# ネットワーク プロキシミティに使用するルーティング プロトコルの設定

NPS では、データセンターを選択するプロセスで IS-IS、OSPF、および BGP の各ルーティング プロトコルを使用してネットワーク プロキシミティを計算します。 このモジュールの説明に従って、これらのプロトコルが目的のルータで稼働している必要があります。


- ネットワーク ルーティング プロキシミティに関する情報，1 ページ
- ルーティング プロトコルの設定方法，3 ページ

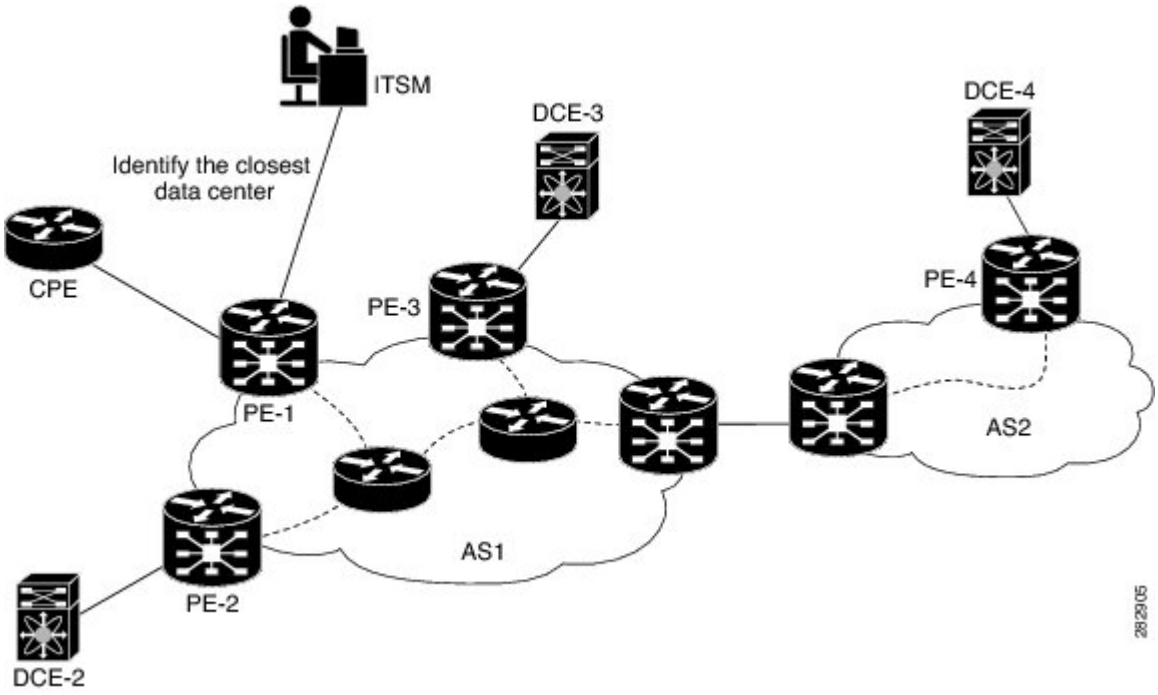

## ネットワークルーティングプロキシミティに関する情報

Cisco NPS のプロキシミティ エンジン（PXE）では、ネットワーク ルーティング プロキシミティを使用し、クライアントからデータセンターまでのトポロジ距離とパス距離に基づいてデータセンターを選択します。 PXE は IGP（IS-IS、OSPF）と EGP（BGP）の両方からトポロジとパス情報を収集します。 次に、固定ソース（プロキシミティ ソースアドレス（PSA））からのトポロジ距離の順に、リスト（プロキシミティ ターゲット リスト（PTL））の中で宛先をランク付けします。 サービス解決エンジン（SRE）から、クライアント アドレス（PSA）とデータセンター候補リスト（PTL）が PXE に送信されます。 PXE は、そのリストをネットワーク プロキシミティによってランク付けして SRE に返します。 PXE は、PSA とプロキシミティ ターゲット アドレス（PTA）とのトポロジ距離を常時計算しています。PTA は、PTL に存在する単一の要素です。

次の図に、基本的なプロキシミティ機能を示します。 2 つの自律システム（AS）を持つこのネットワークには、それぞれ DCE-2、DCE-3、および DCE-4 でホストされている 3 つのデータセンターがあります。 PTL にはこれらの DCE が記述されます。 PE-1 の背後でホストされている CPE がサービス解決に要求を送信します。 サービス解決は、AS1 にあるどのプロバイダーエッジルータ（PE）上でも動作できます。 PE-1 は PSA を形成します。 PXE は IGP プロキシミティ アルゴ

リズムを実行し、DCE-2 を「最も近いデータセンター」として選択したうえで、ランキング リスト DCE-2、DCE-3、DCE-4 を返します。

**図 *1*：ネットワーク プロキシミティによるデータセンターの選択**




# PXE のデータ要素

PXE の動作は、以下のデータ要素によって決まります。

- PSA：プロキシミティ ソース アドレス。 対象とするひと揃いのデータセンターの場所を得るためにプロキシミティの計算を要求しているエンドユーザまたはクライアントの送信元IP アドレスです。
- PTA：プロキシミティ ターゲット アドレス。 所定のデータ センタの場所を示す IP アドレスです。 所定の PSA と PAT のペアについてプロキシミティが計算されます。
- PTL：プロキシミティ ターゲット リスト。 PTA の集合です（ランク付けがある場合とない場合があります）。

PSA または PTA の従来の値は IP アドレスとマスクの組み合わせです。 Cisco NPS では、PXE で IP アドレスを扱うことを想定しているので、IP アドレスではない形式の識別情報はすべて、PXE の外部で IP アドレスとマスクの組み合わせに変換する必要があります。

# PXE のピアリングおよびランキング

PXE は、ネットワーク上の他のルータとパッシブにピアリングします。つまり、PXE はルートのみを学習します。 PXE から何らかのルートがネットワークに追加されることはありません。 PXE は、適切な IGP/EGP コントロール プレーンの動作に全面的に参加します。 ただし、PXE は学習したルートをプロキシミティの計算にのみ使用し、メイン ルータ コントロール プレーンの Routing Information Base（RIB）には干渉しません。 この目的で、PXE は RIB のコピーを別途保持しています。

### IGP のプロキシミティ

Cisco NPS は、IGP プロトコルとして OSPF と IS-IS をサポートしています。 そのアルゴリズムは、逆方向の Shortest Path First（SPF）の計算に依存しているので、たとえばリンク コストは PTA から PSA の方向に評価されます。

### EGP のプロキシミティ

BGP は、事実上の EGP 標準であり、BGP のプロキシミティの基本アルゴリズムは AS PATH 属性に依存しています。 この計算は、IGP の場合に非常によく似ていますが、リンク コストの代わりにパス コストを使用する点が異なります。 Cisco NPS では、AS 間のトポロジでプロキシミティを計算できません。

### プロキシミティ ソース アドレスのルート オリジン

PXE は、正しいプロキシミティ アルゴリズムを適用するために、PSA の学習で使用したものと同じルーティング プロトコルを選択します。 たとえば、PXE が OSPF を使用して PSA を学習している場合、プロキシミティの計算は IGP のプロキシミティに依存し、BGP から学習した PTA は自動的に低位にランクされます。 同じ OSPF エリアにある PTA が別の AS では PTA よりも優先するので、この手法は良好に機能します。 IGP のプロキシミティおよび BGP のプロキシミティは最も頻繁に適用されます。

# ルーティング プロトコルの設定方法

## プロキシミティの計算に向けた OSPF の設定

次のタスクで、プロキシミティ エンジンで実行するプロキシミティの計算に使用する Open Shortest Path First（OSPF）ルーティング プロセスを設定します。

### はじめる前に

OSPF プロキシミティの計算機能を使用する場合は、使用しているルータの integrated-service インターフェイスに対し、インターフェイス コンフィギュレーション モードで **ip ospf priority** コマン

ドを設定することを推奨します。 このコマンドは、指定ルータ（DR）/バックアップ DR の選択を判断する上で効果的です。

```
Router# config
Router(config)# interface interface-service 0
Router(config-if)# ip ospf priority 1
```

## 手順の概要

1. **router ospf** *process-id*
2. **network** *ip-address wildcard-mask* **area** *area-id*
3. **area** *area-id* {**stub** | **nssa**}
4. **area** *area-id* **authentication** {**message-digest** | **cleartext**}
5. **log-adjacency-changes**
6. **router-id** *ip-address*
7. **interface eth0 priority** *priority*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **router ospf** *process-id*<br><br>例：<br>`switch(config)# router ospf 123` | OSPF ルーティング プロセスを設定し、ルーティング コンフィギュレーション モードを開始します。<br>• *process-id*：OSPF ルーティング プロセスにローカルで割り当てて内部使用する識別情報。 任意の正の整数が使用できます。 |
| ステップ 2 | **network** *ip-address wildcard-mask* **area** *area-id*<br><br>例：<br>`switch(config-router)# network 26.0.0.0 255.0.0.0 area 1` | OSPF を実行するインターフェイスと、そのインターフェイスのエリア識別情報を定義します。<br>• *ip-address*：OSPF に関連付けるエリアの IP アドレス。<br>• *wildcard-mask*：IP アドレスの範囲を定義するために IP アドレスに適用するワイルドカード マスク。<br>• *area-id*：OSPF アドレス範囲に関連付けるエリア。 |
| ステップ 3 | **area** *area-id* {**stub** | **nssa**}<br><br>例：<br>`switch(config-router)# area 1 stub` | スタブ エリアまたは Not-So-Stubby Area として OSPF エリアを設定します。 |
| ステップ 4 | **area** *area-id* **authentication** {**message-digest** | **cleartext**}<br><br>例：<br>`switch(config-router)# area 0 authentication message-digest` | OSPF エリアの認証をイネーブルにします。<br>• *area-id*：認証をイネーブルにするエリアの識別情報。 10 進数値で指定する必要があります。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • **message-digest**：指定のエリアに対して Message Digest 5（MD5）認証をイネーブルにします。<br>• **cleartext**：指定のエリアに対してクリア テキスト認証をイネーブルにします。 |
| ステップ 5 | **log-adjacency-changes**<br>例：<br>`switch(config-router)# log-adjacency-changes` | OSPF ネイバーが起動または停止したときに、ルータが syslog メッセージを送信するように設定します。 |
| ステップ 6 | **router-id** *ip-address*<br>例：<br>`switch(config-router)# router-id 26.0.0.2` | 固定ルータ ID を使用することを指定します。<br>• *ip-address*：IP アドレス形式で記述したルータ ID。 |
| ステップ 7 | **interface eth0 priority** *priority*<br>例：<br>`switch(config-router)# interface eth0 priority 2` | OSPF のプライオリティを指定します。 0 ～ 255 の範囲で値を指定できます。 デフォルトは 1 です。 |

# プロキシミティの計算に向けた BGP の設定

次のタスクで、プロキシミティ エンジンでプロキシミティを計算できるようにボーダー ゲートウェイ プロトコル（BGP）のルーティング プロセスを設定します。

## 手順の概要

1. **router bgp** *as-no*
2. **location-community** *community-string* **weight** *weight*
3. **log-neighbor-changes**
4. **neighbor** *ip-address* **timers** *keepalives holdtime*
5. **neighbor** *ip-address* **ebgp-multihop**
6. **neighbor** *ip-address* **remote-as** *as-no*
7. **neighbor** *ip-address* **password** *string*
8. **ip urib bgp bestpath**

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **router bgp** *as-no*<br><br>例：<br>`switch(config)# router bgp 3` | BGP ルーティング プロセスを設定し、ルーティング コンフィギュレーション モードを開始します。<br>• *as-nc*：他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする、自律システムの番号。 |
| ステップ 2 | **location-community** *community-string* **weight** *weight*<br><br>例：<br>`switch(config-router)# location-community 11:222 weight 100` | プロキシミティ エンジンに関連付けたコミュニティ値を設定します。<br>• *community-string*：プロキシミティ エンジンに関連付けた文字列。<br>• *weight*：コミュニティに関連付けた重み。 |
| ステップ 3 | **log-neighbor-changes**<br><br>例：<br>`switch(config-router)# log-neighbor-changes` | BGP ネイバー リセットのロギングをイネーブルにします。 |
| ステップ 4 | **neighbor** *ip-address* **timers** *keepalives holdtime*<br><br>例：<br>`switch(config-router)# neighbor 26.0.0.1 timers 30 100` | 特定の BGP ピアまたは BGP ピア グループのタイマーを設定します。<br>• *ip-address*：BGP ピアまたは BGP ピア グループの IP アドレス。<br>• *keepalives*：BGP プロセスからそのピアにキープアライブ メッセージを送信する時間間隔（秒）。 デフォルトは 60 です。 有効な範囲は 0 ～ 65535 です。<br>• *holdtime*：キープアライブ メッセージを受信できない状態が継続して、ピアがデッドであるとプロセスで宣言するまでの時間（秒）。 デフォルト値は 180 です。 有効な範囲は 3 ～ 65535 です。 この保持時間は、キープアライブ メッセージの時間間隔の 2 倍より長くする必要があります。 |
| ステップ 5 | **neighbor** *ip-address* **ebgp-multihop**<br><br>例：<br>`switch(config-router)# neighbor 26.0.0.1 ebgp-multihop` | 直接接続されていないネットワークに存在する外部ピアとの BGP 接続を受け入れ、またその接続を試行するように BGP ルーティング プロセスを設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 6 | **neighbor** *ip-address* **remote-as** *as-no*<br><br>例：<br>`switch(config-router)# neighbor 26.0.0.1 remote-as 1` | BGP ネイバーテーブルまたはマルチプロトコル BGP ネイバーテーブルにエントリを追加します。 |
| ステップ 7 | **neighbor** *ip-address* **password** *string*<br><br>例：<br>`switch(config-router)# neighbor 26.0.0.1 password 123` | 2 つの BGP ピア間の TCP 接続上で Message Digest 5（MD5）認証をイネーブルにします。<br>• *ip-address*：BGP スピーキング ネイバーの IP アドレス。<br>• *string*：大文字と小文字が区別される、最大 25 文字のパスワード。最初の文字を数値にはできません。この文字列には、スペースも含め、あらゆる英数字を使用できます。数字-スペース-任意の文字の形式でパスワードを指定することはできません。数字の後にスペースを使用すると、認証に失敗する原因となることがあります。 |
| ステップ 8 | **ip urib bgp bestpath**<br><br>例：<br>`switch(config)# ip urib bgp bestpath` | BGP 自律システム（AS）のパスの長さに基づくプロキシミティを使用するプロキシミティ アルゴリズムをイネーブルにします。 |

# プロキシミティの計算に向けた IS-IS の設定

次のタスクで、プロキシミティ エンジンで実行するプロキシミティの計算に使用する Intermediate System-to-Intermediate System（IS-IS）ルーティング プロセスを設定します。

## 手順の概要

1. **router isis** *process-name*
2. **net** *network-entity-title*
3. **lsp-mtu** *max-lsp-size*
4. **log-adjacency-changes**
5. **is-type** {**level-1** | **level-2** | **level -1-2**}
6. **authentication-check** {**level-1** | **level-2**}
7. **authentication-type** {**md5** | **text**} {**level-1** | **level-2**}
8. **authentication key-chain** *name-of-chain* {**level-1** | **level-2**}
9. **interface eth0 priority** *priority*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **router isis** *process-name*<br><br>例：<br>`switch(config)# router isis 123` | IS-IS ルーティング プロセスを設定し、ルーティング コンフィギュレーション モードを開始します。<br>• *process-name*：ルーティング プロセスを表す名前。 この名前は、指定ルータでのすべての IP、またはコネクションレス型ネットワーク サービス（CLNS）ルータ プロセス内で一意でなければなりません。 |
| ステップ 2 | **net** *network-entity-title*<br><br>例：<br>`switch(config-router)# net 26.0.0.0` | CLNS ルーティング プロセスの IS-IS Network Entity Title（NET）を設定します。<br>• *network-entity-title*：CLNS ルーティング プロセスのエリア アドレスおよびシステム ID。 この引数には、IP アドレスまたは名前を指定できます。 |
| ステップ 3 | **lsp-mtu** *max-lsp-size*<br><br>例：<br>`switch(config-router)# lsp-mtu 1000` | IS-IS リンク ステート パケット（LSP）の最大伝送単位（MTU）サイズをバイトの単位で設定します。 指定できる値は 0 ～ 2147483647 です。 |
| ステップ 4 | **log-adjacency-changes**<br><br>例：<br>`switch(config-router)# log-adjacency-changes` | IS-IS ネイバーが起動または停止したときに syslog メッセージを送信するようにルータを設定します。 |
| ステップ 5 | **is-type** {**level-1** \| **level-2** \| **level -1-2**}<br><br>例：<br>`(config-router)# is-type level-1-2` | IS-IS ルーティング プロセスのインスタンスのルーティング レベルを設定します。<br>• **level-1**：レベル 1（エリア内）ルーティングのみの実行を指定します。 このルータが学習するのはそのエリア内の宛先だけです。 レベル 2（エリア間）ルーティングは、最も近いレベル 1 ～ 2 ルータによって実行されます。<br>• **level-2**：レベル 1 とレベル 2 の両方のルーティングを実行します。 このルータは、ルーティング プロセスのインスタンスを 2 つ実行します。 このルータは、エリア内（レベル 1 ルーティング）の宛先について 1 つのリンクステート パケット データベース（LSDB）を持っており、Shortest Path First（SPF）の計算を実行してエリア トポロジを検出します。 また、他のすべてのバックボーン（レベル 2）ルータの LSP による別の LSDB も備え、別の SPF 計算を実行してバックボーンのトポロジと他のすべてのエリアの存在を検出します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • **level-1-2**：ルーティング プロセスはレベル 2（エリア間）ルータとしてのみ機能します。 このルータはバックボーンの一部であり、レベル 1 とは通信せずに、自身のエリアに存在するルータとのみ通信します。 |
| ステップ 6 | **authentication-check** {**level-1** \| **level-2**}<br><br>例：<br>`(config-router)# authentication-check level1` | 該当のレベルで受信パケットのチェックをイネーブルにします。<br>• **level-1**：レベル 1 の LSP、CSNP、および PSNP の認証タイプ。<br>• **level-2**：レベル 2 の LSP、CSNP、および PSNP の認証タイプ。 |
| ステップ 7 | **authentication-type** {**md5** \| **text**} {**level-1** \| **level-2**}<br><br>例：<br>`(config-router)# authentication-type md5 level-2` | IS-IS で使用する認証のタイプを指定します。<br>• **md5**：Message Digest 5 認証。<br>• **text**：クリア テキスト認証。<br>• **level-1**：レベル 1 パケットでのみ、指定の認証をイネーブルにします。<br>• **level-2**：レベル 2 パケットでのみ、指定の認証をイネーブルにします。 |
| ステップ 8 | **authentication key-chain** *name-of-chain* {**level-1** \| **level-2**}<br><br>例：<br>`(config-router)# authentication key-chain abc level-2` | IS-IS に対して認証をイネーブルにします。<br>• *name-of-chain*：有効なキーのグループを特定します。<br>• **level-1**：レベル 1 パケットでのみ認証をイネーブルにします。<br>• **level-2**：レベル 2 パケットでのみ認証をイネーブルにします。 |
| ステップ 9 | **interface eth0 priority** *priority*<br><br>例：<br>`switch(config-router)# interface eth0 priority 2` | IS-IS プライオリティを指定します。 0 ～ 255 の範囲で値を指定できます。 デフォルトは 1 です。 |