高度な BGP の設定

この章は、次の項で構成されています。

拡張 BGP について

BGP は、組織または自律システム間のループフリー ルーティングを実現する、インタードメイン ルーティング プロトコルです。Cisco NX-OS は BGP バージョン 4 をサポートしています。BGP v4 に組み込まれているマルチプロトコル拡張機能を使用すると、IP マルチキャスト ルートおよび複数のレイヤ 3 プロトコル アドレス ファミリに関するルーティング情報を BGP に伝送させることができます。BGP では、他の BGP 対応デバイス（BGP ピア）との間で TCP セッションを確立するために、信頼できるトランスポート プロトコルとして TCP を使用します。外部組織に接続するときには、ルータが外部 BGP（eBGP）ピアリング セッションを作成します。同じ組織内の BGP ピアは、内部 BGP（iBGP）ピアリング セッションを通じて、ルーティング情報を交換します。

ピア テンプレート

BGP ピア テンプレートを使用すると、類似した BGP ピア間で再利用できる共通のコンフィギュレーションブロックを作成できます。各ブロックでは、ピアに継承させる一連の属性を定義できます。継承した属性の一部を上書きすることもできるので、非常に柔軟性のある方法で、繰り返しの多い BGP の設定を簡素化できます。

Cisco NX-OS は、3 種類のピア テンプレートを実装します。

- peer-session テンプレートでは、トランスポートの詳細、ピアのリモート自律システム番号、セッション タイマーなど、BGP セッション属性を定義します。peer-session テンプレートは、別の peer-session テンプレートから属性を継承することもできます（ローカル定義の属性によって、継承した peer-session 属性は上書きされます）。

- peer-policy テンプレートでは、着信ポリシー、発信ポリシー、フィルタ リスト、プレフィックス リストを含め、アドレス ファミリに依存する、ピアのポリシー要素を定義します。peer-policy テンプレートは、一連の peer-policy テンプレートからの継承が可能です。Cisco NX-OS は、継承設定のプリファレンス値で指定された順序で、これらの peer-policy テンプレートを評価します。最小値が大きい値よりも優先されます。

- peer テンプレートは、peer-session および peer-policy テンプレートからの継承が可能であり、ピアの定義を簡素化できます。peer テンプレートの使用は必須ではありませんが、

peer テンプレートによって再利用可能なコンフィギュレーション ブロックが得られるので、BGP の設定を簡素化できます。

認証

BGP ネイバー セッションに認証を設定できます。この認証方式によって、ネイバーに送られる各 TCP セグメントに MD5 認証ダイジェストが追加され、不正なメッセージや TCP セキュリティ アタックから BGP が保護されます。

（注） MD5 パスワードは、BGP ピア間で一致させる必要があります。

ルート ポリシーおよび BGP セッションのリセット

BGP ピアにルート ポリシーを関連付けることができます。ルート ポリシーではルート マップを使用して、BGP が認識するルートを制御または変更します。着信または発信ルート アップデートに関するルート ポリシーを設定できます。ルート ポリシーはプレフィックス、AS_path 属性など、さまざまな条件で一致が必要であり、ルートを選択して受け付けるかまたは拒否します。ルート ポリシーでパス属性を変更することもできます。

BGP ピアに適用するルート ポリシーを変更する場合は、そのピアの BGP セッションをリセットする必要があります。Cisco NX-OS は、BGP セッションをリセットするため、次の 3 つのメカニズムをサポートしています。

- ハード リセット：ハード リセットでは、指定されたピアリング セッションが TCP 接続を含めて切断され、指定のピアからのルートが削除されます。このオプションを使用すると、BGP ネットワーク上のパケット フローが中断します。ハード リセットは、デフォルトでディセーブルです。
- ソフト再構成着信：ソフト再構成着信によって、セッションをリセットすることなく、指定されたピアのルーティング アップデートが開始されます。このオプションを使用できるのは、着信ルート ポリシーを変更する場合です。ソフト再構成着信の場合、ピアから受け取ったすべてのルートのコピーを保存したあとで、着信ルート ポリシーを介してルートが処理されます。着信ルート ポリシーを変更する場合、Cisco NX-OS は変更された着信ルート ポリシーを介して保存ルートを渡し、既存のピアリング セッションを切断することなく、ルート テーブルをアップデートします。ソフト再構成着信の場合、まだフィルタリングされていない BGP ルートの保存に、大量のメモリ リソースを使用する可能性があります。ソフト再構成着信は、デフォルトでディセーブルです。
- ルート リフレッシュ：ルート リフレッシュでは、着信ルート ポリシーの変更時に、サポートするピアにルート リフレッシュ要求を送信することによって、着信ルーティング テーブルがダイナミックにアップデートされます。リモート BGP ピアは新しいルート コピーで応答し、ローカル BGP スピーカが変更されたルート ポリシーでそれを処理します。Cisco NX-OS は自動的に、プレフィックスのアウトバウンド ルートの更新をピアに送信します。

- BGP ピアは、BGP ピア セッションの確立時に、BGP 機能ネゴシエーションの一部として、ルート リフレッシュ機能をアドバタイズします。ルート リフレッシュは優先オプションであり、デフォルトでイネーブルです。

（注） BGP はさらに、ルート再配布、ルート集約、ルート ダンプニングなどの機能にルート マップを使用します。ルート マップの詳細については、Route Policy Manager の設定を参照してください。

eBGP

eBGP を使用すると、異なる AS からの BGP ピアを接続し、ルーティング アップデートを交換できます。外部ネットワークへの接続によって、自分のネットワークから他のネットワークへ、またインターネットを介して、トラフィックを転送できます。

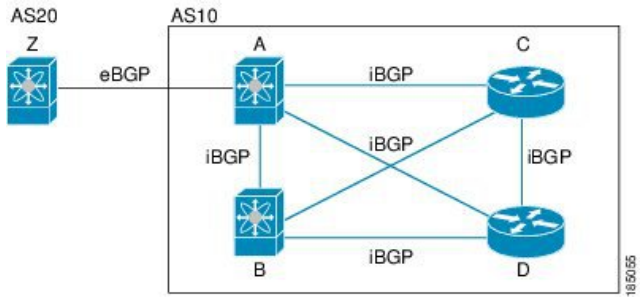
通常、eBGP ピアリングは、インターフェイスがダウンしたときにコンバージェンスが高速になるように、直接接続されたインターフェイス上で行う必要があります。

iBGP

iBGP を使用すると、同じ自律システム内の BGP ピアを接続できます。iBGP はマルチホーム BGP ネットワーク（同じ外部自律システムに対して複数の接続があるネットワーク）に使用できます。

図に、大きい BGP ネットワークの中の iBGP ネットワークを示します。

図 *1 : iBGP* ネットワーク



iBGP ネットワークはフルメッシュです。各 iBGP ピアは、ネットワーク ループを防止するために、他のすべての iBGP ピアに対して直接接続されています。

ネイバー コンフィギュレーション モードで update-source が設定された単一ホップ iBGP ピアでは、ピアは高速外部フェールオーバーをサポートします。

iBGP ピアリング セッションの確立には、ループバック インターフェイスを使用します。ループバック インターフェイスは、インターフェイス フラップが発生する可能性が小さいからです。インターフェイス フラップが発生するのは、障害またはメンテナンスが原因で、インターフェイスが管理上アップまたはダウンになったときです。マルチホップ、高速外部フォール

オーバー、AS パス属性のサイズ制限については、eBGP の設定 （48 ページ） セクションを参照してください。

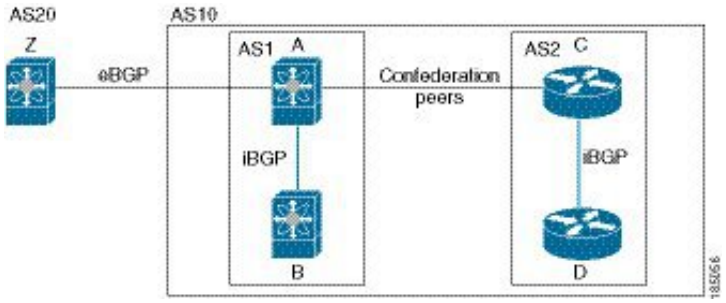
（注） iBGP ネットワークでは別個のインテリア ゲートウェイ プロトコルを設定する必要があります。

AS 連合

フルメッシュの iBGP ネットワークは、iBGP ピア数が増えるにしたがって複雑になります。自律システムを複数のサブ自律システムに分割し、それを 1 つの連合としてまとめることによって、iBGP メッシュを緩和できます。連合は、同じ自律システム番号を使用して外部ネットワークと通信する、iBGP ピアからなるグループです。各サブ AS はその中ではフルメッシュであり、同じ連合内の他のサブ AS に対する少数の接続があります。

図に BGP ネットワークが 2 つのサブ AS と 1 つの連合に分けられて表示されます。

図 *2 : AS* 連合



この例では、AS10 が 2 つの AS（AS1 および AS2）に分割されています。各サブ AS はフルメッシュですが、サブ AS 間のリンクは 1 つだけです。AS 連合を使用することによって、 のフルメッシュ AS に比べて、リンク数を少なくできます。

ルート リフレクタ

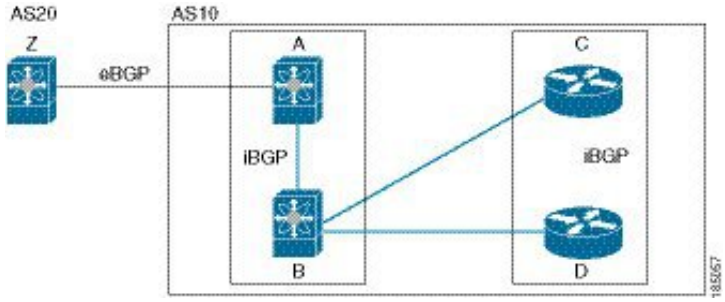
すべての iBGP ピアが完全に一致する必要がないように、ルート リフレクタが学習したルートをネイバーに渡すルート リフレクタ構成を使用することによって、iBGP メッシュを削減できます。

ある iBGP ピアをルート リフレクタとして設定すると、そのピアが iBGP で学習したルートを一連の iBGP ネイバーに渡す役割を担います。

図に、メッシュの iBGP スピーカを 4 つ（ルータ A、B、C、D）使用する、単純な iBGP 構成を示します。ルート リフレクタを使用しなかった場合、外部ネイバーからルートを受け取ったルータ A は、3 つの iBGP ネイバーのすべてにルートをアドバタイズします。

図では、ルータ B がルート リフレクタです。ルート リフレクタは、ルータ A からアドバタイズされたルートを受信すると、ルータ C と D へのルートをアドバタイズ（リフレクト）します。ルータ A は、ルータ C と D の両方にアドバタイズする必要がなくなります。

図 *3:* ルート リフレクタ



ルート リフレクタおよびそのクライアント ピアは、クラスタを形成します。ルート リフレクタのクライアント ピアとして動作するように、すべての iBGP ピアを設定する必要はありません。ただし、完全な BGP アップデートがすべてのピアに届くように、非クライアント ピアはフルメッシュとして設定する必要があります。

機能ネゴシエーション

BGP スピーカは機能ネゴシエーション機能を使用することによって、ピアでサポートされている BGP 拡張機能を学習できます。機能ネゴシエーションによって、リンクの両側の BGP ピアがサポートする機能セットだけを BGP に使用させることができます。

BGP ピアが機能ネゴシエーションをサポートしない場合で、なおかつアドレス ファミリが IPv4 として設定されている場合、Cisco NX-OS は機能ネゴシエーションを行わずに、ピアとの新規セッションを試みます。他のマルチプロトコル設定（IPv6 など）の場合は、機能ネゴシエーションが不可欠です。

ルート ダンプニング

ルート ダンプニングは、インターネットワーク上でのフラッピング ルートの伝搬を最小限に抑える BGP 機能です。ルート フラップが発生するのは、使用可能ステートと使用不能ステートが短時間で次々切り替わる場合です。

AS1、AS2、および AS3 という 3 つの BGP 自律システムからなるネットワークの場合について考えてみます。AS1 のルートがフラップした（使用不能になった）とします。ルート ダンプニングを使用しない場合、AS1 は AS2 に回収メッセージを送信します。AS2 は AS3 にその回収メッセージを伝達します。フラッピング ルートが再び発生すると、AS1 から AS2 にアドバタイズメント メッセージを送信し、AS2 は AS3 にそのアドバタイズメントを送信します。ルートの使用不能と使用可能が繰り返されると、AS1 は多数の回収メッセージおよびアドバタイズメント メッセージを送信することになり、それが他の自律システムに伝播します。

ルートダンプニングによって、フラッピングを最小限に抑えることができます。ルートフラップが発生したとします。（ルート ダンプニングがイネーブルの）AS2 がルートにペナルティとして 1000 を割り当てます。AS2 は引き続き、ネイバーにルートの状態をアドバタイズします。ルート フラップが発生するたびに、AS2 がペナルティ値を追加します。ルート フラップが頻繁に発生して、ペナルティが設定可能な抑制限度を超えると、AS2 はフラップ回数に関係

なく、ルートのアドバタイズを中止します。その結果、ルートが減衰（ダンプニング）します。

ルートに与えられたペナルティは、再使用限度に達するまで減衰します。その時点で、AS2 は再びルートをアドバタイズします。再使用限度が 50% になると、AS2 はそのルートのダンプニング情報を削除します。

（注） ルート ダンプニングがイネーブルの場合は、ピアのリセットによってルートが回収されても、リセット中の BGP にはペナルティは適用されません。

ロード シェアリングおよびマルチパス

BGP はルーティング テーブルに、同じ宛先プレフィックスに到達する複数の等コスト eBGP または iBGP パスを組み込むことができます。その場合、宛先プレフィックスへのトラフィックは、組み込まれたすべてのパス間で共有されます。

BGP ベストパス アルゴリズムでは、次の属性が同じ場合に、等コスト パスと見なされます。

- 重量
- ローカル プリファレンス
- AS_path
- オリジン コード
- Multi-Exit Discriminator（MED）
- BGP ネクスト ホップまでの IGP コスト

BGP はこれら複数のパスの中から、ベスト パスとして 1 つだけ選択し、そのパスを BGP ピアにアドバタイズします。詳細については、「BGP の追加パス」の項を参照してください。

（注） 異なる AS 連合から受け取ったパスは、外部 AS_path 値およびその他の属性が同じ場合に、等コスト パスと見なされます。

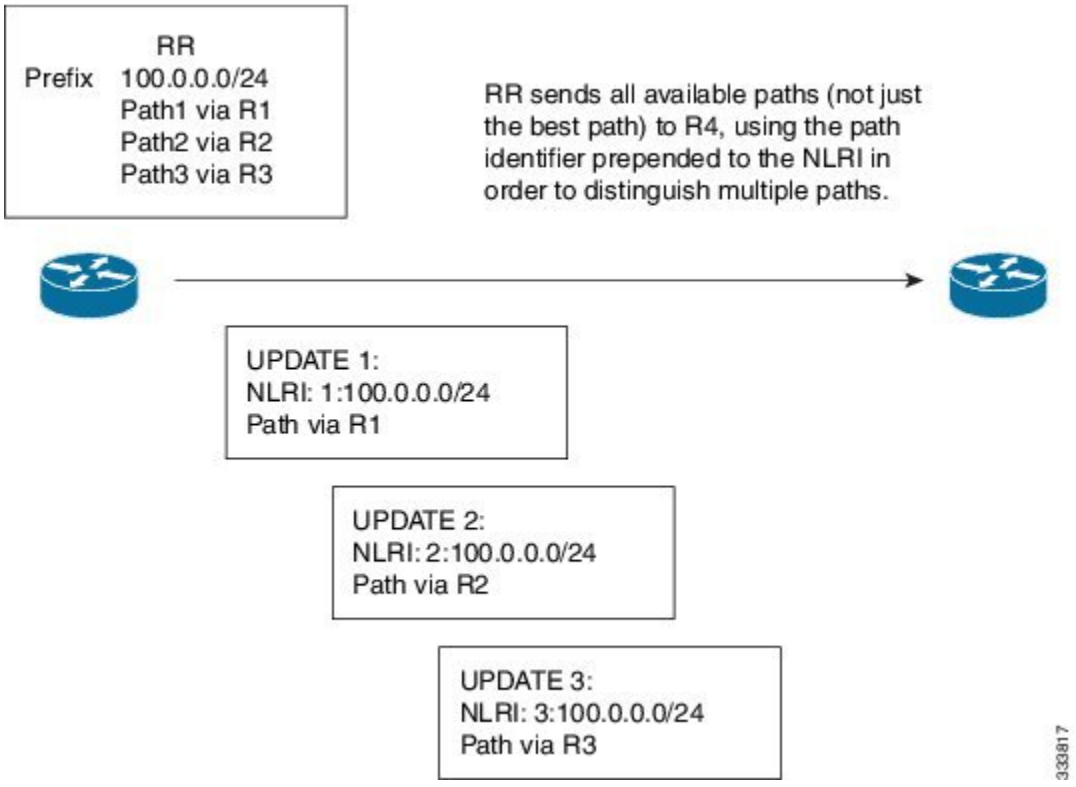
（注） iBGP マルチパスに関してルート リフレクタを設定すると、ルート リフレクタが、選択されたベスト パスをピアにアドバタイズします。そのパスのネクスト ホップは変更されません。

BGP の追加パス

1 つの BGP 最良パスだけがアドバタイズされ、BGP スピーカは特定ピアからの特定プレフィックスの 1 パスだけを受け入れます。BGP スピーカが同じセッション内で同じプレフィックスの複数のパスを受信した場合、最新のアドバタイズメントを使用します。

BGP は、以前のパスに代わる新しいパスなしで、BGP スピーカが同じプレフィックスに対して複数のパスを伝播し、受け入れることを可能にする追加のパス機能をサポートします。この機能は、BGP スピーカのピアが、プレフィックスごとの複数パスのアドバタイズおよび受信をサポートし、また、そのパスのアドバタイズをサポートするかどうかネゴシエートすることを可能にします。特別な 4 バイトのパス ID は、ピア セッションを介して送信される同じプレフィックスに対して複数のパスを区別するため、ネットワーク層到達可能性情報（NLRI）に追加されます。次の図に、追加の BGP パス機能を示します。

図 *4:* 追加パスの機能を持つ *BGP* ルート アドバタイズメント



BGP 追加パス設定の詳細については、BGP 追加パスの設定 （44 ページ） の項を参照してください。

ルート集約

集約アドレスを設定できます。ルート集約を使用すると、固有性の強い一連のアドレスをすべての固有アドレスを代表する 1 つのアドレスに置き換えることによって、ルート テーブルを簡

素化できます。たとえば、10.1.1.0/24、10.1.2.0/24、および 10.1.3.0/24 という固有性の強い 3 つのアドレスを 1 つの集約アドレス 10.1.0.0/16 に置き換えることができます。

アドバタイズされるルートが少なくなるように、BGP ルート テーブル内には集約プレフィックスが存在します。

（注） Cisco NX-OS は、自動ルート集約をサポートしません。

ルート集約はフォワーディング ループにつながる可能性があります。この問題を回避するために、集約アドレスのアドバタイズメントを生成するときに、BGP はローカル ルーティング テーブルに、その集約アドレスに対応するサマリー廃棄ルートを自動的に組み込みます。BGP はサマリー廃棄のアドミニストレーティブ ディスタンスを 220 に設定し、ルート タイプを廃棄に設定します。BGP はネクストホップ解決に廃棄ルートを使用しません。

ユーザが **aggregate-address** コマンドを発行すると、BGP テーブルにサマリー エントリが作成されますが、サマリーエントリは、集約のサブセットがテーブルで見つかるまでアドバタイズできません。

BGP 条件付きアドバタイズメント

BGP 条件付きアドバタイズメントを使用すると、プレフィックスが BGP テーブルに存在するかどうかに基づいてルートをアドバタイズまたは撤回するように BGP を設定できます。この機能は、たとえば、BGP でいずれかのプロバイダーにプレフィックスをアドバタイズするようなマルチホーム ネットワーク（他のプロバイダーからの情報が存在しない場合のみ）で便利です。

AS1、AS2、および AS3 という 3 つの BGP 自律システムからなるネットワークの例について考えてみます。この例で、AS1 と AS3 はインターネットと AS2 に接続しています。条件付きアドバタイズメントを使用しない場合、AS2 はすべてのルートを AS1 と AS3 の両方にプロパゲートします。条件付きアドバタイズメントを使用すれば、AS1 からのルートが存在しない場合のみ（たとえば AS1 へのリンクがダウンした場合）、特定のルートを AS3 にアドバタイズするように AS2 を設定できます。

BGP 条件付きアドバタイズメントでは、設定されたルート マップに一致する各ルートに、存在テストまたは非存在テストが追加されます。「BGP 条件付きアドバタイズメントの設定」を参照してください。

BGP ネクスト ホップ アドレス トラッキング

BGP は、インストールされているルートのネクスト ホップ アドレスをモニタして、ネクストホップの到達可能性の確認、および BGP ベストパスの選択、インストール、検証を行います。BGP ネクストホップ アドレスのトラッキングを行うと、ネクストホップの到達可能性に影響を及ぼす可能性のあるルート変更がルーティング情報ベース（RIB）で行われたときに確認プロセスをトリガーすることで、このようなネクストホップ到達可能性テストの速度が向上します。

ネクスト ホップ情報が変更されると、BGP は RIB から通知を受信します（イベント駆動型の通知）。BGP は、次のいずれかのイベントが発生したときに通知を受けます。

- ネクスト ホップが到達不能になった。
- ネクスト ホップが到達可能になった。
- ネクスト ホップへの完全再帰のインテリア ゲートウェイ プロトコル（IGP）メトリックが変更された。
- ファースト ホップの IP アドレスまたはファースト ホップのインターフェイスが変更された。
- ネクスト ホップが接続された。
- ネクスト ホップが接続解除された。
- ネクスト ホップがローカル アドレスになった。
- ネクスト ホップが非ローカル アドレスになった。

（注） 到達可能性および再帰メトリック イベントは、最適パスの再計算をトリガーします。

RIB からのイベント通知は、クリティカルおよび非クリティカルとして分類されます。クリティカルおよび非クリティカルイベントの通知は、別々のバッチで送信されます。ただし、非クリティカルイベントが保留中であり、クリティカルイベントを読み込む要求がある場合は、非クリティカル イベントがクリティカル イベントとともに送信されます。

- クリティカルなイベントとは、異なるパスに対してスイッチオーバーの原因となるネクスト ホップの消失など、ネクスト ホップの到達可能性に関連しています。異なるパスに対してスイッチオーバーの原因となるネクストホップの IGP メトリックの変更は、クリティカルなイベントと見なすことができます。
- 非クリティカルなイベントとは、最適パスに影響を与えたり、単一のネクスト ホップに IGP メトリックを変更したりせずに追加されるネクスト ホップに関連しています。

詳細については、「BGP ネクスト ホップ アドレス トラッキングの設定」を参照してください。

ルートの再配布

スタティック ルートまたは他のプロトコルからのルートを再配布するように、BGP を設定できます。再配布を指定したルート マップを設定して、どのルートが BGP に渡されるかを制御する必要があります。ルートマップを使用すると、宛先、送信元プロトコル、ルート タイプ、ルート タグなどの属性に基づいて、ルートをフィルタリングできます。詳細については、Route Policy Manager の設定を参照してください。

ルート マップを使用して両シナリオのデフォルト動作を無効にできますが、ルート マップの正しくない使用によってネットワーク ループが発生することがあるため、そうする場合は注意が必要です。次に、デフォルトの動作の変更にルート マップを使用する例を示します。

ルート マップの変更によって、シナリオ 1 のデフォルトの動作を次のように変更できます。

```
route-map foo permit 10
   match route-type internal
router ospf 1
   redistribute bgp 100 route-map foo
```

同様に、ルートマップの変更によって、シナリオ2のデフォルトの動作を次のように変更できます。

```
route-map foo deny 10
  match route-type internal
router ospf 1
   vrf bar
     redistribute bgp 100 route-map foo
```

ラベル付きユニキャスト ルートとラベルなしユニキャスト ルート

リリース7.0(3)I7(6) では、SAFI-1（ラベルなしユニキャスト）およびSAFI-4（ラベル付きユニキャスト ルーティング）が単一セッションの IPv4 BGP でサポートされるようになりました。詳細については、『*Cisco Nexus 9000 Series NX-OS Label Switching Configuration Guide*、*Release 7.x*』を参照してください。

BFD

この機能では、IPv4およびIPv6用の双方向フォワーディング検出（BFD）をサポートします。BFD は、転送パスの障害を高速で検出することを目的にした検出プロトコルです。BFD は 2 台の隣接デバイス間のサブセカンド障害を検出し、BFD の負荷の一部を、サポートされるモジュール上のデータ プレーンに分散できるため、プロトコル hello メッセージよりも CPU を使いません。

BGP の BFD は eBGP ピアおよび iBGP シングルホップ ピアでサポートされます。BFD を使用している iBGP シングル ホップ ピアのネイバー コンフィギュレーション モードで **update-source** オプションを設定します。

Cisco NX-OS リリース 9.3(3) 以降では、BGP の BFD は BGP IPv4 と IPv6 のプレフィックス ピアでもサポートされます。このサポートにより、BGP はマルチホップ BFD を使用できるようになり、BGP コンバージェンス時間が改善されます。プレフィックス ピアでは、シングルホップ BGP とマルチホップ BGP の両方がサポートされます。

Cisco NX-OS リリース 9.3(3) 以降、BFD は IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカルを介した BGP インターフェイス ピアリングをサポートします。ただし、BFD マルチホップはアンナンバード BGP ではサポートされません。

詳細については、『Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide』を参照してください。

BGP の調整

BGP タイマーによって、さらにベストパス アルゴリズムの調整によって、BGP のデフォルト動作を変更できます。

BGP タイマー

BGP では、ネイバー セッションおよびグローバル プロトコル イベントにさまざまなタイプのタイマーを使用します。確立されたセッションごとに、最低限2つのタイマーがあります。定期的にキープアライブ メッセージを送信するためのタイマー、さらに想定時間内にピアのキープアライブが届かなかった場合に、セッションをタイムアウトさせるためのタイマーです。また、個々の機能を処理するための、その他のタイマーがあります。これらのタイマーは通常、秒単位で設定します。タイマーには、異なる BGP ピアで同じタイマーが異なるタイミングでスタートするように、ランダム アジャストメントが組み込まれています。

ベストパス アルゴリズムの調整

オプションの設定パラメータによって、ベストパスアルゴリズムのデフォルト動作を変更できます。たとえば、アルゴリズムでの Multi-Exit Discriminator（MED）属性およびルータ ID の扱い方を変更できます。

マルチプロトコル BGP

Cisco NX-OS の BGP は、複数のアドレス ファミリをサポートします。マルチプロトコル BGP（MP-BGP）は、アドレス ファミリに応じて異なるルート セットを伝送します。BGP ではたとえば、IPv4 ユニキャスト ルーティング用のルート セットを 1 つ、IPv4 マルチキャスト ルーティング用のルート セットを 1 つ、さらに IPv6 マルチキャスト ルーティング用のルート セットを 1 つ伝送できます。IP マルチキャスト ネットワークではリバース パス フォワーディング（RPF）のチェックに MP-BGP を使用できます。

（注） マルチキャスト BGP ではマルチキャスト状態情報をプロパゲートしないため、プロトコル独立マルチキャスト（PIM）などのマルチキャスト プロトコルが必要です。

マルチプロトコル BGP 設定をサポートするには、ルータ アドレスファミリおよびネイバー アドレス ファミリの各コンフィギュレーション モードを使用します。MP-BGP では、設定されたアドレス ファミリごとに別々の RIB が維持されます（ユニキャスト RIB と、BGP のマルチキャスト RIB など）。

マルチプロトコル BGP ネットワークは下位互換性がありますが、マルチプロトコル拡張機能をサポートしない BGP ピアは、アドレス ファミリ ID 情報など、マルチプロトコル拡張機能が伝送するルーティング情報を転送できません。

RFC 5549

BGP は RFC 5549 をサポートしており、IPv4 プレフィックスを IPv6 ネクスト ホップで伝送できます。BGP はすべてのホップで実行されるため、すべてのルータが IPv4 および IPv6 トラフィックを転送できます。したがって、ルータ間で IPv6 トンネルをサポートする必要はありません。BGP は、IPv6 ルートを介した IPv4 を Unicast Route Information Base（URIB）にインストールします。

Cisco NX-OS リリース9.2(2) 以降では、-R タイプのライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチは、RFC 5549 をサポートします。

現在、NX-OS は IPv4 ルートの IPv6 再帰ネクストホップ（RNH）をサポートしていません。

RFC 6368

はじめに

このセクションでは、Cisco NX-OS のプロバイダー エッジ（PE）機能とカスタマー エッジ（CE）機能間で内部ボーダー ゲートウェイ プロトコル（iBGP）がどのように実装されているかについて説明します。

現在の展開で、プロバイダー/カスタマーエッジのルーティングプロトコルとしてBGPを使用すると、VPN プロバイダー自律システム（AS）とカスタマー ネットワーク自律システム間の外部ピアリングとしてピアリング セッションが設定されます。

RFC 6368では、これらのピアがiBGPピアとして設定されるようになりました。

Cisco NX-OSリリース10.1（2）以降では、EVPN-VxLANv4およびEVPN-VxLANv6のRFC 6368サポートが有効になっています。

フレームワーク

Cisco NX-OSリリース10.1（2）以降では、iBGP PE-CE機能を導入しています。

- as-override を使用した外部 Border Gateway Protocol（eBGP）を展開せずに、VRF の複数のサイトで単一の自律システム番号（ASN）を持つことができます。
- プロバイダー コアがまるで 1 つの透過ルート リフレクタ（RR）のように機能する、CE ルータへの内部ルート リフレクションを提供したいと考えます。

この機能を使用VRF サイトは、プロバイダー コアと同じ ASN を持つことができます。ただし、VRF サイトの ASN が プロバイダー コアの ASN と異なっている場合は、この機能のローカル自律システム（AS）を使用して、同じであるように表示できます。

iBGP PE-CE の実装

この機能を動作させるのは、次の 2 つの主要部分です。

- プロバイダー コアでVPN BGP 属性を透過的に伝送するために、新しい属性である `ATTR_SET` が BGP プロトコルに追加されました。
- PE ルータを、VRF 内の CE ルータへの iBGP セッションの RR にします。

新しい ATTR_SET 属性ではプロバイダーがカスタマーの BGP 属性すべてを透過的に伝送でき、プロバイダー属性や BGP ポリシーに干渉することがありません。こうした属性にはクラスタリスト、ローカル設定などがあります。

BGP カスタマー ルート属性

ATTR_SET は、プロバイダー カスタマーの VPN BGP 属性を伝送するために使用される、新しい BGP 属性です。これは過渡的なオプション属性です。この属性では、Local Preference、Med、Origin、AS Path、Originator ID、Cluster list属性がプロバイダーネットワーク全体で伝送されます。ATTR_SET 属性の形式は次のとおりです。

```
+-----------------------------+
| Attr Flags□O | T□Code = 128  |
+-----------------------------+
| Attr. Length (1 or 2 octets) |
+-----------------------------+
| Origin AS (4 octets)        |
+-----------------------------+
|Path Attributes (variable)    |
+-----------------------------+
```

- 属性フラグは、通常の BGP 属性フラグです。
- 属性の長さは、この属性の長さが 1 オクテットであるか 2 オクテットであるかを示します。
- Origin AS フィールドある AS で発生するルートが、適切な AS_PATH 操作を行われずに、別の AS にリークされないようにします。
- 可変長-のパス属性フィールドには、プロバイダー コアで伝送されなければならない VPN BGP 属性が含まれます。

iBGP PE-CE の実装の詳細については、「iBGP PE-CE 機能の IOS 実装」を参照してください。

次に、iBGPカスタマーエッジデバイスのPEデバイスでのBGPネイバー設定の例を示します。

```
router bgp 200
vrf nxbgp3-leaf2-2
address-family ipv4 unicast
redistribute static route-map ALLOW-ALL
address-family ipv6 unicast
redistribute static route-map ALLOW-ALL
neighbor 101.101.101.101 remote-as 200
description ibgp sample config
internal-vpn-client (1)
address-family ipv4 unicast
route-reflector-client (2)
next-hop-self (3)
```

BGP モニタリング プロトコル

BGP モニタリング プロトコル（BMP）は、BGP アップデートとピア統計情報をモニタし、すべての Cisco Nexus 9000 シリーズ スイッチでサポートされます。

このプロトコルを使用して、BGP スピーカーは外部 BMP サーバに接続し、BGP イベントに関する情報を送信します。1つの BGP スピーカーに最大 2 つの BMP サーバを設定でき、各 BGP ピアは BMP サーバのすべてまたはサブセットによるモニタリング用に設定できます。BGP スピーカーは、BMP サーバからの情報を受け入れません。

グレースフル リスタートおよびハイ アベイラビリティ

Cisco NX-OS は、BGP に対してノンストップ フォワーディングとグレースフル リスタートをサポートしています。

BGP ルーティング プロトコル情報がフェールオーバー後に復元されている間に、転送情報ベース（FIB）内の既知のルートでデータ パケットを転送するように、BGP の無停止フォワーディング（NSF）を使用できます。NSF では、BGP ピアはルーティング フラップと無縁です。フェールオーバー時に、データ トラフィックはインテリジェント モジュール経由で転送され、スタンバイ スーパーバイザがアクティブになります。

Cisco NX-OS ルータでコールド リブートが発生した場合、ネットワークはルータへのトラフィック転送を中止し、ネットワーク トポロジからルータを削除します。この状況では、BGP は非グレースフル リスタートになり、すべてのルートが削除されます。Cisco NX-OS がスタートアップ コンフィギュレーションを適用すると、BGP はピアリング セッションを再び確立して、ルートを再学習します。

Cisco NX-OS デュアル スーパーバイザ構成のルータでは、ステートフル スーパーバイザ スイッチオーバーが実行されます。スイッチオーバーの間、BGP は無停止フォワーディングを使用し、FIB の情報に基づいてトラフィックを転送します。システムがネットワーク トポロジから取り除かれることはありません。ネイバーが再起動しているルータは、「ヘルパー」と呼ばれます。スイッチオーバー後、グレースフル リスタート動作が開始されます。この処理が進行中の際、2 つのルータはネイバー関係を再確立し、これらの BGP ルートを交換します。それらネイバー関係が再起動したとしても、ヘルパーは再起動中のピアを指すプレフィックスを転送し続け、再起動中のルータはピアへトラフィックを転送し続けます。再起動中のルータがグレースフル リスタート可能なすべての BGP ピアを持つ場合、グレースフル リスタートが完了し、BGP は再び動作可能なネイバーを通知します。

グレースフル リスタート動作中であることがルータで検出されると、両方のルータがそれぞれのトポロジ テーブルを交換します。すべての BGP ピアからルート アップデートを受信したルータは、古いルートをすべて削除し、アップデートされたルートでベストパス アルゴリズムを実行します。

スイッチオーバーが完了すると、Cisco NX-OS は実行コンフィギュレーションを適用し、BGP は自身が再度使用可能になったことをネイバーに通知します。

ネイバー コンフィギュレーション モードで update-source が設定された単一ホップ iBGP ピアでは、ピアは高速外部フェールオーバーをサポートします。

Cisco NX-OS リリース 9.3(3) 以降、BGP プレフィックス ピアはグレースフル リスタートをサポートします。

追加 BGP パス機能により、特定のプレフィックスにアドバタイズされるパス数が再起動の前後で同じ場合、パス ID の選択は古いパスの最終状態および削除を保証します。いくつかのパスが指定されたプレフィックスにアドバタイズされる場合、古いパスがグレースフル リスタート ヘルパー ピアに発生する可能性があります。

メモリ不足の処理

BGP は、次の条件でメモリ不足に対処します。

- マイナー アラート：BGP は新しい eBGP ピアを確立しません。BGP は新しい iBGP ピアおよび連合ピアの確立は続行します。ピアは存続しますが、リセット ピアは再確立されません。
- 重大アラート：BGP は、メモリ アラートがマイナーになるまで、選択した確立済み eBGP ピアを 2 分おきにシャット ダウンします。eBGP ピアごとに、受信したパスの合計数と最適パスとして選択されたパスの数の比率が計算されます。比率が最高のピアが、メモリ使用状況を削減するためのシャットダウン対象として選択されます。オシレーションを回避するために、シャットダウンされた eBGP ピアを復帰する前にその eBGP ピアをクリアする必要があります。

（注） 重要な eBGP ピアをこの選択プロセスから除外できます。

- クリティカル アラート：BGP は確立されたすべてのピアを正常にシャットダウンします。シャットダウンされた eBGP ピアを復帰する前にその eBGP ピアをクリアする必要があります。

メモリ不足状態によるシャットダウンから BGP ピアを除外する方法の詳細については、「BGP の調整 」を参照してください。

仮想化のサポート

1 個の BGP インスタンスを設定できます。BGP は、仮想ルーティングおよび転送（VRF）インスタンスをサポートします。

拡張 BGP の前提条件

拡張 BGP の前提条件は次のとおりです。

- BGP を有効にする必要があります（「BGP の有効化」の項を参照）。
- システムに有効なルータ ID を設定しておく必要があります。

- Regional Internet Registry（RIR）によって割り当てられたか、またはローカル管理の AS 番号を取得しておく必要があります。
- ネイバー関係を作成しようとするピアに到達可能でなければなりません（Interior Gateway Protocol（IGP）、スタティック ルート、直接接続など）。
- BGP セッションを確立するネイバー環境で、アドレス ファミリを明示的に設定する必要があります。

拡張 BGP に関する注意事項と制限事項

拡張 BGP 設定時の注意事項および制約事項は、次のとおりです。

- Cisco NX-OS リリース 9.3(5) 以降、コマンドの動作が変更された 3 つのシナリオがあります。

  - ```
    Router bgp 1
         Template peer abc
             Ttl-security hops 30
         Neighbor 1.2.3.4
             Inherit peer abc
    ```

    後で **ebgp-multihop 20** コマンドを入力すると、**ttl-security hops 30** コマンドが存在するため、設定はブロックされます。Cisco NX-OS リリース 9.3(5) 以降、設定はブロックされなくなりました。ただし、**ttl-security hops** コマンドが優先され、有効な機能になります。

  - ```
    Router bgp 1
         Template peer abc
             Ebgp-multihops 20
         Neighbor 1.2.3.4
             Inherit peer abc
    ```

    後で **ttl-security hops 30** コマンドを入力すると、**ebgp-multihop 20** コマンドが存在するため、設定はブロックされます。Cisco NX-OS リリース 9.3(5) 以降、設定はブロックされなくなりました。ただし、ここでも **ttl-security hops** コマンドが優先され、有効な機能なります。

  - ```
    Router bgp 1
         Template peer abc
             Remote-as 1
         Neighbor 1.2.3.4
             Inherit peer abc
    ```

    後で **ttl-security hops 30** または **ebgp-multihop 20** コマンドを入力すると、ブロックされます。Cisco NX-OS リリース 9.3(5) 以降、設定はブロックされなくなりました。ただし、ピアが iBGP ピアになる **remote-as** コマンドが優先されるため、これらの機能はオフになります。

- プレフィックス ピアリングは、パッシブ TCP モードでのみ動作します。ピア アドレスがプレフィックス内にある場合、リモート ピアからの着信接続を受け入れます。
    ```

- Cisco NX-OS 9.3(5) 以降、vPC ピアへの TTL 値が 1 のパケットは、転送されるハードウェアです。
- **advertise-maps** コマンドを複数回設定することはサポートされていません。
- プレフィックス リスト内の名前は、大文字と小文字が区別されません。一意の名前を使用することを推奨します。大文字と小文字を変更して同じ名前を使用しないでください。たとえば、CTCPrimaryNetworks と CtcPrimaryNetworks は 2 つの異なるエントリではありません。
- ダイナミック AS 番号プレフィックス ピア設定は、BGP テンプレートから継承した個々の AS 番号の設定よりも優先します。
- AS 連合でプレフィックス ピアにダイナミック AS 番号を設定した場合、BGP はローカル連合の AS 番号のみでセッションを確立します。
- ダイナミック AS 番号プレフィックス ピアで作成された BGP セッションは、設定済みの eBGP マルチホップ存続可能時間（TTL）値や直接接続ピアに対するディセーブル済みのチェックを無視します。
- ルータ ID の自動変更およびセッション フラップを避けるために、BGP 用のルータ ID を設定します。
- ピアごとに最大プレフィックス設定オプションを使用し、受信するルート数および使用するシステム リソース数を制限してください。
- update-source を設定し、eBGP マルチホップ セッションでセッションを確立します。
- 再配布を設定する場合は、BGP ルート マップを指定します。
- VRF 内で BGP ルータ ID を設定します。
- キープアライブおよびホールドタイマーの値を小さくすると、ネットワークでセッション フラップが発生する可能性があります。
- BGP を IGP に再配布するとき、iBGP も再配布されます。この動作を無効にするには、ルート マップに追加 deny 文を挿入します。
- iBGP の単一ホップ ピアに対して BFD を有効にするには、物理インターフェイスの **update-source** オプションを設定します。
- Cisco NX-OS リリース 9.3(3) 以降では、BGP の BFD は BGP IPv4 と IPv6 のプレフィックス ピアでサポートされます。
- VLAN には、次の注意事項および制約事項が **remove-private-as** コマンドに適用されます。
  - これは、eBGP ピアにだけ適用されます。
  - これは、パブリック AS のみのルータのみに適用されます。この制約事項を回避するには、ネイバー単位で **neighbor local-as** コマンドを適用し、ローカル AS 番号をパブリック AS 番号として指定することです。

- ネイバー コンフィギュレーション モードだけで設定可能となり、ネイバー アドレス ファミリ モードでは設定できません。
- AS パスにプライベートとパブリック AS 番号を含める場合、プライベート AS 番号は削除されません。
- AS パスに eBGP ネイバーの AS 番号が含まれている場合、プライベート AS 番号は削除されません。
- その AS パス内のすべての AS 番号がプライベート AS 番号範囲に属する場合のみ、プライベート AS 番号は削除されます。ピアの AS 番号または非プライベート AS 番号が AS パス セグメントに存在する場合、プライベート AS 番号は削除されません。

- **aggregate-address** を使用する場合 コマンドを使用して集約アドレスを設定し、**suppress-fib-pending** コマンドを使用して BGP ルートを抑制するコマンドを使用する場合、集約のロスレス トラフィックを BGP またはシステム トリガーで保証できません。
- スイッチでFIB抑制をイネーブルにし、ルートプログラミングがハードウェアで失敗すると、BGPはハードウェアでローカルにプログラミングされていないルートをアドバタイズします。
- ネイバー、テンプレート ピア、テンプレート ピアセッション、またはテンプレート ピア ポリシー コンフィギュレーション モードでコマンドを無効にした場合（**inherit peer** または **inherit peer-session** コマンドが存在する場合）、**default** キーワードを使用してコマンドをデフォルトの状態に戻す必要があります。たとえば、実行コンフィギュレーションから **default update-source loopback 0** コマンドを無効にするには、**update-source loopback 0** コマンドを入力する必要があります。
- route-reflector クライアントに next-hop-self が設定されている場合、ルート リフレクタは自身をネクスト ホップとしてクライアントにルートをアドバタイズします。
- 重み付き ECMP に次の注意事項および制約事項が適用されます。
  - 重み付き ECMP 機能は、IPv4 アドレス ファミリでのみサポートされます。
  - BGPは、draft-ietf-idr-link-bandwidth-06.txtで定義されているリンク帯域幅EXTCOMMを使用して、重み付けECMP機能を実装します。
  - BGP は、eBGP ピアと iBGP ピアの両方から受け入れることができます。
- IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカルを介した BGP インターフェイス ピアリングには、次の注意事項と制限事項が適用されます。
  - この機能は、複数のインターフェイス間で同じリンクローカルアドレスを設定することをサポートしていません。
  - この機能は、論理インターフェイス（ループバック）ではサポートされていません。イーサネット インターフェイス、ポートチャネル インターフェイス、サブインターフェイス、およびブレークアウト インターフェイスのみがサポートされます。
  - Cisco NX-OS リリース 9.3(6) 以降では、VLAN インターフェイスがサポートされます。

- この機能は、リンクローカルアドレスを持つIPv6対応インターフェイスでのみサポートされます。
- この機能は、設定されたプレフィックス ピアとインターフェイスのリモート ピアが同じ場合はサポートされません。
- 次のコマンドはネイバーインターフェイス コンフィギュレーション モードではサポートされていません。
  - **disable-connected-check**
  - **maximum-peers**
  - **update-source**
  - **ebgp-multihop**
- BFD マルチホップおよび次のコマンドは、IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカルを介したBGPインターフェイス ピアリングではサポートされません。
  - **bfd-multihop**
  - **bfd multihop interval**
  - **bfd multihop authentication**
- BGPでは、ルートアドバタイズメントのコンバージェンス時間が短縮されます。ルート アドバタイズメント（RA）リンクレベル プロトコルの検出を高速化するには、IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカル経由 BGP インターフェイス ピアリングを使用する各IPv6対応インターフェイスで次のコマンドを入力します。

```
interface Ethernet port/slot
ipv6 nd ra-interval 4 min 3
ipv6 nd ra-lifetime 10
```

- リンクローカルでBGPネイバーを設定する場合は、TCAM「in-sup」を512から768にカスタマイズする必要があります。
- **[maximum-paths eibgp]** コマンドは、MPLS 環境でのみサポートされています。
- ルート マップ削除機能は、BGP に関連付けられたルート マップ全体の削除をブロックするメカニズムを追加します。ルート マップの削除がブロックされても、ルート マップ ステートメントへの変更は引き続き許可されます。
- ルートマップに複数のシーケンスがある場合、少なくとも1つのシーケンスが使用可能になるまで、ユーザーはルート マップ シーケンスを削除できます。
- ユーザーは、クライアントからのルートマップの前方参照ケースを持つことができます。ただし、ルート マップが作成されて関連付けられると、ルート マップの削除はブロックされます。
- ブロック削除機能は、ノブを使用して動的に構成できます。

- ルート マップへの BGP アソシエーションを削除すること、および単一のトランザクション ペイロードでルート マップ自体を削除することは許可されています。
- ルート マップに BGP アソシエーションを追加することが許可されており、ルート マップの削除に対してエラーをスローする必要があります。
- 以下は、デュアル ステージに関連する動作のリストです。
  - ノブと削除が同時に発生した場合、デュアル ステージは検証し、コミットせずにエラーをスローする必要があります。
  - ノブはすでに存在し、ルートマップ削除がデュアルステージで発生する場合、エラーをスローする必要があります。
  - ルート マップと CLI ノブが異なる順序のシングル コミットである場合、エラーをスローする必要があります。
  - ノブが有効になっておらず、ルート マップの削除がデュアル ステージで発生した場合は、正常に実行する必要があります。
  - 1回のベリファイで、「cliノブが無効かつルートマップの削除」が実行された場合、ルートマップの削除が許可されます。
- BGP テンプレートで使用されるルート マップがいずれの BGP ネイバーにも継承されない場合、ルート マップ全体の削除は引き続きブロックされます。
- BGP によって所有されているが、bgpInst の一部ではない、vrf コンテキストの下にいくつかのコマンドがあります。
- Cloudscale IPv6 リンクローカル BGP のサポートには、512 を超える ing-sup TCAM リージョンを切り分ける必要があります (これを有効にするには、リロードが必要です)。
- VPN アドレス ファミリ (L3VPN および EVPN) がサポートされていないため、同盟ピアから受信したルートは VPN アドレス ファミリでアドバタイズされません。
- Cisco NX-OS リリース 10.3(1)F 以降、BGP は Cisco Nexus 9808 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、BGP は Cisco Nexus 9804 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.3(1)F 以降、VXLAN EVPN は、Cisco Nexus 9808 プラットフォーム スイッチで、トランジットとしてのみサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、VXLAN EVPN は、Cisco Nexus 9804 プラットフォーム スイッチで、トランジットとしてのみサポートされます。
- Cisco NX-OS リリース 10.3(3)F 以降、BGP パスワードのタイプ 6 暗号化は、次の制限付きで Cisco NX-OS スイッチでサポートされます。
  - タイプ 6 暗号化が構成されている場合、既存のタイプ 6 暗号化パスワードをタイプ 0/タイプ 3/タイプ 7 パスワードに変更することはできません。

- タイプ6暗号化がサポートされていない古いイメージでコールドリブートによってシステムをダウングレードする場合は、タイプ6構成を削除して、それからコールドリブートを実行してください。そうしないと、構成が失われ、ネイバーの構成がなくなります。
- プライマリ キーの設定は、スイッチに対してローカルです。あるスイッチからタイプ 6 に構成された実行データを取得し、別のプライマリ キーが構成されている別のスイッチに適用すると、新しいスイッチでの復号化は失敗します。
- ISSU 中に、古いイメージ（タイプ 0/タイプ 3/タイプ 7 暗号化キーが構成に存在する）から新しいイメージ（タイプ 6 暗号化がサポートされている）に移行する場合、BGP は既存**encryption re-encrypt obfuscated** のコマンドを使用して再暗号化が適用されるまで、または適用されない限り、タイプ 6 の暗号化に既存のキーを変換しません。
- BGP タイプ 6 パスワードは、非 DME プラットフォームではサポートされません。
- ネイバーまたはテンプレートのパスワードをプログラム（RESTCONF、NETCONF など）で構成する場合は、パスワードのタイプとパスワードを指定することを強くお勧めします。プログラム コールでいずれかのプロパティが欠落している場合、BGP は欠落しているプロパティのすでに使用可能な（またはデフォルトの）値を使用して、ネイバーまたはテンプレートのパスワードを構成します。

  ユーザーがプロパティを指定せずに構成する必要がある場合、ユーザーは両方のピアルータで同じ手順を実行する必要があります。

- Cisco NX-OS リリース 10.4(1)F 以降、BGP は、Cisco Nexus 9808 および 9804 スイッチを搭載した N9KX98900CD-A および N9KX9836DM-A ライン カードでサポートされます。

デフォルト設定

高度な BGP パラメータのデフォルト設定値を表に示します。

パラメータ	デフォルト
BGP 機能	ディセーブル
BGP の追加パス	ディセーブル
キープアライブインターバル	60 秒
ホールド タイマー	180 秒
ダイナミック機能	有効（Enabled）

高度な BGP の設定

インターフェイスでの IP 転送の有効化

RFC 5549 を使用するには、少なくとも 1 つの IPv4 アドレスを設定する必要があります。IPv4 アドレスを設定しない場合は、RFC 5549 を使用するように IP転送機能を有効にする必要があります。

手順の概要

1. **configure terminal**
2. **interface** *type slot/port*
3. **ip forward**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス設定モードを開始します。

ステップ 3	**ip forward** 例： `switch(config-if)# ip forward`	インターフェイスにIPアドレスが設定されていない場合でも、そのインターフェイスでIPv4トラフィックを許可します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	この設定変更を保存します。

BGP セッション テンプレートの設定

BGP セッション テンプレートを使用すると、類似した設定が必要な複数の BGP ピアで、BGP の設定を簡素化できます。BGPテンプレートによって、共通のコンフィギュレーションブロックを再利用できます。先に BGP テンプレートを設定し、BGP ピアにテンプレートを適用します。

BGP セッション テンプレートでは、継承、パスワード、タイマー、セキュリティなどのセッション属性を設定できます。

peer-session テンプレートは、別の peer-session テンプレートからの継承が可能です。第 3 のテンプレートから継承するように第2テンプレートを設定できます。さらに最初のテンプレートもこの第3のテンプレートから継承させることができます。この間接継承を続けることができる peer-session テンプレートの数は、最大 7 つです。

ネイバーに設定した属性は、ネイバーが BGP テンプレートから継承した属性よりも優先されます。

始める前に

BGP を有効にする必要があります（「BGP の有効化」の項を参照）。

（注）

- テンプレートを編集するときには、ピアまたはテンプレートのレベルで **no** 形式のコマンドを使用すると、テンプレートの設定を明示的に上書きできます。属性をデフォルトの状態にリセットするには、default 形式のコマンドを使用する必要があります。
- BGP ピア テンプレートを使用する場合、テンプレート内で使用されるコマンドをチェックして、そのコマンドが iBGP / eBGP ピアに適用されるかどうかを確認することはありません。たとえば、テンプレートを作成し、テンプレート内に「Remove-private-as」コマンドを追加し、このテンプレートを iBGP ピアに割り当てた場合、このコマンド「Remove-private-as」は適用されないというエラーは出力されません。 iBGP ピア。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **template peer-session** *template-name*
4. （任意） **password** *number password*
5. （任意） **timers** *keepalive hold*
6. **exit**
7. **neighbor** *ip-address* **remote-as** *as-number*
8. **inherit peer-session** *template-name*
9. （任意） **description** *text*
10. （任意） **show bgp peer-session** *template-name*
11. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	

ステップ 2	**router bgp** *autonomous-system-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP を有効にして、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**template peer-session** *template-name*
	例：
	`switch(config-router)# template peer-session BaseSession`
	`switch(config-router-stmp)#`
		peer-session テンプレート 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	（任意） **password** *number password* 例： `switch(config-router-stmp)# password 0 test`	ネイバーにクリアテキストのパスワード「test」を追加します。パスワードは 3DES（タイプ 3 暗号形式）で保存および表示されます。
ステップ 5	（任意） **timers** *keepalive hold* 例： `switch(config-router-stmp)# timers 30 90`	peer-session テンプレートに BGP キープアライブおよびホールドタイマー値を追加します。 デフォルトのキープアライブ インターバルは 60 です。デフォルトのホールド タイムは 180 です。

ステップ 6	**exit**
	例：
	`switch(config-router-stmp)# exit`
	`switch(config-router)#`
		peer-session テンプレート コンフィギュレーション モードを終了します。

ステップ 7	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 65535`
	`switch(config-router-neighbor)#`
		BGP ルーティング用のネイバー コンフィギュレーション モードを開始し、ネイバー IP アドレスを設定します。

ステップ 8	**inherit peer-session** *template-name*
	例：
	`switch(config-router-neighbor)# inherit peer-session BaseSession`
	`switch(config-router-neighbor)#`
		ピアに peer-session テンプレートを適用します。

ステップ 9	（任意） **description** *text*
	例：
	`switch(config-router-neighbor)# description Peer Router A`
	`switch(config-router-neighbor)#`
		ネイバーの説明を追加します。

	コマンドまたはアクション	目的
ステップ 10	（任意） **show bgp peer-session** *template-name* 例： `switch(config-router-neighbor)# show bgp peer-session BaseSession`	peer-policy テンプレートを表示します。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor)# copy running-config startup-config`	この設定変更を保存します。 **show bgp neighbor** コマンドを使用し、 コマンドを実行して、適用されたテンプレートを確認します。

例

BGP peer-session テンプレートを設定して、BGP ピアに適用する例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer-session BaseSession
switch(config-router-stmp)# timers 30 90
switch(config-router-stmp)# exit
switch(config-router)# neighbor 192.168.1.2 remote-as 65536
switch(config-router-neighbor)# inherit peer-session BaseSession
switch(config-router-neighbor)# description Peer Router A
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# copy running-config startup-config
```

BGP peer-policy テンプレートの設定

peer-policyテンプレートを設定すると、特定のアドレス ファミリに対応する属性を定義できます。各 peer-policyテンプレートにプリファレンスを割り当て、指定した順序でテンプレートが継承されるようにします。ネイバー アドレス ファミリでは最大 5 つの peer-policy テンプレートを使用できます。

Cisco NX-OS は、プリファレンス値を使用して、アドレス ファミリの複数のピア ポリシーを評価します。プリファレンス値が最小のものが最初に評価されます。ネイバーに設定した属性は、ネイバーが BGP テンプレートから継承した属性よりも優先されます。

peer-policy テンプレートでは、AS-path フィルタ リスト、プレフィックス リスト、ルート リフレクション、ソフト再構成など、アドレス ファミリ固有の属性を設定できます。

（注） **show bgp neighbor** コマンドを使用し、 コマンドを実行して、適用されたテンプレートを確認します。テンプレートで使用できる全コマンドの詳細については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Command Reference*』を参照してください。

始める前に

BGP を有効にする必要があります（「BGP の有効化」の項を参照）。

（注） テンプレートを編集するときには、ピアまたはテンプレートのレベルで **no** 形式のコマンドを使用すると、テンプレートの設定を明示的に上書きできます。属性をデフォルトの状態にリセットするには、default 形式のコマンドを使用する必要があります。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **template peer-session** *template-name*
4. （任意） **advertise-active-only**
5. （任意） **maximum-prefix** *number*
6. **exit**
7. **neighbor** *ip-address* **remote-as** *as-number*
8. **address-family** {**ipv4** | **ipv6**} {**multicast** | **unicast**}
9. **inherit peer-policy** *template-name preference*
10. （任意） **show bgp peer-policy** *template-name*
11. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードに入ります。

ステップ 2	**router bgp** *autonomous-system-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP を有効にして、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**template peer-session** *template-name*
	例：
	`switch(config-router)# template peer-policy BasePolicy`
	`switch(config-router-ptmp)#`
		peer-policy テンプレートを作成します。

ステップ 4	（任意） **advertise-active-only** 例： `switch(config-router-ptmp)# advertise-active-only`	アクティブ ルートのみをピアにアドバタイズします。

	コマンドまたはアクション	目的
ステップ 5	（任意） **maximum-prefix** *number* 例： `switch(config-router-ptmp)# maximum-prefix 20`	このピアに認めるプレフィックスの最大数を設定します。

ステップ 6	**exit**
	例：
	`switch(config-router-ptmp)# exit`
	`switch(config-router)#`
		peer-policy テンプレート コンフィギュレーション モードを終了します。

ステップ 7	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 65535`
	`switch(config-router-neighbor)#`
		BGP ルーティング用のネイバー コンフィギュレーション モードを開始し、ネイバー IP アドレスを設定します。

ステップ 8	**address-family** {**ipv4** | **ipv6**} {**multicast** | **unicast**}
	例：
	`switch(config-router-neighbor)# address-family ipv4 unicast`
	`switch(config-router-neighbor-af)#`
		指定のアドレス ファミリに対しグローバル アドレス ファミリ設定モードを開始します。

ステップ 9	**inherit peer-policy** *template-name preference* 例： `switch(config-router-neighbor-af)# inherit peer-policy BasePolicy 1`	ピア アドレス ファミリ設定に peer-policy テンプレートを適用し、このピア ポリシーのプリファレンス値を割り当てます。
ステップ 10	（任意） **show bgp peer-policy** *template-name* 例： `switch(config-router-neighbor-af)# show bgp peer-policy BasePolicy`	peer-policy テンプレートを表示します。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。 **show bgp neighbor** コマンドを使用し、 コマンドを実行して、適用されたテンプレートを確認します。

例

BGP peer-policy テンプレートを設定して、BGP ピアに適用する例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer-session BasePolicy
switch(config-router-ptmp)# maximum-prefix 20
switch(config-router-ptmp)# exit
switch(config-router)# neighbor 192.168.1.1 remote-as 65536
```

```
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# inherit peer-policy BasePolicy
switch(config-router-neighbor-af)# copy running-config startup-config
```

BGP peer テンプレートの設定

BGP peer テンプレートを設定すると、1 つの再利用可能なコンフィギュレーション ブロックで、セッション属性とポリシー属性を結合することができます。peer テンプレートも、peer-session または peer-policy テンプレートを継承できます。ネイバーに設定した属性は、ネイバーが BGP テンプレートから継承した属性よりも優先されます。ネイバーに設定できる peer テンプレートは 1 つだけですが、peer テンプレートは peer-session および peer-policy テンプレートを継承できます。

peer テンプレートは、eBGP マルチホップ TTL、最大プレフィックス数、ネクストホップ セルフ、タイマーなど、セッション属性およびアドレス ファミリ属性をサポートします。

始める前に

BGP を有効にする必要があります（「BGP の有効化」の項を参照）。

（注） テンプレートを編集するときには、ピアまたはテンプレートのレベルで **no** 形式のコマンドを使用すると、テンプレートの設定を明示的に上書きできます。属性をデフォルトの状態にリセットするには、default 形式のコマンドを使用する必要があります。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **template peer** *template-name*
4. （任意） **inherit peer-session** *template-name*
5. （任意） **address-family** {**ipv4**|**ipv6**} {**multicast**|**unicast**}
6. （任意） **inherit peer-policy** *template-name*
7. **exit**
8. （任意） **timers** *keepalive hold*
9. **exit**
10. **neighbor** *ip-address* **remote-as** *as-number*
11. **inherit peer** *template-name*
12. （任意） **timers** *keepalive hold*
13. （任意） **show bgp peer-template** *template-name*
14. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# 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 2	**router bgp** *autonomous-system-number* 例： `switch(config)# router bgp 65535`	BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。
ステップ 3	**template peer** *template-name* 例： `switch(config-router)# template peer BasePeer`	peer テンプレート 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	（任意） **inherit peer-session** *template-name* 例： `switch(config-router-neighbor)# inherit peer-session BaseSession`	ピア テンプレートに peer-session テンプレートを適用します。
ステップ 5	（任意） **address-family** {**ipv4**|**ipv6**} {**multicast**|**unicast**} 例： `switch(config-router-neighbor)# address-family ipv4 unicast switch(config-router-neighbor-af)`	指定のアドレス ファミリに対しグローバル アドレス ファミリ コンフィギュレーション モードを設定します。
ステップ 6	（任意） **inherit peer-policy** *template-name* 例： `switch(config-router-neighbor-af)# inherit peer-policy BasePolicy 1`	ネイバー アドレス ファミリ設定に peer-policy テンプレートを適用します。
ステップ 7	**exit** 例： `switch(config-router-neighbor-af)# exit`	BGP ネイバーアドレス ファミリ コンフィギュレーション モードを終了します。
ステップ 8	（任意） **timers** *keepalive hold* 例： `switch(config-router-neighbor)# timers 45 100`	ピアに BGP タイマー値を追加します。 これらの値によって、peer-session テンプレート、BaseSession のタイマー値が上書きされます。
ステップ 9	**exit** 例： `switch(config-router-neighbor)# exit`	BGP ネイバー コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的

ステップ 10	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 65535`
	`switch(config-router-neighbor)#`
		BGP ルーティング用のネイバー設定モードを開始し、ネイバー IP アドレスを設定します。

ステップ 11	**inherit peer** *template-name* 例： `switch(config-router-neighbor)# inherit peer BasePeer`	peer テンプレートを継承します。
ステップ 12	（任意） **timers** *keepalive hold* 例： `switch(config-router-neighbor)# timers 60 120`	このネイバーに BGP タイマー値を追加します。 これらの値によって、peer テンプレートおよび peer-session テンプレートのタイマー値が上書きされます。
ステップ 13	（任意） **show bgp peer-template** *template-name* 例： `switch(config-router-neighbor)# show bgp peer-template BasePeer`	peer テンプレートを表示します。
ステップ 14	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor)# copy running-config startup-config`	この設定変更を保存します。 **show bgp neighbor** コマンドを使用し、 コマンドを実行して、適用されたテンプレートを確認します。

例

BGP peer テンプレートを設定して、BGP ピアに適用する例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer BasePeer
switch(config-router-neighbor)# inherit peer-session BaseSession
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# inherit peer-policy BasePolicy 1
switch(config-router-neighbor-af)# exit
switch(config-router-neighbor)# exit
switch(config-router)# neighbor 192.168.1.2 remote-as 65536
switch(config-router-neighbor)# inherit peer BasePeer
switch(config-router-neighbor)# copy running-config startup-config
```

プレフィックス ピアリングの設定

BGP では、IPv4 と IPv6 の両方のプレフィックスを使用してピア セットを定義できます。この機能を使用すると、各ネイバーを設定に追加する必要がありません。

プレフィックス ピアリングを定義する場合は、プレフィックスとともにリモート AS 番号を指定する必要があります。プレフィックス ピアリングが設定されている許容最大ピア数を超えない場合、BGP はプレフィックスおよび自律システムから接続するピアを受け付けます。

プレフィックス ピアリングに含まれている BGP ピアが切断されると、Cisco NX-OS は定義されているプレフィックス ピア タイムアウト値まで、ピア構造を維持します。この場合、そのプレフィックス ピアリングのすべてのスロットを他のピアが使い果たした結果、ブロックされるという危険性を伴わずに、確立されたピアのリセットまたは再接続が可能になります。

手順の概要

1. **timers prefix-peer-timeout** *value*
2. **maximum-peers** *value*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**timers prefix-peer-timeout** *value* 例： `switch(config-router-neighbor)# timers prefix-peer-timeout 120`	ルータ コンフィギュレーション モードで BGP プレフィックス ピアリングのタイムアウト値を設定します。有効な範囲は 0 ～ 1200 秒です。デフォルト値は 30 です。 （注） プレフィックス ピアの場合は、プレフィックス ピア タイムアウトを、設定されたグレースフル リスタート タイマーよりも大きく設定します。プレフィックス ピア タイムアウトがグレースフル リスタート タイマーよりも大きければ、ピアのルートは再起動中に保持されます。プレフィックス ピア タイムアウトがグレースフル リスタート タイマーよりも小さいと、ピアのルートはプレフィックス ピア タイムアウトによって消去されます。これは、再起動が完了する前に発生する可能性があります。
ステップ 2	**maximum-peers** *value* 例： `switch(config-router-neighbor)# maximum-peers 120`	ネイバー設定モードのこのプレフィックス ピアリングの最大ピア数を設定します。範囲は 1 ～ 1000 です。

例

最大 10 のピアを受け付けるプレフィックス ピアリングの設定例を示します。

```
switch(config)# router bgp 65536
switch(config-router)# timers prefix-peer-timeout 120
```

```
switch(config-router)# neighbor 10.100.200.0/24 remote-as 65536
switch(config-router-neighbor)# maximum-peers 10
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)#
```

show bgp ipv4 unicast neighbors コマンドを使用し、すると、所定のプレフィックス ピアリングの設定の詳細とともに、現在受け付けられているインスタンスのリスト、アクティブ ピア数、最大同時ピア数、および受け付けたピアの合計数を表示できます。

IPv4 および IPv6 アドレス ファミリ向け IPv6 リンク ローカル経由の BGP インターフェイス ピアリングの設定

アンナンバード インターフェイスを使用した自動 BGP ネイバー探索のために、IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカルを経由して、BGP インターフェイス ピアリングを設定できます。これにより、インターフェイス名を（インターフェイススコープのアドレスではなく）BGP ピアとして使用する BGP セッションを設定できます。この機能は、ICMPv6 ネイバー探索（ND）のルート アドバタイズメント（RA）を使用して自動ネイバー探索を行い、RFC 5549 を使用して IPv6 ネクスト ホップで IPv4 ルートを送信します。

始める前に

BGP をイネーブルにする必要があります（「BGP のイネーブル化」の項を参照）。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	コンフィギュレーション モードに入ります。

ステップ 2	**router bgp** *autonomous-system-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP を有効にして、ローカル BGP スピーカに自律システム番号を割り当てます。AS 番号は 16 ビット整数または 32 ビット整数にできます。上位 16 ビット 10 進数と下位 16 ビット 10 進数による xx.xx という形式です。

ステップ 3	**neighbor** *interface-name* **remote-as** {*as-number* | **route-map** *map-name*} 例：	BGP ルーティングのためにルータをネイバー設定モードにして、インターフェイスを BGP ピア用に設定します。

	コマンドまたはアクション	目的
	`switch(config-router)# neighbor Ethernet1/1 remote-as 65535` `switch(config-router-neighbor)#`	（注） 指定できるのは、イーサネットインターフェイス、ポートチャネルインターフェイス、サブインターフェイス、およびブレークアウト インターフェイスだけです。 Cisco NX-OS リリース 9.3(6) 以降では、ルート マップを指定でき、AS リストを含められるルート マップを指定できます。ダイナミック AS 番号の使用の詳細については、プレフィックス ピアおよびインターフェイス ピアのダイナミック AS 番号 を参照してください。 設定を複数のインターフェイスに適用する必要がある場合、*interface-name* は範囲にすることができます。
ステップ 4	**inherit peer** *template-name* 例： `switch(config-router-neighbor)# inherit peer PEER`	peer テンプレートを継承します。

ステップ 5	**address-family** {**ipv4** | **ipv6**} **unicast**
	例：
	`switch(config-router-neighbor)#`
	`address-family ipv4 unicast`
	`switch(config-router-neighbor-af)#`
		指定のアドレス ファミリに対しグローバル アドレス ファミリ設定モードを開始します。

ステップ 6	（任意） **show bgp** {**ipv4** | **ipv6**} **unicast neighbors** *interface*
	例：
	`switch(config-router-neighbor-af)# show bgp ipv4 unicast neighbors e1/25`
	例：
	`switch(config-router-neighbor-af)# show bgp ipv6 unicast neighbors 3FFE:700:20:1::11`
		BGP ピアに関する情報を表示します。

ステップ 7	（任意） **show ip bgp neighbors** *interface-name* 例： `switch(config-router-neighbor-af)# show ip bgp neighbors Ethernet1/1`	BGP ピアとして使用されるインターフェイスを表示します。

	コマンドまたはアクション	目的
ステップ 8	（任意） **show ipv6 routers** [**interface** *interface*] 例： `switch(config-router-neighbor-af)# show ipv6 routers interface Ethernet1/1`	IPv6 ICMP ルータ アドバタイズメントによって学習されたリモート IPv6 ルータのリンク ローカル アドレスを表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。

例

次に、IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカル経由で、BGP インターフェイス ピアリングを設定する例を示します。

リーフ 1 の iBGP インターフェイス ピアリング設定：

```
switch# configure terminal
switch(config)# router bgp 65000
switch(config-router)# neighbor Ethernet1/1 remote-as 65000
switch(config-router-neighbor)# inherit peer PEER
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor)# address-family ipv6 unicast
switch(config-router-neighbor-af)# copy running-config startup-config
```

次に、IPv4 および IPv6 アドレス ファミリの IPv6 リンクローカル経由での、BGP インターフェイス ピアリングのサンプル出力例を示します。

```
switch(config-router-neighbor)# show bgp ipv4 unicast neighbors e1/15.1
BGP neighbor is fe80::2, remote AS 100, ibgp link, Peer index 4
Peer is an instance of interface peering Ethernet1/15.1
BGP version 4, remote router ID 5.5.5.5
Neighbor previous state = OpenConfirm
BGP state = Established, up for 2d16h
Neighbor vrf: default
Peer is directly attached, interface Ethernet1/15.1
Last read 00:00:54, hold time = 180, keepalive interval is 60 seconds
Last written 00:00:08, keepalive timer expiry due 00:00:51
Received 3869 messages, 0 notifications, 0 bytes in queue
Sent 3871 messages, 0 notifications, 0(0) bytes in queue
Enhanced error processing: On
0 discarded attributes
Connections established 2, dropped 1
Last reset by peer 2d16h, due to session closed
Last error length received: 0
Reset error value received 0
Reset error received major: 104 minor: 0
Notification data received:
Last reset by us never, due to No error
Last error length sent: 0
Reset error value sent: 0
Reset error sent major: 0 minor: 0
--More--
```

インターフェイス コンフィギュレーション：

次のいずれかのコマンドを使用して、対応するインターフェイスでIPv6を有効にする必要があります。

- **ipv6 address** *ipv6-address*
- **ipv6 address use-link-local-only**
- **ipv6 link-local** *link-local-address*

```
switch# configure terminal
switch(config)# interface Ethernet1/1
switch(config-if)# ipv6 address use-link-local-only
```

（注） インターフェイスで IPv4 アドレスが設定されていない場合は、**ip forward** コマンドをインターフェイスで設定して IPv4 転送を有効にする必要があります。

（注） IPv6 ND タイマーを調整して、ネイバー探索を高速化し、BGP のルートコンバージェンスを高速化できます。

```
switch(config-if)# ipv6 nd ra-interval 4 min 3
switch(config-if)# ipv6 nd ra-lifetime 10
```

（注） Cisco NX-OS リリース 9.3(6) 以降で、パラレル リンクを使用するカスタマーの導入では、インターフェイス モードで次のコマンドを追加する必要があります。

```
switch(config-if)# ipv6 link-local use-bia
```

このコマンドは、異なるインターフェイス間での IPv6 LLA を一意にします。

BGP 認証の設定

MD5ダイジェストを使用してピアからのルート更新を認証するように、BGPを設定できます。

Cisco NX-OS リリース 10.3(3)F 以降では、BGP パスワードのタイプ 6 暗号化が Cisco NX-OS スイッチでサポートされています。以下の暗号化タイプがサポートされます。

- AES ベースの暗号化
- 秘密の暗号化と復号には、プライマリキーと呼ばれる構成可能な暗号キーが使用されます。

MD5ダイジェストを使用するようにBGPを設定するには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

始める前に

- プライマリキーが Cisco NX-OS スイッチで **key config-key ascii** <*primary_key*> コマンドを使用して構成されていることを確認します。
- タイプ 6 暗号化を適切に機能させるには、Cisco NX-OS スイッチで **feature password encryption aes** が有効になっていることを確認します。

手順の概要

1. **key config-key ascii** <*primary_key*>
2. **configure terminal**
3. **feature password encryption aes**
4. **router bgp** *AS* 番号
5. **template peer**テンプレート名
6. **password** {**0** | **3** | **7** | **6**} *string*
7. （任意） **encryption re-encrypt obfuscated**
8. （任意） **encryption delete type-6**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**key config-key ascii** <*primary_key*> 例： `switch# key config-key ascii 0123456789012345`	プライマリ キーを構成します。 （注） • このコマンドは、プライマリ キーが構成されていない場合にのみ入力します。 • プライマリ キーがすでに構成されている場合にこのコマンドを入力すると、実際には既存のプライマリ キー値が変更されます。新しい値に変更するには、プロンプトが表示されたら既存のプライマリ キー値を入力します。
ステップ 2	**configure terminal** 例： `switch# 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	**feature password encryption aes** 例： `switch(config)# feature password encryption aes`	AES パスワード暗号化を有効にします。

	コマンドまたはアクション	目的
ステップ 4	**router bgp** *AS* 番号 例： `switch(config-router)# router bgp 1`	BGP ルータ モードを開始します。
ステップ 5	**template peer**テンプレート名 例： `switch(config-router-neighbor)# template peer abc`	BGP ネイバー モードを開始します。
ステップ 6	**password** {**0** | **3** | **7** | **6**} *string* 例： `switch(config-router-neighbor)# password 6 [redacted]`	MGP ネイバー セッションの MD5 パスワードを設定します。 （注） タイプ 0/タイプ 3/タイプ 7 を新しく構成する場合、プライマリ キーが構成されていて **feature password encryption aes** が有効になっている場合、タイプ 0/3/7 はタイプ 6 パスワードに自動的に暗号化されます。
ステップ 7	（任意） **encryption re-encrypt obfuscated** 例： `switch# encryption re-encrypt obfuscated`	既存のタイプ 0/タイプ 3/タイプ 7 パスワードをタイプ 6 パスワードに暗号化します。
ステップ 8	（任意） **encryption delete type-6** 例： `switch# encryption delete type-6`	タイプ 6 暗号化パスワードを削除します。

BGP セッションのリセット

BGP のルート ポリシーを変更した場合は、関連付けられた BGP ピア セッションをリセットする必要があります。BGP ピアがルート リフレッシュをサポートしない場合は、着信ポリシー変更に関するソフト再構成を設定できます。Cisco NX-OS は自動的に、セッションのソフト リセットを試みます。

ソフト再構成着信を設定するには、ネイバー アドレス ファミリ設定モードで次のコマンドを使用します。

手順の概要

1. **soft-reconfiguration inbound**
2. （任意） **clear bgp** {**ipv4** | **ipv6** } {**unicast** | **multicast** *ip-address* **soft** {**in** | **out**}
3. **clear bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} *ip-address* **soft** (**in** | **out**)

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**soft-reconfiguration inbound** 例： `switch(config-router-neighbor-af)# soft-reconfiguration inbound`	着信 BGP ルート アップデートを格納するために、ソフト再構成をイネーブルにします。このコマンドによって、BGP ネイバー セッションの自動ソフト クリアまたはリフレッシュが開始されます。
ステップ 2	（任意） **clear bgp** {**ipv4** | **ipv6** } {**unicast** | **multicast** *ip-address* **soft** {**in** | **out**} 例： `switch# clear bgp ip unicast 192.0.2.1 soft in`	TCP セッションを切断しないで、BGP セッションをリセットします。
ステップ 3	**clear bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} *ip-address* **soft** (**in** | **out**) 例： `switch# clear bgp ip unicast 192.0.2.1 soft in`	TCP セッションを切断しないで、BGP セッションをリセットします。

ネクストホップ アドレスの変更

次の方法で、ルートアドバタイズメントで使用するネクストホップアドレスを変更できます。

- ネクストホップ計算をディセーブルにして、ローカル BGP スピーカ アドレスをネクスト ホップ アドレスとして使用します。
- ネクストホップ アドレスをサードパーティ アドレスとして設定します。この機能は、元のネクスト ホップ アドレスがルートの送り先のピアと同じサブネット上にある場合に使用します。この機能を使用すると、フォワーディング時に余分なホップを節約できます。

ネクストホップ アドレス トラッキングを変更するには、アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **next-hop-self**
2. **next-hop-third-party**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**next-hop-self** 例： `switch(config-router-neighbor-af)# next-hop-self`	ルート アップデートのネクストホップ アドレスとして、ローカル BGP スピーカ アドレスを使用します。このコマンドによって、BGP ネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。

	コマンドまたはアクション	目的
ステップ 2	**next-hop-third-party** 例： `switch(config-router-neighbor-af)# next-hop-third-party`	ネクストホップ アドレスをサードパーティ アドレスとして設定します。このコマンドは、**next-hop-self** が設定されていないシングル ホップの EBGP ピアに使用します。 configured.

BGP ネクスト ホップ アドレス トラッキングの設定

BGP ネクスト ホップ アドレス トラッキングはデフォルトで有効であり、無効にすることができません。

BGP ネクスト ホップ トラッキングのパフォーマンスを向上するために、RIB チェック間の遅延インターバルを変更できます。

BGP ネクストホップアドレス トラッキングを変更するには、アドレス ファミリ設定モードで次のコマンドを使用します。

手順の概要

1. **nexthop trigger-delay** {**critical** | **non-critical**} *milliseconds*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**nexthop trigger-delay** {**critical**	**non-critical**} *milliseconds* 例： `switch(config-router-af)# nexthop trigger-delay critical 5000`	クリティカルなネクストホップの到達可能性ルートおよび非クリティカルなルートについて、ネクスト ホップ アドレス トラッキングの遅延タイマーを指定します。指定できる範囲は 1 ～ 4294967295 ミリ秒です。クリティカル タイマーのデフォルトは 3000 です。非クリティカル タイマーのデフォルトは 10000 です。

ネクスト ホップ フィルタリングの設定

BGP ネクストホップ フィルタリングを使用すると、RIB でネクストホップ アドレスがチェックされるときにそのネクストホップ アドレスの基盤となるルートがルート マップを経由します。ルート マップでそのルートが拒否されると、ネクストホップ アドレスは到達不能として扱われます。

BGP は、ルート ポリシーによって拒否されたすべてのネクストホップを無効であるとマークし、無効なネクストホップ アドレスを使用するルートについてベスト パスを計算しません。

BGP ネクストホップ フィルタリングを設定するには、アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **nexthop route-map** *name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**nexthop route-map** *name* 例： `switch(config-router-af)# nexthop route-map nextHopLimits`	BGP ネクストホップ ルートが一致するルート マップを指定します。63 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

デフォルト ルートによるネクストホップ解決の設定

BGP ネクストホップ解決では、IP デフォルト ルートを BGP ネクストホップ解決に使用するかどうかを指定できます。

BGP ネクストホップ解決を設定するには、ルータ設定モードで次のコマンドを使用します。

手順の概要

1. [**no**] **nexthop suppress-default-resolution**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **nexthop suppress-default-resolution** 例： `switch(config-router)# nexthop suppress-default-resolution`	IP デフォルト ルートを介した BGP ネクストホップの解決を防止します。 このコマンドを有効にすると、以下のようになります。 • **show bgp process detail** コマンドの出力には、次の行が含まれます。 Use default route for nexthop Resolution：No • **show routing clients bgp** コマンドの出力には、次の行が含まれます。 Owned rnh will never resolve to 0.0.0.0/0

ネクストホップセルフによるリフレクト ルートの制御

NX-OS では、**next-hop-self** [all] 引数を使用して特定のピアに送信する際の iBGP ルートを制御できます。これらの引数を使用すると、ルートのリフレクトが実施されている場合でも、ルートのネクストホップを選択的に変更できます。

コマンド	目的
next-hop-self [all] 例： `switch(config-router-af)# next-hop-self all`	ルートアップデートのネクストホップアドレスとして、ローカルBGPスピーカアドレスを使用します。 allキーワードはオプションです。allを指定すると、すべてのルートがnext-hop-selfを使用するピアに送信されます。allを指定しなかった場合、リフレクトしたルートのネクストホップは変更されません。

セッションがダウンした場合のネクストホップ グループの縮小

セッションがダウンしたときに迅速な方法で ECMP グループを縮小するように BGP を設定できます。

この機能は、次の BGP パス障害イベントに適用されます。

- 1 つまたは複数のレイヤ 3 リンクの障害
- ライン カード障害
- BGP ネイバーの BFD 障害検出
- BGP ネイバーの管理上のシャットダウン（shutdown コマンドを使用）

最初の2つのイベント（レイヤ3リンク障害とラインカード障害）の迅速な処理はデフォルトでイネーブルになっており、イネーブルにするための設定コマンドは必要ありません。

最後の2つのイベントの迅速な処理を設定するには、ルータ設定モードで次のコマンドを使用します。

手順の概要

1. **neighbor-down fib-accelerate**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**neighbor-down fib-accelerate** 例： `switch(config-router)# neighbor-down fib-accelerate`	BGPセッションがダウンするたびに、すべてのネクストホップグループ（ECMPグループと単一のネクストホップルート）から対応する次のネクストホップを取り消します。 （注） このコマンドは、IPv4ルートとIPv6ルートの両方に適用されます。

機能ネゴシエーションのディセーブル化

機能ネゴシエーションをディセーブルにすると、機能ネゴシエーションをサポートしない古い BGP ピアとの相互運用が可能です。

機能ネゴシエーションをディセーブルにするには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **dont-capability-negotiate**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**dont-capability-negotiate** 例： `switch(config-router-neighbor)# dont-capability-negotiate`	機能ネゴシエーションをディセーブルにします。このコマンドの設定後、BGP セッションを手動でリセットする必要があります。

ポリシーのバッチ処理の無効化

プレフィックスに一意の属性がある BGP 展開では、BGP は、同じ BGP アップデートメッセージでバンドルする類似の属性を持つルートを識別しようとします。この追加の BGP 処理のオーバーヘッドを回避するには、バッチ処理をディセーブルにします。

固有のネクスト ホップを持つ多数のルートがある BGP 展開では、ポリシーバッチ処理を無効にすることを推奨します。

ポリシー バッチ処理を無効にするには、ルータ設定モードで次のコマンドを使用します。

手順の概要

1. **disable-policy-batching**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**disable-policy-batching** 例： `switch(config-router)# disable-policy-batching`	すべてのピアへのプレフィックスアドバタイズメントのバッチ評価をディセーブルにします。

BGP 追加パスの設定

BGPは、プレフィックスごとの複数パスの送受信と、このパスのアドバタイジングをサポートします。

追加パスの送受信機能のアドバタイズ

BGP ピア間の追加パスの送受信機能をアドバタイズするように BGP を設定できます。これを行うには、ネイバー アドレス ファミリ設定モードで次のコマンドを使用します。

手順の概要

1. [**no**] **capability additional-paths send** [**disable**]
2. [**no**] **capability additional-paths receive** [**disable**]
3. **show bgp neighbor**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **capability additional-paths send** [**disable**] 例： `switch(config-router-neighbor-af)# capability addtional-paths send`	BGP ピアに追加パスを送信する機能をアドバタイズします。**disable** オプションは、追加パス送信機能のアドバタイズをディセーブルにします。 このコマンドの **no** 形式を使用すると、追加パスの送信機能がディセーブルになります。
ステップ 2	[**no**] **capability additional-paths receive** [**disable**] 例： `switch(config-router-neighbor-af)# capability addtional-paths receive`	BGP ピアから追加パスを受信する機能をアドバタイズします。**disable** オプションは、追加パス受信機能のアドバタイズをディセーブルにします。 このコマンドの **no** 形式は、追加パスの受信機能をディセーブルにします。
ステップ 3	**show bgp neighbor** 例： `switch(config-router-neighbor-af)# show bgp neighbor`	ローカル ピアがリモート ピアへの追加パス送受信機能をアドバタイズしたかを表示します。

例

BGP ピアに追加のパスを送受信する機能をアドバタイズする BGP の設定例を示します。

```
switch# configure terminal
switch(config)# router bgp 100
switch(config-router)# neighbor 10.131.31.2 remote-as 100
```

```
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# capability additional-paths send
switch(config-router-neighbor-af)# capability additional-paths receive
```

追加パスの送受信の設定

BGP ピア間の追加パスの送受信機能を設定できます。これを行うには、アドレス ファミリ設定モードで次のコマンドを使用します。

手順の概要

1. [**no**] **additional-paths send**
2. [**no**] **additional-paths receive**
3. **show bgp neighbor**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **additional-paths send** 例： `switch(config-router-af)# additional-paths send`	機能が無効になっていないこのアドレス ファミリで、すべてのネイバーの追加パスの送信機能を有効にします。 このコマンドの **no** 形式を使用すると、送信機能が無効になります。
ステップ 2	[**no**] **additional-paths receive** 例： `switch(config-router-af)# additional-paths receive`	機能が無効になっていないこのアドレス ファミリで、すべてのネイバーの追加パスの受信機能を有効にします。 このコマンドの **no** 形式を使用すると、受信機能が無効になります。
ステップ 3	**show bgp neighbor** 例： `switch(config-router-af)# show bgp neighbor`	ローカル ピアがリモート ピアへの追加パス送受信機能をアドバタイズしたものとして表示します。

例

機能が無効になっていない指定されたアドレス ファミリで、すべてのネイバーの追加パスの受信機能を有効にする例を示します。

```
switch# configure terminal
switch(config)# router bgp 100
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# additional-paths send
switch(config-router-af)# additional-paths receive
```

アドバタイズされるパスの設定

BGPにアドバタイズされたパスを指定できます。これを行うには、ルートマップ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. [**no**] **set ip next-hop unchanged**
2. [**no**] **set path-selection { all | backup | best2 | multipaths}** | advertise
3. **show bgp** {**ipv4** | **ipv6**} **unicast** [*ip-address* | *ipv6-prefix*] [**vrf** *vrf-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **set ip next-hop unchanged** 例： `switch(config-route-map)# set ip next-hop unchanged`	不変のネクストホップ IP アドレスを指定します。
ステップ 2	[**no**] **set path-selection { all | backup | best2 | multipaths}** | advertise 例： `switch(config-route-map)# set path-selection all advertise`	すべてのパスが指定されたプレフィックスにアドバタイズされるように指定します。次のいずれかのオプションを選択できます。 • all：使用可能なすべての有効なパスをアドバタイズします。 • backup：バックアップパスとしてマークされたパスをアドバタイズします。このオプションでは、additional-path install backup コマンドを使用してバックアップパスを有効にする必要があります。 • best2：2 番目に最適なパスをアドバタイズします。これは、すでに計算されているベストパスを除き、残りの使用可能なパスのベストパスです。 • multipaths：すべてのマルチパスをアドバタイズします。このオプションでは、maximum-paths コマンドを使用してマルチパスを有効にする必要があります。

	コマンドまたはアクション	目的
		（注） マルチパスがない場合、backup オプションと best2 オプションは同じです。マルチパスがある場合、best2 はマルチパスのリストの最初のパスで、バックアップは計算されたベスト パスとマルチパスを除くすべての使用可能なパスのベストパスです。 このコマンドの **no** 形式は、最適パスだけがアドバタイズされるように指定します。
ステップ 3	**show bgp** {**ipv4** | **ipv6**} **unicast** [*ip-address* | *ipv6-prefix*] [**vrf** *vrf-name*] 例： `switch(config-route-map)# show bgp ipv4 unicast`	プレフィックスの追加パスのパス ID とこれらのパスのアドバタイズメント情報を表示します。

例

すべてのパスがプレフィックス リスト p1 にアドバタイズされるよう指定する例を示します。

```
switch# configure terminal
switch(config)# route-map PATH_SELECTION_RMAP
switch(config-route-map)# match ip address prefix-list p1
switch(config-route-map)# set path-selection all advertise
```

追加パス選択の設定

プレフィックスに追加のパスを選択する機能を設定できます。これを行うには、アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. [**no**] **additional-paths selection route-map** *map-name*
2. {| } [ip-address | ipv6-prefix] [vrf-name] **show bgpipv4ipv6unicastvrf**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **additional-paths selection route-map** *map-name* 例： `switch(config-router-af)# additional paths selection route-map map1`	プレフィックスに追加のパスを選択する機能を設定します。 このコマンドの **no** 形式は、追加パス選択機能をディセーブルにします。

	コマンドまたはアクション	目的
ステップ 2	{| } [ip-address | ipv6-prefix] [vrf-name] **show bgpipv4ipv6unicastvrf** 例： `switch(config-route-af)# show bgp ipv4 unicast`	プレフィックスの追加パスのパス ID とこれらのパスのアドバタイズメント情報を表示します。

例

指定されたアドレス ファミリで追加パス選択を設定する例を示します。

```
switch# configure terminal
switch(config)# router bgp 100
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# additional-paths selection route-map PATH_SELECTION_RMAP
```

eBGP の設定

eBGP シングルホップ チェックの無効化

シングルホップ eBGP ピアがローカル ルータに直接接続されているかどうかのチェック機能を無効にするように、eBGP を設定できます。このオプションは、直接接続されたスイッチ間のシングルホップ ループバック eBGP セッションの設定に使用します。

シングルホップ eBGP ピアが直接接続されているかどうかのチェックを無効にするには、ネイバー設定モードで次のコマンドを使用します。

手順の概要

1. **disable-connected-check**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**disable-connected-check** 例： `switch(config-router-neighbor)# disable-connected-check`	シングルホップ eBGP ピアが直接接続されているかどうかのチェックを無効にします。このコマンドの使用後、BGP セッションを手動でリセットする必要があります。

TTL セキュリティ ホップの構成

IP パケット ヘッダーの TTL 値が BGP ネイバー セッション用に設定された TTL 値以上の場合のみ BGP がセッションを確立または維持できるようにするには、次の作業を実行します。

始める前に

TTL セキュリティ チェックに対する BGP サポート機能の効果を最大化するために、参加している各ルータでこの機能を設定することを推奨します。この機能を有効にすると、eBGP セッションが受信方向だけ保護され、送信 IP パケットまたはリモート ルータは影響を受けません。

（注）
- TTL セキュリティ チェックに対する BGP サポート機能がマルチホップ ネイバー セッション用に構成されている場合、**neighbor ebgp-multihop** コマンドは必要なく、この機能を構成する前にこのコマンドをディセーブルにする必要があります。
- 大きい直径のマルチホップ ピアリングでは、TTL セキュリティ チェックに対する BGP サポート機能の効果は下がります。大きい直径のピアリング用に設定された BGP ルータに対する CPU 利用率に基づく攻撃の場合は、影響を受けたネイバー セッションをシャットダウンして、この攻撃に対処する必要がある場合があります。
- この機能は、ローカル ネットワークおよびリモート ネットワーク内部が損なわれているピアからの攻撃には効果的ではありません。この制約事項には、ローカル ネットワークとリモート ネットワークの間のネットワーク セグメント上のピアも含まれます。

手順の概要

1. **enable**
2. **trace** *[protocol] destination*
3. **configure terminal**
4. **router bgp** *autonomous-system-number*
5. **neighbor** *ip-address*
6. **ttl-security hops** *hop-count*
7. **end**
8. **show running-config**
9. **show ip bgp neighbors** *[ip-address]*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch(config)# enable`	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	**trace** *[protocol] destination* 例： `switch(config)# trace ip 10.1.1.1`	パケットが宛先に移動中、実際に通過する指定されたプロトコルのルートを検出します。 trace コマンドを入力して、指定されたピアへのホップ カウントを決定します。

	コマンドまたはアクション	目的
ステップ 3	**configure terminal** 例： `switch(config)# 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 4	**router bgp** *autonomous-system-number* 例： `switch(config)# router bgp 65000`	ルータ コンフィギュレーションモードを開始して、BGP ルーティング プロセスを作成します。
ステップ 5	**neighbor** *ip-address* 例： `switch(config)# neighbor 10.1.1.1`	ネイバー IP アドレスを構成します。
ステップ 6	**ttl-security hops** *hop-count* 例： `switch(config)# ttl-security hops 2`	2 つのピアを区切るホップの最大数を設定します。 hop-count 引数は、ローカル ピアとリモート ピアを区切るホップ カウントに設定されます。IP パケット ヘッダーの予想される TTL 値が 254 の場合、数値 1 を hop-count 引数に設定する必要があります。値の範囲は、1 ～ 254 の数番です。 TTL セキュリティ チェックに対する BGP サポート機能が有効な場合、BGP は、予想値以上の TTL 値を持つ着信 IP パケットを受け入れます。受け入れられないパケットは廃棄されます。 この設定例では、予想される着信 TTL 値が 253（255 引く TTL 値の 2）以上に設定されます。これは、BGP ピアから予想される最小 TTL 値です。ローカル ルータは、10.1.1.1 ネイバーが 1 または 2 ホップ離れている場合だけ、このネイバーからのピアリング セッションを受け入れます。
ステップ 7	**end** 例： `switch(config)# end`	ルータ コンフィギュレーションモードを終了して、特権 EXEC モードを開始します。
ステップ 8	**show running-config** 例： `switch(config)# show running-config | begin bgp`	（任意）現在実行中のコンフィギュレーション ファイルの内容を表示します。 このコマンドの出力は、各ピアの neighbor ttl-security コマンドの設定を出力の BGP コンフィギュレーション セクションの下に表示します。そのセクションには、ネイバー アドレスおよび構成されたホップ カウントが含まれます。

	コマンドまたはアクション	目的
		（注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS IP Routing: BGP Command Reference』を参照してください。
ステップ 9	**show ip bgp neighbors** *[ip-address]* 例： `switch(config)# show ip bgp neighbors 10.4.9.5`	（任意）ネイバーへの TCP 接続および BGP 接続の情報を表示します。 このコマンドは、TTL セキュリティ チェックに対する BGP サポート機能が有効になっている場合、「External BGP neighbor may be up to number hops away」と表示します。この number 値は、ホップ カウントを表します。これは、1 ～ 254 の数値です。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS IP Routing: BGP Command Reference』を参照してください。

eBGP マルチホップの設定

eBGP マルチホップをサポートする eBGP 存続可能時間（TTL）値を設定できます。eBGP ピアは状況によって、別の eBGP ピアに直接接続されず、リモート eBGP ピアに到達するために複数のホップを必要とします。ネイバーセッションに eBGP TTL 値を設定すると、このようなマルチホップ セッションが可能になります。

（注） この設定は、BGP インターフェイス ピアリングではサポートされません。

eBGP マルチホップを設定するには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **ebgp-multihop** *ttl-value*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**ebgp-multihop** *ttl-value* 例：	eBGP マルチホップの eBGP TTL を設定します。有効な範囲は 2 ～ 255 です。このコマンドの使用後、

	コマンドまたはアクション	目的
	`switch(config-router-neighbor)# ebgp-multihop 5`	BGP セッションを手動でリセットする必要があります。

高速外部フォールオーバーの無効化

Cisco NX-OS デバイスは、すべての VRF のネイバーおよびアドレス ファミリ（IPv4 または IPv6）の高速外部フォールオーバーをデフォルトでサポートします。通常、BGP ルータと直接接続 eBGP ピア間の接続が失われると、ピアとの eBGP セッションをリセットすることによって、BGP が高速外部フォールオーバーを開始します。この高速外部フォールオーバーをディセーブルにすると、リンク フラップが原因の不安定さを制限できます。

高速外部フォールオーバーをディセーブルにするには、ルータ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **no fast-external-fallover**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**no fast-external-fallover** 例： `switch(config-router)# no fast-external-fallover`	eBGP ピアの高速外部フォールオーバーをディセーブルにします。このコマンドは、デフォルトでイネーブルになっています。

AS パス属性の制限

AS パス属性で自律システム番号が高いルートを廃棄するように eBGP を設定できます。

AS パス属性で AS 番号の多いルートを廃棄するには、ルータ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **maxas-limit** *number*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**maxas-limit** *number* 例： `switch(config-router)# maxas-limit 50`	AS パス セグメントの番号が指定された上限を超えている eBGP ルートを廃棄します。指定できる範囲は 1 ～ 2000 です。

ローカル AS サポートの設定

ローカル AS 機能では、ルータが実際の AS に加えて、2 番めの自律システム（AS）のメンバであるように見せることができます。ローカル AS を使用すると、ピアリングの調整を変更せずに 2 つの ISP をマージできます。マージされた ISP 内のルータは、新しい自律システムのメンバになりますが、使用者に対しては古い自律システム番号を使用し続けます。

この機能は、正しい eBGP ピアにしか使用できません。別のコンフェデレーションのサブ自律システムのメンバである 2 ピアに対しては、この機能は使用できません。

さらに、remote-asコマンドで設定されたリモートピアのASNは、local-asコマンドで設定されたローカルデバイスのASNと同一にすることはできません。

eBGP ローカル AS のサポートを設定するには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **local-as** *number* [**no-prepend** [**replace-as** [**dual-as**]]]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**local-as** *number* [**no-prepend** [**replace-as** [**dual-as**]]] 例： `switch(config-router-neighbor)# local-as 1.1`	AS_PATH 属性にローカル AS の *number* を付加するよう eBGP を設定します。AS 番号は 16 ビット整数または 32 ビット整数にできます。上位 16 ビット 10 進数と下位 16 ビット 10 進数による xx.xx という形式です。

例

次に、VRF のローカル AS サポートを設定する例を示します。

```
switch# configure terminal
switch(config)# router bgp 1
switch(config-router)# vrf test
switch(config-router-vrf)# local-as 1
switch(config-router-vrf)# show running-config bgp
```

AS 連合の設定

AS連合を設定するには、連合識別情報を指定する必要があります。AS連合内の自律システムグループは、自律システム番号として連合 ID を持つ、1 つの自律システムとして外部で認識されます。

BGP 連合 ID を設定するには、ルータ設定モードで次のコマンドを使用します。

手順の概要

1. **confederation identifier** *as-number*
2. **bgp confederation peers** *as-number* [*as-number2...*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**confederation identifier** *as-number* 例： `switch(config-router)# confederation identifier 4000`	ルータ設定モードで、このコマンドは BGP 連合 ID を設定します。 このコマンドによって、BGP ネイバーセッションの自動通知およびセッション リセットが開始されます。
ステップ 2	**bgp confederation peers** *as-number* [*as-number2...*] 例： `switch(config-router)# bgp confederation peers 5 33 44`	ルータ設定モードで、このコマンドは AS 連合に属する自律システムを設定します。 このコマンドは、連合に属する自律システムのリストを指定し、BGP ネイバーセッションの自動通知とセッションリセットをトリガーします。

ルート リフレクタの設定

ルート リフレクタとして動作するローカル BGP スピーカに対するルート リフレクタ クライアントとして、iBGP ピアを設定できます。ルート リフレクタとそのクライアントがともにクラスタを形成します。クライアントからなるクラスタには通常、ルート リフレクタが 1 つ存在します。このような状況では、ルート リフレクタのルータ ID でクラスタを識別します。ネットワークの冗長性を高め、シングル ポイント障害を回避するために、複数のルート リフレクタからなるクラスタを設定できます。クラスタ内のすべてのルート リフレクタは、同じ 4 バイト クラスタ ID で設定する必要があります。これは、ルート リフレクタが同じクラスタ内のルート リフレクタからのアップデートを認識できるようにするためです。

始める前に

BGPをイネーブルにする必要があります。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **cluster-id** *cluster-id*
4. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
5. （任意） **client-to-client reflection**
6. **exit**
7. **neighbor** *ip-address* **remote-as** *as-number*

8. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
9. **route-reflector-client**
10. （任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} **neighbors**
11. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# 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 2	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**cluster-id** *cluster-id* 例： `switch(config-router)# cluster-id 192.0.2.1`	クラスタに対応するルート リフレクタの 1 つとして、ローカル ルータを設定します。クラスタを識別するクラスタ ID を指定します。このコマンドによって、BGP ネイバー セッションの自動ソフト クリアまたはリフレッシュが開始されます。

ステップ 4	**address-family** {**ipv4**	**ipv6**} {**unicast**	**multicast**}
			例：
			`switch(config-router)# address-family ipv4 unicast`
			`switch(config-router-af)#`
				指定のアドレス ファミリに対応するグローバル アドレス ファミリ コンフィギュレーション モードを開始します。

ステップ 5	（任意） **client-to-client reflection** 例： `switch(config-router-af)# client-to-client reflection`	クライアント間のルート リフレクションを設定します。この機能は、デフォルトでイネーブルになっています。このコマンドによって、BGP ネイバー セッションの自動ソフトクリアまたはリフレッシュが開始されます。

ステップ 6	**exit**
	例：
	`switch(config-router-af)# exit`
	`switch(config-router)#`
		ルータ アドレス コンフィギュレーション モードを終了します。

ステップ 7	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.0.2.10 remote-as 65535`
	`switch(config-router-neighbor)#`
		リモート BGP ピアの IP アドレスおよび AS 番号を設定します。

	コマンドまたはアクション	目的

ステップ 8	**address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
	例：
	`switch(config-router-neighbor)# address-family ipv4 unicast`
	`switch(config-router-neighbor-af)#`
		ユニキャスト IPv4 アドレス ファミリに対応するネイバー アドレス ファミリ コンフィギュレーション モードを開始します。

ステップ 9	**route-reflector-client** 例： `switch(config-router-neighbor-af)# route-reflector-client`	BGP ルート リフレクタとしてデバイスを設定し、そのクライアントとしてネイバーを設定します。このコマンドによって、BGP ネイバー セッションの自動通知およびセッション リセットが開始されます。
ステップ 10	（任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} **neighbors** 例： `switch(config-router-neighbor-af)# show bgp ipv4 unicast neighbors`	BGP ピアを表示します。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。

例

次に、ルート リフレクタとしてルータを設定し、クライアントとしてネイバーを 1 つ追加する例を示します。

```
switch(config)# router bgp 65536
switch(config-router)# neighbor 192.0.2.10 remote-as 65536
switch(config-router-neighbor)# address-family ip unicast
switch(config-router-neighbor-af)# route-reflector-client
switch(config-router-neighbor-af)# copy running-config startup-config
```

アウトバウンド ルート マップを使用した、反映されたルートのネクスト ホップの設定

アウトバウンド ルート マップを使用して、BGP ルート リフレクタの反映されたルートのネクスト ホップを変更できます。ネクスト ホップ アドレスとしてピアのローカル アドレスを指定するため、アウトバウンド ルート マップを設定できます。

（注） この項で説明している **next-hop-self** コマンドは、ルート リフレクタによってクライアントに反映されるルートに対してこの機能を有効にしません。この機能は、アウトバウンド ルート マップを使用した場合にだけ有効にできます。

始める前に

BGP を有効にする必要があります（「BGP の有効化」の項を参照）。

正しいVDCを使用していることを確認します（または**switchto vdc**コマンドを使用します）。

set next-hop を入力する必要があります コマンドを入力して、アドレス ファミリ固有のネクスト ホップ アドレスを設定する必要があります。たとえば、IPv6アドレスファミリの場合は、**set ipv6 next-hop peer-address** コマンドを入力する必要があります。

- ルートマップを使用してIPv4ネクストホップを設定する場合：**set ip next-hop peer-address** がルートマップと一致する場合、ネクストホップはピアのローカルアドレスに設定されます。ネクスト ホップがルート マップで設定されていない場合、ネクスト ホップはパスに保存されているネクスト ホップに設定されます。
- ルートマップを使用してIPv6ネクストホップを設定する場合：**set ipv6 next-hop peer-address** がルートマップと一致する場合、ネクスト ホップは次のように設定されます。
  - IPv6 ピアでは、ネクスト ホップはピアのローカル IPv6 アドレスに設定されます。
  - IPv4 ピアの場合、 **update-source** が設定されている場合、ネクスト ホップは、該当する場合、発信元インターフェイスの IPv6 アドレスに設定されます。IPv6 アドレスが設定されていない場合、ネクスト ホップは設定されません。
  - IPv4 ピアの場合、 **update-source** が設定されていない場合、ネクスト ホップは、該当する場合、送信先インターフェイスの IPv6 アドレスに設定されます。IPv6 アドレスが設定されていない場合、ネクスト ホップは設定されません。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **neighbor** *ip-address* **remote-as** *as-number*
4. （任意） **update-source** *interface number*
5. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
6. **route-reflector-client**
7. **route-map** *map-name* **out**
8. （任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} [**ip-address** | **ipv6-prefix**] **route-map** *map-name* [**vrf** *vrf-name*]
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 200`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.0.2.12 remote-as 200`
	`switch(config-router-neighbor)#`
		リモート BGP ピアの IP アドレスおよび AS 番号を設定します。

ステップ 4	（任意） **update-source** *interface number* 例： `switch(config-router-neighbor)# update-source loopback 300`	BGP セッションの送信元を指定し、更新します。

ステップ 5	**address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
	例：
	`switch(config-router-neighbor)# address-family ipv4 unicast`
	`switch(config-router-neighbor-af)#`
		指定のアドレス ファミリに対応するグローバル アドレス ファミリ コンフィギュレーション モードを開始します。

ステップ 6	**route-reflector-client** 例： `switch(config-router-neighbor-af)# route-reflector-client`	BGP ルート リフレクタとしてデバイスを設定し、そのクライアントとしてネイバーを設定します。このコマンドによって、BGP ネイバー セッションの自動通知およびセッション リセットが開始されます。
ステップ 7	**route-map** *map-name* **out** 例： `switch(config-router-neighbor-af)# route-map setrrnh out`	発信ルートに設定された BGP ポリシーを適用します。
ステップ 8	（任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} [**ip-address** | **ipv6-prefix**] **route-map** *map-name* [**vrf** *vrf-name*] 例： `switch(config-router-neighbor-af)# show bgp ipv4 unicast route-map setrrnh`	ルートマップと一致する BGP ルートを表示します。

	コマンドまたはアクション	目的
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。

例

アウトバウンド ルート マップを使用して、BGP ルート リフレクタの反映されたルートのネクスト ホップを設定する例を示します。

```
switch(config)# interface loopback 300
switch(config-if)# ip address 192.0.2.11/32
switch(config-if)# ipv6 address 2001::a0c:1a65/64
switch(config-if)# ip router ospf 1 area 0.0.0.0
switch(config-if)# exit
switch(config)# route-map setrrnh permit 10
switch(config-route-map)# set ip next-hop peer-address
switch(config-route-map)# exit
switch(config)# route-map setrrnhv6 permit 10
switch(config-route-map)# set ipv6 next-hop peer-address
switch(config-route-map)# exit
switch(config)# router bgp 200
switch(config-router)# neighbor 192.0.2.12 remote-as 200
switch(config-router-neighbor)# update-source loopback 300
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# route-reflector-client
switch(config-router-neighbor-af)# route-map setrrnh out
switch(config-router-neighbor-af)# exit
switch(config-router-neighbor)# address-family ipv6 unicast
switch(config-router-neighbor-af)# route-reflector-client
switch(config-router-neighbor-af)# route-map setrrnhv6 out
```

ルート ダンプニングの設定

iBGP ネットワーク上でのルート フラップの伝播を最小限に抑えるために、ルート ダンプニングを設定できます。

ルート ダンプニングを設定するには、アドレス ファミリまたは VRF アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **dampening** [{*half-life reuse-limit suppress-limit max-suppress-time* | **route-map** *map-name*}]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**dampening** [{*half-life reuse-limit suppress-limit max-suppress-time* | **route-map** *map-name*}] 例： `switch(config-router-af)# dampening route-map bgpDamp`	機能ネゴシエーションをディセーブルにします。パラメータ値は次のとおりです。 • *half-life*：指定できる範囲は 1 ～ 45 です。 • *resuse-limit*指定できる範囲は 1 ～ 20000 です。 • *suppress-limit*：指定できる範囲は 1 ～ 20000 です。 • *max-suppress-time*：指定できる範囲は 1 ～ 20000 です。

ロード シェアリングおよび ECMP の設定

等コスト マルチパス ロード バランシング用に BGP がルート テーブルに追加するパスの最大数を設定できます（EXMP）。

パスの最大数を設定するには、ルータ アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **maximum-paths** [**ibgp**] *maxpaths*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**maximum-paths** [**ibgp**] *maxpaths* 例： `switch(config-router-af)# maximum-paths 8`	ロード シェアリング用の等コスト パスの最大数を設定します。デフォルトは 1 です。

BGP 経由不等コストマルチパス（UCMP）

UCMP は加重 ECMP とも呼ばれます。これは、ネクスト ホップごとに異なる重みを持つ、同じ宛先への複数のルートを許可し、ルーティングされたトラフィックをそれらの複数のネクスト ホップにロード バランシングするメカニズムです。基本的な UCMP は、ほとんどの顧客の要件に対応します。負荷エントロピーは、リンク使用効率を最大化する最良の方法です。

多くの場合、ネットワーク内のアプリケーションの分散は不均衡になりがちです。新しいクラスタは、古いクラスタとは異なるオーバーサブスクリプション率でロールインします。新しい

クラスタには、古いクラスタよりも強力なサーバーがあり、CPUごとにより多くの負荷を処理できます。ネットワークは完全ではないため、ルーティング動作をある程度制御する必要があります。トラフィックの負荷を分散し、ルーティング動作の制御を管理するために、BGP経由の加重 ECMP を構成できます。

（注） リンク帯域幅拡張コミュニティは、非推移的な属性として定義されていますが、eBGPセッション全体でアドバタイズする必要があります。

Next-hop-self は、アドバタイズから Link-Bandwidth Extended Community を取り除く必要があります。

UCMP over BGP の有効化

ユースケースでリソースの不均等な分散と最適ではないトラフィック分散が発生している場合の解決策は、BGP 上で重み付き ECMP を構成することです。各インスタンスの重みは、（ホストまたはコントローラーから）ルートを挿入して通知できます。その後、インフラストラクチャ全体の重みを集計し、アプリケーション展開の分布に比例するようにトラフィックを配信できます。

BGP 経由 UCMP の注意事項と制限事項

- BGPは、draft-ietf-idr-link-bandwidth-06.txtで定義されているリンク帯域幅拡張コミュニティを使用して、重み付けECMP機能を実装します。リンク帯域幅拡張コミュニティは、次ホップが変更されていない限り、非推移的な属性として定義されていますが、eBGP セッション全体でアドバタイズされます。
- iBGP ピアと eBGP ピアの両方からリンク帯域幅拡張コミュニティを受け入れることができます。
- 重み付けプログラミングの場合、リンク帯域幅拡張コミュニティには、RIBにダウンロードする前に 0 ～ 1000 の間で正規化された 4 バイトの浮動小数点整数としてバイト/秒でエンコードされたリンク帯域幅があります。
- ハードウェア ECMP 幅は 64 サイズに固定されています。

最大プレフィックス数の設定

BGPがBGPピアから受け取ることのできるプレフィックスの最大数を設定できます。任意で、プレフィックス数がこの値を超えた場合に、BGPに警告メッセージを生成させる、またはピアとの BGP セッションを切断させることを設定できます。

BPG ピアに認めるプレフィックスの最大数を設定するには、ネイバー アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **maximum-prefix** *maximum* [*threshold*] [**restart** *time* | **warning-only**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**maximum-prefix** *maximum* [*threshold*] [**restart** *time* | **warning-only**] 例： `switch(config-router-neighbor-af)# maximum-prefix 12`	ピアからのプレフィックスの最大数を設定します。パラメータの範囲は次のとおりです。 • *maximum*：指定できる範囲は 1 ～ 300000 です。 • *threshold*：指定できる範囲は 1 ～ 100 % です。デフォルトは 75% です。 • *time*：指定できる範囲は 1 ～ 65535 分です。 このコマンドによって、プレフィックス限度を超えた場合に、BGP ネイバーセッションの自動通知およびセッション リセットが開始されます。

DSCP の設定

ネイバーの differentiated services code point（DSCP）を設定します。IPv4 または IPv6 のローカル発信パケットの DSCP 値を指定できます。

DSCP 値を設定するには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **dscp** *dscp_value*

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**dscp** *dscp_value*
	例：
	`switch(config-router-neighbor)# dscp 63`
	次に、対応する **show** コマンドの例を示します。
	`show ipv6 bgp neighbors`
	`BGP neighbor is 10.1.1.1, remote AS 0, unknown`
		ネイバーの Differentiated Services Code Point（DSCP）の値を設定します。DSCP 値には、0 ～ 63 の数字、または、**ef**、**af11**、**af12**、**af13**、**af21**、**af22**、**af23**、**af31**、**af32**、**af33**、**af41**、**af42**、**af43**、**cs1**、**cs2**、**cs3**、**cs4**、**cs5**、**cs6**、または **cs7** のいずれかのキーワードを指定できます。
		デフォルト値は cs6 です。

	コマンドまたはアクション	目的
	link, Peer index 4 BGP version 4, remote router ID 0.0.0.0 BGP state = Idle, down for 00:13:34, retry in 0.000000 DSCP (DiffServ CodePoint): 0 Last read never, hold time = 180, keepalive interval is 60 seconds	

ダイナミック機能の設定

BGP ピアのダイナミック機能を設定できます。

ダイナミック機能を設定するには、ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **dynamic-capability**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**dynamic-capability** 例： `switch(config-router-neighbor)# dynamic-capability`	ダイナミック機能をイネーブルにします。このコマンドによって、BGP ネイバーセッションの自動通知およびセッション リセットが開始されます。

集約アドレスの設定

BGP ルート テーブルの集約アドレス エントリを設定できます。

集約アドレスを設定するには、ルータ アドレス ファミリ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **aggregate-address** *ip-prefix/length* [**as-set**] [**summary-only**] [**advertise-map** *map-name*] [**attribute-map** *map-name*] [**suppress-map** *map-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**aggregate-address** *ip-prefix/length* [**as-set**] [**summary-only**] [**advertise-map** *map-name*] [**attribute-map** *map-name*] [**suppress-map** *map-name*]	集約アドレスを作成します。このルートに関してアドバタイズされるパスは、集約されているすべての

	コマンドまたはアクション	目的
	例： `switch(config-router-af)# aggregate-address 192.0.2.0/8 as-set`	パスに含まれるすべての要素からなる、自律システム セットです。 • **as-set** キーワードは、関係するパスから自律システム セット パス情報およびコミュニティ情報を生成します。 • **summary-only** キーワードは、アップデートから具体的なルートをすべてフィルタリングします。 • **advertise-map** キーワードおよび引数では、選択されたルートから属性情報を選択するためのルート マップを指定します。 • **attribute-map** キーワードおよび引数では、集約から属性情報を選択するためのルートマップを指定します。 • **suppress-map** キーワードおよび引数によって、固有性の強いルートを条件付きでフィルタリングします。BGP ルート集約の実行中に **suppress-map** オプションを指定すると、BGP ルート更新のコミュニティ属性を設定できます。このオプションを使用すると、より具体的なルートにコミュニティ属性を設定できます。 • **suppress-map** キーワードおよび引数によって、固有性の強いルートを条件付きでフィルタリングします。BGPルート集約の実行中に **suppress-map** オプションを指定すると、特定のより具体的なルートがピアにアドバタイズされないように抑制したり、suppress-map route-map 設定に応じて、いくつかのコミュニティ属性が設定されたより具体的なルートをアドバタイズしたりすることができます。match 句だけで設定されたルートマップは、一致基準を満たすより具体的なルートを抑制します。ただし、ルートマップが match および set 句で設定されている場合、一致基準を満たすルートは、ルートマップによって変更された適切な属性でアドバタイズされます。2 番目のオプションでは、より具体的なルートにコミュニティ属性を設定できます。

BGP ルートの抑制

新しく学習されたBGPルートが転送情報ベース（FIB）により確認され、ハードウェアでプログラミングされた後にのみ、これらのルートをアドバタイズするようにCisco NX-OSを設定できます。ルートがプログラミングされた後は、これらのルートに対する以降の変更にはこのハードウェア プログラミングのチェックは必要ありません。

BGP ルートを抑制するには、ルータ コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **suppress-fib-pending**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**suppress-fib-pending** 例： `switch(config-router)# suppress-fib-pending`	新しく学習された BGP ルート（IPv4 または IPv6）がハードウェアでプログラミングされるまで、ダウンストリームのBGPネイバーにアドバタイズされることを抑制します。

BGP 条件付きアドバタイズメントの設定

BGP がプロパゲートするルートを制限するように BGP 条件付きアドバタイズメントを設定できます。次の 2 つのルート マップを定義します。

- アドバタイズ マップ：BGP が条件付きアドバタイズメントを考慮する前にルートが一致する必要のある条件を指定します。このルートマップには、適切なmatch文を含めることができます。
- 存在マップまたは非存在マップ：BGP がアドバタイズ マップに一致するルートをプロパゲートする前に BGP テーブルに存在する必要のあるプレフィックスを定義します。非存在マップは、BGP がアドバタイズ マップに一致するルートをプロパゲートする前に BGP テーブルに存在してはならないプレフィックスを定義します。BGP は、これらのルートマップでプレフィックス リストの match 文内にある permit 文のみを処理します。

ルートが条件を渡さない場合、そのルートがBGPテーブルにあればBGPによってルートが取り消されます。

始める前に

BGP を有効にする必要があります（「BGP の有効化」のセクションを参照）。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **neighbor** *ip-address* **remote-as** *as-number*
4. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
5. **advertise-map** *adv-map* {**exist-map** *exist-rmap*|**non-exist-map** *nonexist-rmap*}
6. （任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} **neighbors**
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		コンフィギュレーション モードに入ります。

ステップ 2	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 65534`
	`switch(config-router-neighbor)#`
		BGP ルーティング用のネイバー設定モードを開始し、ネイバー IP アドレスを設定します。

ステップ 4	**address-family** {**ipv4**	**ipv6**} {**unicast**	**multicast**}
			例：
			`switch(config-router-neighbor)# address-family ipv4 multicast`
			`switch(config-router-neighbor-af)#`
				アドレス ファミリ設定モードを開始します。

ステップ 5	**advertise-map** *adv-map* {**exist-map** *exist-rmap*|**non-exist-map** *nonexist-rmap*} 例： `switch(config-router-neighbor-af)# advertise-map advertise exist-map exist`	2 つの設定済みルート マップに従い、ルートを条件付きでアドバタイズするように BGP を設定します。 • *adv-map*：BGP がルートを次のルート マップに渡す前に、そのルートが渡す必要のある **match** 文を含むルート マップを指定します。*adv-map* には最大 63 文字の英数字を使用できます。大文字と小文字は区別されます。 • *exist-rmap*：プレフィックス リストの match ステートメントを使用してルートマップを指定します。BGP テーブル内のプレフィックスは、BGP がルートをアドバタイズする前に、プレ

	コマンドまたはアクション	目的
		フィックスリスト内のプレフィックスと一致する必要があります。*exist-rmap* には最大 63 文字の英数字を使用できます。大文字と小文字は区別されます。 • *nonexist-rmap*：プレフィックス リストの match ステートメントを使用してルートマップを指定します。BGP テーブル内のプレフィックスは、BGP がルートをアドバタイズする前に、プレフィックスリスト内のプレフィックスと一致してはいけません。*nonexist-rmap* には最大 63 文字の英数字を使用できます。大文字と小文字は区別されます。 （注） BGP 条件付きアドバタイズメント機能の場合、exist マップまたは nonexist マップに関連付けられている場合、プレフィックスリストで「le」または「ge」ステートメントが使用されていないことを確認します。
ステップ 6	（任意） **show bgp** {**ipv4** | **ipv6**} {**unicast** | **multicast**} **neighbors** 例： `switch(config-router-neighbor-af)# show ip bgp neighbor`	BGP に関する情報、および設定した条件付きアドバタイズメントのルートマップに関する情報を表示します。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。

例

次に、BGP 条件付きアドバタイズメントを設定する例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# neighbor 192.0.2.2 remote-as 65537
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# advertise-map advertise exist-map exist
switch(config-router-neighbor-af)# exit
switch(config-router-neighbor)# exit
switch(config-router)# exit
switch(config)# route-map advertise
switch(config-route-map)# match as-path pathList
switch(config-route-map)# exit
```

```
switch(config)# route-map exit
switch(config-route-map)# match ip address prefix-list plist
switch(config-route-map)# exit
switch(config)# ip prefix-list plist permit 209.165.201.0/27
```

ルートの再配布の設定

別のルーティング プロトコルからのルーティング情報を受け入れて、BGP ネットワークを通じてその情報を再配布するように、BGP を設定できます。任意で、再配布ルートのためのデフォルト ルートを割り当てることができます。

始める前に

BGPを有効にする必要があります。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **address-family** {**ipv4** | **ipv6** } {**unicast** | **multicast**}
4. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
5. **redistribute** {**direct** | {**eigrp** | **isis** | **ospf** | **ospfv3** | **rip**} *instance-tag* | **static** | **icmpv6**} **route-map** *map-name*
6. （任意） **default-metric** *value*
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**address-family** {**ipv4**	**ipv6** } {**unicast**	**multicast**}
			例：
			`switch(config-router)# address-family vpnv4 unicast`
			`switch(config-router-af)#`
				アドレス ファミリ 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 4	**address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
	例：
	`switch(config-router)# address-family ipv4 unicast`
	`switch(config-router-af)#`
		アドレスファミリ コンフィギュレーション モードに入ります。

ステップ 5	**redistribute** {**direct** | {**eigrp** | **isis** | **ospf** | **ospfv3** | **rip**} *instance-tag* | **static** | **icmpv6**} **route-map** *map-name* 例： `switch(config-router-af)# redistribute eigrp 201 route-map Eigrpmap`	他のプロトコルからのルートを BGP に再配布します。 Cisco NX-OS リリース 10.3(3)F 以降では、icmpv6 ルートを他のプロトコルから BGP に再配布するために **icmpv6** キーワードがサポートされています。
ステップ 6	（任意） **default-metric** *value* 例： `switch(config-router-af)# default-metric 33`	BGP へのデフォルト ルートを生成します。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-router-af)# copy running-config startup-config`	この設定変更を保存します。

例

次に、EIGRP を BGP に再配布する例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# redistribute eigrp 201 route-map Eigrpmap
switch(config-router-af)# copy running-config startup-config
```

デフォルト ルートのアドバタイズ

デフォルトのルート（ネットワーク 0.0.0.0）をアドバタイズするように BGP を設定できます。

始める前に

BGP をイネーブルにする必要があります（「BGP のイネーブル化」の項を参照）。

手順の概要

1. **configure terminal**
2. **route-map allow permit**

3. **exit**
4. **ip route** *ip-address network-mask* **null** *null-interface-number*
5. **router bgp** *as-number*
6. **address-family** {**ipv4** | **ipv6**} **unicast**
7. **default-information originate**
8. **redistribute static route-map allow**
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**route-map allow permit**
	例：
	`switch(config)# route-map allow permit`
	`switch(config-route-map)#`
		ルータのマップ コンフィギュレーション モードを開始し、ルートを再配布する条件を定義します。。

ステップ 3	**exit**
	例：
	`switch(config-route-map)# exit`
	`switch(config)#`
		ルータのマップ設定モードを終了します。

ステップ 4	**ip route** *ip-address network-mask* **null** *null-interface-number* 例： `switch(config)# ip route 192.0.2.1 255.255.255.0 null 0`	IP アドレスを設定します。

ステップ 5	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP モードを開始し、AS 番号をローカルの BGP スピーカに割り当てます。

ステップ 6	**address-family** {**ipv4**	**ipv6**} **unicast**
		例：
		`switch(config-router)# address-family ipv4 unicast`
		`switch(config-router-af)#`
			アドレスファミリ設定モードに入ります。

ステップ 7	**default-information originate** 例： `switch(config-router-af)# default-information originate`	デフォルトのルートをアドバタイズします。

	コマンドまたはアクション	目的
ステップ 8	**redistribute static route-map allow** 例： `switch(config-router-af)# redistribute static route-map allow`	デフォルトのルートを再配布します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config-router-af)# copy running-config startup-config`	この設定変更を保存します。

BGP 属性フィルタリングの設定とエラー処理

Cisco NX-OS リリース 9.3(3) 以降では、BGP属性フィルタリングとエラー処理を設定して、セキュリティ レベルを向上させることができます。次の機能を利用でき、次の順序で実装されます。

- **パス属性 treat-as-withdraw:** アップデートに指定した属性タイプが含まれている場合に、指定したネイバーから受け取った BGP アップデートを treat-as-withdraw とすることを許可します。アップデートに含まれるプレフィックスは、ルーティング テーブルから削除されます。
- **パス属性 discard:**BGP アップデートの特定のパス属性を特定のネイバーから削除できます。
- **拡張属性エラー処理:** 形式が誤っているアップデートに起因するピア セッションのフラッピングを防止します。

属性タイプ 1、2、3、4、8、14、15、16 は、パス属性 treat-as-withdraw とパス属性 discard に対して設定できません。属性タイプ 9（Originator）、タイプ 10（Cluster-id）は、eBGP ネイバーでのみ設定できます。

BGP 更新メッセージからのパス属性の取り消しとしての処理

特定のパス属性を含むBGP更新を「扱うように」処理するには、ルータネイバーコンフィギュレーションモードで次のコマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **path-attribute treat-as-withdraw** [*value* | **range** *start end*] **in** 例：	指定されたパス属性またはパス属性の範囲を含む着信BGP更新メッセージをすべて取り消すものとして扱い、ルーティングテーブルが最新であることを確認するために着信ルートリフレッシュをトリガーし

	コマンドまたはアクション	目的
	switch#(config-router)# neighbor 10.20.30.40 switch(config-router-neighbor)# path-attribute treat-as-withdraw 100 in 例： switch#(config-router)# neighbor 10.20.30.40 switch(config-router-neighbor)# path-attribute treat-as-withdraw range 21 255 in	ます。treat-as-withdraw である BGP 更新のプレフィックスは、BGP ルーティング テーブルから削除されます。 このコマンドは、BGP テンプレート ピアおよび BGP テンプレート ピア セッションでもサポートされます。

BGP 更新メッセージからのパス属性の破棄

特定のパス属性を含む BGP アップデートを廃棄するには、ルータ ネイバー コンフィギュレーション モードで次のコマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **path-attribute discard** [*value* | **range** *start end*] **in** 例： switch#(config-router)# neighbor 10.20.30.40 switch(config-router-neighbor)# path-attribute discard 100 in 例： switch#(config-router)# neighbor 10.20.30.40 switch(config-router-neighbor)# path-attribute discard range 100 255 in	指定されたネイバーの BGP アップデート メッセージ内の指定されたパス属性をドロップし、ルーティング テーブルが最新であることを確認するために着信ルート リフレッシュをトリガーします。特定の属性または不要な属性の範囲全体を設定できます。 このコマンドは、BGP テンプレート ピアおよび BGP テンプレート ピア セッションでもサポートされます。 （注） discard と treat-as-withdaw の両方に同じパス属性が設定されている場合、treat-as-withdaw の優先順位が高くなります。

拡張属性エラー処理のイネーブル化またはディセーブル化

BGP 拡張属性エラー処理はデフォルトで有効になっていますが、無効にすることもできます。この機能は、RFC 7606 に準拠しており、不正な更新によるピア セッションのフラッピングを防止します。デフォルトの動作は、eBGP ピアと iBGP ピアの両方に適用されます。

拡張エラー処理を無効または再度有効にするには、ルータ設定モードで次のコマンドを使用します。

手順

	コマンドまたはアクション	目的

ステップ 1	[**no**] **enhanced-error**
	例：
	`switch(config)# router bgp 1000`
	`switch(config-router)# enhanced-error`
		BGP 拡張属性エラー処理をいネーブルまたはディセーブルにします。

取り消されたパス属性または破棄されたパス属性の表示

廃棄または不明なパス属性に関する情報を表示するには、次のいずれかのタスクを実行します。

コマンド	目的
show bgp {**ipv4** | **ipv6**} **unicast path-attribute discard**]	属性が破棄されたすべてのプレフィックスを表示します。
show bgp {**ipv4** | **ipv6**} **unicast path-attribute unknown**]	不明な属性を持つすべてのプレフィックスを表示します。
show bgp {**ipv4** | **ipv6**} **unicast** *ip-address*	プレフィックスに関連付けられている不明な属性および破棄された属性を表示します。

次の例は、属性が廃棄されたプレフィックスを示しています。

```
switch# show bgp ipv4 unicast path-attribute discard
Network            Next Hop
1.1.1.1/32         20.1.1.1
1.1.1.2/32         20.1.1.1
1.1.1.3/32         20.1.1.1
```

次の例は、不明な属性を持つプレフィックスを示しています。

```
switch# show bgp ipv4 unicast path-attribute unknown
Network            Next Hop
2.2.2.2/32         20.1.1.1
2.2.2.3/32         20.1.1.1
```

次の例は、プレフィックスに関連付けられている不明な属性および破棄された属性を表示します。

```
switch# show bgp ipv4 unicast 2.2.2.2
BGP routing table entry for 2.2.2.2/32, version 6241
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 1
  1000
    20.1.1.1 from 20.1.1.1 (20.1.1.1)
      Origin IGP, localpref 100, valid, external, best
      unknown transitive attribute: flag 0xE0 type 0x62 length 0x64
```

```
  value 0000 0000 0100 0000 0200 0000 0300 0000
        0400 0000 0500 0000 0600 0000 0700 0000
        0800 0000 0900 0000 0A00 0000 0B00 0000
        0C00 0000 0D00 0000 0E00 0000 0F00 0000
        1000 0000 1100 0000 1200 0000 1300 0000
        1400 0000 1500 0000 1600 0000 1700 0000
        1800 0000
rx pathid: 0, tx pathid: 0x0
Updated on Jul 20 2019 07:50:43 PST
```

BGP の調整

一連のオプション パラメータを使用することによって、BGP 特性を調整できます。

BGP を調整するには、ルータ コンフィギュレーション モードで次のオプション コマンドを使用します。

コマンド	目的
bestpath [**always-compare-med** | **as-pathmultipath-relax** | **compare-routerid** |**cost-community ignore** | **igp-metric ignore** |**med** {**confed** |**missing-as-worst**| **non-deterministic**}] 例： `switch(config-router)# bestpath always-compare-med`	ベストパス アルゴリズムを変更します。オプション パラメータは次のとおりです。 • **always-compare-med** ：異なる自律システム（AS）からのパスの MED を比較します。 • **as-path multipath-relax** ：異なる（ただし長さが等しい）AS パスを持つプロバイダー間でのロード シェアリングを許可します。このオプションを指定しないと、ASパスはロードシェアリングの場合に同一である必要があります。 • **compare-routerid** ：同一の eBGP パスのルータ ID を比較します。 • **cost-community ignore** ：BGP ベストパス計算のコストコミュニティを無視します。 • **igp-metric ignore** ：ベスト パス選択時に内部ゲートウェイ プロトコル（IGP）メトリックを無視します。このオプションは、Cisco NX-OS リリース 9.2(2) 以降で使用可能です。 • **med confed** ：コンフェデレーション内からのパス間のみでMEDを比較するように最適なパスを強制します。 • **med missing-as-worst** ：消失 MED を最高の MED と見なします。 • **med non-deterministic** ：同じ自律システムからのパスの中から最適なMEDパスを決して選択しません。
enforce-first-as 例： `switch(config-router)# enforce-first-as`	ネイバー自律システムを eBGP の AS_path 属性で指定する最初の AS 番号にします。

コマンド	目的
log-neighbor-changes 例： `switch(config-router)# log-neighbor-changes`	ネイバーでステートが変化したときに、システム メッセージを生成します。 （注） 特定のネイバーのネイバー ステータス変化に関するメッセージを抑制するには、ルータ アドレスファミリ コンフィギュレーション モードで **log-neighbor-changes disable** コマンドを使用できます。
router-id *id* 例： `switch(config-router)# router-id 10.165.20.1`	この BGP スピーカのルータ ID を手動で設定します。
timers [*prefix-peer-wait* | *bgp holdtime* | **prefix-peer-timeout** *timeout* | **bestpath-limit** *bestpath-timeout*] 例： `switch(config-router)# timers bestpath-limit 300`	BGP タイマー値を設定します。オプション パラメータは次のとおりです。 • *prefix-peer-wait* ：プレフィックス ピアの待機タイマー。有効な範囲は 0 ～ 1200 秒です。デフォルトは 90 です。 • *bgp* ：BGP セッション キープアライブ時間。有効な範囲は 0 ～ 3600 秒です。デフォルト値は 60 です。 • *holdtime* ：異なる bgp キープアライブとホールド時間。範囲は 0 ～ 3600 秒で、デフォルト値は 60 秒です。 • *timeout* ：プレフィックス ピア タイムアウト値。有効な範囲は 0 ～ 1200 秒です。デフォルト値は 30 です。 • *bestpath-timeout* ：ベストパス タイムアウトを秒単位で設定します。デフォルト値は 300 です。大規模な BGP セットアップが予想される場合、スケールに基づいて、タイムアウト値を 480～1200 に設定する必要があります。 このコマンドの設定後、BGP セッションを手動でリセットする必要があります。

BGP を調整するには、ルータ アドレス ファミリ設定モードで次のオプション コマンドを使用します。

コマンド	目的
distance *ebgp-distance ibgp-distance local-distance* **例：** `switch(config-router-af)# distance 20 100 200`	BGP のアドミニストレーティブディスタンスを設定します。範囲は 1 ～ 255 です。デフォルトの設定は次のとおりです。 • *ebgp-distance* —20 • *ibgp-distance* —200 • *local-distance* —220 ローカル ディスタンスは、集約廃棄ルートが RIB に組み込まれている場合に、集約廃棄ルートに使用するアドミニストレーティブディスタンスです。 外部アドミニストレーティブディスタンスの値を入力したら、要件に応じて内部ルートのアドミニストレーティブディスタンスの値またはローカルルートのアドミニストレーティブディスタンスの値を入力する必要があります。内部/ローカルルートもルート管理で考慮されます。
log-neighbor-changes [**disable**] **例：** `switch(config-router-af)# log-neighbor-changes disable`	この特定のネイバーの状態が変化すると、システム メッセージを生成します。 **disable** オプションを使用すると、この特定のネイバーのネイバー ステータス変化に関するメッセージが抑制されます。

BGP を調整するには、ネイバー コンフィギュレーション モードで次のオプション コマンドを使用します。

コマンド	目的
description *string* **例：** `switch(config-router-neighbor)# description main site`	この BGP ピアを説明するストリングを設定します。ストリングには最大 80 の英数字を使用できます。
low-memory exempt **例：** `switch(config-router-neighbor)# low-memory exempt`	メモリ不足状態によるシャットダウンからこの BGP ネイバーを除外します。

コマンド	目的
transport connection-mode passive 例： `switch(config-router-neighbor)# transport connection-mode passive`	受動接続の確立だけが可能です。この BGP スピーカは BGP ピアへの TCP 接続を開始しません。このコマンドの設定後、BGP セッションを手動でリセットする必要があります。
[**no** | **default**] **remove-private-as** [**all** | **replace-as**] 例： `switch(config-router-neighbor)# remove-private-as`	eBGP ピアへの発信ルート アップデートからプライベート AS 番号を削除します。このコマンドによって、BGP ネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。 オプション パラメータは次のとおりです。 • **no** ：コマンドをディセーブルにします。 • **default** ：デフォルト モードにコマンドを移動します。 • **all** ：AS パスからすべてのプライベート AS 番号を削除します。 • **replace-as** ：すべてのプライベート AS 番号を replace-as AS-path 値に置き換えます。 このコマンドの詳細については、拡張 BGP に関する注意事項と制限事項 （17 ページ） を参照してください。
update-source *interface-type number* 例： `switch(config-router-neighbor)# update-source ethernet 2/1`	ピアとの BGP セッション用に設定されたインターフェイスの送信元 IP アドレスを使用するように、BGP スピーカを設定します。このコマンドによって、BGP ネイバーセッションの自動通知およびセッション リセットが開始されます。単一ホップ iBGP ピアでは、**update-source** が設定されている場合に、高速外部フォールオーバーをサポートします。

BGP を調整するには、ネイバー アドレス ファミリ コンフィギュレーション モードで次のオプション コマンドを使用します。

コマンド	目的
allowas in 例： `switch(config-router-neighbor-af)# allowas in`	BRIP にインストールする AS パスにルート自体の AS を持つことを可能にします。

コマンド	目的
default-originate [**route-map** *map-name*] 例： `switch(config-router-neighbor-af)# default-originate`	BGP ピアへのデフォルト ルートを作成します。
disable-peer-as-check 例： `switch(config-router-neighbor-af)# disable-peer-as-check`	デバイスが同じASパスで一方のノードからもう一方のノードに学習されたルートをアドバタイズすると同時に、ピアAS番号のチェックをディセーブルにします。
filter-list *list-name* {**in** | **out**} 例： `switch(config-router-neighbor-af)# filter-list BGPFilter in`	着信または発信ルートアップデートに関して、このBGPピアにAS_pathフィルタリストを適用します。このコマンドによって、BGP ネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。
prefix-list *list-name* {**in** | **out**} 例： `switch(config-router-neighbor-af)# prefix-list PrefixFilter in`	着信または発信ルートアップデートに関して、このBGPピアにプレフィックスリストを適用します。このコマンドによって、BGPネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。
send-community 例： `switch(config-router-neighbor-af)# send-community`	このBGPピアにコミュニティ属性を送信します。このコマンドによって、BGP ネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。
send-community extended 例： `switch(config-router-neighbor-af)# send-community extended`	このBGPピアに拡張コミュニティ属性を送信します。このコマンドによって、BGPネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。
suppress-inactive 例： `switch(config-router-neighbor-af)# suppress-inactive`	ベスト（アクティブ）ルートだけをBGPピアにアドバタイズします。このコマンドによって、BGPネイバーセッションの自動ソフトクリアまたはリフレッシュが開始されます。
[no | default] as-override 例： `switch(config-router-neighbor-af)# as-override`	**no**-（オプション）コマンドを無効にします。 **default**：（オプション）デフォルトモードにコマンドを移動します。 **as-override**：eBGP ピアに更新を送信する際に、パス属性内のピアのAS番号をすべてローカルAS 番号に置き換えます。

ポリシーベースのアドミニストレーティブディスタンスの設定

設定されたルートマップで説明されているポリシーに一致する外部BGP（eBGP）と内部BGP（iBGP）の距離を設定できます。ルート マップで設定された距離は、一致するルートとともにユニキャストRIBにダウンロードされます。BGPは最適パスを使用して、ユニキャストRIBテーブルのネクスト ホップをダウンロードするときのアドミニストレーティブ ディスタンスを決定します。ポリシーに match 句または deny 句がない場合、BGP は distance コマンドで設定された距離またはルートのデフォルトの距離を使用します。

ポリシーベースのアドミニストレーティブディスタンス機能は、2つの異なるルーティングプロトコルから同じ宛先に 2 つ以上のルートが存在する場合に役立ちます。

始める前に

BGP を有効にする必要があります。

手順の概要

1. switch# **configure terminal**
2. switch(config)# **ip prefix-list** *name* **seq** *number* **permit** *prefix-length*
3. switch(config)# **route-map** *map-tag* **permit** *sequence-number*
4. switch(config-route-map)# **match ip address prefix-list** *prefix-list-name*
5. switch(config-route-map)# **set distance** *value1 value2 value3*
6. switch(config-route-map)# **exit**
7. switch(config)# **router bgp** *as-number*
8. switch(config-router)# **address-family** {**ipv4** | **ipv6** | **vpnv4** | **vpnv6**} **unicast**
9. switch(config-router-af)# **table-map** *map-name*
10. （任意） switch(config-router-af)# **show forwarding distribution**
11. （任意） switch(config)# **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **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 2	switch(config)# **ip prefix-list** *name* **seq** *number* **permit** *prefix-length*	permit キーワードを使用して、IP パケットまたはルートを照合するためのプレフィクス リストを作成します。
ステップ 3	switch(config)# **route-map** *map-tag* **permit** *sequence-number*	permit キーワードを使用してルート マップを作成し、ルート マップ コンフィギュレーション モードを開始します。ルートの一致基準がポリシー内で満

	コマンドまたはアクション	目的
		たされると、パケットはポリシーでルーティングされます。
ステップ 4	switch(config-route-map)# **match ip address prefix-list** *prefix-list-name*	プレフィクス リストに基づいて IPv4 ネットワーク ルートを照合します。プレフィクス リスト名には最大 63 文字の英数字を使用できます。
ステップ 5	switch(config-route-map)# **set distance** *value1 value2 value3*	ローカル自律システムから発信される内部 BGP（iBGP）または外部 BGP （eBGP）ルートおよび BGP ルートのアドミニストレーティブ ディスタンスを指定します。範囲は 1 ～ 255 です。 外部アドミニストレーティブ ディスタンスの値を入力したら、要件に応じて内部ルートのアドミニストレーティブディスタンスの値またはローカルルートのアドミニストレーティブ ディスタンスの値を入力する必要があります。内部/ローカル ルートもルート管理で考慮されます。
ステップ 6	switch(config-route-map)# **exit**	ルート マップ設定モードを終了します。
ステップ 7	switch(config)# **router bgp** *as-number*	BGP モードを開始し、AS 番号をローカルの BGP スピーカに割り当てます。
ステップ 8	switch(config-router)# **address-family** {**ipv4** | **ipv6** | **vpnv4** | **vpnv6**} **unicast**	アドレス ファミリ設定モードを開始します。
ステップ 9	switch(config-router-af)# **table-map** *map-name*	BGP ルートを RIB テーブルに転送する前にそのルートのルート マップの選択的アドミニストレーティブ ディスタンスを設定します。テーブル マップ名には最大 63 文字の英数字を使用できます。 （注） VRF アドレスファミリ設定モードで **table-map** コマンドを設定することもできます。
ステップ 10	（任意） switch(config-router-af)# **show forwarding distribution**	フォワーディング情報の配布を表示します。
ステップ 11	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

マルチプロトコル BGP の設定

複数のアドレス ファミリ（IPv4 および IPv6 のユニキャストおよびマルチキャスト ルートを含む）をサポートするように MP-BGP を設定できます。

始める前に

BGPをイネーブルにする必要があります。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. **neighbor** *ip-address* **remote-as** *as-number*
4. **address-family** {**ipv4** | **ipv6**} {**unicast** | **multicast**}
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。

ステップ 3	**neighbor** *ip-address* **remote-as** *as-number*
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 65534`
	`switch(config-router-neighbor)#`
		BGP ルーティング用のネイバー設定モードを開始し、ネイバー IP アドレスを設定します。

ステップ 4	**address-family** {**ipv4**	**ipv6**} {**unicast**	**multicast**}
			例：
			`switch(config-router-neighbor)# address-family ipv4 multicast`
			`switch(config-router-neighbor-af)#`
				アドレス ファミリ コンフィギュレーション モードを開始します。

ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor-af)# copy running-config startup-config`	この設定変更を保存します。

例

次に、ネイバーのマルチキャスト RPF に対して IPv4 および IPv6 ルートのアドバタイズおよび受信をイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 2/1
switch(config-if)# ipv6 address 2001:0DB8::1
switch(config-if)# router bgp 65536
switch(config-router)# neighbor 192.168.1.2 remote-as 35537
switch(config-router-neighbor)# address-family ipv4 multicast
switch(config-router-neighbor-af)# exit
switch(config-router-neighbor)# address-family ipv6 multicast
switch(config-router-neighbor-af)# copy running-config startup-config
```

BMP の設定

Cisco NX-OS リリース 7.0(3)I5(2) 以降では、デバイスに BMP を設定できます。

始める前に

BGP をイネーブルにする必要があります（「BGP のイネーブル化」の項を参照）。

手順の概要

1. **configure terminal**
2. **router bgp as-number**
3. **bmp server** *server-number*
4. **address ip-address port-number port-number**
5. **description** *string*
6. **initial-refresh** { *skip* | *delay time*}
7. **initial-delay** *time*
8. **stats-reporting-period** *time*
9. shutdown
10. **vrf** *vrf-name*
11. **update-source** <*interface-name*>
12. **neighbor ip-address**
13. **remote-as** *as-number*
14. **bmp-activate-server** *server-number*
15. （任意） **show bgp bmp** *server [server-number] [detail]*
16. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# 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 2	**router bgp as-number** 例： `switch(config)# router bgp 200`	BGP モードを開始し、ローカル BGP スピーカに自律システム番号を割り当てます。
ステップ 3	**bmp server** *server-number* 例： `switch(config-router-bmp)# bmp-server 1`	BGP が情報を送信する BMP サーバを設定します。サーバ番号がキーとして使用されます。 （注） 最大 2 つの BMP サーバを設定できます。
ステップ 4	**address ip-address port-number port-number** 例： `switch(config-router-bmp)# address 10.1.1.1 port-number 2000`	ホストの IPv4 または IPv6 アドレスと、BMP スピーカーが BMP サーバに接続するポート番号を設定します。
ステップ 5	**description** *string* 例： `switch(config-router-bmp)# description BMPserver1`	BMP サーバの説明を設定します。最大 256 文字の英数字を入力できます。
ステップ 6	**initial-refresh** { *skip* | *delay time*} 例： `switch(config-router-bmp)# initial-refresh delay 100`	BGP がコンバージされ、後で BMP サーバ接続が確立されたときにルート リフレッシュを送信するオプションを設定します。 skip オプションは、BMP サーバ接続が後でアップした場合にルート リフレッシュを送信しないことを指定します。 delay オプションは、ルート更新を送信するまでの時間を秒単位で指定します。有効範囲は 30 ～ 720 秒で、デフォルトは 30 秒です。
ステップ 7	**initial-delay** *time* 例： `switch(config-router-bmp)# initial-delay 120`	BMP サーバへの接続が試行されるまでの遅延を設定します。有効範囲は 30 ～ 720 秒で、デフォルトは 45 秒です。
ステップ 8	**stats-reporting-period** *time* 例： `switch(config-router-bmp)# stats-reporting-period 50`	BMP サーバが BGP ネイバーから統計レポートを受信する時間間隔を設定します。有効範囲は 30 ～ 720 秒で、デフォルトはディスエーブルです。

	コマンドまたはアクション	目的
ステップ 9	shutdown 例： `switch(config-router-bmp)# shutdown`	BMP サーバへの接続を無効にします。
ステップ 10	**vrf** *vrf-name* 例： `switch(config-router-bmp)# vrf BMP`	BMP サーバーが到達可能な VRF を選択します。
ステップ 11	**update-source** <*interface-name*> 例： `switch(config-router-bmp)# update-source ethernet4/2`	BMP サーバー接続の確立に使用するローカル インターフェイスを選択します。
ステップ 12	**neighbor ip-address** 例： `switch(config-router-bmp)# neighbor 192.168.1.2`	BGP ルーティング用のネイバー コンフィギュレーション モードを開始し、ネイバー IP アドレスを設定します。
ステップ 13	**remote-as** *as-number* 例： `switch(config-router-neighbor)# remote-as 65535`	リモート BGP ピアの AS 番号を設定します。
ステップ 14	**bmp-activate-server** *server-number* 例： `switch(config-router-neighbor)# bmp-activate-server 1`	ネイバーの情報の送信先となる BMP サーバを設定します。
ステップ 15	（任意） **show bgp bmp** *server [server-number] [detail]* 例： `switch(config-router-neighbor)# show bgp bmp server`	BMP サーバ情報を表示します。
ステップ 16	（任意） **copy running-config startup-config** 例： `switch(config-router-neighbor)# copy running-config startup-config`	この設定変更を保存します。

BGP ローカル ルート リーク

BGP ローカル ルート リークについて

リリース 9.3(1) 以降、NX-OS BGP は、次の間のインポートされた VPN ルートのリークをサポートします。

- VPN ルート テーブルとデフォルト VRF ルート テーブル
- VPN ルート テーブルと VRF-Lite ルート テーブル
- リーフからリーフへの接続用のボーダー リーフ（BL）スイッチ ルート テーブル

この機能により、ルート テーブル間のルートの伝播が可能になります。インポート マップまたはエクスポート マップを設定することで、VRF のルート リークを制御できます。このマップには、ローカルで発生した着信ルートを許可または禁止し、アドバタイズするかどうかを指定するオプションが含まれています。ローカル ルート リークは双方向であるため、ローカルに発信されたルートはVRFからBGP VPNにリークされ、BGP VPNからインポートされたルートは VRF にリークされます。

（注） NX-OS は、中央集中型ルート リークと呼ばれる同様の機能をサポートしています。詳細については、レイヤ 3 仮想化の設定を参照してください。

BGP ローカル ルート リークの注意事項と制約事項

BGPローカルルートリーク機能の注意事項と制約事項は次のとおりです。

- この機能は、次のシスコ ハードウェアによりサポートされます。
  - この機能は、Cisco Nexus 9332C、9364C、9300-EX、9300-FX/FXP/FX2/FX3、および 9300-GX プラットフォーム スイッチと、9700-EX/FX ラインカードを備えた Cisco Nexus 9500 プラットフォーム スイッチに導入されました。
  - -R ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ
- ルート ターゲットを使用する場合、同じルート ターゲットが同じリモート パスを指す重複パスを持っている可能性があり、これがスイッチのメモリとパフォーマンスに悪影響を及ぼす可能性があります。ルート ターゲットを使用する場合は注意してください。
- 同じ VRF 間で境界リーフ ルータ（BL）がリークするリーフツーリーフの場合に、ローカル ルート リークを使用する場合は注意してください。このシナリオでは、ルーティング ループが発生しやすくなります。インポートされたルートを他の BL から除外するには、インバウンド ルート マップを使用することを推奨します。
- リモート パスが取り消された後、BGP がパスを完全にクリーンアップするまでにさらに 20 秒かかることがあります。

デフォルトVRFにリークするためにVPNからインポートされたルートを設定する

VRF を設定して、BGP VPN からインポートされたルートが、デフォルトの VRF へエクスポートされることを許可することができます。この手順は、デフォルト以外のVRFに使用します。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **config terminal**
2. **vrf context** *vrf-name*
3. **address-family** *address-family sub family*
4. **export vrf default [***prefix-limit***] map***route-map* **allow-vpn**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`
		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 2	**vrf context** *vrf-name*
	例：
	`switch-1(config)# vrf context vpn1`
	`switch-1(config-vrf)#`
		新しい VRF を作成し、VRF 設定モードを開始します。32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

ステップ 3	**address-family** *address-family sub family*
	例：
	`switch-1(config-vrf)# address-family ipv4 unicast`
	`switch-1(config-vrf-af-ipv4)#`	

ステップ 4	**export vrf default [***prefix-limit***] map***route-map* **allow-vpn**
	例：
	`switch-1(config-vrf-af-ipv4)# export vrf default map vpnmap1 allow-vpn`
	`switch-1(config-vrf-af-ipv4)#`
		現在の VRF を設定して、BGP VPN からインポートされたルートが、デフォルトの VRF へエクスポートされることを許可します。

デフォルト VRF からリークされたルートを VPN にエクスポートするための設定

デフォルト VRF からリークされたルートを BGP VPN にエクスポートできるように VRF を設定できます。この手順は、デフォルト以外の VRF に使用します。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **config terminal**
2. **vrf context** *vrf-name*
3. **address-family** *address-family sub family*
4. **import vrf default [***prefix-limt***] map***route-map* **advertise-vpn**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`
		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 2	**vrf context** *vrf-name*
	例：
	`switch-1(config)# vrf context vpn1`
	`switch-1(config-vrf)#`
		新しい VRF を作成し、VRF 設定モードを開始します。32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

ステップ 3	**address-family** *address-family sub family*
	例：
	`switch-1(config-vrf)# address-family ipv4 unicast`
	`switch-1(config-vrf-af-ipv4)#`	

ステップ 4	**import vrf default [***prefix-limt***] map***route-map* **advertise-vpn**
	例：
	`switch-1(config-vrf-af-ipv4)# import vrf map vpnmap1 advertise-vpn`
	`switch-1(config-vrf-af-ipv4)#`
		デフォルト VRF からインポートされたルートを BGP VPN にエクスポートできるように現在の VRF を設定します。

VRF にエクスポートするために VPN からインポートしたルートの設定

VPN でインポートされたルートを別の VRF にエクスポートできるように VRF を設定できます。この手順は、デフォルト以外の VRF に使用してください。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **config terminal**
2. **vrf context** *vrf-name*

3. **address-family** *address-family sub family*
4. **export vrf allow-vpn**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`
		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 2	**vrf context** *vrf-name*
	例：
	`switch-1(config)# vrf context vpn1`
	`switch-1(config-vrf)#`
		新しい VRF を作成し、VRF 設定モードを開始します。32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

ステップ 3	**address-family** *address-family sub family*
	例：
	`switch-1(config-vrf)# address-family ipv4 unicast`
	`switch-1(config-vrf-af-ipv4)#`	

ステップ 4	**export vrf allow-vpn**
	例：
	`switch-1(config-vrf-af-ipv4)# export vrf allow-vpn`
	`nxosv2(config-vrf-af-ipv4)#`
		BGP VPM からインポートしたルートをデフォルト以外の VRF にエクスポートできるように VRF を設定します。

VRF からインポートして VPN にエクスポートするルートの設定

VRF は、別の VRF からインポートされたルートを BGP VPN にエクスポートできるように設定することができます。この手順は、デフォルト以外の VRF に使用してください。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **config terminal**
2. **vrf context** *vrf-name*
3. **address-family** *address-family sub family*
4. **import vrf advertise-vpn**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# config terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch-1(config)#`
		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 2	**vrf context** *vrf-name*
	例：
	`switch-1(config)# vrf context vpn1`
	`switch-1(config-vrf)#`
		新しい VRF を作成し、VRF 設定モードを開始します。32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

ステップ 3	**address-family** *address-family sub family*
	例：
	`switch-1(config-vrf)# address-family ipv4 unicast`
	`switch-1(config-vrf-af-ipv4)#`	

ステップ 4	**import vrf advertise-vpn**
	例：
	`switch-1(config-vrf-af-ipv4)# import vrf advertise-vpn`
	`nxosv2(config-vrf-af-ipv4)#`
		別の VRF からインポートされたルートを BGP VPN にエクスポートできるように現在の VRF を設定します。

設定例

次に、BGP ローカル ルート リーク機能の設定例を示します。

BGP VPN からデフォルト VPN への到達可能性の設定

この例では、VPN とデフォルト VRF の間にある、VRF_A と呼ばれる中間 VRF を介して、ルートの再インポートを有効にします。

```
vrf context VRF_A
   address-family ipv4 unicast
   route-target both auto evpn
   import vrf default map MAP_1 advertise-vpn
   export vrf default map MAP_1 allow-vpn
```

ルートの再インポートは、VPN から VRF_A へのルートのインポートを制御する **advertise-vpn** オプションを使用して、また、VRF_A からデフォルト VRF への VPN インポート ルートのエクスポートを制御する、エクスポート マップのための **allow-vpn** を使用して有効にできます。設定は中間 VRF で行われます。

VPN から VRF-Lite への到達可能性の設定

この例では、VPN は VRF_A と呼ばれるテナント VRF に接続します。VRF_A は、VRF-B と呼ばれる VRF-Lite に接続します。この設定により、VPN でインポートされたルートを VRF_A から VRF_B にリークできます。

```
vrf context VRF_A
   address-family ipv4 unicast
   route-target both auto
   route-target both auto evpn
   route-target import 3:3
   route-target export 2:2
   import vrf advertise-vpn
   export vrf allow-vpn
vrf context VRF_B
   address-family ipv4 unicast
   route-target both 1:1
   route-target import 2:2
   route-target export 3:3
```

2 つの間のルート リークは、VRF_A（テナント）で設定されたエクスポート マップで **allow-vpn** を使用してイネーブルにします。VRF_A のエクスポート マップでは、VPN からインポートされたルートを VRF_B にリークできます。エクスポート マップによって処理されたルートは、ルート ターゲットのルート セットに追加される、**route-mapexport** および **export-map** 属性を持ちます。インポート マップは、**advertise-vpn** を使用して、VRF-Lite からインポートされたルートを VPN にエクスポートできるようにします。

VRF 間でルート リークが発生すると、ルートは再発信され、そのルート ターゲットは、新しい VRF の設定で指定されたルート ターゲット エクスポートおよびエクスポート マップ属性で置き換えられます。

リーフからリーフへの到達可能性

この例では、2 つの VPN と 2 つの VRF が存在します。VPN_1 は VRF_A に接続され、VPN_2 は VRF_B に接続されます。両方の VRF はルート識別子（RD）です。

```
vrf context VRF_A
   address-family ipv4 unicast
   route-target both auto
   route-target both auto evpn
   route-target import 3:3
   route-target export 2:2
   import vrf advertise-vpn
   export vrf allow-vpn
vrf context VRF_B
   address-family ipv4 unicast
   route-target both 1:1
   route-target import 2:2
   route-target export 3:3
   import vrf advertise-vpn
   export vrf allow-vpn
```

この 2 つの間のルート リークは、VRF_A および VRF_B で設定されたエクスポート マップの **allow-vpn** で有効にされます。VPN によってインポートされたルートには、ルート ターゲットのルート セットに追加された **route-mapexport** と **export-map** 属性があります。インポート マップのマップは、各 VRF からインポートされたルートが VPN にエクスポートされるようにする advertise-vpn オプションを使用します。

VRF 間でルート リークが発生すると、ルートは再発信され、そのルート ターゲットは、新しいVRF の設定で指定されたルート ターゲット エクスポートおよびエクスポート マップ属性で置き換えられます。

ループ防止付きリーフツーリーフ

リーフツーリーフ設定では、ルート マップに注意を払わないでいると、同じ VRF 間でリークしている BL 間のループが誤って発生する可能性があります。

- 各 BL でインバウンド ルート マップを使用すれば、他のすべての BL からの更新を拒否できます。
- BL がルートを発信する場合には、標準コミュニティを適用できます。これにより、他の BL はルートを受け入れることができます。このコミュニティは、受信側の BL で削除されます。

次の例では、VTEP 3.3.3.3、4.4.4.4、および 5.5.5.5 が BL です。

```
ip prefix-list BL_PREFIX_LIST seq 5 permit 3.3.3.3/32
ip prefix-list BL_PREFIX_LIST seq 10 permit 4.4.4.4/32
ip prefix-list BL_PREFIX_LIST seq 20 permit 5.5.5.5/32
ip community-list standard BL_COMMUNITY seq 10 permit 123:123
route-map INBOUND_MAP permit 5
  match community BL_COMMUNITY
  set community none
route-map INBOUND_MAP deny 10
  match ip next-hop prefix-list BL_PREFIX_LIST
route-map INBOUND_MAP permit 20
route-map OUTBOUND_SET_COMM permit 10
  match evpn route-type 2 mac-ip
  set community 123:123
route-map SET_COMM permit 10
  set community 123:123
route-map allow permit 10

vrf context vni100
  vni 100
  address-family ipv4 unicast
    route-target import 2:2
    route-target export 1:1
    route-target both auto
    route-target both auto evpn
    import vrf advertise-vpn
    export vrf allow-vpn

vrf context vni200
  vni 200
  address-family ipv4 unicast
    route-target import 1:1
    route-target export 2:2
    route-target both auto
    route-target both auto evpn
    import vrf advertise-vpn
    export vrf allow-vpn

router bgp 100
  template peer rr
    remote-as 100
    update-source loopback0
    address-family l2vpn evpn
```

```
      send-community
      send-community extended
      route-map INBOUND_MAP in
      route-map OUTBOUND_SET_COMM out
  neighbor 101.101.101.101
    inherit peer rr
  neighbor 102.102.102.102
    inherit peer rr
  vrf vni100
    address-family ipv4 unicast
      network 3.3.3.100/32 route-map SET_COMM
  vrf vni200
    address-family ipv4 unicast
      network 3.3.3.200/32 route-map SET_COMM
```

この例では、ボーダーリーフ（BL）ルータのテナント VRF は追加のインポート エクスポート フローを有効にすることで、トラフィックをリークできます。ルート マップ内のルート ターゲットは、ルートのインポート元またはエクスポート先を決定します。

VRF のマルチパス

この例では、VPN に複数の着信パスがあります。この設定により、VRF_A と呼ばれる中間 VRF（VPN と別の VRF の間にあり、VRF_B と呼ばれるもの）を介したルート リークが可能になります。マルチパスが VRF_A で有効になっているとします。

```
vrf context VRF_A
   address-family ipv4 unicast
   route-target both auto evpn
   route-target export 3:3
   export vrf allow-vpn
vrf context VRF_B
   address-family ipv4 unicast
   route-target import 3:3
```

ルートリークは、VRF_A で設定されたエクスポート マップの **allow-vpn** で有効になっています。特定のプレフィックスの 2 つのパスが VPN から学習されて VRF_A にインポートされると、同じ送信元 RD（VRF_A のローカル RD）を持つ 2 つの異なるパスが VRF_B に存在するようになります。各ルートは、元の送信元 RD（リモート RD）によって区別されます。

パスの重複

この例では、設定により単一の VPN パスを VRF_A と VRF_B の両方にインポートできるようになっています。VRF_A は **export vrf allow-vpn** で設定されているため、VRF_A もそのルートを VRF_B にリークします。VRF_B には同じ送信元 RD（VRF_A のローカル RD）を持つ 2 つのパスがありますが、それらは元の送信元 RD（リモート RD）によって区別されます。

```
vrf context VRF_A
   address-family ipv4 unicast
      route-target import 1:1 evpn
      route-target export 1:1 evpn
      route-target export 2:2
      export vrf allow-vpn
vrf context VRF_B
   address-family ipv4 unicast
     route-target import 1:1 evpn
     route-target import 2:2
```

この設定では、マルチパスが存在しない状況が発生します。

BGP ローカル ルート リーク情報の表示

次の show コマンドには、BGP ローカル ルート リーク機能に関する情報が含まれています。

コマンド	アクション
show bgp vrf *vrf-name* **process**	デフォルトまたはデフォルト以外のVRFの場合、**import advertise-vpn** および **export allow-vpn** オプションのイネーブル状態（YesまたはNo）が表示されます。
show bgp vrf *vrf-name* **ipv4 unicast** *prefix*	ルートのインポート元の宛先のリストなど、インポートされたパスに関する情報を表示します。

BGP グレースフル シャットダウン

BGP グレース フル シャットダウンに関する情報

リリース 9.3(1) 以降、BGP はグレースフル シャットダウン機能をサポートしています。この BGP 機能は、BGP **shutdown** コマンドと連携して次のことを行います。

- ルータまたはリンクがオフラインになったときのネットワーク コンバージェンス時間を大幅に短縮します。
- ルータまたはリンクがオフラインになったときに、転送中のドロップされたパケットを削減または排除します。

名前にかかわらず、BGP グレースフル シャットダウンは実際にはシャットダウンを引き起こしません。代わりに、ルータまたはリンクが間もなくダウンすることを、接続されているルータに通知します。

グレースフル シャットダウン機能は、GRACEFUL_SHUTDOWN ウェルノウン コミュニティ（0xFFFF0000 または 65535:0）を使用します。これは、IANA および IETF によって RFC 8326 によって識別されます。この既知のコミュニティは任意のルートにアタッチでき、ルートの他の属性と同様に処理されます。

この機能は、ルータまたはリンクがダウンすることを通知するため、メンテナンス時間帯または計画停止の準備に役立ちます。トラフィックへの影響を制限するには、BGP をシャットダウンする前にこの機能を使用します。

グレースフル シャットダウンの認識とアクティブ化

BGPルータは、すべてのルートの優先事項を、GRACEFUL SHUTDOWN 対応というコンセプトを通し、GRACEFUL_SHUTDOWNコミュニティによって制御できます。グレースフル シャッ

トダウン対応は、デフォルトでイネーブルになっています。これにより、受信側ピアは、GRACEFUL_SHUTDOWN コミュニティを伝える着信ルートを優先しなくなります。一般的な使用例ではありませんが、**graceful-shutdown aware** コマンドを使用して、グレースフル シャットダウン対応を無効にしてから再度有効にすることもできます。

グレースフル シャットダウン対応は、BGP グローバル コンテキストでのみ適用されます。コンテキストの詳細については、グレースフル シャットダウンのコンテキスト （95 ページ）を参照してください。対応のためのオプションは、**activate** という別のオプションと一緒に動作します。このオプションをルートマップに割り当てると、グレースフル シャットダウンのルートをより詳細に制御できます。

グレースフル シャットダウン対応オプションとアクティブ化オプションの協同作用

グレースフル シャットダウンがアクティブな場合、**activate** キーワードを指定した場合にのみ、GRACEFUL_SHUTDOWN コミュニティがルート更新に追加されます。この時点で、コミュニティを含む新しいルート更新が生成され、送信されます。**graceful-shutdown aware** コマンドが設定されると、コミュニティを受信するすべてのルータは、アップデート内のルートの優先を解除します（そのルート優先度を下げます）。**graceful-shutdown aware** コマンドを使用しなかった場合、BGPはGRACEFUL_SHUTDOWN コミュニティの設定されたルートの優先度を下げません。

この機能がアクティブになり、ルータがグレースフル シャットダウンの対応状態になった場合でも、BGP は引き続き、GRACEFUL_SHUTDOWN コミュニティが有効だとしてルートを考慮します。ただし、これらのルートには、最適パスの計算で最低の優先度が与えられます。代替パスが使用可能な場合は、新しい最適パスが選択され、まもなくダウンするルータまたはリンクに対応するためのコンバージェンスが行われます。

グレースフル シャットダウンのコンテキスト

BGPのグレースフルシャットダウン機能には、機能の影響と使用可能な機能を決定する2つのコンテキストがあります。

コンテキスト	影響	コマンド
グローバル	スイッチ全体と、スイッチによって処理されるすべてのルート。たとえば、GRACEFUL_SHUTDOWN コミュニティを持つすべてのルートを再アドバタイズします。	**graceful-shutdown activate [route-map** ルート マップ**]** **graceful-shutdown aware**

コンテキスト	影響	コマンド
Peer	BGP ピアまたはネイバー間のリンク。たとえば、ピア間のリンクを 1 つだけ GRACEFUL_SHUTDOWNコミュニティでアドバタイズします。	**graceful-shutdown activate [route-map** ルート マップ**]**

ルート マップによるグレースフル シャットダウン

グレースフル シャットダウンは、ルート ポリシー マネージャ（RPM）機能と連携して、スイッチの BGP ルータが GRACEFUL_SHUTDOWN コミュニティを使用してルートを送受信する方法を制御します。ルート マップは、インバウンドおよびアウトバウンド方向でコミュニティとのルート更新を処理できます。通常、ルートマップは必要ありません。ただし、必要に応じて、グレースフルシャットダウンルートの制御をカスタマイズするために使用できます。

通常のインバウンド ルート マップ

通常のインバウンド ルート マップは、BGP ルータに着信するルートに影響します。ルータはデフォルトでグレースフル シャットダウンを認識するため、通常のインバウンド ルート マップはグレースフル シャットダウン機能では一般的に使用されません。

Cisco NX-OS リリース 9.3 (1) 以降を実行している Cisco Nexus スイッチでは、グレースフル シャットダウン機能のインバウンドルートマップは必要ありません。Cisco NX-OSリリース9.3（1）以降には、BGPルータがグレースフルシャットダウン対応である場合にGRACEFUL_SHUTDOWNコミュニティを持つすべてのルートを自動的に非優先にする、暗黙のインバウンドルート マップがあります。

通常のインバウンド ルート マップは、既知の GRACEFUL_SHUTDOWN コミュニティと一致するように設定できます。これらの着信ルートマップは一般的ではありませんが、使用される場合があります。

- スイッチが 9.3 (1) よりも前の Cisco NX-OS リリースを実行している場合、NX-OS 9.3 (1) には暗黙的なインバウンド ルート マップがありません。これらのスイッチでグレースフル シャットダウン機能を使用するには、グレースフル シャットダウン インバウンド ルートマップを作成する必要があります。ルートマップは、既知の GRACEFUL_SHUTDOWN コミュニティを持つインバウンドルートと一致し、それらを許可し、それらを非優先にする必要があります。着信ルート マップが必要な場合は、9.3 (1) より前のバージョンの NX-OS を実行し、グレースフル シャットダウン ルートを受信しているBGPピアで作成します。
- グレースフル シャットダウン認識をディセーブルにし、一部の BGP ネイバーからの GRACEFUL_SHUTDOWN コミュニティを持つ着信ルートでルータを動作させる場合は、それぞれのピアでインバウンド ルート マップを設定できます。

通常のアウトバウンド ルート マップ

通常のアウトバウンド ルート マップは、BGP ルータが送信するルートの転送を制御します。通常のアウトバウンド ルート マップは、グレースフル シャットダウン機能に影響を与える可能性があります。たとえば、GRACEFUL_SHUTDOWN コミュニティで一致するようにアウトバウンド ルート マップを設定し、属性を設定できます。これは、グレースフル シャットダウン アウトバウンド ルート マップよりも優先されます。

グレースフル シャットダウン アウトバウンド ルート マップ

アウトバウンド グレースフル シャットダウン ルート マップは、グレースフル シャットダウン機能のアウトバウンド ルート マップの特定のタイプです。これらはオプションですが、ルート マップに関連付けられているコミュニティ リストがすでにある場合に役立ちます。通常のグレースフル シャットダウン アウトバウンド ルート マップには、特定の属性を設定または変更するための `set` 句のみが含まれています。

アウトバウンド ルート マップは、次の方法で使用できます。

- 既存のアウトバウンド ルート マップをすでに持っている顧客の場合は、より大きいシーケンス番号を持つ新しいエントリを追加し、GRACEFUL_SHUTDOWN ウェルノウンコミュニティで照合し、必要な属性を追加できます。
- **graceful-shutdown activate route-map** *name* オプションを使用してグレースフル シャットダウン アウトバウンドルート マップを使用することもできます。これが一般的な使用例です。

  このルート マップには match 句が必要ないため、ルート マップはネイバーに送信されるすべてのルートで一致します。

ルート マップの優先順位

同じルータ上に複数のルートマップが存在する場合は、次の優先順位が適用されて、コミュニティとのルートの処理方法が決定されます。次の例を考慮してください。60 のローカル設定を設定する標準の発信ルート マップ名 Red があるとします。また、Blue という名前のピア グレースフル シャットダウン ルート マップがあり、local-pref が 30 に設定されているとします。ルート更新が処理されると、Red は Blue を上書きするため、ローカル プリファレンスは 60 に設定されます。

- 通常の発信ルート マップは、ピア グレースフル シャットダウン マップよりも優先されます。
- ピア グレースフル シャットダウン マップは、グローバル グレースフル シャットダウン マップよりも優先されます。

注意事項と制約事項

BGP グローバル シャットダウンの制限事項と注意事項は、次のとおりです。

- グレースフル シャットダウン機能は、影響を受けるルータの代替ルートがネットワークに存在する場合にのみ、トラフィック損失を回避するのに役立ちます。ルータに代替ルートがない場合は、GRACEFUL_SHUTDOWN コミュニティを伝送するルートが使用可能な唯一のルートであるため、最適パスの計算に使用されます。この状況では、機能の目的が失われます。
- GRACEFUL_SHUTDOWN コミュニティを送信するには、BGP 送信コミュニティの設定が必要です。
- ルート マップの場合:
  - グローバル ルート マップとネイバー ルート マップが設定されている場合、ネイバー単位のルート マップが優先されます。
  - 発信ルート マップは、グレースフル シャットダウン用に設定されたグローバル ルート マップよりも優先されます。
  - 発信ルート マップは、グレースフル シャットダウン用に設定されたピア ルート マップよりも優先されます。
  - レガシー（既存の）インバウンド ルート マップにグレースフル シャットダウン機能を追加するには、次の手順を実行します。
    1. graceful shutdown match 句をルート マップの先頭に追加します。これには、句に低いシーケンス番号（たとえば、シーケンス番号 0）を設定します。
    2. graceful shutdown 句の後に continue ステートメントを追加します。continue ステートメントを省略すると、graceful shutdown 句と一致するルートマップ処理が停止します。シーケンス番号が大きい他の句（たとえば、1以上）は処理されません。

グレースフル シャットダウン タスクの概要

グレースフル シャットダウン機能を使用するには、通常、すべての Cisco Nexus スイッチでグレースフル シャットダウン対応をイネーブルにし、機能をイネーブルのままにします。BGP ルータをオフラインにする必要がある場合は、graceful-shutdown activate を設定します。

次の詳細に、グレースフル シャットダウン機能を使用するためのベスト プラクティスを示します。

ルータまたはリンクをダウンさせるには、次の手順を実行します。

1. グレースフル シャットダウン機能を設定します。
2. ネイバーでベスト パスを確認します。
3. 最適パスが再計算されたら、BGP を無効にする **shutdown** コマンドを発行します。
4. ルータまたはリンクをシャットダウンする必要がある作業を実行します。

ルータまたはリンクをオンラインに戻すには、次の手順を実行します。

1. シャットダウンが必要な作業が完了したら、BGP を再度イネーブルにします（**no shutdown**）。
2. グレースフル シャットダウン機能を無効にします（config モードの **no graceful-shutdown activate**）。

リンクのグレースフル シャットダウンの設定

この作業では、2 つの BGP ルータ間の特定のリンクでグレースフル シャットダウンを設定できます。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **config terminal**
2. **router bgp** *autonomous-system-number*
3. **neighbor {** *ipv4-address*|*ipv6-address* **} remote-as** *as-number*
4. **graceful-shutdown activate [route-map** *map-name***]**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**config terminal**
	例：
	`switch-1# configure terminal`
	`switch-1(config)#`
		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 2	**router bgp** *autonomous-system-number*
	例：
	`switch-1(config)# router bgp 110`
	`switch-1(config-router)#`
		ルータ コンフィギュレーション モードを開始して、BGP ルーティング プロセスを作成または設定します。

ステップ 3	**neighbor {** *ipv4-address*|*ipv6-address* **} remote-as** *as-number*
	例：
	`switch-1(config-router)# neighbor 10.0.0.3 remote-as 200`
	`switch-1(config-router-neighbor)#`
		ネイバーが属する自律システム（AS）を設定します。

ステップ 4	**graceful-shutdown activate [route-map** *map-name***]**
	例：
	`switch-1(config-router-neighbor)# graceful-shutdown activate route-map gshutPeer`
	`switch-1(config-router-neighbor)#`
		ネイバーへのリンクでグレースフル シャットダウンを設定します。また、既知の GRACEFUL_SHUTDOWN コミュニティを使用してルートをアドバタイズし、アウトバウンド ルート更新にルートマップを適用します。

	コマンドまたはアクション	目的
		ルートは、デフォルトでグレースフル シャットダウン コミュニティでアドバタイズされます。この例では、ルートは gshutPeer という名前のルート マップを使用して、グレースフル シャットダウン コミュニティを持つネイバーにアドバタイズされます。 gshut コミュニティを受信したデバイスは、ルートのコミュニティを確認し、オプションでコミュニティを使用してルーティング ポリシーを適用します。

GRACEFUL_SHUTDOWN コミュニティに基づく BGP ルートのフィルタリングとローカルプリファレンスの設定

まだ 9.3(1) を実行していないスイッチには、GRACEFUL_SHUTDOWN コミュニティ名と一致するインバウンドルートマップがありません。したがって、正しいルートを識別して先送りする方法はありません。

9.3(1) よりも前のリリースの NX-OS を実行しているスイッチでは、グレースフル シャットダウン（65535:0）のコミュニティ値と一致するインバウンド ルート マップを設定し、ルートを非優先にする必要があります。

スイッチが 9.3(1) 以降を実行している場合、着信ルートマップを設定する必要はありません。

手順の概要

1. **configure terminal**
2. **ip community list standard** *community-list-name* **seq** *sequence-number* **{ permit | deny }** *value*
3. **route map** *map-tag* **{deny | permit}** *sequence-number*
4. **match community** *community-list-name*
5. **set local-preference** *local-pref-value*
6. **exit**
7. **router bgp** *community-list-name*
8. **neighbor {** *ipv4-address|ipv6-address* **}**
9. **address-family {** *address-family sub family* **}**
10. **send community**
11. **route map** *map-tag* **in**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch-1# configure terminal`
	`switch-1<config)#`
		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 2	**ip community list standard** *community-list-name* **seq** *sequence-number* **{ permit | deny }** *value*
	例：
	`switch-1(config)# ip community-list standard GSHUT seq 10 permit 65535:0`
	`switch-1(config)#`
		コミュニティ リストを設定し、よく知られたグレースフル シャットダウン コミュニティ値を持つルートを許可または拒否します。

ステップ 3	**route map** *map-tag* **{deny | permit}** *sequence-number*
	例：
	`switch-1(config)# route-map RM_GSHUT permit 10`
	`switch-1(config-route-map)#`
		ルート マップをシーケンス 10 として設定し、GRACEFUL_SHUTDOWN コミュニティを持つルートを許可します。

ステップ 4	**match community** *community-list-name*
	例：
	`switch-1(config-route-map)# match community GSHUT`
	`switch-1(config-route-map)#`
		IP コミュニティ リスト GSHUT に一致するルートがルート ポリシー マネージャ（RPM）により処理されるように設定します。

ステップ 5	**set local-preference** *local-pref-value*
	例：
	`switch-1(config-route-map)# set local-preference 10`
	`switch-1(config-route-map)#`
		IP コミュニティ リスト GSHUT に一致するルートに、指定されたローカル プリファレンスが与えられるように設定します。

ステップ 6	**exit**
	例：
	`switch-1(config-route-map)# exit`
	`switch-1(config)#`
		ルート マップ設定モードを終了し、グローバル設定モードに戻ります。

ステップ 7	**router bgp** *community-list-name*
	例：
	`switch-1(config)# router bgp 100`
	`switch-1(config-router)#`
		ルータ設定モードを開始し、BGP インスタンスを作成します。

ステップ 8	**neighbor {** *ipv4-address*|*ipv6-address* **}**
	例：
	`switch-1(config-router)# neighbor 10.0.0.3`
	`switch-1(config-router-neighbor)#`
		指定したネイバーのルート BGP ネイバー モードを開始します。

ステップ 9	**address-family {** *address-family sub family* **}**
	例：
	`nxosv2(config-router-neighbor)# address-family ipv4 unicast`
	`nxosv2(config-router-neighbor-af)#`
		ネイバーをアドレス ファミリ（AF）設定モードにします。

ステップ 10	**send community** 例：	ネイバーとの BGP コミュニティ交換を可能にします。

	コマンドまたはアクション	目的
	nxosv2(config-router-neighbor-af)# **send-community** nxosv2(config-router-neighbor-af)#	
ステップ 11	**route map** *map-tag* **in** 例： nxosv2(config-router-neighbor-af)# **route-map RM_GSHUT in** nxosv2(config-router-neighbor-af)#	ネイバーからの着信ルートにルート マップを適用します。この例では、RM_GSHUT という名前のルート マップは、ネイバーからの GRACEFUL_SHUTDOWN コミュニティを持つルートを許可します。

すべての BGP ネイバーのグレースフル シャットダウンの設定

グレースフル シャットダウン イニシエータのすべてのネイバーに GRACEFUL_SHUTDOWN ウェルノウン コミュニティを手動で適用できます。

すべての BGP ネイバーに対して、グローバル レベルでグレースフル シャットダウンを設定できます。

始める前に

BGP をまだ有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **configure terminal**
2. **router bgp** *autonomous-system-number*
3. **graceful-shutdown activate [route-map** *map-name***]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： switch-1# **configure terminal** switch-1(config)#	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 2	**router bgp** *autonomous-system-number* 例： switch-1(config)# **router bgp 110** switch-1(config-router)#	ルータ コンフィギュレーションモードを開始して、BGP ルーティング プロセスを作成または設定します。
ステップ 3	**graceful-shutdown activate [route-map** *map-name***]** 例： switch-1(config-router-neighbor)# **graceful-shutdown activate route-map gshutPeer** switch-1(config-router-neighbor)#	すべてのネイバーへのリンクのグレースフル シャットダウン ルート マップを設定します。また、既知の GRACEFUL_SHUTDOWN コミュニティを持つすべてのルートをアドバタイズし、ルートマップをアウトバウンド ルート アップデートに適用します。

	コマンドまたはアクション	目的
		ルートはデフォルトで GRACEFUL_SHUTDOWN コミュニティでアドバタイズされます。この例では、ルートが gshutPeer という名前のルートマップを持つコミュニティを持つすべてのネイバーにアドバタイズされます。ルートマップには set 句のみを含める必要があります。 GRACEFUL_SHUTDOWN コミュニティを受信したデバイスは、ルートのコミュニティを確認し、オプションでコミュニティを使用してルーティング ポリシーを適用します。

GRACEFUL_SHUTDOWNコミュニティを使用したすべてのルートのプリファレンスの制御

Cisco NX-OS では、GRACEFUL_SHUTDOWN コミュニティを持つ着信ルートの優先順位を下げることができます。**graceful shutdown aware** が有効になっている場合、最適パス計算時に、BGP はコミュニティを伝送するルートを最も低い優先順位と見なします。デフォルトでは、プレファレンスの引き下げが有効になっていますが、このオプションを選択的に無効にすることもできます。

このオプションをイネーブルまたはディセーブルにするたびに、BGP のベストパス計算がトリガーされます。このオプションを使用すると、グレースフル シャットダウンのウェルノウンコミュニティにおける BGP のベストパス計算の動作を柔軟に制御できます。

始める前に

BGPを有効にしていない場合は、ここで有効にします（**feature bgp**）。

手順の概要

1. **configure terminal**
2. **router bgp** *autonoums-system*
3. （任意） **no graceful-shutdown aware**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch-1(config)# config terminal`
	`switch-1(config)#`
		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 2	**router bgp** *autonoums-system*
	例：
	`switch-1(config)# router bgp 100`
	`switch-1(config-router)#`
		ルータ コンフィギュレーション モードを開始し、BGP ルーティング プロセスを設定します。

ステップ 3	（任意） **no graceful-shutdown aware**
	例：
	`switch-1(config-router)# no graceful-shutdown aware`
	`switch-1(config-router)#`
		このBGPルータでは、GRACEFUL_SHUTDOWN コミュニティを持つすべてのルートに低い優先順位を指定しないという意味です。グレースフル シャットダウン認識機能がディセーブルになっている場合、デフォルト アクションはルートを非優先にします。そのため、コマンドには **no** 形式というオプションが存在しており、これを使用すると、グレースフル シャットダウン ルートは非優先になりません。

GRACEFUL_SHUTDOWN コミュニティのピアへの送信の防止

発信ルート更新にルート属性として追加されたGRACEFUL_SHUTDOWNコミュニティが不要になった場合は、コミュニティを削除して、指定されたネイバーに送信しなくなります。1 つの使用例は、ルータが自律システム境界にあり、グレースフル シャットダウン機能が自律システム境界の外部に伝播しないようにする場合です。

GRACEFUL_SHUTDOWN がピアに送信されないようにするには、send community オプションを無効にするか、コミュニティを発信ルート マップから削除します。

次の方法の中から 1 つを選択してください。

- 実行コンフィギュレーションで send-community を無効にします。

  例：

  ```
  nxosv2(config-router-neighbor-af)# no send-community standard
  nxosv2(config-router-neighbor-af)#
  ```

  このオプションを使用すると、スイッチはGRACEFUL_SHUTDOWNコミュニティを受信しますが、発信ルート マップを介してダウンストリーム ネイバーに送信されません。すべての標準コミュニティも送信されません。

- 次の手順に従って、発信ルートマップを介してGRACEFUL_SHUTDOWNコミュニティを削除します。

  1. GRACEFUL_SHUTDOWNコミュニティと一致するIPコミュニティ リストを作成します。
  2. GRACEFUL_SHUTDOWN コミュニティと照合する発信ルート マップを作成します。
  3. **set community-list delete** 句を使用して GRACEFUL_SHUTDOWN コミュニティを削除します。

このオプションを使用すると、コミュニティ リストはGRACEFUL_SHUTDOWN コミュニティと一致し、許可されます。その後、発信ルートマップはコミュニティと照合され、発信ルート マップから削除されます。他のすべてのコミュニティは、問題なく発信ルートマップを通過します。

グレースフル シャットダウン情報の表示

グレースフル シャットダウン機能に関する情報は、次の **show** コマンドで確認できます。

コマンド	アクション
show ip bgp community-list graceful-shutdown	GRACEFUL_SHUTDOWN コミュニティを持つ BGP ルーティング テーブル内のすべてのエントリを表示します。
show running-config bgp	実行中の BGP のデフォルト設定を示します。
show running-config bgp all	グレースフル シャットダウン機能に関する情報など、実行中の BGP 設定のすべての情報を表示します。
show bgp *address-family* **neighbors** *neighbor-address*	機能がピアに設定されている場合、次のように表示されます。 • 指定されたネイバーの graceful-shutdown-activate 機能の状態 • 指定されたネイバーに設定されたグレースフル シャットダウン ルートマップの名前
show bgp process	コンテキストに応じて異なる情報を表示します。 graceful-shutdown-activate オプションがピア コンテキストで設定されている場合、`graceful-shutdown-active` を介して機能の有効または無効状態を示します。 graceful-shutdown-activate オプションがグローバル コンテキストで設定され、graceful-shutdown ルートマップがある場合は、次のように機能の有効状態が表示されます。 • `graceful-shutdown-active` • `graceful-shutdown-aware` • `graceful-shutdown route-map`

コマンド	アクション
show ip bgp *address*	指定されたアドレスについて、次を含む BGP ルーティング テーブル情報を表示します。 • 最適パスとして指定されたアドレスの状態 • 指定されたアドレスが GRACEFUL_SHUTDOWN コミュニティの一部であるかどうか

グレースフル シャットダウンの設定例

次に、グレースフル シャットダウン機能を使用するための設定例を示します。

BGP リンクのグレースフル シャットダウンの設定

次に、ローカル プリファレンスとコミュニティを設定しながらグレースフル シャットダウンを設定する例を示します。

- 指定されたネイバーへのリンクのグレースフル シャットダウン アクティブ化の設定
- ルートへの GRACEFUL_SHUTDOWN コミュニティの追加
- コミュニティとのアウトバウンド ルートに対して set 句のみを使用して gshutPeer という名前のルートマップを設定します。

```
router bgp 100
    neighbor 20.0.0.3 remote-as 200
         graceful-shutdown activate route-map gshutPeer
         address-family ipv4 unicast
              send-community

route-map gshutPeer permit 10
    set local-preference 0
    set community 200:30
```

All-Neighbor BGP リンクのグレースフル シャットダウンの設定

次に例を示します。

- ローカル ルータとそのすべてのネイバーを接続するすべてのリンクに対してグレース フルシャットダウン アクティブ化を設定します。
- GRACEFUL_SHUTDOWN コミュニティをルートに追加しています。
- すべての発信ルートに対して set 句のみを使用して gshutAall という名前のルートマップを設定します。

```
router bgp 200
   graceful-shutdown activate route-map gshutAll
```

```
route-map gshutAll permit 10
   set as-path prepend 10 100 110
   set community 100:80

route-map Red permit 10
   set local-pref 20

router bgp 100
   graceful-shutdown activate route-map gshutAll
      router-id 2.2.2.2
         address-family ipv4 unicast
         network 2.2.2.2/32
         neighbor 1.1.1.1 remote-as 100
         update-source loopback0
         address-family ipv4 unicast
            send-community
         neighbor 20.0.0.3 remote-as 200
         address-family ipv4 unicast
            send-community
               route-map Red out
```

この例では、ネイバー 1.1.1.1 に対して `gshutAll` ルート マップが有効になりますが、ネイバー 20.0.0.3 で設定された発信ルートマップ `Red` が優先されるため、ネイバー 20.0.0.3 に対しては有効になりません。

ピアテンプレートでのグレースフル シャットダウンの設定

この例では、ピアセッションテンプレートでグレースフルシャットダウン機能を設定します。これはネイバーによって継承されます。

```
router bgp 200
   template peer-session p1
      graceful-shutdown activate route-map gshut_out
   neighbor 1.1.1.1 remote-as 100
      inherit peer-session p1
      address-family ipv4 unicast
         send-community
```

GRACEFUL_SHUTDOWN コミュニティの使用およびインバウンドルートマップに基づく BGP ルートのフィルタリングとローカル プリファレンスの設定

次に、コミュニティ リストを使用して、GRACEFUL_SHUTDOWN コミュニティを持つ着信ルートをフィルタリングする例を示します。この設定は、Cisco NX-OS 9.3(1) を最小バージョンとして実行していないレガシー スイッチに役立ちます。

次に例を示します。

- GRACEFUL_SHUTDOWN コミュニティを持つルートを許可する IP コミュニティ リスト。
- RM_GSHUT という名前のルート マップは、GSHUT という名前の標準コミュニティ リストに基づいてルートを許可します。
- また、ルートマップは、処理するルートの優先順位を 0 に設定します。これにより、ルータがオフラインになったときに、それらのルートに最適パス計算の優先順位が低くなります。ネイバー (20.0.0.2) からの着信 IPv4 ルートにルート マップが適用されます。

```
ip community-list standard GSHUT permit 65535:0
```

```
route-map RM_GSHUT permit 10
   match community GSHUT
   set local-preference 0

router bgp 200
   neighbor 20.0.0.2 remote-as 100
      address-family ipv4 unicast
         send-community
         route-map RM_GSHUT in
```

グレースフル リスタートの設定

グレースフル リスタートを設定し、BGP に対してグレースフル リスタート ヘルパー機能をイネーブルにできます。

（注） Cisco NX-OS リリース 10.1(1) は、より多くの BFD セッションをサポートします。BGP セッションが BFD に関連付けられている場合、ISSU 中にピア接続を維持するために BGP **restart-time** を増やす必要が生じることがあります。

始める前に

BGP をイネーブルにする必要があります（「BGP のイネーブル化」の項を参照）。

VRF を作成します。

手順の概要

1. **configure terminal**
2. **router bgp** *as-number*
3. （任意） **timers prefix-peer-timeout** *timeout*
4. **graceful-restart**
5. **graceful-restart** {**restart-time** *time*|**stalepath-time** *time*}
6. **graceful-restart-helper**
7. （任意） **show running-config bgp**
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		コンフィギュレーション モードに入ります。

ステップ 2	**router bgp** *as-number* 例：	自律システム番号を設定して、新しいBGPプロセスを作成します。

	コマンドまたはアクション	目的
	`switch(config)# router bgp 65535` `switch(config-router)#`	
ステップ 3	（任意） **timers prefix-peer-timeout** *timeout* 例： `switch(config-router)# timers prefix-peer-timeout 20`	BGP プレフィックス ピアのタイムアウト値を設定します（秒単位）。デフォルト値は 90 秒です。 （注） このコマンドは、Cisco NX-OS リリース 9.3(3) 以降でサポートされます。
ステップ 4	**graceful-restart** 例： `switch(config-router)# graceful-restart`	グレースフル リスタートおよびグレースフル リスタート ヘルパー機能をイネーブルにします。このコマンドは、デフォルトでイネーブルになっています。 このコマンドによって、BGP ネイバー セッションの自動通知およびセッション リセットが開始されます。
ステップ 5	**graceful-restart** {**restart-time** *time*|**stalepath-time** *time*} 例： `switch(config-router)# graceful-restart restart-time 300`	グレースフル リスタート タイマーを設定します。 オプション パラメータは次のとおりです。 • **restart-time**：BGP ピアに送信されたリスタートの最大時間。有効な範囲は 1 ～ 3600 秒です。デフォルトは 120 です。 （注） Cisco NX-OS リリース 10.1(1) は、より多くの BFD セッションをサポートします。BGP セッションが BFD に関連付けられている場合、ISSU 中にピア接続を維持するために BGP **restart-time** を増やす必要が生じることがあります。 • **stalepath-time**：BGP が再起動中の BGP ピアからの古いルートを維持する最大時間有効な範囲は 1 ～ 3600 秒です。デフォルトは 300 です。 NX-OS ソフトウェア リリース 10.2(1) では、BGP セッションがグレースフル リスタート機能をアドバタイズするために、BGP セッションの手動リセットが必要です。NX-OS ソフトウェア リリース 10.2(2) 以降では、このコマンドが有効になっている場合、BGP セッションは、BGP セッションを再起動する必要なく、グレースフル リスタート機能を動的にアドバタイズします。

	コマンドまたはアクション	目的
ステップ 6	**graceful-restart-helper** 例： `switch(config-router)# graceful-restart restart-time 300`	BGP GR が無効になっている場合、SSO や BGP プロセスの再起動などの特定の GR 対応イベントが N9K でローカルに発生している間、N9K 自体は必ずしも自身の転送状態を保持しません。ただし、GR ヘルパーとして、GR 機能をアドバタイズして再起動しているピアをサポートします。つまり、N9K は、ピアリングがダウンしたことを検出すると（ホールドタイマーの期限切れまたは通知メッセージの受信以外）、ピアを指すルートを失効させ、ピアの EOR（または失効パス タイムアウト）を待機します。ピアが再起動して N9K とのピアリングを再確立すると、ピアは自身のすべてのルートを再アドバタイズし、N9K は BGP およびルーティング テーブルでそれらのルートを更新します。ピアから EOR を受信するか、または古いパス タイムアウト（どちらか先に発生した方）を受信すると、N9K はそのピアから残りの古いルートをフラッシュします。ヘルパー モードがない場合、N9K は再起動中のリモート ピアから学習したルートを即座にクリアし、トラフィック損失につながる可能性があります。
ステップ 7	（任意） **show running-config bgp** 例： `switch(config-router)# show running-config bgp`	BGP の設定を表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config-router)# copy running-config startup-config`	この設定変更を保存します。

例

次に、グレースフル リスタートを有効にする例を示します。

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# graceful-restart
switch(config-router)# graceful-restart restart-time 300
switch(config-router)# copy running-config startup-config
```

仮想化の設定

1 つの BGP プロセスを設定し、複数の VRF を作成できます。また、各 VRF で同じ BGP プロセスを使用できます。

始める前に

BGPを有効にする必要があります。

手順の概要

1. **configure terminal**
2. **vrf context** *vrf-name*
3. **exit**
4. **router bgp** *as-number*
5. **vrf** *vrf-name*
6. **neighbor** *ip-address* **remote-as** *as-number*
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**vrf context** *vrf-name*
	例：
	`switch(config)# vrf context RemoteOfficeVRF`
	`switch(config-vrf)#`
		新しい VRF を作成し、VRF 設定モードを開始します。

ステップ 3	**exit**
	例：
	`switch(config-vrf)# exit`
	`switch(config)#`
		VRF設定モードを終了します。

ステップ 4	**router bgp** *as-number*
	例：
	`switch(config)# router bgp 65535`
	`switch(config-router)#`
		自律システム番号を設定して、新しいBGPプロセスを作成します。

ステップ 5	**vrf** *vrf-name* 例：	ルータ VRF設定モードを開始し、この BGP インスタンスと VRF を関連付けます。

	コマンドまたはアクション	目的
	`switch(config-router)# vrf RemoteOfficeVRF switch(config-router-vrf)#`	
ステップ 6	**neighbor** *ip-address* **remote-as** *as-number* 例： `switch(config-router-vrf)# neighbor 209.165.201.1 remote-as 65535 switch(config-router--vrf-neighbor)#`	リモート BGP ピアの IP アドレスおよび AS 番号を設定します。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-router-vrf-neighbor)# copy running-config startup-config`	この設定変更を保存します。

例

次に、VRF を作成し、VRF でルータ ID を設定する例を示します。

```
switch# configure terminal
switch(config)# vrf context NewVRF
switch(config-vrf)# exit
switch(config)# router bgp 65536
switch(config-router)# vrf NewVRF
switch(config-router-vrf)# neighbor 209.165.201.1 remote-as 65536
switch(config-router-vrf-neighbor)# copy running-config startup-config
```

拡張 BGP の設定の確認

BGP の設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show bgp all [**summary**] [**vrf** *vrf-name*]	すべてのアドレス ファミリについて、BGP 情報を表示します。
show bgp convergence [vrf *vrf-name]*	すべてのアドレス ファミリについて、BGP 情報を表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community** {**regexp** *expression* | [**community**] [**no-advertise**] [**no-export**] [**no-export-subconfed**]} [**vrf** *vrf-name*]	BGP コミュニティと一致する BGP ルートを表示します。

コマンド	目的
show bgp [**vrf** *vrf-name*] {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community-list** *list-name* [**vrf** *vrf-name*]	BGP コミュニティ リストと一致する BGP ルートを表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **extcommunity** {**regexp** *expression* | **generic** [**non-transitive** | **transitive**] *aa4:nn* [**exact-match**]} [**vrf** *vrf-name*]	BGP 拡張コミュニティと一致する BGP ルートを表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **extcommunity-list** *list-name* [**exact-match**]} [**vrf** vrf-name]	BGP 拡張コミュニティ リストと一致する BGP ルートを表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **extcommunity-list** *list-name* [**exact-match**]} [**vrf** vrf-name]	BGP ルート ダンプニングの情報を表示します。ルート フラップ ダンプニング情報を消去するには、**clear bgp dampening** コマンドを使用します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**dampening dampened-paths** [**regexp** *expression*]} [**vrf** vrf-name]	BGP ルート ヒストリ パスを表示します。
show bgp {**ipv4** | **ipv6** | **vpnv4** | **vpnv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **filter-list** *list-name* [**vrf** *vrf-name*]	BGP フィルタ リストの情報を表示します。
show bgp {**ipv4** | **ipv6** | **vpnv4** | **vpnv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **neighbors** [*ip-address* | *ipv6-prefix*] [**vrf** *vrf-name*]	BGP ピアの情報を表示します。これらのネイバーを消去するには、**clear bgp neighbors** コマンドを使用します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**nexthop** | **nexthop-database**} [**vrf** *vrf-name*]	BGP ルート ネクスト ホップの情報を表示します。
show bgp paths	BGP パス情報を表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **policy** *name* [**vrf** *vrf-name*]	BGP ポリシー情報を表示します。ポリシー情報を消去するには、**clear bgp policy** コマンドを使用します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **prefix-list** *list-name* [**vrf** *vrf-name*]	プレフィックス リストと一致する BGP ルートを表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **received-paths** [**vrf** *vrf-name*]	ソフト再構成用に保管されている BGP パスを表示します。

コマンド	目的
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **regexp** *expression* [**vrf** *vrf-name*]	AS_path 正規表現と一致する BGP ルートを表示します。
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **route-map** *map-name* [**vrf** *vrf-name*]	ルート マップと一致する BGP ルートを表示します。
show bgp peer-policy *name* [**vrf** *vrf-name*]	BGP ピア ポリシー情報を表示します。
show bgp peer-session *name* [**vrf** *vrf-name*]	BGP ピア セッション情報を表示します。
show bgp peer-template *name* [**vrf** *vrf-name*]	BGP ピア テンプレート情報を表示します。ピア テンプレートのすべてのネイバーを消去するには、**clear bgp peer-template** コマンドを使用します。
show bgp process	BGP プロセス情報を表示します。
show bgp {**ipv4** | **ipv6**} **unicast neighbors** *interface*	指定されたインターフェイスの BGP ピアに関する情報を表示します。
show ip bgp neighbors *interface-name*	BGP ピアとして使用されるインターフェイスを表示します。
show ip route *ip-address* **detail vrf all | i bw**	リンク帯域幅の EXTCOMM フィールドを表示します。出力の bw：xx（bw：40 など）は、BGP ピアが帯域幅付きの BGP 拡張属性を送信していることを示します（重み付け ECMP の場合）。
show {**ipv4** | **ipv6**} **bgp** *options*	BGP のステータスと構成情報を表示します。
show {**ipv4** | **ipv6**} **mbgp** *options*	BGP のステータスと構成情報を表示します。

コマンド	目的
show ipv6 routers interface *interface*	IPv6 ICMP ルータ アドバタイズメントによって学習されたリモート IPv6 ルータのリンクローカル アドレスを表示します。
show running-configuration bgp	現在実行中の BGP コンフィギュレーションを表示します。

BGP 統計情報のモニタリング

BGP の統計情報を表示するには、次のコマンドを使用します。

コマンド	目的
show bgp {**ipv4** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **flap-statistics** [**vrf** *vrf-name*]	BGP ルート フラップの統計情報を表示します。これらの統計情報をクリアするには、**clear bgp flap-statistics** コマンドを使用します。
show bgp {**ipv4** | **ipv6**} **unicast injected-routes**	ルーティング テーブルに挿入されたルートを表示します。
show bgp sessions [**vrf** *vrf-name*]	すべてのピアの BGP セッションを表示します。これらの統計情報をクリアするには、**clear bgp sessions** コマンドを使用します。
show bgp statistics	BGP 統計情報を表示します。

設定例

この例は、個々の BGP ネイバーの BFD をイネーブルにする方法を示します。

```
router bgp 400
  router-id 2.2.2.2
  neighbor 172.16.2.3
    bfd
    remote-as 400
    update-source Vlan1002
    address-family ipv4 unicast
```

この例は、BGP プレフィックス ピアの BFD をイネーブルにする方法を示します。

```
router bgp 400
  router-id 1.1.1.1
  neighbor 172.16.2.0/24
    bfd
```

```
    remote-as 400
    update-source Vlan1002
    address-family ipv4 unicast
```

プレフィックス ベース ネイバーの MD5 認証を設定する例を示します。

```
template peer BasePeer-V6
    description BasePeer-V6
    password 3 f4200cfc725bbd28
    transport connection-mode passive
    address-family ipv6 unicast
template peer BasePeer-V4
    bfd
    description BasePeer-V4
    password 3 f4200cfc725bbd28
    address-family ipv4 unicast
--
    neighbor fc00::10:3:11:0/127 remote-as 65006
      inherit peer BasePeer-V6
    neighbor 10.3.11.0/31 remote-as 65006
      inherit peer BasePeer-V4
```

次に、ネイバー ステータスの変化に関するメッセージをグローバルに有効にし、特定のネイバーについてはメッセージを抑制する方法を示します。

```
router bgp 65100
   log-neighbor-changes
     neighbor 209.165.201.1 remote-as 65535
       description test
       address-family ipv4 unicast
         soft-reconfiguration inbound
          disable log-neighbor-changes
```

関連項目

BGP の詳細については、次の項目を参照してください。

- 基本的 BGP の設定
- Route Policy Manager の設定

その他の参考資料

BGP の実装に関連する詳細情報については、次の項を参照してください。

MIB

MIB	MIB のリンク
BGP に関連する MIB	サポートされている MIB を検索およびダウンロードするには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

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。