバー間の時間同期交換がセキュアで信頼できるものになるように、信頼できるキーが認証目的で使用されます。信頼できるキーは、不正なデバイスや悪意のある攻撃者による時間同期データのスプーフィングを防ぐという重要な役割を果たします。信頼できるキー設定の範囲とは、認証用の NTP 設定で指定できるキー ID を指します。キー ID は通常整数であり、その有効な範囲は使用されている特定の NTP 実装によって異なります。

信頼できるキーのキー ID 範囲

標準範囲：

- キー ID は、1 ～ 65535 の範囲の整数です。
- この範囲は、参照 NTP 実装である ntpd を含むほとんどの NTP 実装と一致します。

コンフィギュレーション コンテキスト：

- 信頼できるキーは、trustedkey ディレクティブを使用して NTP 構成ファイルで指定します。
- 信頼できるキーとして 1 つ以上のキー ID をリストできます。

IPv6 での NTP サポート

ネットワーク時間プロトコル（NTP）は、IPv4 と IPv6 の両方をサポートしており、いずれかのプロトコルを使用したネットワーク内の時間同期が可能です。RFC 5905 で規定されている NTPv4 以降、NTP では IPv6 の完全なサポートが含まれます。このことは、IPv4 の制約事項に対処するためにより多くの組織が IPv6 を導入する中で、特に重要です。

IPv6 をサポートする NTP の主な機能

1. デュアルスタックのサポート：
   - NTPv4 は、IPv4、IPv6、またはその両方を使用するネットワーク（デュアルスタックネットワーク）で機能します。
   - NTP サーバーとクライアントは、IPv4 と IPv6 の両方で同時に動作できます。
2. グローバルアドレス指定可能：IPv6 では、デバイスに固有のグローバルアドレスがあるため、大規模な分散環境での NTP の設定が容易になります。
3. ネットワーク効率の向上：IPv6 マルチキャストおよびエニーキャスト機能を使用して、ネットワーク全体に時刻同期を効率的に配信できます。
4. セキュリティ：IPv6 は、NTP トラフィックも保護できる IPsec などの機能により、全体的なネットワークのセキュリティを向上させます。
   -

IPv6 で NTP を使用する利点

1. 拡張性：IPv6 では、アドレスの数が事実上無制限であるため、大規模な展開に最適です。
2. 効率的なマルチキャスト：IPv6 マルチキャストにより、効率的な時間配信が可能になり、NTPサーバーの負荷を軽減できます。
3. ネットワーク効率の向上：IPv6 マルチキャストおよびエニーキャスト機能を使用して、ネットワーク全体に時刻同期を効率的に配信できます。
4. 将来に向けた信頼性：IPv6 の導入が増加しており、NTP の IPv6 サポートにより、将来のネットワークとの互換性が確保されます。

NTP IPv6 の課題

1. IPv6 の導入：すべてのネットワークが完全に IPv6 対応になっているわけではなく、一部のデバイスは IPv4 に依存する場合もあります。

2. DNS 設定：NTP で使用されるホスト名に DNS レコード（AAAA レコードなど）が正しく設定されていることを確認します。

NTP 設定に関するガイドライン

Network Time Protocol（NTP）パッケージには、認証されていないリモート攻撃者がサービス妨害（DoS）状態を発生させる可能性がある脆弱性が含まれています。NTP バージョン 4.2.4p7 以前は脆弱です。

この脆弱性は、特定の不正メッセージの処理におけるエラーによるものです。認証されていないリモート攻撃者は、スプーフィングされた送信元 IP アドレスを使用して、悪意ある NTP パケットを脆弱なホストに送信する可能性があります。このパケットを処理するホストは、送信者に応答パケットを返信します。この処理により、2 つのホスト間でメッセージのループが開始される可能性があります。その結果、両方のホストは、過剰な CPU リソースを消費し、ログファイルへのメッセージの書き込みにディスク スペースを使い切り、ネットワーク帯域幅を消費します。これにより、影響を受けたホスト上で DoS 状態が発生する可能性があります。

NTPv4 をサポートしている Cisco ソフトウェア リリースは影響を受けません。この問題は、その他すべての Cisco ソフトウェア バージョンに影響を及ぼします。

デバイスが NTP を使用するように設定されているかどうかを表示するには、show running-config | include ntp コマンドを使用します。出力に次のいずれかのコマンドが返された場合、そのデバイスは DoS 攻撃に対して脆弱です。

- ntp broadcast client
- ntp primary
- ntp multicast client
- ntp peer
- ntp server

デバイスで NTP を無効にする以外にこの脆弱性に対する回避策はありません。この脆弱性を悪用できるのは、デバイス上の設定済み IP アドレスに宛てられたパケットだけです。中継トラフィックは、この脆弱性を悪用しません。

リリースによっては NTP モード 7 パケットが処理され、NTP のデバッグが有効になっている場合は「NTP: Receive: dropping message: Received NTP private mode 7 packet」というメッセージが表示されることがあります。NTP モード 7 パケットを処理するには、ntp allow mode private コマンドを設定します。このコマンドは、デフォルトで無効になっています。

（注） NTP ピア認証は回避策ではなく、脆弱な設定です。

NTP サービスは、デフォルトではすべてのインターフェイスで無効になっています。

NTP を実行しているネットワーキング デバイスは、時刻を基準時刻源と同期する際にさまざまなアソシエーション モードで動作するように設定できます。ネットワーキング デバイスは、2 つの方法でネットワーク上の時刻情報を取得できます。それらは、ホスト サービスのポーリングと NTP ブロードキャストのリスニングです。

Line Aux 0 オプションはデフォルトで無効になっています。Cisco IOS XE で同じ NTP サーバーの IP アドレスと FQDN の両方を設定すると、FQDN が同じ IP アドレスに解決された後、FQDN 設定のみが コマンド出力に表示されます。

ポーリング ベースの NTP アソシエーションの設定

ポーリングベースの NTP アソシエーションを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. [**no**] **ntp peer** *ip-address* [**version** *number*] [**key** *key-id*] [**source** *interface*] [**prefer**]
4. [**no**] **ntp server** [**vrf** *vrf-name*] *ip-address* [**version** *number*] [**key** *key-id*] [**source** *interface*] [**prefer**]
5. **end**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	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	**configure terminal** 例： `Device# configure terminal`	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	[**no**] **ntp peer** *ip-address* [**version** *number*] [**key** *key-id*] [**source** *interface*] [**prefer**] 例： `Device(config)# ntp peer 172.16.22.44 version 2`	ピアを同期化するか、またはピアによって同期化されるように、デバイスのシステムクロックを設定します（ピアアソシエーション）。 • *ip-address*：クロック同期を提供する、またはクロック同期を提供されるピアの IP アドレス。

	コマンドまたはアクション	目的
		• *number*：NTP バージョン番号。指定できる範囲は 1 ～ 4 です。デフォルトでは、バージョン 4 が選択されています。 • *key-id*：**ntp authentication-key** コマンドで定義された認証キー。 • *interface*：IP の送信元アドレスを取得するインターフェイス。デフォルトでは、送信元 IP アドレスは発信インターフェイスから取得します。 • **prefer**：このピアを、同期を提供する優先ピアにします。このキーワードにより、ピア間の切り替えが減少します。 ピアアソシエーションを解除するには、このコマンドの **no** 形式を使用します。
ステップ 4	[**no**] **ntp server** [**vrf** *vrf-name*] *ip-address* [**version** *number*] [**key** *key-id*] [**source** *interface*] [**prefer**] 例： `Device(config)# ntp server 172.16.22.44 version 2`	タイムサーバーによって同期化されるように、デバイスのシステムクロックを設定します（サーバーアソシエーション）。 • *vrf-name*：クロック同期を提供するサーバーの仮想ルーティングおよび転送（VRF）アドレス。 （注） このコマンドを設定する前に、VRF を設定する必要があります。 • *ip-address*：クロック同期を提供するタイムサーバーの IP アドレス。 • *number*：NTP バージョン番号。指定できる範囲は 1 ～ 4 です。デフォルトでは、バージョン 4 が選択されています。 • *key-id*：**ntp authentication-key** コマンドで定義された認証キー。 • *interface*：IP の送信元アドレスを取得するインターフェイス。デフォルトでは、送信元 IP アドレスは発信インターフェイスから取得します。 • **prefer**：このピアを、同期を提供する優先ピアにします。このキーワードは、ピア間のクロックホップを減らします。

	コマンドまたはアクション	目的
		サーバーアソシエーションを解除するには、このコマンドの **no** 形式を入力します。
ステップ 5	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

ブロードキャストベースの NTP アソシエーションの設定

ブロードキャストベースの NTP アソシエーションを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. [**no**] **ntp broadcast** [**version** *number*] [**key** *key-id*] [*destination-address*]
5. [**no**] **ntp broadcast client**
6. **exit**
7. [**no**] **ntp broadcastdelay** *microseconds*
8. **end**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	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	**configure terminal** 例： `Device# configure terminal`	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	**interface** *interface-id* 例：	インターフェイスを設定し、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。

	コマンドまたはアクション	目的
	`Device(config)# interface gigabitethernet1/0/1`	
ステップ 4	[**no**] **ntp broadcast** [**version** *number*] [**key** *key-id*] [*destination-address*] 例： `Device(config-if)# ntp broadcast version 2`	NTP ブロードキャスト パケットをピアに送信するインターフェイスをイネーブルにします。 • *number*：NTP バージョン番号。指定できる範囲は 1 ～ 4 です。デフォルトでは、バージョン 4 が使用されます。 • *key-id*：認証キー。 • *destination-address*：このスイッチに対してクロックを同期しているピアの IP アドレス。 インターフェイスでの NTP ブロードキャスト パケットの送信を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 5	[**no**] **ntp broadcast client** 例： `Device(config-if)# ntp broadcast client`	インターフェイスが NTP ブロードキャスト パケットを受信できるようにします。 インターフェイスでの NTP ブロードキャスト パケットの受信を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 6	**exit** 例： `Device(config-if)# exit`	特権 EXEC モードに戻ります。
ステップ 7	[**no**] **ntp broadcastdelay** *microseconds* 例： `Device(config)# ntp broadcastdelay 100`	（任意）デバイスと NTP ブロードキャスト サーバー間のラウンドトリップ遅延の予測値を変更します。 デフォルトは 3000 マイクロ秒です。範囲は 1 ～ 999999 です。 インターフェイスでの NTP ブロードキャスト パケットの受信を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 8	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

NTP 認証の設定

NTP 認証を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. [**no**] **ntp authenticate**
4. [**no**] **ntp authentication-key** *number* {**md5** | **cmac-aes-128** | **hmac-sha1** | **hmac-sha2-256**} *value*
5. [**no**] **ntp trusted-key** *key-number*
6. [**no**] **ntp server** *ip-address* **key** *key-id* [**prefer**]
7. **end**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	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	**configure terminal** 例： `Device# configure terminal`	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	[**no**] **ntp authenticate** 例： `Device(config)# ntp authenticate`	NTP 認証をイネーブルにします。 NTP 認証を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 4	[**no**] **ntp authentication-key** *number* {**md5** | **cmac-aes-128** | **hmac-sha1** | **hmac-sha2-256**} *value* 例： `Device(config)# ntp authentication-key 42 md5 aNiceKey`	認証キーを定義します。 • キーごとに、キー番号、タイプ、および値を 1 つずつ指定します。 • キーは次のいずれかのタイプになります。 • **md5**：MD5 アルゴリズムを使用した認証。

	コマンドまたはアクション	目的
		• **cmac-aes-128**：AES-128 アルゴリズムによる暗号ベースメッセージ承認コード（CMAC）を使用した認証。ダイジェストの長さは 128 ビットで、キーの長さは 16 バイトまたは 32 バイトです。 • **hmac-sha1**：SHA1ハッシュ関数を使用したハッシュベースメッセージ承認コード（HMAC）を使用した認証。ダイジェストの長さは 128 ビットで、キーの長さは 1 ー 32 バイトです。 • **hmac-sha2-256**：SHA2 ハッシュ関数を使用した HMAC を使用した認証。ダイジェストの長さは 256 ビットで、キーの長さは 1 ー 32 バイトです。 SNTP の認証キーを削除する場合は、このコマンドの **no** 形式を使用します。
ステップ 5	[**no**] **ntp trusted-key** *key-number* **例**： `Device(config)# ntp trusted-key 42`	このデバイスと同期できるようにするために、ピア NTP デバイスが NTP パケットで提供する必要がある信頼できる認証キーを定義します。 信頼できる認証を無効にするには、このコマンドの **no** 形式を使用します。
ステップ 6	[**no**] **ntp server** *ip-address* **key** *key-id* [**prefer**] **例**： `Device(config)# ntp server 172.16.22.44 key 42`	NTP タイム サーバーによってソフトウェア クロックが同期されるように設定します。 • *ip-address*：クロック同期を提供するタイムサーバーの IP アドレス。 • *key-id*：**ntp authentication-key** コマンドで定義された認証キー。 • **prefer**：このピアを、同期を提供する優先ピアにします。このキーワードは、ピア間のクロックホップを減らします。 サーバーアソシエーションを解除するには、このコマンドの **no** 形式を入力します。
ステップ 7	**end** **例**：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	`Device(config)# end`	

外部基準クロックの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **line aux***line-number*
4. **end**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**line aux***line-number* 例： `Device(config)# line aux 0`	補助ポート 0 のライン 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	**end** 例： `Device(config)# end`	回線コンフィギュレーション モードを終了します。続いて、特権 EXEC モードに戻ります。

孤立モードの設定

始める前に

孤立モードを設定するには、少なくとも 2 つのクライアントが必要です。次のタスクは、1 つのクライアントで孤立モードを設定する方法を示しています。他のクライアントで手順を繰り返します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ntp server** *ip-address*
4. **ntp peer** *ip-address*
5. **ntp orphan**ストラタム

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	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	**ntp server** *ip-address* 例： `Device(config)# ntp server 10.1.1.1`	他のシステムとのサーバーアソシエーションを形成します。
ステップ 4	**ntp peer** *ip-address* 例： `Device(config)# ntp peer 172.16.0.1`	他のシステムとのピアアソシエーションを形成します。 （注） 他のクライアントでピアを設定するときに、設定したばかりの IP アドレスとは異なる IP アドレス（172.16.0.2 など）を使用します。

	コマンドまたはアクション	目的
ステップ 5	**ntp orphan**ストラタム 例： `Device(config)# ntp orphan 4`	ホストで孤立モードを有効にします。

NTP のトラブルシューティング

適切な時刻同期は、ネットワーク運用の完全性を維持するのに重要です。ただし、不良構成、ネットワークの問題、またはサーバーの問題が原因で、NTPに関する問題が発生する可能性があります。以下に、NTPをトラブルシュートするための構造化されたアプローチを示します。これには、一般的な問題、解決策、および問題の特定と解決に役立つツールが含まれます。

- NTPサービスを開始する前に、システムクロックが正しい時刻に合理的に近づいていることを確認してください（必要に応じて、日付を使用して手動で確認および設定できます）。
- 複数のアップストリームサーバーを構成して冗長性を提供し、精度を向上させます。
- NTPパフォーマンスとログを定期的にモニターして、潜在的な問題を早期に特定します。

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。

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。

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。