Cisco NX-OS Unicast Routing コンフィギュレーション ガイド Release 4.0

March 31, 2008

Text Part Number: OL-12912-01-J

【注意】シスコ製品をご使用になる前に、安全上の注意（www.cisco.com/jp/go/safety_warning/）をご確認ください。

本書は、米国シスコシステムズ発行ドキュメントの参考和訳です。
米国サイト掲載ドキュメントとの差異が生じる場合があるため、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。
また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコシステムズおよびこれら各社は、商品性や特定の目的への準拠性、権利を侵害しないことに関する、または取り扱い、使用、または取引によって発生する、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコシステムズおよびその代理店は、このマニュアルの使用またはこのマニュアルを使用できないことによって起こる制約、利益の損失、データの損傷など間接的で偶発的に起こる特殊な損害のあらゆる可能性がシスコシステムズまたは代理店に知らされていても、それらに対する責任を一切負いかねます。

CCDE, CCENT, Cisco Eos, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn is a service mark; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0805R)

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco NX-OS Unicast Routing コンフィギュレーション ガイド Release 4.0
Copyright © 2008 Cisco Systems, Inc.
All rights reserved.

Copyright © 2008, シスコシステムズ合同会社 .
All rights reserved.

CONTENTS

はじめに xxi
対象読者 xxi
マニュアルの構成 xxi
表記法 xxiii
関連資料 xxiv
マニュアルの入手方法とサービス リクエストの提出方法 xxv
Japan TAC Web サイト xxv

CHAPTER 1 概要 1-1
レイヤ 3 ユニキャスト ルーティングについて 1-2
ルーティングの基本 1-2
パケット交換 1-2
ルーティング メトリック 1-4
パスの長さ 1-4
信頼性 1-4
ルーティング遅延 1-4
帯域幅 1-4
負荷 1-5
通信コスト 1-5
ルータ ID 1-5
自律システム 1-5
収束 1-6
ロード バランシングおよび等コスト マルチパス 1-6
ルート再配布 1-6
管理ディスタンス 1-7
スタブ ルーティング 1-7
ルーティング アルゴリズム 1-8
スタティック ルートおよびダイナミック ルーティング プロトコル 1-8
内部および外部ゲートウェイ プロトコル 1-9
ディスタンス ベクトル プロトコル 1-9
リンクステート プロトコル 1-9
レイヤ 3 仮想化 1-11
Cisco NX-OS 転送アーキテクチャ 1-12

ユニキャスト RIB 1-12
隣接マネージャ 1-13
ユニキャスト転送分散モジュール 1-13
ユニキャスト FIB 1-13
ルーティング情報と転送情報の表示 1-13
ハードウェア転送 1-14
ソフトウェア転送 1-15
レイヤ 3 ユニキャスト ルーティング機能のまとめ 1-15
IPv4 と IPv6 1-15
IP サービス 1-15
OSPF 1-16
EIGRP 1-16
IS-IS 1-16
BGP 1-16
RIP 1-16
スタティック ルーティング 1-17
レイヤ 3 仮想化 1-17
Route Policy Manager 1-17
ポリシーベース ルーティング 1-17
First-Hop Redundancy Protocol（FHRP） 1-17
オブジェクト追跡 1-18
関連資料 1-18

PART 1 IP

CHAPTER 2 IPv4 の設定 2-1
IPv4 について 2-2
複数の IPv4 アドレス 2-2
Address Resolution Protocol 2-3
ARP キャッシング 2-4
ARP キャッシュのスタティック エントリおよびダイナミック エントリ 2-4
ARP を使用しないデバイス 2-4
Inverse ARP 2-4
Reverse ARP 2-5
プロキシ ARP 2-5
ローカル プロキシ ARP 2-6
ICMP 2-6
仮想化のサポート 2-6

IPv4 のライセンス要件 2-7
IPv4 の前提条件 2-7
注意事項および制約事項 2-7
IPv4 の設定 2-8
IPv4 アドレス指定の設定 2-8
複数の IP アドレスの設定 2-9
スタティック ARP エントリの設定 2-10
プロキシ ARP の設定 2-11
ローカル プロキシ ARP の設定 2-12
IPv4 設定の確認 2-13
IPv4 設定の例 2-13
デフォルト設定 2-13
その他の関連資料 2-14
関連資料 2-14
標準 2-14

CHAPTER 3 IPv6 の設定 3-1
IPv6 について 3-2
IPv6 アドレス フォーマット 3-2
IPv6 ユニキャスト アドレス 3-3
集約可能グローバル アドレス 3-4
リンクローカル アドレス 3-5
IPv4 互換の IPv6 アドレス 3-6
ユニーク ローカル アドレス 3-6
サイトローカル アドレス 3-7
IPv6 エニーキャスト アドレス 3-7
IPv6 マルチキャスト アドレス 3-8
IPv4 パケット ヘッダー 3-9
簡易 IPv4 パケット ヘッダー 3-10
IPv6 の DNS 3-12
IPv6 のパス MTU 探索 3-13
Cisco Discovery Protocol（CDP）IPv6 アドレスのサポート 3-13
IPv6 の ICMP 3-13
IPv6 近隣探索 3-14
IPv6 ネイバー送信要求メッセージ 3-14
IPv6 ルータ アドバタイズメント メッセージ 3-16
IPv6 ネイバー リダイレクト メッセージ 3-17
仮想化のサポート 3-18
IPv6 のライセンス要件 3-19

IPv6 の前提条件 3-19
IPv6 の注意事項および制約事項 3-19
IPv6 の設定 3-20
IPv6 アドレス指定の設定 3-20
IPv6 近隣探索の設定 3-22
選択可能なその他の IPv6 近隣探索 3-23
IPv6 設定の確認 3-25
IPv6 設定の例 3-25
デフォルト設定 3-25
その他の関連資料 3-26
関連資料 3-26
標準 3-26

CHAPTER 4 DNS の設定 4-1
DNS クライアントについて 4-2
DNS クライアントの概要 4-2
ネーム サーバ 4-2
DNS の動作 4-2
ハイ アベイラビリティ 4-3
仮想化のサポート 4-3
DNS クライアントのライセンス要件 4-4
DNS クライアントの前提条件 4-4
設定時の注意事項および制約事項 4-4
DNS クライアントの設定 4-5
DNS クライアントの設定 4-5
仮想化の設定 4-7
DNS クライアント設定の確認 4-9
DNS クライアント設定の例 4-9
デフォルト設定 4-9
その他の関連資料 4-10
関連資料 4-10
標準 4-10

PART 2 ルーティング

CHAPTER 5 OSPFv2 の設定 5-1
OSPFv2 について 5-2
hello パケット 5-2
ネイバー 5-3

隣接関係 5-3
代表ルータ 5-4
エリア 5-5
Link-State Advertisement 5-6
LSA タイプ 5-6
リンク コスト 5-6
フラッディングと LSA グループ ペーシング 5-7
リンクステート データベース 5-7
不透明 LSA 5-7
OSPFv2 とユニキャスト RIB 5-7
認証 5-8
簡易パスワード認証 5-8
MD5 認証 5-8
高度な機能 5-8
スタブ エリア 5-9
Not-So-Stubby エリア 5-9
仮想リンク 5-10
ルート再配布 5-10
ルート集約 5-11
ハイ アベイラビリティとグレースフル リスタート 5-11
OSPFv2 スタブ ルータ アドバタイズメント 5-12
複数の OSPFv2 インスタンス 5-12
SPF 最適化 5-13
仮想化のサポート 5-13
OSPFv2 のライセンス要件 5-14
OSPFv2 の前提条件 5-14
設定時の注意事項および制約事項 5-14
基本的 OSPFv2 の設定 5-15
OSPFv2 機能のイネーブル化 5-15
OSPFv2 インスタンスの作成 5-16
OSPFv2 インスタンス上のオプション パラメータの設定 5-17
OSPFv2 でのネットワークの設定 5-18
エリアの認証の設定 5-20
インターフェイスの認証の設定 5-22
高度な OSPFv2 の設定 5-24
境界ルータのフィルタ リストの設定 5-24
スタブ エリアの設定 5-25
Totally Stubby エリアの設定 5-26

NSSA の設定 5-27
仮想リンクの設定 5-29
再配布の設定 5-31
ルート集約の設定 5-32
スタブ ルート アドバタイズメントの設定 5-34
デフォルト タイマーの変更 5-35
グレースフル リスタートの設定 5-37
OSPFv2 インスタンスの再起動 5-39
仮想化による OSPFv2 の設定 5-39
OSPFv2 設定の確認 5-42
OSPFv2 統計情報の表示 5-42
OSPFv2 設定の例 5-43
デフォルト設定 5-43
その他の関連資料 5-44
関連資料 5-44
MIB 5-44

CHAPTER 6 OSPFv3 の設定 6-1
OSPFv3 について 6-2
OSPFv3 と OSPFv2 の比較 6-2
hello パケット 6-3
ネイバー 6-3
隣接関係 6-4
代表ルータ 6-4
エリア 6-5
Link-State Advertisement 6-6
LSA タイプ 6-6
リンク コスト 6-7
フラッディングと LSA グループ ペーシング 6-7
リンクステート データベース 6-8
OSPFv3 と IPv6 ユニキャスト RIB 6-8
認証 6-8
アドレス ファミリのサポート 6-8
高度な機能 6-9
スタブ エリア 6-9
Not-So-Stubby エリア 6-10
仮想リンク 6-10
ルート再配布 6-11
ルート集約 6-11

ハイ アベイラビリティとグレースフル リスタート 6-11
複数の OSPFv3 インスタンス 6-12
SPF 最適化 6-12
仮想化のサポート 6-13
OSPFv3 のライセンス要件 6-13
OSPFv3 の前提条件 6-13
設定時の注意事項および制約事項 6-13
基本的 OSPFv3 の設定 6-14
OSPFv3 機能のイネーブル化 6-14
OSPFv3 インスタンスの作成 6-15
OSPFv3 でのネットワークの設定 6-17
高度な OSPFv3 の設定 6-20
境界ルータのフィルタ リストの設定 6-20
スタブ エリアの設定 6-21
Totally Stubby エリアの設定 6-23
NSSA の設定 6-23
仮想リンクの設定 6-25
再配布の設定 6-27
ルート集約の設定 6-28
デフォルト タイマーの変更 6-30
グレースフル リスタートの設定 6-32
OSPFv3 インスタンスの再起動 6-34
仮想化による OSPFv3 の設定 6-34
OSPFv3 設定の確認 6-37
OSPFv3 統計情報の表示 6-37
OSPFv3 設定の例 6-38
関連資料 6-38
デフォルト設定 6-38
その他の関連資料 6-39
関連資料 6-39
MIB 6-39

CHAPTER 7 EIGRP の設定 7-1
EIGRP について 7-2
EIGRP のコンポーネント 7-2
Reliable Transport Protocol 7-2
近隣探索および近隣回復 7-2
DUAL 7-3
EIGRP ルート更新 7-3

内部ルート メトリック 7-4
外部ルート メトリック 7-4
EIGRP とユニキャスト RIB 7-4
高度な EIGRP 7-4
認証 7-5
スタブ ルータ 7-5
ルート集約 7-5
ルート再配布 7-6
ロード バランシング 7-6
スプリット ホライズン 7-6
仮想化のサポート 7-7
グレースフル リスタートとハイ アベイラビリティ 7-7
EIGRP のライセンス要件 7-9
EIGRP の前提条件 7-9
設定時の注意事項および制約事項 7-9
基本的 EIGRP の設定 7-10
EIGRP 機能のイネーブル化 7-10
EIGRP インスタンスの作成 7-11
EIGRP インスタンスの再起動 7-13
EIGRP インスタンスのディセーブル化 7-13
インターフェイスでの EIGRP のディセーブル化 7-13
高度な EIGRP の設定 7-14
EIGRP での認証の設定 7-14
EIGRP スタブ ルーティングの設定 7-16
EIGRP の集約アドレスの設定 7-16
EIGRP へのルート再配布 7-17
EIGRP でのロード バランシングの設定 7-18
EIGRP のグレースフル リスタートの設定 7-19
hello パケットの間隔と保留時間の調整 7-21
スプリット ホライズンのディセーブル化 7-21
EIGRP の調整 7-22
EIGRP の仮想化の設定 7-24
EIGRP 設定の確認 7-26
EIGRP 統計情報の表示 7-26
EIGRP 設定の例 7-26
関連資料 7-27
デフォルト設定 7-27
その他の関連資料 7-28

関連資料 7-28

CHAPTER 8 IS-IS の設定 8-1
IS-IS 情報 8-2
IS-IS の概要 8-2
IS-IS エリア 8-2
NET およびシステム ID 8-3
DIS 8-3
IS-IS 認証 8-4
メッシュ グループ 8-4
過負荷ビット 8-4
ルート集約 8-5
ルートの再配布 8-5
ロード バランシング 8-5
仮想化サポート 8-5
ハイ アベイラビリティおよびグレースフル リスタート 8-6
複数の IS-IS インスタンス 8-6
IS-IS のライセンス要件 8-7
IS-IS の前提条件 8-7
設定時の注意事項および制約事項 8-7
IS-IS の設定 8-8
IS-IS コンフィギュレーション モード 8-8
IS-IS 機能のイネーブル化 8-9
IS-IS インスタンスの作成 8-10
IS-IS インスタンスの再起動 8-12
インターフェイス上での IS-IS の設定 8-12
エリアでの IS-IS 認証の設定 8-14
インターフェイス上での IS-IS 認証の設定 8-15
メッシュ グループの設定 8-16
DIS の設定 8-16
ダイナミック ホスト交換の設定 8-17
過負荷ビットの設定 8-17
サマリー アドレスの設定 8-17
再配布の設定 8-18
グレースフル リスタートの設定 8-20
仮想化の設定 8-21
IS-IS の調整 8-24
IS-IS の設定確認 8-26
IS-IS 統計情報の表示 8-27

IS-IS の設定例 8-27
関連項目 8-28
デフォルト設定 8-28
その他の関連資料 8-29
関連資料 8-29
規格 8-29

CHAPTER 9 ベーシック BGP の設定 9-1
ベーシック BGP の概要 9-2
BGP AS 9-2
管理ディスタンス 9-2
BGP ピア 9-3
BGP ルータ ID 9-3
BGP パス選択 9-3
パス ペアの比較 9-4
比較順序 9-5
ベスト パス変更の抑制 9-6
BGP およびユニキャスト RIB 9-6
BGP の仮想化 9-6
ベーシック BGP のライセンス要件 9-7
BGP の前提条件 9-7
BGP に関する注意事項および制限事項 9-7
CLI コンフィギュレーション モード 9-8
グローバル コンフィギュレーション モード 9-8
アドレス ファミリ コンフィギュレーション モード 9-8
ネイバー コンフィギュレーション モード 9-9
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 9-9
ベーシック BGP の設定 9-10
BGP 機能のイネーブル化 9-10
BGP インスタンスの作成 9-11
BGP インスタンスの再起動 9-12
BGP ピアの設定 9-13
ベーシック BGP の設定確認 9-15
BGP 統計情報の表示 9-16
ベーシック BGP の設定例 9-16
関連項目 9-16
関連情報 9-17
デフォルト設定 9-17
その他の関連資料 9-18

関連資料 9-18
MIB 9-18
技術サポート 9-18

CHAPTER 10 拡張 BGP の設定 10-1
拡張 BGP の概要 10-2
ピア テンプレート 10-2
認証 10-3
ルート ポリシーおよび BGP セッションのリセット 10-3
eBGP 10-4
iBGP 10-4
AS 連合 10-5
ルータ リフレクタ 10-5
機能ネゴシエーション 10-6
ルート ダンプニング 10-6
ロード シェアリングおよびマルチパス 10-7
ルート集約 10-7
ルートの再配布 10-8
BGP の調整 10-8
BGP タイマー 10-8
ベストパス アルゴリズムの調整 10-8
マルチプロトコル BGP 10-8
グレースフル リスタートおよびハイ アベイラビリティ 10-9
ISSU 10-9
仮想化サポート 10-10
拡張 BGP のライセンス要件 10-11
BGP の前提条件 10-11
BGP に関する注意事項および制限事項 10-11
拡張 BGP の設定 10-12
BGP セッション テンプレートの設定 10-12
BGP peer-policy テンプレートの設定 10-15
BGP ピア テンプレートの設定 10-17
プレフィクス ピアリングの設定 10-19
BGP 認証の設定 10-20
BGP セッションのリセット 10-20
ネクストホップ アドレスの変更 10-21
機能ネゴシエーションのディセーブル化 10-21
eBGP の設定 10-22
eBGP シングルホップ チェックのディセーブル化 10-22

eBGP マルチホップの設定 10-22
高速外部フェールオーバーのディセーブル化 10-23
AS 連合の設定 10-23
ルータ リフレクタの設定 10-23
ルート ダンプニングの設定 10-25
ロード シェアリングおよび ECMP の設定 10-26
最大プレフィクス数の設定 10-26
ダイナミック機能の設定 10-27
集約アドレスの設定 10-27
ルートの再配布の設定 10-27
BGP の調整 10-29
グレースフル リスタートの設定 10-31
仮想化の設定 10-33
拡張 BGP の設定確認 10-35
BGP 統計情報の表示 10-36
関連項目 10-36
デフォルト設定 10-36
その他の関連資料 10-37
関連資料 10-37
MIB 10-37

CHAPTER 11 RIP の設定 11-1
RIP 情報 11-2
RIP の概要 11-2
RIPv2 の認証 11-2
スプリット ホライズン 11-3
ルート フィルタリング 11-3
ルート集約 11-3
ルートの再配布 11-4
ロード バランシング 11-4
ハイ アベイラビリティ 11-4
仮想化サポート 11-4
RIP のライセンス要件 11-5
RIP の前提条件 11-5
設定時の注意事項および制約事項 11-5
RIP の設定 11-6
RIP 機能のイネーブル化 11-6
RIP インスタンスの作成 11-7
RIP インスタンスの再起動 11-8

インターフェイス上での RIP の設定 11-9
RIP 認証の設定 11-10
受動インターフェイスの設定 11-11
ポイズン リバースを指定した スプリット ホライズンの設定 11-11
ルート集約の設定 11-11
ルートの再配布の設定 11-12
仮想化の設定 11-13
RIP の調整 11-16
RIP の設定確認 11-17
RIP 統計情報の表示 11-17
RIP の設定例 11-18
関連項目 11-18
関連情報 11-18
デフォルト設定 11-18
その他の関連資料 11-19
関連資料 11-19
規格 11-19

CHAPTER 12 スタティック ルーティングの設定 12-1
スタティック ルーティングの概要 12-2
管理ディスタンス 12-2
直接接続のスタティック ルート 12-3
完全指定のスタティック ルート 12-3
フローティング スタティック ルート 12-3
スタティック ルートのリモート ネクスト ホップ 12-3
仮想化サポート 12-3
スタティック ルーティングのライセンス要件 12-4
スタティック ルーティングの前提条件 12-4
注意事項および制約事項 12-4
スタティック ルーティングの設定 12-5
スタティック ルートの設定 12-5
仮想化の設定 12-6
スタティック ルーティングの設定確認 12-8
スタティック ルーティングの設定例 12-8
デフォルト設定 12-8
その他の関連資料 12-8
関連資料 12-8

CHAPTER 13 レイヤ 3 仮想化の設定 13-1
レイヤ 3 仮想化 13-2
レイヤ 3 仮想化の概要 13-2
VRF およびルーティング 13-3
VRF 認識サービス 13-3
到達可能性 13-4
フィルタリング 13-5
到達可能性とフィルタリングの組み合わせ 13-5
VRF のライセンス要件 13-6
VRF の前提条件 13-6
注意事項および制約事項 13-6
VRF の設定 13-7
VRF の作成 13-7
インターフェイスへの VRF メンバシップの割り当て 13-8
ルーティング プロトコルに関する VRF パラメータの設定 13-9
VRF 認識サービスの設定 13-11
VRF スコープの設定 13-12
VRF の設定確認 13-13
VRF の設定例 13-13
関連資料 13-13
デフォルト設定 13-13
その他の関連資料 13-14
関連資料 13-14
規格 13-14

CHAPTER 14 Route Policy Manager の設定 14-1
Route Policy Manager の概要 14-2
プレフィクス リスト 14-2
ルート マップ 14-2
一致基準 14-3
設定変更 14-3
アクセス リスト 14-3
BGP の AS パス リスト 14-4
BGP のコミュニティ リスト 14-4
ルートの再配布およびルート マップ 14-4
ポリシーベース ルーティング 14-4
Route Policy Manager のライセンス要件 14-5
Route Policy Manager の前提条件 14-5
注意事項および制約事項 14-5

Route Policy Manager の設定 14-6
IP プレフィクス リストの設定 14-6
AS パス リストの設定 14-7
コミュニティ リストの設定 14-8
ルート マップの設定 14-9
Route Policy Manager の設定確認 14-14
Route Policy Manager の設定例 14-14
関連項目 14-14
デフォルト設定 14-14
その他の関連資料 14-15
関連資料 14-15
規格 14-15

CHAPTER 15 ポリシーベース ルーティングの設定 15-1
ポリシーベース ルーティングの概要 15-2
ポリシー ルート マップ 15-2
ポリシーベース ルーティングの set 基準 15-3
ポリシーベース ルーティングのライセンス要件 15-4
ポリシーベース ルーティングの前提条件 15-4
注意事項および制約事項 15-4
ポリシーベース ルーティングの設定 15-5
ポリシーベース ルーティング機能のイネーブル化 15-5
ルート ポリシーの設定 15-6
ポリシーベース ルーティングの設定確認 15-9
ポリシーベース ルーティングの設定例 15-9
関連項目 15-9
デフォルト設定 15-9
その他の関連資料 15-10
関連資料 15-10
規格 15-10

PART 3 FHRP

CHAPTER 16 GLBP の設定 16-1
GLBP の概要 16-2
GLBP の概要 16-2
GLBP アクティブ バーチャル ゲートウェイ 16-2
GLBP バーチャル MAC アドレス割り当て 16-3
GLBP によるバーチャル ゲートウェイの冗長性 16-3

GLBP によるバーチャル フォワーダの冗長性 16-3
GLBP 認証 16-4
GLBP ロード バランシングおよびトラッキング 16-5
ハイ アベイラビリティ 16-6
仮想化サポート 16-6
GLBP のライセンス要件 16-7
GLBP の前提条件 16-7
注意事項および制約事項 16-7
GLBP の設定 16-8
GLBP 機能のイネーブル化 16-8
GLBP 認証の設定 16-8
GLBP ロード バランシングの設定 16-10
GLBP 重み付けおよびトラッキングの設定 16-11
GLBP のカスタマイズ 16-13
GLBP グループのイネーブル化 16-14
GLBP の設定確認 16-16
GLBP の設定例 16-16
デフォルト設定 16-17
その他の関連資料 16-18
関連資料 16-18
規格 16-18

CHAPTER 17 HSRP の設定 17-1
HSRP 情報 17-2
HSRP の概要 17-2
HSRP のバージョン 17-3
HSRP 認証 17-4
HSRP アドレッシング 17-4
HSRP メッセージ 17-4
HSRP ロード シェアリング 17-5
オブジェクト トラッキングおよび HSRP 17-5
ハイ アベイラビリティ 17-6
仮想化サポート 17-6
HSRP のライセンス要件 17-7
HSRPP の前提条件 17-7
注意事項および制約事項 17-7
HSRP の設定 17-8
HSRP 機能のイネーブル化 17-8
HSRP グループの設定 17-8

HSRP バージョン設定 17-10
HSRP バーチャル MAC アドレスの設定 17-10
HSRP の認証 17-11
HSRP オブジェクト トラッキングの設定 17-13
HSRP のカスタマイズ 17-15
HSRP の設定確認 17-17
HSRP の設定例 17-17
デフォルト設定 17-18
その他の関連資料 17-18
関連資料 17-18
規格 17-18

CHAPTER 18 VRRP の設定 18-1
VRRP の概要 18-2
VRRP の動作 18-2
VRRP の利点 18-3
マルチ VRRP グループ 18-4
VRRP ルータのプライオリティおよびプリエンプト 18-4
VRRP のアドバタイズメント 18-5
VRRP 認証 18-5
VRRP トラッキング 18-5
ハイ アベイラビリティ 18-6
仮想化サポート 18-6
VRRP のライセンス要件 18-7
注意事項および制約事項 18-7
VRRP の設定 18-8
VRRP 機能のイネーブル化 18-8
VRRP グループの設定 18-8
VRRP プライオリティの設定 18-10
VRRP 認証の設定 18-11
アドバタイズメント パケットのタイム インターバル設定 18-13
プリエンプトのディセーブル化 18-14
VRRP インターフェイス ステート トラッキングの設定 18-16
VRRP の設定確認 18-18
VRRP 統計情報の表示 18-18
VRRP の設定例 18-19
デフォルト設定 18-20
その他の関連資料 18-20
関連資料 18-20

CHAPTER 19 オブジェクト トラッキングの設定 19-1
オブジェクト トラッキング情報 19-2
オブジェクト トラッキングの概要 19-2
ハイ アベイラビリティ 19-2
仮想化サポート 19-3
オブジェクト トラッキングのライセンス要件 19-4
オブジェクト トラッキングの前提条件 19-4
注意事項および制約事項 19-4
オブジェクト トラッキングの設定 19-5
インターフェイスのオブジェクト トラッキング設定 19-5
ルート到達可能性のオブジェクト トラッキング設定 19-6
非デフォルト VRF のオブジェクト トラッキング設定 19-7
オブジェクト トラッキングの設定確認 19-8
オブジェクト トラッキングの設定例 19-8
関連項目 19-8
デフォルト設定 19-9
その他の関連資料 19-9
関連資料 19-9
規格 19-9

APPENDIX A Cisco NX-OS Unicast Features Release 4.x がサポートする IETF RFC A-1
BGP の FRC A-1
First-Hop Redundancy Protocol の RFC A-1
IP サービスの RFC A-2
IPv6 の RFC A-2
IS-IS の RFC A-3
OSPF の RFC A-3
RIP の RFC A-3

APPENDIX B Cisco NX-OS レイヤ 3 ユニキャスト機能、リリース 4.x の設定の制限 B-1

GLOSSARY 用語集

INDEX 索引

はじめに

このマニュアルでは、Cisco NX-OS ユニキャスト ルーティングの設定の詳細について説明します。

対象読者

このマニュアルを使用するには、IP およびルーティングのテクノロジに関する詳しい知識が必要です。

マニュアルの構成

このマニュアルは、次の章で構成されています。

タイトル	説明
第 1 章「概要」	ユニキャスト ルーティングの概要と各機能の簡単な説明を示します。
第 2 章「IPv4 の設定」	ARP と ICMP を含む IPv4 を設定し、管理する手順について説明します。
第 3 章「IPv6 の設定」	Neighbor Discovery Protocol と ICMPv6 を含む IPv6 を設定し、管理する手順について説明します。
第 4 章「DNS の設定」	DHCP および DNS クライアントを設定する手順について説明します。
第 5 章「OSPFv2 の設定」	IPv4 ネットワークのための OSPFv2 ルーティング プロトコルを設定する手順について説明します。
第 6 章「OSPFv3 の設定」	IPv6 ネットワークのための OSPFv3 ルーティング プロトコルを設定する手順について説明します。
第 7 章「EIGRP の設定」	IPv4 ネットワークのための Cisco EIGRP ルーティング プロトコルを設定する手順について説明します。
第 8 章「IS-IS の設定」	IPv4 および IPv6 ネットワークのための IS-IS ルーティング プロトコルを設定する手順について説明します。
第 9 章「ベーシック BGP の設定」	IPv4 および IPv6 ネットワークのための BGP ルーティング プロトコルの基本機能を設定する手順について説明します。
第 10 章「拡張 BGP の設定」	ルート再配布とルート集約を含む、IPv4 および IPv6 ネットワークのための BGP ルーティング プロトコルの高度な機能を設定する手順について説明します。

タイトル	説明
第 11 章「RIP の設定」	IPv4 および IPv6 ネットワークのための RIP および RIPng ルーティング プロトコルを設定する手順について説明します。
第 12 章「スタティック ルーティングの設定」	IPv4 および IPv6 ネットワークのためのスタティック ルーティングを設定する手順について説明します。
第 13 章「レイヤ 3 仮想化の設定」	レイヤ 3 仮想化を設定する手順について説明します。
第 14 章「Route Policy Manager の設定」	フィルタリングおよび再配布用の IP プレフィクス リストとルートマップを含む Route Policy Manager を設定する手順について説明します。
第 15 章「ポリシーベース ルーティングの設定」	ポリシーベース ルーティング用ルートマップを設定する手順について説明します。
第 16 章「GLBP の設定」	GLBP を設定する手順について説明します。
第 17 章「HSRP の設定」	Hot Standby Router Protocol を設定する手順について説明します。
第 18 章「VRRP の設定」	Virtual Router Redundancy Protocol を設定する手順について説明します。
第 19 章「オブジェクト トラッキングの設定」	オブジェクト追跡を設定する手順について説明します。
付録 A「Cisco NX-OS Unicast Features Release 4.x がサポートする IETF RFC」	Cisco NX-OS でサポートされる IETF RFC の一覧です。
付録 B「Cisco NX-OS レイヤ 3 ユニキャスト機能、リリース 4.x の設定の制限」	Cisco NX-OS リリース 4.x 設定の制限について説明します。

表記法

コマンドの説明では、次の表記法を使用しています。

表記	説明
太字	コマンドおよびキーワードは**太字**で示しています。
イタリック体	ユーザが値を指定する引数は、*イタリック体*で示しています。
[ ]	角カッコの中の要素は、省略可能です。
[x|y|z]	どれか 1 つを選択できる省略可能なキーワードは、角カッコで囲み、縦棒で区切って示しています。
ストリング	引用符を付けない一組の文字。ストリングの前後には引用符を使用しません。引用符を使用すると、その引用符も含めてストリングとみなされます。

出力例では、次の表記法を使用しています。

`screen` フォント	スイッチが表示する端末セッションおよび情報は、`screen` フォントで示しています。
太字の **`screen`** フォント	ユーザが入力しなければならない情報は、**太字**の **`screen`** フォントで示しています。
*イタリック体*の *`screen`* フォント	ユーザが値を指定する引数は、*イタリック体*の *`screen`* フォントで示しています。
< >	パスワードのように出力されない文字は、かぎカッコ（<>）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

このマニュアルでは、次の表記法を使用しています。

（注） 「*注釈*」です。役立つ情報や、このマニュアル以外の参照資料などを紹介しています。

注意 「*要注意*」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

関連資料

Cisco NX-OS のマニュアルは、次の URL から入手できます。

http://www.cisco.com/en/US/products/ps9372/tsd_products_support_series_home.html

Cisco NX-OS のマニュアル セットには、次の資料が含まれます。

リリース ノート

『*Cisco NX-OS Release Notes*』Release 4.0

NX-OS コンフィギュレーション ガイド

『*Cisco NX-OS Getting Started with Virtual Device Contexts*』Release 4.0

『*Cisco NX-OS Fundamentals Configuration Guide*』Release 4.0

『*Cisco NX-OS Interfaces Configuration Guide*』Release 4.0

『*Cisco NX-OS Layer 2 Switching Configuration Guide*』Release 4.0

『*Cisco NX-OS Quality of Service Configuration Guide*』Release 4.0

『*Cisco NX-OS Unicast Routing Configuration Guide*』Release 4.0

『*Cisco NX-OS Multicast Routing Configuration Guide*』Release 4.0

『*Cisco NX-OS Security Configuration Guide*』Release 4.0

『*Cisco NX-OS Virtual Device Context Configuration Guide*』Release 4.0

『*Cisco NX-OS Software Upgrade Guide*』Release 4.0

『*Cisco NX-OS Licensing Guide*』Release 4.0

『*Cisco NX-OS High Availability and Redundancy Guide*』Release 4.0

『*Cisco NX-OS System Management Configuration Guide*』Release 4.0

『*Cisco NX-OS XML Management Interface User Guide*』Release 4.0

『*Cisco NX-OS System Messages Reference*』

『*Cisco NX-OS MIB Quick Reference*』

NX-OS コマンド リファレンス

『*Cisco NX-OS Command Reference Master Index*』Release 4.0

『*Cisco NX-OS Fundamentals Command Reference*』Release 4.0

『*Cisco NX-OS Interfaces Command Reference*』Release 4.0

『*Cisco NX-OS Layer 2 Switching Command Reference*』Release 4.0

『*Cisco NX-OS Quality of Service Command Reference*』Release 4.0

『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0

『*Cisco NX-OS Multicast Routing Command Reference*』Release 4.0

『*Cisco NX-OS Security Command Reference*』Release 4.0

『*Cisco NX-OS Virtual Device Context Command Reference*』Release 4.0

『*Cisco NX-OS High Availability and Redundancy Command Reference*』Release 4.0

『*Cisco NX-OS System Management Command Reference*』Release 4.0

その他のソフトウェアのマニュアル

『*Cisco NX-OS Troubleshooting Guide*』Release 4.0

マニュアルの入手方法とサービス リクエストの提出方法

マニュアルの入手方法、サービス リクエストの提出、および追加情報の収集については、次の URL で毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。このページには、シスコの新規および改訂版の技術マニュアルの一覧も掲載されています。

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

リーダー アプリケーションを使用して、『*What's New in Cisco Product Documentation*』を Really Simple Syndication（RSS）フィードとして購読すると、情報がデスクトップに直接配信されるようになります。RSS フィードの提供は無償です。現在、シスコでは RSS バージョン 2.0 をサポートしています。

Japan TAC Web サイト

Japan TAC Web サイトでは、利用頻度の高い TAC Web サイト（http://www.cisco.com/tac）のドキュメントを日本語で提供しています。Japan TAC Web サイトには、次の URL からアクセスしてください。

http://www.cisco.com/jp/go/tac

サポート契約を結んでいない方は、「ゲスト」としてご登録いただくだけで、Japan TAC Web サイトのドキュメントにアクセスできます。

Japan TAC Web サイトにアクセスするには、Cisco.com のログイン ID とパスワードが必要です。ログイン ID とパスワードを取得していない場合は、次の URL にアクセスして登録手続きを行ってください。

http://www.cisco.com/jp/register/

CHAPTER 1

概要

この章では、Cisco NX-OS でのレイヤ 3 ユニキャスト ルーティング プロトコルの基盤となる概念を紹介します。

ここでは、次の内容を説明します。

- レイヤ 3 ユニキャスト ルーティングについて（p.1-2）
- ルーティング アルゴリズム（p.1-8）
- レイヤ 3 仮想化（p.1-11）
- Cisco NX-OS 転送アーキテクチャ（p.1-12）
- レイヤ 3 ユニキャスト ルーティング機能のまとめ（p.1-15）
- 関連資料（p.1-18）

レイヤ 3 ユニキャスト ルーティングについて

レイヤ 3 ユニキャスト ルーティングには、最適なルーティング パスの決定とパケットの交換という、2 つの基本的動作があります。ルーティング アルゴリズムを使用すると、ルータから宛先までの最適なパス（経路）を計算できます。この計算方法は、選択したアルゴリズム、ルート メトリック、そしてロード バランシングや代替パスの探索などの考慮事項により異なります。

ここでは、次の内容について説明します。

- ルーティングの基本（p.1-2）
- パケット交換（p.1-2）
- ルーティング メトリック（p.1-4）
- ルータ ID（p.1-5）
- 自律システム（p.1-5）
- 収束（p.1-6）
- ロード バランシングおよび等コスト マルチパス（p.1-6）
- ルート再配布（p.1-6）
- 管理ディスタンス（p.1-7）
- スタブ ルーティング（p.1-7）

ルーティングの基本

ルーティング プロトコルは、*メトリック*を使用して、宛先までの最適なパスを調べます。メトリックとは、パス帯域幅などの、ルーティング アルゴリズムが宛先までの最適なパスを決定するために使用する測定基準です。パスを決定しやすいように、ルーティング アルゴリズムは、ルート情報（IP 宛先アドレス、および次のルータまたは*ネクスト ホップ*のアドレスなど）を含むルーティング テーブルを初期化して維持します。宛先とネクストホップの関連付けにより、ルータは、宛先までの途中にあるネクストホップとなる特定のルータにパケットを送信すると、最適なパスで IP 宛先まで届けられることを判定できます。ルータは、着信パケットを受信すると、宛先アドレスをチェックし、このアドレスをネクストホップと関連付けようとします。ルーティング テーブルの詳細については、「ユニキャスト RIB」（p.1-12）を参照してください。

ルーティング テーブルには、パスの優先度に関するデータなどのその他の情報も含まれる場合があります。ルータはメトリックを比較して、最適なルートを決定します。また、これらのメトリックは、使用されるルーティング アルゴリズムの設計により異なります。「ルーティング メトリック」（p.1-4）を参照してください。

各ルータは互いに通信し、さまざまなメッセージを送信して、そのルーティング テーブルを維持します。ルーティング更新メッセージは、ルーティング テーブルの全部または一部で構成されるメッセージです。ルータは、他のすべてのルータからのルーティング更新情報を分析して、ネットワーク トポロジの詳細な図を構築できます。ルータ間で送信されるもう 1 つのメッセージであるリンク ステート アドバタイズメントは、他のルータに、送信側ルータのリンク状態を通知します。リンク情報を使用して、ルータが、ネットワーク宛先までの最適なルートを決定できるようにすることもできます。詳細については、「ルーティング アルゴリズム」（p.1-8）を参照してください。

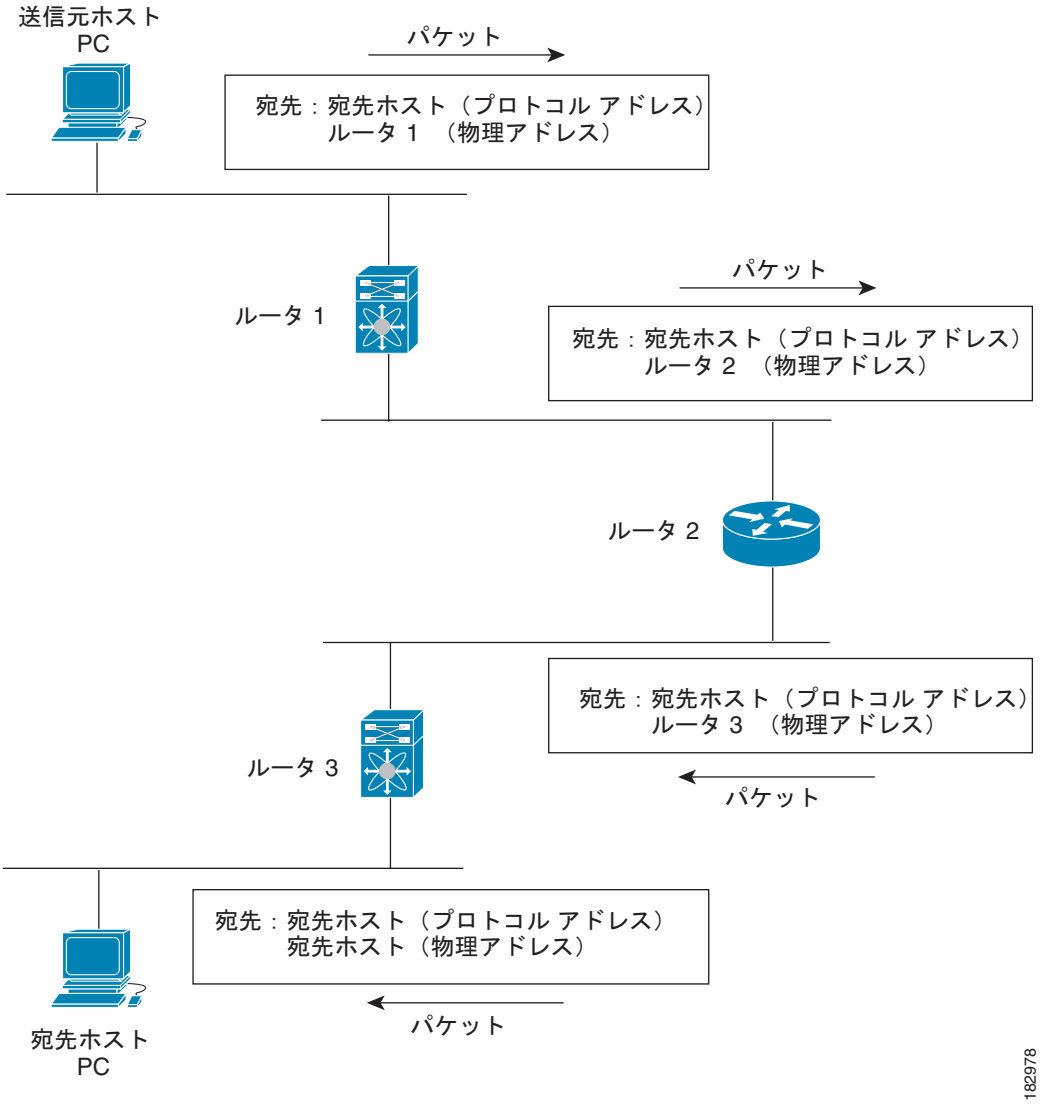
パケット交換

パケット交換では、ホストが、パケットを別のホストに送信する必要があることを決定します。なんらかの方法でルータのアドレスを入手したら、送信元ホストはパケットを明確に、宛先ホストの IP（ネットワーク レイヤ）アドレスを含むルータの物理（Media Access Control [MAC] レイヤ）アドレス宛てに送信します。

ルータは宛先の IP アドレスを調べ、ルーティング テーブルでその IP アドレスを探します。ルータでパケットの転送方法がわからない場合は、パケットは通常、廃棄されます。パケットの転送方法がわかった場合、ルータは、宛先の MAC アドレスをネクストホップ ルータの MAC アドレスに変更し、パケットを送信します。

ネクストホップが宛先のホストである場合や、同じ交換決定処理を行う別のルータである場合があります。パケットがネットワーク間を移動するにつれ、その物理アドレスは変更されますが、そのプロトコル アドレスは変わりません（図 1-1 を参照）。

図 1-1　ネットワーク上でのパケット ヘッダーの更新



ルーティング メトリック

ルーティング アルゴリズムは、多くの異なるメトリックを使用して最適なルートを決定します。高度なルーティング アルゴリズムは、複数のメトリックに基づいてルートを選択している場合があります。

ここでは、次のメトリックについて説明します。

- パスの長さ（p.1-4）
- 信頼性（p.1-4）
- ルーティング遅延（p.1-4）
- 帯域幅（p.1-4）
- 負荷（p.1-5）
- 通信コスト（p.1-5）

パスの長さ

*パス長*は、最も一般的なルーティング メトリックです。一部のルーティング プロトコルでは、各ネットワーク リンクに恣意的なコストの割り当てが可能です。この場合、パスの長さは、経由した各リンクに関連付けられたコストの合計となります。それ以外のルーティング プロトコルでは、パケットが送信元から宛先までに経由する必要のある、ルータなどのネットワーク間製品の通過回数を指定するメトリックであるホップ数が定義されます。

信頼性

ルーティング アルゴリズムとの関連における*信頼性*は、各ネットワーク リンクの信頼性（ビット誤り率で示される）です。一部のネットワーク リンクは、他のネットワーク リンクよりダウンする頻度が高い場合があります。ネットワークがダウンしたあと、特定のネットワーク リンクが他のリンクより容易に、または短時間に修復される場合もあります。信頼性のランクを割り当てるときに考慮できる信頼性係数は、一般的にネットワーク リンクに割り当てる任意の数値です。

ルーティング遅延

ルーティング*遅延*は、送信元から宛先に、ネットワーク間を通過してパケットを移動するために必要な時間の長さです。遅延は、中間のネットワーク リンクの帯域幅、経由する各ルータでのポート キュー、中間の全ネットワーク リンクでのネットワークの混雑状況、パケットが移動する物理的な距離など、多くの要素に応じて異なります。ルーティング遅延はいくつかの重要な変数の組み合わせであるため、一般的で便利なメトリックです。

帯域幅

*帯域幅*は、リンクで使用可能なトラフィック容量です。たとえば、10 ギガビット イーサネット リンクは 1 ギガビット イーサネット リンクより容量が大きく、優れています。帯域幅は、リンクで達成可能な最大スループットですが、帯域幅のより大きいリンクを経由するルートが、帯域幅のより小さいリンクを経由するルートより優れているとは限りません。たとえば、帯域幅の大きいリンクの方が混雑していると、実際には、パケットを宛先に送信するためにさらに長い時間がかかる場合があります。

負荷

*負荷*は、ルータなどのネットワーク リソースが使用状況の程度です。負荷は、CPU 使用状況や処理される 1 秒あたりのパケット数など、さまざまな方法で計算できます。これらのパラメータを継続的に監視すると、リソースに負担がかかる場合があります。

通信コスト

*通信コスト*は、リンク上でルーティングするための稼働コストの測定単位です。通信コストは重要なメトリックの 1 つで、特にパフォーマンスより稼働コストの削減が優先される場合に使用されます。たとえば、専用回線での回線遅延が公衆回線より大きくても、使用時間に応じて課金される公衆回線上でなく、自身の専用回線上でパケットを送信できます。

ルータ ID

各ルーティング処理には、*ルータ ID* が関連付けられています。ルータ ID は、システムのあらゆるインターフェイスに設定できます。ルータ ID を設定しないと、Cisco NX-OS が次の基準に基づいて、ルータ ID を選択します。

- Cisco NX-OS は、他のあらゆるインターフェイス上で loopback0 を優先します。loopback0 が存在しない場合、Cisco NX-OS は、他のあらゆるインターフェイス タイプ上で最初のループバックを優先します。
- 設定済みのループバック インターフェイスがない場合、Cisco NX-OS は、コンフィギュレーション ファイル中の最初のインターフェイスをルータ ID として使用します。Cisco NX-OS がルータ ID を選択した後にいずれかのループバック インターフェイスを設定した場合は、ループバック インターフェイスがルータ ID となります。ループバック インターフェイスが loopback0 ではなく、あとで loopback0 を IP アドレスで設定した場合は、ルータ ID が loopback0 の IP アドレスに変更されます。
- ルータ ID の元であるインターフェイスが変更されると、新しい IP アドレスがルータ ID となります。他のどのインターフェイスの IP アドレスが変更されても、ルータ ID はまったく変更されません。

自律システム

自律システム（AS）とは、単一の技術的管理エンティティにより制御されるネットワークです。AS により、グローバルな外部ネットワークが個々のルーティング ドメインに分割され、これらのドメインでは、ローカルのルーティング ポリシーが適用されます。この構成により、ルーティング ドメインの管理と一貫したポリシー設定が簡素化されます。

各 AS は、ルート*再配布*により動的にルーティング情報を交換する、複数の内部ルーティング プロトコルをサポートできます。地域インターネット レジストリにより、インターネットに直接接続する各公共 AS に一意の番号が割り当てられます。この一意の番号で、ルーティング処理と AS の両方が識別されます。表 1-1 は、AS 番号の範囲を示します。

表 1-1　AS 番号

16 ビット番号	32 ビット番号	目的
1 ～ 64511	0.1 ～ 0.64511	公共 AS（RIR により割り当てられる）[1]
64512 ～ 65534	0.64512 ～ 0.65534	専用 AS（ローカルの管理者により割り当てられる）
65535	0.65535	予約済み
該当なし	1.0 ～ 65535.65535	公共 AS（RIR により割り当てられる）

1. RIR = 地域インターネット レジストリ（Regional Internet Registries）

専用 AS 番号は内部ルーティング ドメインに使用されますが、インターネット上にルーティングされたトラフィック向けに、ルータにより変換される必要があります。ルーティング プロトコルを、専用 AS 番号が外部ネットワークにアドバタイズされるように設定しないでください。デフォルトでは、Cisco NX-OS は専用 AS 番号をルーティング更新情報から削除しません。

（注） 公共ネットワークおよび専用ネットワークの AS 番号は、Internet Assigned Number Authority（IANA; インターネット割り当て番号局）により管理されています。予約済み番号の割り当てを含む AS 番号の詳細について、または、AS 番号の登録を申請するには、次の URL を参照してください。
http://www.iana.org/

収束

ルーティング アルゴリズム測定の鍵となる要素の 1 つは、ルータがネットワーク トポロジの変化に対応するために要する時間です。リンク障害など、なんらかの理由でネットワークの一部が変化すると、さまざまなルータのルーティング情報が一致しなくなる場合があります。一部のルータでは、変化したトポロジに関する情報が更新されているが、その他のルータでは古い情報が残っているためです。*コンバージェンス*は、ネットワーク内のすべてのルータが更新され、ルーティング情報が一致するまでにかかる時間の長さです。収束時間は、ルーティング アルゴリズムによって異なります。収束が速い場合は、不正確なルーティング情報によるパッケージ損失の可能性が小さくなります。

ロード バランシングおよび等コスト マルチパス

ルーティング プロトコルでは、*ロード バランシング*または等コスト マルチパス（ECMP）を使用して、複数のパス上のトラフィックを共有できます。ルータは、特定のネットワークへのルートを複数検出すると、もっとも管理ディスタンスの低いルートをルーティング テーブルにインストールします。ルータが、同じ管理ディスタンスと宛先までのコストを持つ複数のパスを受信し、インストールすると、ロード バランシングが発生する場合があります。ロード バランシングでは、すべてのパス上にトラフィックが配布され、負荷が共有されます。使用されるパスの数は、ルーティング プロトコルによりルーティング テーブルに配置されるエントリの数に制限されます。Cisco NX-OS は、宛先までの 16 のパスをサポートします。

Enhanced IGRP（EIGRP）は、等コストでないロード バランシングもサポートしています。詳細については、第 7 章「EIGRP の設定」を参照してください。

ルート再配布

ネットワークに複数のルーティング プロトコルが設定されている場合は、各プロトコルでルート再配布を設定して、ルーティング情報を共有するように設定できます。たとえば、OSPF（Open Shortest Path First）を設定して、Border Gateway Protocol（BGP）で検出したルートをアドバタイズできます。また、スタティック ルートを、どのダイナミック ルーティング プロトコルにも再配布できます。別のプロトコルからのルートを再配布しているルータは、その再配布ルートに対する固定ルート メトリックを設定します。これにより、異なるルーティング プロトコル間で互換性のないルート メトリックの問題が回避されます。たとえば、EIGRP から OSPF に再配布されたルートには、OSPF が認識できる固定リンク コスト メトリックが割り当てられます。

ルート再配布では、管理ディスタンス（「管理ディスタンス」[p.1-7] を参照）の使用によっても、2 つの異なるルーティング プロトコルで検出されたルートが区別されます。優先ルーティング プロトコルには、より低い管理ディスタンスが与えられており、そのルートが、より高い管理ディスタンスが割り当てられた他のプロトコルからのルートに優先して選択されます。

管理ディスタンス

*管理ディスタンス*は、ルーティング情報の送信元の信頼性のランクです。値が高いほど、信頼性のランクは低くなります。一般的にルートは、複数のプロトコルを通じて検出されます。管理ディスタンスは、複数のプロトコルで検出されたルートを区別するために使用されます。もっとも管理ディスタンスが低いルートが IP ルーティング テーブルにインストールされます。

スタブ ルーティング

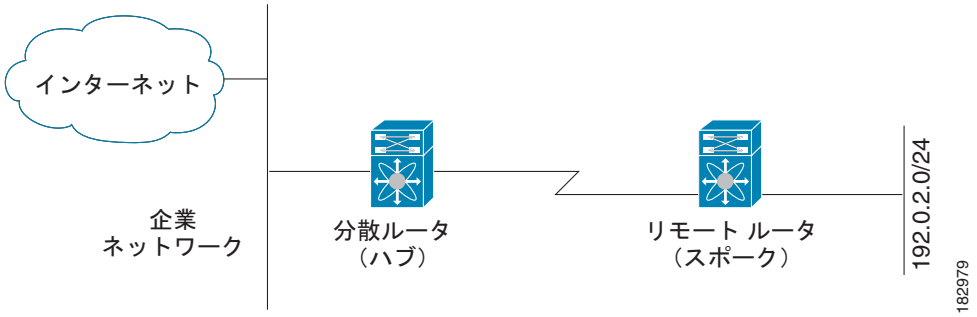
スタブ ルーティングはハブ アンド スポーク型ネットワーク トポロジで使用できます。このトポロジでは、1 つ以上の終端（スタブ）ネットワークが、1 つ以上の分散ルータ（ハブ）に接続されたリモート ルータ（スポーク）に接続されています。リモート ルータは、1 つ以上の分散ルータにのみ隣接しています。リモート ルータへと流れる IP トラフィックのルートは、分散ルータ経由のルートのみです。このタイプの設定は、分散ルータが直接 WAN に接続されている WAN トポロジで使用されるのが一般的です。分散ルータは、さらに多くのリモート ルータに接続できます。分散ルータが 100 台以上のリモート ルータに接続されていることも、よくあります。ハブ アンド スポーク型トポロジでは、リモート ルータがすべての非ローカル トラフィックを分散ルータに転送する必要があります。これにより、リモート ルータが完全なルーティング テーブルを保持する必要はなくなります。通常、分散ルータは、デフォルトのルートのみをリモート ルータに送信します。

指定されたルートのみが、リモート（スタブ）ルータから伝播されます。スタブ ルータは、要約、接続したルート、再配布されたスタティック ルート、外部ルート、内部ルートに対する照会のすべてに、「アクセスできない」メッセージで対応します。スタブとして設定されたルータは、すべての隣接ルータに特別なピア情報パケットを送信して、自身のスタブ ルータとしての状態を報告します。

スタブ ルータの状態を通知するパケットを受信した隣接ルータは、ルートについてはスタブ ルータに照会しません。また、スタブ ピアを持つルータは、そのピアについては照会しません。スタブ ルータが、すべてのピアに適切な更新情報を送信するには、分散ルータが必要です。

図 1-2 は、単純なハブ アンド スポーク型設定を示します。

図 1-2　単純なハブ アンド スポーク ネットワーク



スタブ ルーティングを使用する場合でも、リモート ルータにルータをアドバタイズできます。図 1-2 は、リモート ルータが、分散ルータのみを使用して企業ネットワークとインターネットにアクセスできることを示しています。この例では、企業ネットワークとインターネットへのパスがつねに分散ルータを経由するため、リモート ルータ上の完全なルーティング テーブルの機能は無意味です。より大規模なルーティング テーブルを使用しても、リモート ルータに必要なメモリの量が削減されるだけです。使用される帯域幅とメモリは、分散ルータでルートを要約し、フィルタリン

グすると、削減できます。このネットワーク トポロジでリモート ルータは、他のネットワークから検出されたルートを受信する必要はありません。これは、宛先がどこであっても、リモート ルータは、すべての非ローカル トラフィックを分散ルータに送信する必要があるためです。真のスタブ ネットワークを設定するには、リモート ルータへのデフォルト ルートのみを送信するよう、分散ルータを設定する必要があります。

OSPF はスタブ エリアをサポートしており、EIGRP はスタブ ルータをサポートしています。

ルーティング アルゴリズム

ルーティング アルゴリズムは、ルータが到達可能性の情報を収集し、報告する方法、トポロジの変化に対応する方法、および宛先までの最適なルートを決定する方法を決定します。ルーティング アルゴリズムにはさまざまなタイプがあり、各アルゴリズムがネットワークやルータ リソースに与える影響もさまざまです。ルーティング アルゴリズムは、最適なルートの計算に影響するさまざまなメトリックを使用します。ルーティング アルゴリズムは、スタティックまたはダイナミック、内部または外部など、タイプで分類できます。

ここでは、次の内容について説明します。

- スタティック ルートおよびダイナミック ルーティング プロトコル（p.1-8）
- 内部および外部ゲートウェイ プロトコル（p.1-9）
- ディスタンス ベクトル プロトコル（p.1-9）
- リンクステート プロトコル（p.1-9）

スタティック ルートおよびダイナミック ルーティング プロトコル

スタティック ルートは、手動で設定するルーティング テーブル エントリです。スタティック ルートは、手動で再設定しない限り、変更されません。スタティック ルートは設計が簡単で、ネットワーク トラフィックが比較的予想しやすい環境や、ネットワーク設計が比較的単純な環境での使用に適しています。

スタティック ルーティング システムはネットワークの変化に対応できないため、絶えず変化する、今日の大規模ネットワークには使用しないでください。今日のほとんどのルーティング プロトコルは、着信ルーティング更新メッセージを分析して、ネットワーク状況の変化に合わせて調整されるダイナミック ルーティング アルゴリズムを使用します。メッセージがネットワークの変化を示している場合は、ルーティング ソフトウェアがルートを計算し直して、新しいルーティング更新メッセージを送信します。これらのメッセージがネットワークを通過すると、ルータがそのアルゴリズムを再実行し、それに従ってルーティング テーブルを変更します。

適切であれば、ダイナミック ルーティング アルゴリズムをスタティック ルートで補完することができます。たとえば、各サブネットワークに IP *デフォルト ゲートウェイ*または、最後の手段であるルータ（ルーティングできないすべてのパケットが送信されるルータ）へのスタティック ルートを設定する必要があります。

内部および外部ゲートウェイ プロトコル

ネットワークを、一意のルーティング ドメインまたは AS に分割できます。AS は、管理ガイドラインの特定のセットで規制された共通の管理機関の下の内部ネットワークの一部です。AS 間でのルートを設定するルーティング プロトコルは、外部ゲートウェイ プロトコルまたはドメイン間プロトコルと呼ばれます。BGP は、外部ゲートウェイ プロトコルの例です。1 つの AS 内で使用されるルーティング プロトコルは、内部ゲートウェイ プロトコルまたはドメイン内プロトコルと呼ばれます。EIGRP および OSPF は、内部ゲートウェイ プロトコルの例です。

ディスタンス ベクトル プロトコル

ディスタンス ベクトル プロトコルは*ディスタンス ベクトル*アルゴリズム（Bellman-Ford アルゴリズムとも呼ばれます）を使用します。このアルゴリズムにより、各ルータは、そのルーティング テーブルの一部または全部を隣接ルータに送信します。ディスタンス ベクトル アルゴリズムでは、ルートが、ディスタンス（宛先までのホップ数など）および方向（ネクストホップ ルータなど）により定義されます。その後、これらのルートは、直接接続された隣接ルータにブロードキャストされます。各ルータは、これらの更新情報を使用して、ルーティング テーブルを確認し、更新します。

ルーティング ループを防ぐために、ほとんどのディスタンス ベクトル アルゴリズムは*ポイズン リバースを指定したスプリット ホライズン*を使用します。これは、インターフェイスで検出されたルートを到達不能として設定し、それをそのインターフェイスで、次の定期更新中にアドバタイズするという意味です。これにより、ルータによるルート更新が、そのルータ自体に返信されなくなります。

ディスタンス ベクトル アルゴリズムは、一定の間隔で更新を送信しますが、ルート メトリックの値の変更に応じて、更新を送信することもできます。このように送信された更新により、ルート収束時間の短縮が可能です。Routing Information Protocol（RIP）はディスタンス ベクトル プロトコルの 1 つです。

リンクステート プロトコル

リンクステート プロトコルは、SPF（最短パス優先）とも呼ばれ、情報を隣接ルータと共有します。各ルータは Link State Advertisement（LSA; リンクステート アドバタイズメント）を構築し、ここに、各リンクおよび直接接続された隣接ルータに関する情報が含まれます。

各 LSA にはシーケンス番号があります。ルータが LSA を受信し、そのリンクステート データベースを更新すると、LSA がすべての隣接ルータにフラッディングされます。ルータが同じシーケンス番号の 2 つの LSA（同じルータからの）を受信した場合は、LSA 更新ループを防ぐために、ルータは最後に受信した LSA を隣接ルータにフラッディングしません。ルータは、受信直後に LSA をフラッディングするため、リンクステート プロトコルの収束時間は最小となります。

隣接ルータの探索と隣接関係の確立は、リンクステート プロトコルの重要な部分です。隣接ルータは、特別な hello パケットを使用して探索されます。このパケットは、各隣接ルータのキープアライブ通知としても機能します。隣接関係は、隣接ルータ間のリンクステート プロトコルの一般的な動作パラメータ セットで確立されます。

ルータが受信した LSA は、そのリンクステート データベースに追加されます。各エントリは、次のパラメータで構成されます。

- ルータ ID（LSA を構築したルータの）
- ネイバー ID
- リンク コスト
- LSA のシーケンス番号

- LSA エントリの作成時からの経過時間

ルータは、リンクステート データベース上で SPF アルゴリズムを実行し、そのルータの最短パス ツリーを構築します。この SPF ツリーを使用して、ルーティング テーブルにデータが入力されます。

リンクステート アルゴリズムでは、各ルータがそのルーティング テーブル内に、ネットワーク全体の図を構築します。リンクステート アルゴリズムが小さな更新を全体的に送信するのに対し、ディスタンス ベクトル アルゴリズムは、より大きな更新を隣接ルータのみに送信します。

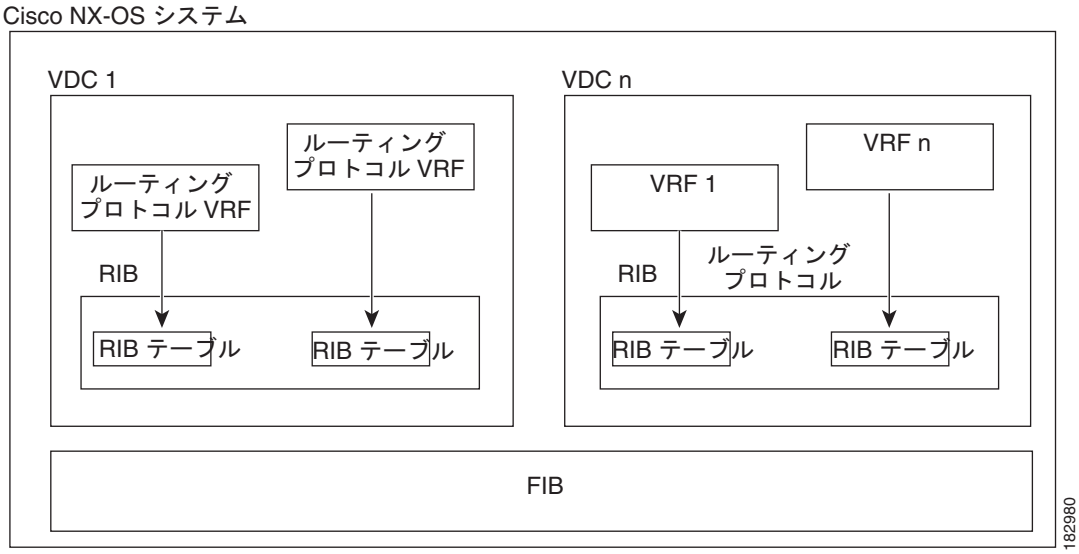
リンクステート アルゴリズムは、より短時間で収束するため、ディスタンス ベクトル アルゴリズムより、ルーティング ループがやや発生しにくくなっています。ただし、リンクステート アルゴリズムはディスタンス ベクトル アルゴリズムより、大きな CPU パワーとメモリを必要とします。リンクステート アルゴリズムは、実装とサポートにより多くの費用がかかる場合があります。リンクステート プロトコルは通常、ディスタンス ベクトル プロトコルよりスケーラブルです。

OSPF は、リンクステート プロトコルの一例です。

レイヤ 3 仮想化

Cisco NX-OS では、Virtual Device Context（VDC; 仮想デバイス コンテキスト）が採用されており、切り離された VDC およびソフトウェア障害ごとに別の管理ドメインを提供します。各 VDC は、複数の Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）および複数の Routing Information Base（*RIB*）をサポートしているため、複数のアドレス ドメインがサポートされます。各 VRF は RIB と関連付けられ、この情報が Forwarding Information Base（FIB; 転送情報ベース）により収集されます。図 1-3 は、VDC、VRF、および Cisco NX-OS システムの間の関係を示します。

図 1-3　レイヤ 3 仮想化の例



VRF は、レイヤ 3 アドレス指定ドメインを表します。各レイヤ 3 インターフェイス（論理または物理）は、1 つの VRF に属します。VRF は、1 つの VDC に属します。各 VDC は複数の VRF をサポートできます。詳細については、第 13 章「レイヤ 3 仮想化の設定」を参照してください。

VDC の詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

Cisco NX-OS 転送アーキテクチャ

Cisco NX-OS 転送アーキテクチャにより、すべてのルーティングの更新処理と、シャーシ内のすべてのモジュールへの転送情報の入力が行われます。

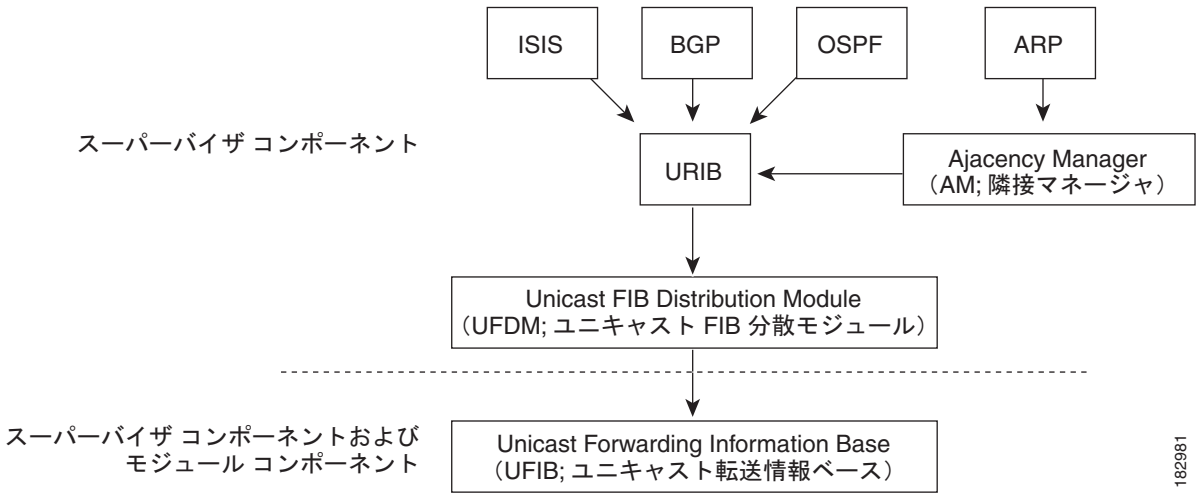
ここでは、次の内容について説明します。

- ユニキャスト RIB（p.1-12）
- 隣接マネージャ（p.1-13）
- ユニキャスト転送分散モジュール（p.1-13）
- ユニキャスト FIB（p.1-13）
- ハードウェア転送（p.1-14）
- ソフトウェア転送（p.1-15）

ユニキャスト RIB

Cisco NX-OS 転送アーキテクチャは、図 1-4 に示すように、複数のコンポーネントで構成されます。

図 1-4　Cisco NX-OS 転送アーキテクチャ



ユニキャスト RIB は、アクティブなスーパーバイザ上にあります。ユニキャスト RIB は、直接接続のルート、スタティック ルート、ダイナミック ユニキャスト ルーティング プロトコルで検出されたルートを含むルーティング テーブルを維持しています。また、Address Resolution Protocol（ARP; アドレス解決プロトコル）などの送信元から、隣接情報を収集します。ユニキャスト RIB は、ルートに最適なネクストホップを決定し、さらに Unicast FIB Distribution Module（UFDM; ユニキャスト FIB 分散モジュール）のサービスを使用して、スーパーバイザおよびモジュール上のユニキャスト FIB にデータを入力します。

各ダイナミック ルーティング プロトコルは、タイム アウトしたあらゆるルートについて、ユニキャスト RIB を更新する必要があります。そのあと、ユニキャスト RIB はそのルートを削除し、そのルートに最適なネクストホップを再計算します（代わりに使用できるパスがある場合）。

隣接マネージャ

隣接マネージャは、アクティブなスーパーバイザ上にあり、ARP、Neighbor Discovery Protocol（NDP; 近接探索プロトコル）、スタティック設定などのさまざまなプロトコルの隣接情報を維持します。もっとも基本的な隣接情報は、これらのプロトコルで探索されたレイヤ 3 からレイヤ 2 へのアドレス マッピングです。発信レイヤ 2 パケットは、隣接情報を使用して、レイヤ 2 ヘッダーの作成を終了します。

隣接マネージャは、ARP 要求による、レイヤ 3 からレイヤ 2 への特定のマッピングの探索をトリガーできます。新しいマッピングは、対応する ARP 返信を受信し、処理すると、使用できるようになります。IPv6 の場合は、AM が NDP からの、レイヤ 3 からレイヤ 2 へのマッピング情報を探索します。第 3 章「IPv6 の設定」を参照してください。

ユニキャスト転送分散モジュール

ユニキャスト 転送分散モジュールはアクティブなスーパーバイザ上にあり、ユニキャスト RIB やその他の送信元からの転送パス情報を配布します。ユニキャスト RIB は転送情報を生成し、この情報がユニキャスト FIB により、スタンバイ スーパーバイザとスタンバイ モジュール上のハードウェア転送テーブルにプログラミングされます。また、ユニキャスト転送分散モジュールは、新規挿入されたモジュールへの FIB 情報のダウンロードも行います。

ユニキャスト転送分散モジュールは、隣接情報を収集し、ユニキャスト FIB でのルートの更新時に、この情報およびその他のプラットフォーム依存の情報を書き直し（リライトし）ます。隣接情報およびリライト情報には、インターフェイス、ネクストホップ、およびレイヤ 3 からレイヤ 2 へのマッピング情報が含まれています。インターフェイスとネクストホップの情報は、ユニキャスト RIB からのルート更新情報で受信します。レイヤ 3 からレイヤ 2 へのマッピングは、隣接マネージャから受信します。

ユニキャスト FIB

ユニキャスト FIB は、スーパーバイザ モジュールとスイッチング モジュール上にあり、ハードウェア転送エンジンが使用する情報を構築します。ユニキャスト FIB は、ユニキャスト転送分散モジュールからルート更新情報を受信し、ハードウェア転送エンジンにプログラミングされるよう、この情報を送信します。ユニキャスト FIB は、ルート、パス、隣接関係の追加、削除、変更を管理します。

ユニキャスト FIB は、VRF ごと、および address-family ごとに維持されます。つまり、設定された各 VRF について、IPv4 用に 1 つ、IPv6 用に 1 つ維持されます。ルート更新メッセージに基づいて、ユニキャスト FIB は、VRF ごとのプレフィクスとネクストホップ隣接情報データベースを維持します。ネクストホップ隣接データ構造には、ネクストホップの IP アドレスとレイヤ 2 リライト情報が含まれます。同じネクストホップ隣接情報構造を複数のプレフィクスで使用できます。

ルーティング情報と転送情報の表示

CLI を使用して、ルーティング テーブルや転送テーブルを表示できます。

次の例は、**show routing** コマンド出力を示します。

```
switch# show routing
IP Route Table for Context "default"
'*' denotes best ucast next-hop       '**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]

0.0.0.0/0, 1 ucast next-hops, 0 mcast next-hops
   *via 10.1.1.1, mgmt0, [1/0], 5d21h, static
0.0.0.0/32, 1 ucast next-hops, 0 mcast next-hops
   *via Null0, [220/0], 1w6d, local, discard
10.1.0.0/22, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.1.55, mgmt0, [0/0], 5d21h, direct
10.1.0.0/32, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.0.0, Null0, [0/0], 5d21h, local
10.1.1.1/32, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.1.1, mgmt0, [2/0], 5d16h, am
10.1.1.55/32, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.1.55, mgmt0, [0/0], 5d21h, local
10.1.1.253/32, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.1.253, mgmt0, [2/0], 5d20h, am
10.1.3.255/32, 1 ucast next-hops, 0 mcast next-hops, attached
   *via 10.1.3.255, mgmt0, [0/0], 5d21h, local
255.255.255.255/32, 1 ucast next-hops, 0 mcast next-hops
   *via Eth Inband Port, [0/0], 1w6d, local
```

次の例は、**show ip adjacency** CLI コマンドで表示される隣接情報を示します。

```
switch# show ip adjacency

IP Adjacency Table for context default
Total number of entries: 2
Address         Age       MAC Address     Pref Source     Interface       Best
10.1.1.1        02:20:54  00e0.b06a.71eb  50   arp        mgmt0           Yes
10.1.1.253      00:06:27  0014.5e0b.81d1  50   arp        mgmt0           Yes
switch
```

また、**show ip fib** コマンドを使用して、ユニキャスト FIB の詳細も表示できます。

ハードウェア転送

Cisco NX-OS は、分散パケット転送をサポートしています。入力ポートは、パケット ヘッダーから該当する情報を取得し、その情報をローカル スイッチング エンジンに渡します。ローカル スイッチング エンジンはレイヤ 3 ルックアップを行い、この情報を使って、パケット ヘッダーをリライトします。入力モジュールは、パケットを出力ポートに転送します。出力ポートが別のモジュール上にある場合は、スイッチ ファブリックを使って、パケットが出力モジュールに転送されます。出力モジュールは、レイヤ 3 転送決定には関与しません。

スーパーバイザ上とすべてのモジュール上の転送テーブルは同じです。

また、**show platform fib** または **show platform forwarding** コマンドを使用すると、ハードウェア転送の詳細が表示されます。

ソフトウェア転送

Cisco NX-OS のソフトウェア転送パスは、おもに、ハードウェアでサポートされない機能、またはハードウェア処理中に発生したエラーへの対処に使用されます。通常、IP オプション付きのパケットまたはフラグメンテーションの必要なパケットは、アクティブなスーパーバイザ上の CPU に渡されます。ソフトウェアでの切り替えが必要なパケットや終端される必要のあるパケットはすべて、スーパーバイザに渡されます。スーパーバイザは、ユニキャスト RIB および隣接マネージャから提供された情報を使用して、転送の決定を下します。モジュールは、ソフトウェア転送パスには関与しません。

ソフトウェア転送は、コントロール プレーン ポリシーおよびレート リミッタによって管理されます（『*Cisco NX-OS Security Configuration Guide, Release 4.0*』を参照）。

レイヤ 3 ユニキャスト ルーティング機能のまとめ

ここでは、Cisco NX-OS でサポートされるレイヤ 3 ユニキャスト機能およびプロトコルを簡単に説明します。

ここでは、次の内容について説明します。

- IPv4 と IPv6（p.1-15）
- IP サービス（p.1-15）
- OSPF（p.1-16）
- EIGRP（p.1-16）
- IS-IS（p.1-16）
- BGP（p.1-16）
- RIP（p.1-16）
- スタティック ルーティング（p.1-17）
- レイヤ 3 仮想化（p.1-17）
- Route Policy Manager（p.1-17）
- ポリシーベース ルーティング（p.1-17）
- First-Hop Redundancy Protocol（FHRP）（p.1-17）
- オブジェクト追跡（p.1-18）

IPv4 と IPv6

レイヤ 3 は、IPv4 プロトコルまたは IPv6 プロトコルを使用します。IPv6 は新しい IP プロトコルで、世界中で広く展開され、使用されているインターネット プロトコルである IPv4 に代わるものとして設計されました。IPv6 では、ネットワーク アドレス ビット数が 32 ビット（IPv4 の場合）から 128 ビットに増やされています。詳細については、第 2 章「IPv4 の設定」または第 3 章「IPv6 の設定」を参照してください。

IP サービス

IP サービスには、DHCP クライアントおよび Domain Name System（DNS; ドメイン ネーム システム）クライアントがあります。詳細については、第 4 章「DNS の設定」を参照してください。

OSPF

OSPF プロトコルは、AS 内のネットワーク到達可能性情報の交換に使用されるリンクステート ルーティング プロトコルです。各 OSPF ルータは、そのアクティブなリンクに関する情報を隣接ルータにアドバタイズします。リンク情報には、リンク タイプ、リンク メトリック、およびリンクに接続された隣接ルータが含まれます。このリンク情報を含むアドバタイズメントは、リンクステート アドバタイズメントと呼ばれます。詳細については、第 5 章「OSPFv2 の設定」を参照してください。

EIGRP

EIGRP プロトコルは、ディスタンス ベクトルとリンクステートの両ルーティング プロトコルの特徴を備えたユニキャスト ルーティング プロトコルです。これは、シスコ専用ルーティング プロトコルである IGRP の改良バージョンです。EIGRP は、典型的なディスタンス ベクトル ルーティング プロトコルのように、ルートを提供するために隣接ルータを必要とします。また、リンクステート プロトコルのように、隣接ルータからアドバタイズされたルートからネットワーク トポロジを構築し、この情報を使用して、ループの発生しない、宛先までのパスを選択します。詳細については、第 7 章「EIGRP の設定」を参照してください。

IS-IS

Intermediate System-to-Intermediate System（IS-IS）プロトコルは、International Organization for Standardization（ISO; 国際標準化機構）10589 で指定されたドメイン内 OSI（Open System Interconnection; オープン システム インターコネクション）ダイナミック ルーティング プロトコルです。IS-IS ルーティング プロトコルはリンクステート プロトコルです。IS-IS には、次の機能があります。

- 階層型ルーティング
- クラスレス動作
- 新情報の高速フラッディング
- 短時間での収束
- 高いスケーラビリティ

詳細については、第 8 章「IS-IS の設定」を参照してください。

BGP

BGP は AS 間ルーティング プロトコルです。BGP ルータは、信頼性の高い転送メカニズムとして Transmission Control Protocol（TCP）を使用し、他の BGP ルータにネットワーク到達可能性情報をアドバタイズします。ネットワーク到達可能性情報には、宛先ネットワーク プレフィクス、宛先に到達するまでに通過する必要のある AS のリスト、およびネクストホップ ルータが含まれます。到達可能性情報には、ルートの優先度、ルートの始点、コミュニティなどの詳細なパス属性が含まれます。詳細については、第 9 章「ベーシック BGP の設定」および第 10 章「拡張 BGP の設定」を参照してください。

RIP

RIP は、ホップ数をメトリックとして使用するディスタンス ベクトル プロトコルです。RIP は、世界中のインターネットでトラフィックのルーティングに広く使用されています。また、IGP であるため、単一の AS 内でルーティングを行います。詳細については、第 11 章「RIP の設定」を参照してください。

スタティック ルーティング

スタティック ルーティングを使用して、宛先までの一定のルートを入力できます。この機能は、単純なトポロジの小規模ネットワークでは便利です。また、スタティック ルーティングは、他のルーティング プロトコルとともに、デフォルト ルートおよびルート配布の管理に使用されます。詳細については、第 12 章「スタティック ルーティングの設定」を参照してください。

レイヤ 3 仮想化

仮想化を使用すると、複数の管理ドメインにわたる物理リソースを共有できます。Cisco NX-OS は VDC をサポートしているため、Cisco NX-OS システム内に複数の仮想システムを作成できます。各 VDC は互いに孤立しているため、1 つの VDC 内に問題が発生しても、他のどの VDC にも影響しません。VDC は、相互にセキュリティが確保されています。各 VDC にそれぞれ別のネットワーク オペレータを割り当てることができます。これらのネットワーク オペレータは、別の VDC の設定を表示することも、管理することもできません。

Cisco NX-OS は、VRF を含むレイヤ 3 仮想化もサポートしています。VRF では、レイヤ 3 ルーティング プロトコルを設定するための別のアドレス ドメインが提供されます。詳細については、第 13 章「レイヤ 3 仮想化の設定」を参照してください。

Route Policy Manager

Route Policy Manager は、Cisco NX-OS でルート フィルタリング機能を提供します。Route Policy Manager はルートマップを使用して、さまざまなルーティング プロトコルや、特定のルーティング プロトコル内のさまざまなエンティティ間で配布されたルートをフィルタリングします。フィルタリングは、特定の一致基準に基づいて行われます。これは、アクセス コントロール リストによるパケット フィルタリングに似ています。詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

ポリシーベース ルーティング

ポリシーベース ルーティングは、Route Policy Manager を使用してポリシー ルート フィルタを作成します。これらのポリシー ルート フィルタでは、パケットの送信元またはパケット ヘッダーのその他フィールドに基づいて、指定されたネクストホップにパケットを転送できます。ポリシー ルートは、拡張 IP アクセス リストにリンクできるため、プロトコル タイプやポート番号などに基づいてルーティングが行われる場合もあります。詳細については、第 15 章「ポリシーベース ルーティングの設定」を参照してください。

First-Hop Redundancy Protocol（FHRP）

FHRP を使用すると、ホストで冗長接続が可能となります。アクティブなファーストホップ ルータがダウンした場合は、それの機能を引き継ぐスタンバイ ルータが、FHRP により自動的に選択されます。アドレスは仮想のものであり、FHRP グループ内の各ルータ間で共有されているため、ホストを新しい IP アドレスで更新する必要はありません。Gateway Load Balancing Protocol（GLBP）の詳細については、第 16 章「GLBP の設定」を参照してください。

オブジェクト追跡

オブジェクト追跡を使用すると、インターフェイス回線プロトコル状態、IP ルーティング、ルート到達可能性などの、ネットワーク上の特定のオブジェクトを追跡し、追跡したオブジェクトの状態が変化したときに対処することができます。この機能により、ネットワークのアベイラビリティの向上し、オブジェクトがダウン状態となった場合の回復時間が短縮されます。詳細については、第 19 章「オブジェクト トラッキングの設定」を参照してください。

関連資料

次のシスコ マニュアルは、レイヤ 3 機能に関連するものです。

- 『*Cisco NX-OS Multicast Routing Configuration Guide*』Release 4.0
- 『*Cisco NX-OS High Availability and Redundancy Guide*』Release 4.0
- 『*Cisco NX-OS Virtual Device Context Configuration Guide*』Release 4.0
- AS 番号の詳細については、次のページを参照してください。
  http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-1/autonomous_system_numbers.html

PART 1

IP

CHAPTER 2

IPv4 の設定

この章では、デバイス上での Internet Protocol version 4（IPv4）（アドレス指定を含む）、Address Resolution Protocol（ARP; アドレス解決プロトコル）、および Internet Control Message Protocol（ICMP）の設定方法を説明します。

ここでは、次の内容を説明します。

- IPv4 について（p.2-2）
- IPv4 のライセンス要件（p.2-7）
- IPv4 の前提条件（p.2-7）
- 注意事項および制約事項（p.2-7）
- IPv4 の設定（p.2-8）
- IPv4 設定の確認（p.2-13）
- IPv4 設定の例（p.2-13）
- デフォルト設定（p.2-13）
- その他の関連資料（p.2-14）

IPv4 について

デバイス上で IP を設定し、ネットワーク インターフェイスに IP アドレスを割り当てることができます。IP アドレスを割り当てると、インターフェイスがイネーブルになり、そのインターフェイス上のホストと通信できるようになります。

IP アドレスは、デバイス上でプライマリまたはセカンダリとして設定できます。インターフェイスには、1 つのプライマリ IP アドレスと複数のセカンダリ アドレスを設定できます。デバイスが生成したパケットは、つねにプライマリ IPv4 アドレスを使用するため、インターフェイス上のすべてのネットワーキング デバイスは、同じプライマリ IP アドレスを共有する必要があります。各 IPv4 パケットは、送信元または宛先の IP アドレスからの情報に基づいています。「複数の IPv4 アドレス」（p.2-2）を参照してください。

サブネットを使用して、IP アドレスをマスクできます。マスクは、IP アドレスがどのサブネットに属するかを決定するために使用されます。IP アドレスには、ネットワーク アドレスとホスト アドレスが含まれます。マスクで、IP アドレス中のネットワーク番号を示すビットが識別できます。マスクを使用してネットワークをサブネット化した場合、そのマスクはサブネット マスクと呼ばれます。サブネット マスクは 32 ビット値で、これにより IP パケットの受信者は、IP アドレスのネットワーク ID 部分とホスト ID 部分を区別できます。

Cisco NX-OS システムの IP 機能は、スーパーバイザ モジュールで終端する IPv4 パケットを処理し、IPv4 パケットを転送する役割を果たしています。この役割には、IPv4 ユニキャスト / マルチキャスト ルート ルックアップ、RPF チェック、およびソフトウェア アクセス コントロール リスト / ポリシーベース ルーティング（ACL/PBR）転送が含まれます。また、IP 機能は、ネットワーク インターフェイス IP アドレス設定、重複アドレス チェック、スタティック ルート、および IP クライアントのパケット送信 / 受信インターフェイスも管理します。

ここでは、次の内容について説明します。

- 複数の IPv4 アドレス（p.2-2）
- Address Resolution Protocol（p.2-3）
- ARP キャッシング（p.2-4）
- ARP キャッシュのスタティック エントリおよびダイナミック エントリ（p.2-4）
- ARP を使用しないデバイス（p.2-4）
- Inverse ARP（p.2-4）
- Reverse ARP（p.2-5）
- プロキシ ARP（p.2-5）
- ローカル プロキシ ARP（p.2-6）
- ICMP（p.2-6）
- 仮想化のサポート（p.2-6）

複数の IPv4 アドレス

Cisco NX-OS システムは、インターフェイスごとに複数の IP アドレスをサポートしています。さまざまな状況に備え、いくつでもセカンダリ アドレスを指定できます。もっとも一般的な状況は次のとおりです。

- 特定のネットワーク インターフェイスのホスト IP アドレスの数が不足している場合。たとえば、サブネット化により、論理サブネットごとに 254 までのホストを使用できるが、物理サブネットの 1 つに 300 のホスト アドレスが必要な場合は、ルータ上またはアクセス サーバ上でセカンダリ IP アドレスを使用して、1 つの物理サブネットで 2 つの論理サブネットを使用できます。

- 1 つのネットワークの 2 つのサブネットは、別の方法で、別のネットワークにより分離できる場合があります。別のネットワークによって物理的に分離された複数のサブネットから、セカンダリ アドレスを使用して、1 つのネットワークを作成できます。このような場合、最初のネットワークは、2 番めのネットワークの上に拡張されます。つまり、上の階層となります。サブネットは、同時に複数のアクティブなインターフェイス上に表示することはできません。

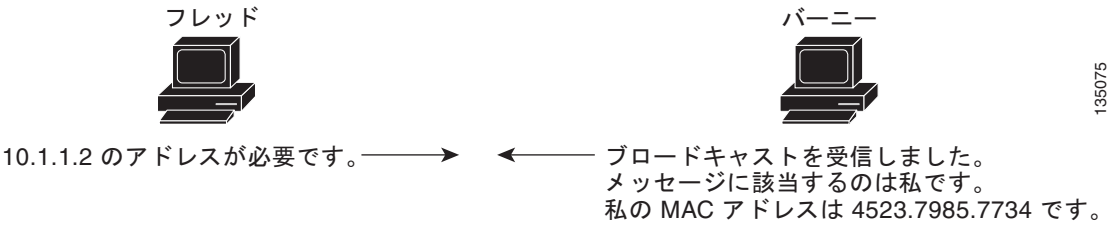
(注)　ネットワーク セグメント上のいずれかのデバイスがセカンダリ IPv4 アドレスを使用している場合は、同じネットワーク インターフェイス上の他のすべてのデバイスも、同じネットワークまたはサブネットからのセカンダリ アドレスを使用する必要があります。ネットワーク セグメント上で、一貫性のない方法でセカンダリ アドレスを使用すると、ただちにルーティング ループが発生する可能性があります。

Address Resolution Protocol

ネットワーキング デバイスおよびレイヤ 3 スイッチは ARP を使用して、IP（ネットワーク レイヤ）アドレスを物理（Media Access Control [MAC] レイヤ）アドレスにマッピングし、IP パケットがネットワーク上に送信されるようにします。デバイスは、他のデバイスにパケットを送信する前に自身の ARP キャッシュを調べて、MAC アドレスまたは対応する宛先デバイスの IP アドレスがないかを確認します。エントリがまったくない場合、送信元のデバイスは、ネットワーク上の全デバイスにブロードキャスト メッセージを送信します。

各デバイスは、問い合わせられた IP アドレスを自身のアドレスと比較します。一致する IP アドレスを持つデバイスのみが、デバイスの MAC アドレスを含むパケットとともにデータを送信したデバイスに返信します。送信元デバイスは、あとで参照できるよう、宛先デバイスの MAC アドレスをその ARP テーブルに追加し、データリンク ヘッダーおよびトレーラを作成してパケットをカプセル化し、データの転送へと進みます。図 2-1 は、ARP ブロードキャストと応答処理を示します。

図 2-1　ARP 処理



宛先デバイスが、別のデバイスを挟んだリモート ネットワーク上にあるときは、同じ処理が行われますが、データを送信するデバイスが、デフォルト ゲートウェイの MAC アドレスを求める ARP 要求を送信する点が異なります。アドレスが解決され、デフォルト ゲートウェイがパケットを受信したあとに、デフォルト ゲートウェイは、接続されているネットワーク上で宛先の IP アドレスをブロードキャストします。宛先デバイスのネットワーク上のデバイスは、ARP を使用して宛先デバイスの MAC アドレスを取得し、パケットを配信します。ARP はデフォルトでイネーブルにされています。

ARP キャッシング

ARP キャッシングにより、ブロードキャストが最小になり、無駄に使用されるネットワーク リソースが制限されます。IP アドレスの MAC アドレスへのマッピングは、ネットワーク間でパケットが送信されるたびに、ネットワーク上の各ホップ（デバイス）で行われるため、ネットワークのパフォーマンスに影響する場合があります。

ARP キャッシングでは、ネットワーク アドレスと関連付けられたデータリンク アドレスが一定の期間、メモリに格納されるため、パケットが送信されるたびに同じアドレスを求めてブロードキャストする場合の、貴重なネットワーク リソースの使用が最小限となります。キャッシュ エントリは、定期的に失効するよう設定されているため、保守が必要です。これは、古い情報が無効となる場合があるためです。ネットワーク上のすべてのデバイスは、アドレスのブロードキャストに従ってアドレス テーブルを更新します。

ARP キャッシュのスタティック エントリおよびダイナミック エントリ

スタティック ルートの使用時には、各デバイスの各インターフェイスの IP アドレス、サブネット マスク、ゲートウェイ、および対応する MAC アドレスを手動で設定する必要があります。スタティック ルーティングを使用すると、管理を強化できますが、より多くのルーティング テーブル保守作業が必要となります。ルートを追加または変更するたびに、テーブルの更新が必要となるためです。

ダイナミック ルーティングは、ネットワーク上のデバイスが相互にルーティング テーブル情報を交換できるプロトコルを使用します。ダイナミック ルーティングは、キャッシュに制限時間を追加しない限り、ルーティング テーブルが自動更新されるため、スタティック ルーティングより効率的です。デフォルトの制限時間は 25 分ですが、キャッシュから追加および削除されるルートがネットワークに数多く存在する場合は、制限時間を変更できます。

ARP を使用しないデバイス

ネットワークが 2 つのセグメントに分割されると ブリッジによりセグメントが結合され、各セグメントへのトラフィックが MAC アドレスに基づいてフィルタリングされます。ブリッジは、MAC アドレスのみを使用する専用アドレス テーブルを構築します。これに対し、デバイスは、IP アドレスおよび対応する MAC アドレスの両方を含む ARP キャッシュを持っています。

パッシブ ハブは、ネットワーク内の他のデバイスを物理的に接続する集中接続デバイスです。パッシブ ハブはそのすべてのポートでデバイスにメッセージを送信し、レイヤ 1 で動作しますが、アドレス テーブルを保持しません。

レイヤ 2 スイッチは、すべてのポートからメッセージを送信するハブとは異なり、メッセージの宛先であるデバイスに接続されるポートを決定し、そのポートにのみ送信します。これに対して、レイヤ 3 スイッチは、ARP キャッシュ（テーブル）を構築するデバイスです。

Inverse ARP

Inverse ARP は、Asynchronous Transfer Mode（ATM; 非同期転送モード）ネットワークではデフォルトでイネーブルにされており、ATM マップ エントリを作成します。また、接続先であるサーバ（またはリレー エージェント）にパケットを送信するために必要です。Inverse ARP は aal5snap カプセル化タイプでのみサポートされています。

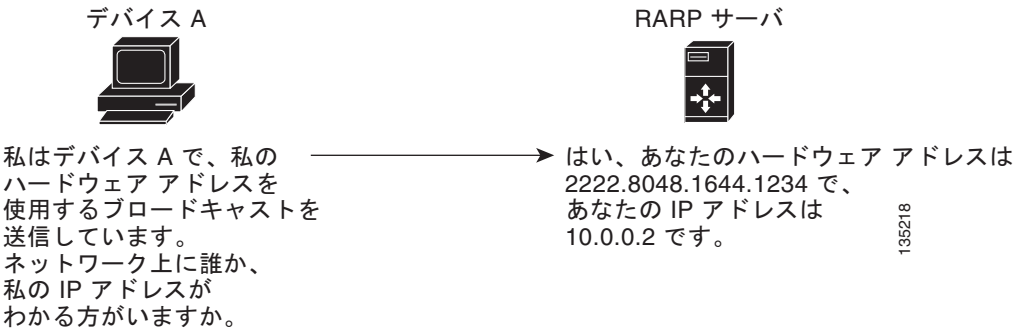
ブロードキャスト パケットが使用されるため、複数のインターフェイス用の他のカプセル化タイプを使用して IP アドレスを取得できます。ただし、ユニキャスト パケットの送信には失敗します。これは、ATM マップ エントリも DHCP 更新も存在せず、リリースも失敗するためです。

Reverse ARP

RFC 903 で定義された Reverse ARP（RARP）は ARP と同様に機能しますが、RARP 要求パケットが MAC アドレスではなく、IP アドレスを要求する点が異なります。RARP は多くの場合、ディスクレス ワークステーションで使用されます。これは、このタイプのデバイスには、起動時に使用する IP アドレスを格納する手段がないためです。認識できるアドレスは MAC アドレスのみで、これはハードウェアに焼き付けられているためです。

RARP を使用するには、ルータ インターフェイスとして、同じネットワーク セグメント上に RARP サーバが必要です。図 2-2 は、RARP の機能の図解です。

図 2-2　Reverse ARP



RARP には、いくつかの制限があります。これらの制限により、ほとんどの企業では、DHCP を使用してダイナミックに IP アドレスを割り当てています。DHCP は、RARP よりコスト効率が高く、必要な保守作業も少ないためです。もっとも重要な制限は次のとおりです。

- RARP はハードウェア アドレスを使用するため、多くの物理ネットワークを含む大規模なネットワークの場合は、各セグメント上に、冗長性のための追加サーバを備えた RARP サーバが必要です。各セグメントに 2 台のサーバを保持すると、コストがかかります。
- 各サーバは、ハードウェア アドレスと IP アドレスのスタティック マッピングのテーブルで設定する必要があります。IP アドレスの保守は困難です。
- RARP は、ホストの IP アドレスのみを提供し、サブネット マスクもデフォルト ゲートウェイも提供しません。

プロキシ ARP

プロキシ ARP を使用すると、物理的に 1 つのネットワーク上に存在するデバイスが、論理的に、同じデバイスまたはファイアウォールに接続された別の物理ネットワークの一部として表示されます。プロキシ ARP で、プライベート ネットワーク上のプライベート IP アドレスを持つデバイスをルータの背後に隠すと同時に、このデバイスを、ルータの前のパブリック ネットワーク上に表示できます。ルータはそのアイデンティティを隠すことにより、実際の宛先までパケットをルーティングする役割を担います。プロキシ ARP を使用すると、サブネット上のデバイスは、ルーティングもデフォルト ゲートウェイも設定せずにリモート サブネットまで到達できます。

複数のデバイスが同じデータリンク レイヤ のネットワークでなく、同じ IP ネットワーク内にある場合、これらのデバイスは相互に、ローカル ネットワーク上にあるかのようにデータを送信しようとします。ただし、これらのデバイスを隔てるルータは、ブロードキャスト メッセージを送信しません。これは、ルータがハードウェア レイヤのブロードキャストを渡さず、アドレスが解決されないためです。

デバイスでプロキシ ARP をイネーブルにし、ARP 要求を受信すると、プロキシ ARP はこれを、ローカル LAN 上にないシステムに対する要求とみなします。デバイスは、ブロードキャストの宛先であるリモートの宛先であるかのように、自身の MAC アドレスをリモートの宛先の IP アドレスに関連付ける ARP 応答で応答します。ローカル デバイスは、自身が宛先に直接、接続されていると認識していますが、実際には、そのパケットは、ローカル デバイスによりローカル サブネットワークから宛先のサブネットワークへと転送されています。デフォルトでは、プロキシ ARP はディセーブルになっています。

ローカル プロキシ ARP

ローカル プロキシ ARP を使用して、通常はルーティングが不要なサブネット内の IP アドレスを求める ARP 要求に対して、デバイスが応答できるようにすることができます。ローカル プロキシ ARP をイネーブルにすると、ARP は、サブネット内の IP アドレスを求めるすべての ARP 要求に応答し、サブネット内のホスト間ですべてのトラフィックを転送します。この機能は、ホストが接続されているデバイスの設定により意図的に、ホストの直接通信が禁止されているサブネットでのみ使用してください。

ICMP

ICMP を使用して、IP 処理に関連するエラーおよびその他の情報を報告するメッセージ パケットを提供できます。ICMP は、ICMP 宛先到達不能メッセージ、ICMP エコー要求（2 つのホスト間でパケットを往復送信する）、およびエコー返信メッセージなどのエラー メッセージを生成します。ICMP は多くの診断機能も備えており、ホストへのエラー パケットの送信およびリダイレクトが可能です。デフォルトでは、ICMP がイネーブルにされています。

次に示すのは、ICMP メッセージ タイプの一部です。

- ネットワーク エラー メッセージ
- ネットワーク混雑メッセージ
- トラブルシューティング情報
- タイムアウト告知

（注） ICMP リダイレクトは、ローカル プロキシ ARP 機能がイネーブルであるインターフェイス上ではディセーブルにされています。

仮想化のサポート

IPv4 は、Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）をサポートしています。VRF は Virtual Device Contexts（VDC; 仮想化デバイス コンテキスト）内にあります。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』の第 13 章「レイヤ 3 仮想化の設定」を参照してください。

IPv4 のライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	IP にはライセンスは不要です。ライセンス パッケージに含まれない機能はいずれも、Cisco NX-OS システム イメージにバンドルされており、無償で提供されます。NX-OS ライセンス方式の詳細説明については、『*Cisco NX-OS Licensing Guide*』を参照してください。

IPv4 の前提条件

IPv4 には、次の前提条件があります。

- レイヤ 3 インターフェイス上でのみ設定可能です。

注意事項および制約事項

IPv4 には、次の注意事項、および制約、制限事項があります。

- セカンダリ IP アドレスは、プライマリ IP アドレスの設定後にのみ設定できます。

IPv4 の設定

ここでは、次の内容について説明します。

- IPv4 アドレス指定の設定（p.2-8）
- 複数の IP アドレスの設定（p.2-9）
- スタティック ARP エントリの設定（p.2-10）
- プロキシ ARP の設定（p.2-11）
- ローカル プロキシ ARP の設定（p.2-12）

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

IPv4 アドレス指定の設定

ネットワーク インターフェイスにプライマリ IP アドレスを割り当てることができます。

操作の前に

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）。

コマンドの一覧

1. ***config t***
2. ***interface ethernet*** *number*
3. **ip address** *ip-address/length*
4. ***show ip interface***
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ethernet** *number*
	
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	**ip address** *ip-address/length* [secondary] **例：** switch(config-if)# ip address 192.2.1.1 255.0.0.0	インターフェイスにプライマリまたはセカンダリ IPv4 アドレスを指定します。 • ネットワーク マスクは、ドットで 4 つの部分に分けられている 10 進数のアドレスです。たとえば、255.0.0.0 は、1 に等しい各ビットが、ネットワーク アドレスに属した対応するアドレス ビットを意味することを示します。 • ネットワーク マスクは、スラッシュ（/）および数字、つまり、プレフィクス長として示される場合もあります。プレフィクス長は、プレフィクス（アドレスのネットワーク部分）を構成する、アドレスの高位隣接ビットの数を示す 10 進数値です。スラッシュは 10 進数値の前に置かれ、IP アドレスとスラッシュの間にスペースは入りません。
ステップ 4	**show ip interface** **例：** switch(config-if)# show ip interface	（任意）IPv4 に設定されたインターフェイスを表示します。
ステップ 5	***copy running-config startup-config*** **例：** switch(config-if)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、IPv4 アドレスを割り当てる例を示します。

```
switch# config t
switch(config)# interface ethernet 2/3
switch(config-if)# ip address 192.2.1.1 255.0.0.0
switch(config-if)# copy running-config startup-config
```

複数の IP アドレスの設定

セカンダリ IP アドレスは、プライマリ IP アドレスの設定後にのみ追加できます。

操作の前に

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）。

コマンドの一覧

1. ***config t***
2. ***interface ethernet*** *number*
3. **ip address** *ip-address/length*
4. ***show ip interface***
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ethernet** *number*
	
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip address** *ip-address/length* [secondary] **例：** `switch(config-if)# ip address 192.2.1.1 255.0.0.0 secondary`	設定したアドレスをセカンダリ IPv4 アドレスとして指定します。
ステップ 4	**show ip interface** **例：** `switch(config-if)# show ip interface`	(任意)IPv4 に設定されたインターフェイスを表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	(任意）この設定変更を保存します。

スタティック ARP エントリの設定

デバイス上でスタティック ARP エントリを設定して、IP アドレスを MAC ハードウェア アドレスにマッピングできます。

操作の前に

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)。

コマンドの一覧

1. **config t**
2. **interface ethernet** *number*
3. **ip arp** *ipaddr mac_addr*
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t**	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 ethernet** *number*
	
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	**ip arp** *ipaddr mac_addr* **例：** `switch(config-if)# ip arp 192.2.1.1 0019.076c.1a78`	IP アドレスを MAC アドレスにスタティック エントリとして関連付けます。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、スタティック ARP エントリを設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 2/3
switch(config-if)# ip arp 92.2.1.1 0019.076c.1a78
switch(config-if)# copy running-config startup-config
```

プロキシ ARP の設定

デバイス上で プロキシ ARP を設定して、他のネットワークまたはサブネット上のホストのメディア アドレスを決定できます。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. **interface ethernet** *number*
3. **ip proxy-arp**
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t**	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 ethernet** *number*
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip proxy-arp** **例：** `switch(config-if)# ip proxy-arp`	インターフェイス上でプロキシ ARP をイネーブルにします。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、プロキシ ARP を設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 2/3
switch(config-if)# ip proxy-arp
switch(config-if)# copy running-config startup-config
```

ローカル プロキシ ARP の設定

デバイス上でローカル プロキシ ARP を設定できます。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. **interface ethernet** *number*
3. **ip local-proxy-arp**
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t**	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 ethernet** *number*
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip local-proxy-arp** **例：** `switch(config-if)# ip local-proxy-arp`	インターフェイス上でローカル プロキシ ARP をイネーブルにします。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ローカル プロキシ ARP を設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 2/3
switch(config-if)# ip local-proxy-arp
switch(config-if)# copy running-config startup-config
```

IPv4 設定の確認

設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show ip adjacency	隣接関係テーブルを表示します。
show ip arp	ARP テーブルを表示します。
show ip interface	IP 関連のインターフェイス情報を表示します。
show ip arp statistics	ARP 統計情報を表示します。

IPv4 設定の例

設定例が、「詳細な手順」の表にあるコマンドの例と矛盾しないようにします。

次に、IPv4 を設定する例を示します。

```
config t
 command keyword argument
```

デフォルト設定

表 2-1 は、IP パラメータのデフォルト設定の一覧です。

表 2-1　デフォルト IP パラメータ

パラメータ	デフォルト
proxy ARP	ディセーブル

その他の関連資料

IP の実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.2-14）
- 標準（p.2-14）

関連資料

関連項目	マニュアル名
IP CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0

標準

標準	タイトル
この機能によりサポートされる新規標準や変更された標準はありません。また、この機能による、既存の標準のサポートの変更もありません。	—

CHAPTER 3

IPv6 の設定

この章では、デバイス上での Internet Protocol version 6（IPv6）（アドレス指定を含む）、Neighbor Discovery Protocol（NDP; 近隣探索プロトコル）、および Internet Control Message Protocol version 6（ICMPv6）の設定方法を説明します。

ここでは、次の内容を説明します。

- IPv6 について（p.3-2）
- IPv6 のライセンス要件（p.3-19）
- IPv6 の前提条件（p.3-19）
- IPv6 の注意事項および制約事項（p.3-19）
- IPv6 の設定（p.3-20）
- IPv6 設定の確認（p.3-25）
- IPv6 設定の例（p.3-25）
- デフォルト設定（p.3-25）
- その他の関連資料（p.3-26）

IPv6 について

IPv6 は、IPv4 の後継として設計されており、ネットワーク アドレス ビット数が 32 ビット（IPv4 の場合）から 128 ビットに増やされています。IPv6 は IPv4 に基づいていますが、アドレス空間が大幅に拡大されており、メイン ヘッダーと拡張ヘッダーの簡素化など、その他の機能強化が含まれています。

拡大された IPv6 アドレス空間により、ネットワークのスケーラビリティが可能となり、グローバルな到達可能性が提供されます。簡素化された IPv6 パケット ヘッダー フォーマットにより、パケットの処理効率が向上しています。柔軟性の高い IPv6 アドレス空間により、プライベート アドレスの必要性と、プライベート（グローバルに一意ではない）アドレスを限られた数のパブリック アドレスに変換する Network Address Translation（NAT; ネットワーク アドレス変換）の使用が削減されます。IPv6 を使用すると、ネットワークの境界にある境界ルータによる特別な処理を必要としない新しいアプリケーション プロトコルがイネーブルになります。

プレフィクス集約、簡易ネットワーク再番号割り当て、IPv6 サイト マルチホーミング機能などの IPv6 機能により、さらに効率的にルーティングが行われます。IPv6 は、Routing Information Protocol（RIP）、Integrated Intermediate System-to-Intermediate System（IS-IS）、IPv6 向け OSPF（Open Shortest Path First）、マルチプロトコル Border Gateway Protocol（BGP）をサポートしています。

ここでは、次の内容について説明します。

- IPv6 アドレス フォーマット（p.3-2）
- IPv6 ユニキャスト アドレス（p.3-3）
- IPv6 エニーキャスト アドレス（p.3-7）
- IPv6 マルチキャスト アドレス（p.3-8）
- IPv4 パケット ヘッダー（p.3-9）
- 簡易 IPv4 パケット ヘッダー（p.3-10）
- IPv6 の DNS（p.3-12）
- IPv6 のパス MTU 探索（p.3-13）
- Cisco Discovery Protocol（CDP）IPv6 アドレスのサポート（p.3-13）
- IPv6 の ICMP（p.3-13）
- IPv6 近隣探索（p.3-14）
- IPv6 ネイバー送信要求メッセージ（p.3-14）
- IPv6 ルータ アドバタイズメント メッセージ（p.3-16）
- IPv6 ネイバー リダイレクト メッセージ（p.3-17）
- 仮想化のサポート（p.3-18）

IPv6 アドレス フォーマット

IPv6 アドレスは、128 ビットまたは 16 バイトです。このアドレスは、x:x:x:x:x:x:x:x のように、コロン（:）で区切られた 16 ビット 16 進数のブロック 8 つに分かれています。次に、IPv6 アドレスの例を 2 つ示します。

```
2001:0DB8:7654:3210:FEDC:BA98:7654:3210
2001:0DB8:0:0:8:800:200C:417A
```

IPv6 アドレスの中には、連続するゼロが含まれます。IPv6 アドレスの先頭、中間、または末尾で、この連続するゼロの代わりに 2 つのコロン（::）を使用できます。表 3-1 は、圧縮された IPv6 アドレス フォーマットの一覧です。

(注)　IPv6 アドレスでは、アドレス中でもっとも長く連続するゼロの代わりに、2 つのコロン（::）を 1 度だけ使用できます。

連続する 16 ビット値がゼロで示されている場合は、2 つのコロンを IPv6 アドレスの一部として使用できます。インターフェイスごとに複数の IPv6 アドレスを設定できますが、設定できるリンクローカル アドレスは 1 つだけです。

IPv6 アドレス中の 16 進数の文字の大文字と小文字は区別されません。

表 3-1　圧縮された IPv6 アドレス フォーマット

IPv6 アドレス タイプ	元のフォーマット	圧縮されたフォーマット
ユニキャスト	2001:0:0:0:0DB8:800:200C:417A	2001::0DB8:800:200C:417A
マルチキャスト	FF01:0:0:0:0:0:0:101	FF01::101
ループバック	0:0:0:0:0:0:0:1	::1
未指定	0:0:0:0:0:0:0:0	::

ノードは、表 3-1 にあるループバック アドレスを使用して、IPv6 パケットを自分宛てに送信できます。IPv6 のループバック アドレスは、IPv4 のループバック アドレスと同じです。詳細については、第 1 章「概要」を参照してください。

(注)　IPv6 ループバック アドレスは、物理インターフェイスには割り当てられません。送信元または宛先のアドレスとして IPv6 ループバック アドレスを含むパケットは、そのパケットを作成したノードの外には転送できません。IPv6 ルータは、送信元または宛先のアドレスとして IPv6 ループバック アドレスを含むパケットを転送しません。

(注)　IPv6 未指定アドレスは、インターフェイスには割り当てられません。未指定 IPv6 アドレスは、IPv6 パケット内の宛先アドレスまたは IPv6 ルーティング ヘッダーとして使用しないでください。

IPv6 プレフィクスは、RFC 2373 で規定された形式です。この形式では、IPv6 アドレスが、コロンに囲まれた 16 ビット値を使用した 16 進数で指定されています。プレフィクス長は、プレフィクス（アドレスのネットワーク部）を構成する、アドレスの高位隣接ビットの数を示す 10 進数値です。たとえば、2001:0DB8:8086:6502::/32 は有効な IPv6 プレフィクスです。

IPv6 ユニキャスト アドレス

IPv6 ユニキャスト アドレスは、単一ノード上の単一インターフェイスの ID です。ユニキャスト アドレス宛てに送信されたパケットは、そのアドレスを持つインターフェイスに配信されます。ここでは、次の内容について説明します。

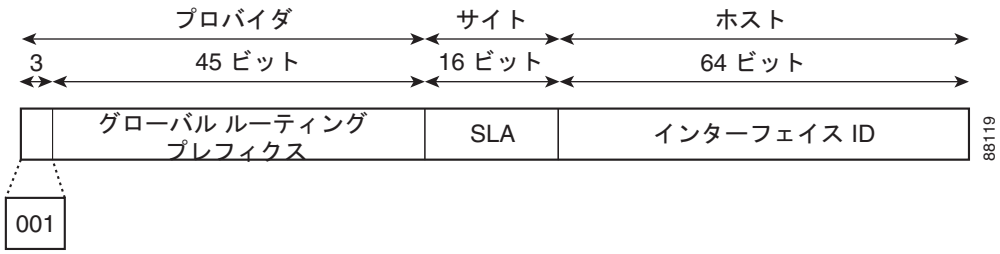
- 集約可能グローバル アドレス（p.3-4）
- リンクローカル アドレス（p.3-5）
- IPv4 互換の IPv6 アドレス（p.3-6）
- ユニーク ローカル アドレス（p.3-6）
- サイトローカル アドレス（p.3-7）

集約可能グローバル アドレス

集約可能グローバル アドレスは、集約可能なグローバル ユニキャスト プレフィクスによる IPv6 アドレスです。集約可能なグローバル ユニキャスト アドレスの構造により、グローバル ルーティング テーブル内のルーティング テーブル エントリ数を制限するルーティング プレフィクスの厳密な集約が可能となります。集約可能なグローバル アドレスは、組織間を上に向かって、最終的に Internet service provider（ISP; インターネット サービス プロバイダー）まで集約されるリンクで使用されます。

集約可能なグローバル IPv6 アドレスは、グローバル ルーティング プレフィクス、サブネット ID、およびインターフェイス ID により定義されます。IPv6 グローバル ユニキャスト アドレスの割り当てには、バイナリ値 001（2000::/3）から始まるアドレスの範囲が使用されます。図 3-1 は、集約可能なグローバル アドレスの構造を示します。

図 3-1　集約可能グローバル アドレスのフォーマット



2000::/3 (001) ～ E000::/3 (111) のプレフィクスを持つアドレスには、Extended Universal Identifier（EUI）-64 フォーマットの 64 ビット インターフェイス ID が必要です。Internet Assigned Numbers Authority（IANA; インターネット割り当て番号局）は、2000::/16 の範囲の IPv6 アドレス空間を地域レジストリに割り当てます。

集約可能なグローバル アドレスは、48 ビット グローバル ルーティング プレフィクスと、16 ビット サブネット ID または Site-Level Aggregator（SLA）で構成されます。IPv6 集約可能なグローバル ユニキャスト アドレス フォーマット文書（RFC 2374）では、グローバル ルーティング プレフィクスには、Top-Level Aggregator（TLA）および Next-Level Aggregator（NLA）という他の 2 つの階層構造のフィールドが含まれるとされていました。TLS フィールドおよび NLA フィールドはポリシーベースであるため、IETF は、これらのフィールドを RFC から削除することを決定しました。この変更以前に展開された既存の IPv6 ネットワークの中には、依然として、古いアーキテクチャ上のネットワークを使用しているものもあります。

個々の組織では、16 ビット サブネット フィールドであるサブネット ID を使用して、ローカル アドレス指定階層構造を作成したり、サブネットを識別したりできます。サブネット ID は、IPv4 でのサブネットに似ていますが、IPv6 サブネット ID を持つ組織では 65,535 のサブネットをサポートできるという点で異なります。

インターフェイス ID で、リンク上のインターフェイスが識別されます。インターフェイス ID は、リンク上では一意です。多くの場合、インターフェイス ID は、インターフェイスのリンクレイヤ アドレスと同じか、リンクレイヤ アドレスに基づいています。集約可能なグローバル ユニキャスト アドレス タイプおよびその他の IPv6 アドレス タイプで使用されるインターフェイス ID は、長さが 64 ビットの変更済み EUI-64 フォーマットです。

インターフェイス ID は、次のいずれかに該当する変更済みの EUI-64 フォーマットです。

- すべての IEEE 802 インターフェイス タイプ（イーサネット、および Fiber Distributed Data インターフェイスなど）の場合は、最初の 3 オクテット（24 ビット）がそのインターフェイスの 48 ビット リンクレイヤ アドレス（MAC アドレス）の Organizationally Unique Identifier（OUI）、4 番めと 5 番めのオクテット（16 ビット）が FFFE の固定 16 進数値、そして、最後の 3 オクテット（24 ビット）が MAC アドレスの最後の 3 オクテットです。最初のオクテットの 7 番めのビットである Universal/Local（U/L）ビットの値は 0 または 1 です。ゼロはローカルに管理されている ID を表し、1 はグローバルに一意の IPv6 インターフェイス ID を表します。
- その他のすべてのインターフェイス タイプ（シリアル、ループバック、ATM、フレームリレー、トンネル インターフェイス タイプなど。ただし、IPv6 オーバーレイ トンネルで使用されるトンネル インターフェイスを除く）の場合、インターフェイス ID は IEEE 802 インターフェイス タイプのインターフェイス ID に似ていますが、ルータの MAC アドレス プールからの最初の MAC アドレスが ID として使用される点が異なります（インターフェイスが MAC アドレスを持たないため）。
- IPv6 オーバーレイ トンネルで使用されるトンネル インターフェイス タイプの場合、インターフェイス ID は、ID の高位 32 ビットがすべてのゼロであるトンネル インターフェイスに割り当てられた IPv4 アドレスです。

（注） PPP（ポイントツーポイント　プロトコル）を使用するインターフェイスの場合は、接続の両端のインターフェイスが同じ MAC アドレスを持つため、接続の両端のインターフェイス ID が、両方の ID が一意となるまでネゴシエートされます（ランダムに選択され、必要に応じて再構築されます）。ルータの最初の MAC アドレスが、PPP を使用するインターフェイスの ID として使用されます。

ルータに IEEE 802 インターフェイス タイプがない場合は、ルータのインターフェイスでリンクローカル IPv6 アドレスが次のシーケンスで生成されます。

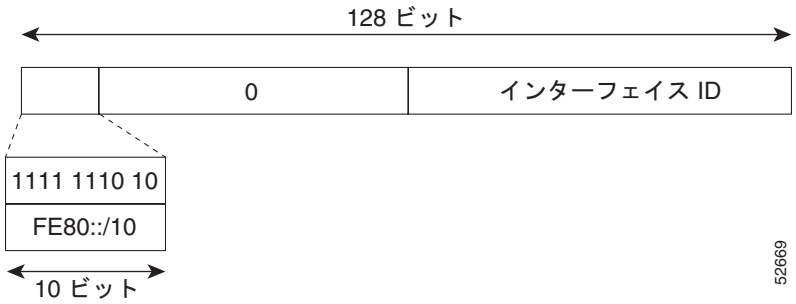
1. ルータに MAC アドレス（ルータ内の MAC アドレス プールの）が照会されます。
2. 使用可能な MAC アドレスがルータにない場合は、ルータのシリアル番号を使用して、リンクローカル アドレスが作成されます。
3. リンクローカル アドレスの作成にルータのシリアル番号を使用できない場合、ルータは MD5 ハッシュを使用して、ルータのホスト名からルータの MAC アドレスを決定します。

リンクローカル アドレス

リンクローカル アドレスは、リンクローカル プレフィクス FE80::/10（1111 1110 10）と、変更済み EUI-64 フォーマットのインターフェイス ID を使用するどのインターフェイスでも自動設定が可能な IPv6 ユニキャスト アドレスです。リンクローカル アドレスは NDP およびステートレス自動設定処理で使用されます。ローカル リンク上のノードは、リンクローカル アドレスを使用して通信でき、通信するためにはグローバルに一意のアドレスを必要としません。図 3-2 は、リンクローカル アドレスの構造を示します。

IPv6 ルータは、送信元または宛先がリンクローカル アドレスであるパケットを他のリンクに転送できません。

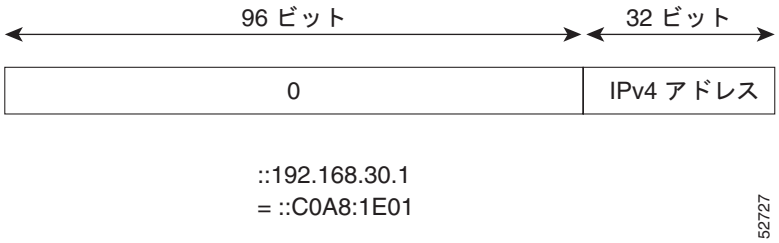
図 3-2　リンクローカル アドレス フォーマット



IPv4 互換の IPv6 アドレス

IPv4 互換 IPv6 アドレスとは、アドレスの高位 96 ビットがゼロであり、アドレスの低位 32 ビットが IPv4 アドレスである IPv6 ユニキャスト アドレスです。IPv4 互換 IPv6 アドレスのフォーマットは 0:0:0:0:0:0:A.B.C.D または ::A.B.C.D です。IPv4 互換 IPv6 アドレスの 128 ビット全体はノードの IPv6 アドレスとして使用され、低位 32 ビットに埋め込まれた IPv4 アドレスはノードの IPv4 アドレスとして使用されます。IPv4 互換 IPv6 アドレスは、IPv4 および IPv6 の両プロトコル スタックをサポートするノードに割り当てられ、自動トンネルで使用されます。図 3-3 は、IPv4 互換 IPv6 アドレスの構造と、許容されるアドレス フォーマットのいくつかを示します。

図 3-3　IPv4 互換 IPv6 アドレスのフォーマット



ユニーク ローカル アドレス

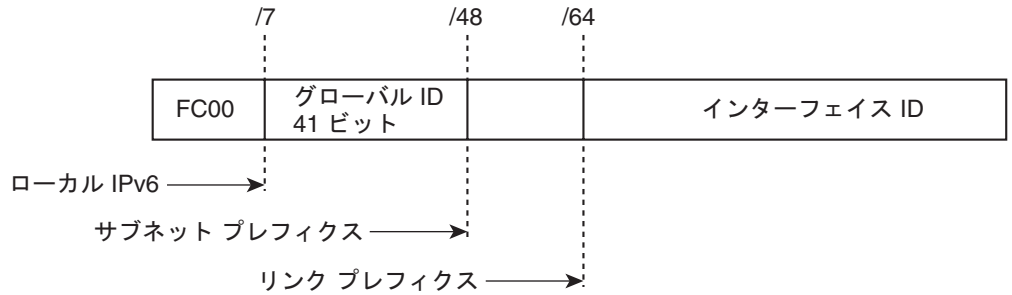
ユニーク ローカル アドレスは、グローバルに一意であり、ローカルでの通信のための IPv6 ユニキャスト アドレスです。グローバルなインターネット上でのルーティングには対応しておらず、サイトなどの限られたエリア内でのみルーティング可能です。限られた複数のサイト間もルーティングできる場合もあります。アプリケーションは、ユニーク ローカル アドレスをグローバルなスコープのアドレスとして扱う場合があります。

ユニーク ローカル アドレスには次の特徴があります。

- グローバルに一意のプレフィクスを持っている（一意である可能性が大）
- 周知のプレフィクスを持つため、サイトの境界でフィルタリングが容易
- アドレスの競合が発生せず、これらのプレフィクスを使用するインターフェイスの再番号割り当ても必要とせずに、サイトを結合したり、プライベートに相互接続したりできる
- ISP に依存せず、永久的または断続的なインターネット接続を使用することなく、サイト内での通信に使用できる
- ルーティングや Domain Name Server（DNS; ドメイン ネーム サーバ）により、誤ってサイト外に漏れても、他のどのアドレスとも競合しない

図 3-4 は、ユニーク ローカル アドレスの構造を示します。

図 3-4　ユニーク ローカル アドレスの構造



- プレフィクス ― ローカル IPv6 ユニキャスト アドレスであることを表す FC00::/7 プレフィクス
- グローバル ID ― グローバルに一意のプレフィクスの作成に使用される 41 ビット グローバル ID
- サブネット ID ― サイト内のサブネットの ID である 16 ビット サブネット ID
- インターフェイス ID ― 64 ビット IID

サイトローカル アドレス

RFC 3879 によりサイトローカル アドレスの使用が廃止されたため、プライベート IPv6 アドレスの設定時には、RFC 4193 で推奨されるユニーク ローカル アドレス（UCA）を使用する必要があります。

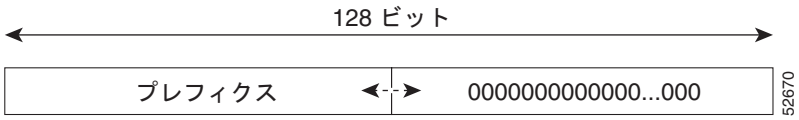
IPv6 エニーキャスト アドレス

エニーキャスト アドレスとは、異なるノードに属するインターフェイス一式に割り当てられたアドレスです。エニーキャスト アドレスに送信されたパケットは、ルーティング プロトコルの定義に従って、そのエニーキャスト アドレスが示す最寄りのインターフェイスに配信されます。エニーキャスト アドレスは、ユニキャスト アドレス空間から割り当てられるため、その構文ではユニキャスト アドレスと区別できません。ユニキャスト アドレスを複数のインターフェイスに割り当てると、ユニキャスト アドレスがエニーキャスト アドレスとなります。エニーキャスト アドレスが割り当てられたノードは、アドレスがエニーキャスト アドレスであることを認識できるよう、設定する必要があります。

（注） エニーキャスト アドレスを使用できるのは、ルータのみです。ホストはエニーキャスト アドレスを使用できません。エニーキャスト アドレスは、IPv6 パケットの送信元アドレスには使用できません。

図 3-5 は、サブネット ルータ エニーキャスト アドレスのフォーマットを示します。アドレスには、連続するゼロに連結されたプリフィクス（インターフェイス ID）があります。サブネット ルータ エニーキャスト アドレスを使用すると、サブネット ルータ エニーキャスト アドレスのプレフィクスが表すリンク上のルータに到達できます。

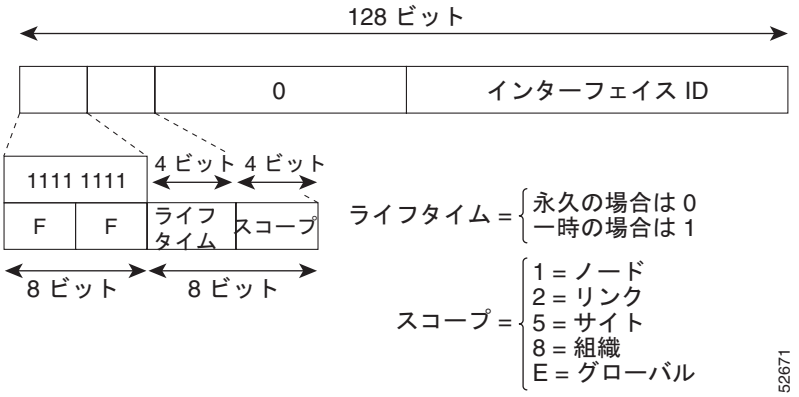
図 3-5　サブネット ルータ エニーキャスト アドレスのフォーマット



IPv6 マルチキャスト アドレス

IPv6 マルチキャスト アドレスとは、FF00::/8（1111 1111）というプレフィクスを持つ IPv6 アドレスです。IPv6 マルチキャスト アドレスは、異なるノードに属するインターフェイス一式の ID です。マルチキャスト アドレスに送信されたパケットは、マルチキャスト アドレスが示すすべてのインターフェイスに配信されます。プレフィクスに続く 2 番めのオクテットで、マルチキャスト アドレスのライフタイムとスコープが定義されます。永久マルチキャスト アドレスはライフタイム パラメータが 0 に等しく、一時マルチキャスト アドレスのライフタイム パラメータは 1 に等しくなっています。ノード、リンク、サイト、または組織のスコープ、またはグローバル スコープを持つマルチキャスト アドレスのスコープ パラメータはそれぞれ、1、2、5、8、または E です。たとえば、FF02::/16 というプレフィクスを持つマルチキャスト アドレスは、リンク スコープを持つ永久マルチキャスト アドレスです。図 3-6 は、IPv6 マルチキャスト アドレスのフォーマットを示します。

図 3-6　IPv6 マルチキャスト アドレスのフォーマット



IPv6 ノード（ホストとルータ）は、（受信パケットの宛先となる）次のマルチキャスト グループに加入する必要があります。

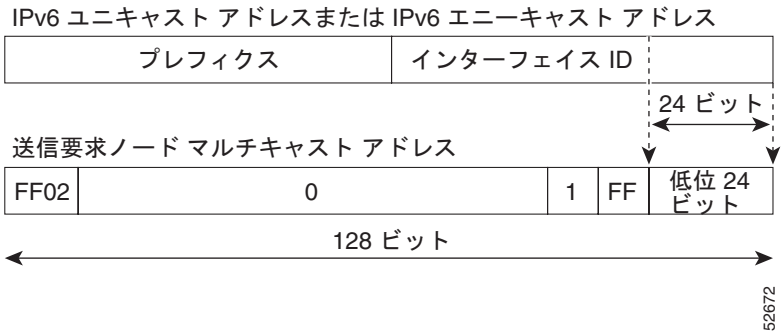
- 全ノード マルチキャスト グループ FF02:0:0:0:0:0:0:1（スコープはリンクローカル）
- 割り当てられたユニキャスト アドレスおよびエニーキャスト アドレスごとの送信要求ノード マルチキャスト グループ FF02:0:0:0:0:1:FF00:0000/104

IPv6 ルータは、全ルータ マルチキャスト グループ FF02:0:0:0:0:0:0:2（スコープはリンクローカル）にも加入する必要があります。

送信要求ノード マルチキャスト アドレスは、IPv6 ユニキャスト アドレスまたは IPv6 エニーキャスト アドレスに対応するマルチキャスト グループです。IPv6 ノードは、割り当てられているユニキャスト アドレスおよびエニーキャスト アドレスごとに、関連付けられた送信要求ノード マルチキャスト グループに加入する必要があります。IPv6 送信要求ノード マルチキャスト アドレスには、対

応する IPv6 ユニキャスト アドレス または IPv6 エニーキャスト アドレスの低位 24 ビットに連結されたプレフィクス FF02:0:0:0:0:1:FF00:0000/104 があります（図 3-7 を参照）。たとえば、IPv6 アドレス　2037::01:800:200E:8C6C　に対応する送信要求ノード　マルチキャスト　アドレスは FF02::1:FF0E:8C6C です。送信要求ノード アドレスは、ネイバー送信要求メッセージで使用されます。

図 3-7　IPv6 送信要求ノード マルチキャスト アドレスのフォーマット

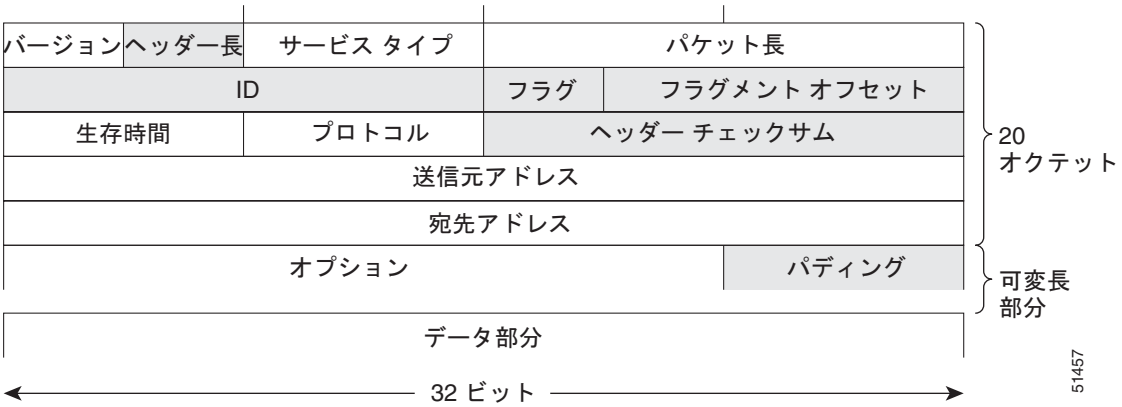


(注)　IPv6 にはブロードキャスト アドレスはありません。IPv6 マルチキャスト アドレスがブロードキャスト アドレスの代わりに使用されます。

IPv4 パケット ヘッダー

基本 IPv4 パケット ヘッダーには、合計サイズが 20 オクテット（160 ビット）の 12 のフィールドがあります（図 3-8 を参照）。この 12 このフィールドの後にはオプション フィールドが、さらにその後に、通常はトランスポート レイヤ パケットであるデータ部分が続く場合があります。オプション フィールドの可変長部分は、IPv4 パケット ヘッダーの合計サイズに加算されます。IPv4 パケット ヘッダーのグレーの部分のフィールドは、IPv6 パケット ヘッダーに含まれません。

図 3-8　IPv4 パケット ヘッダーのフォーマット



簡易 IPv4 パケット ヘッダー

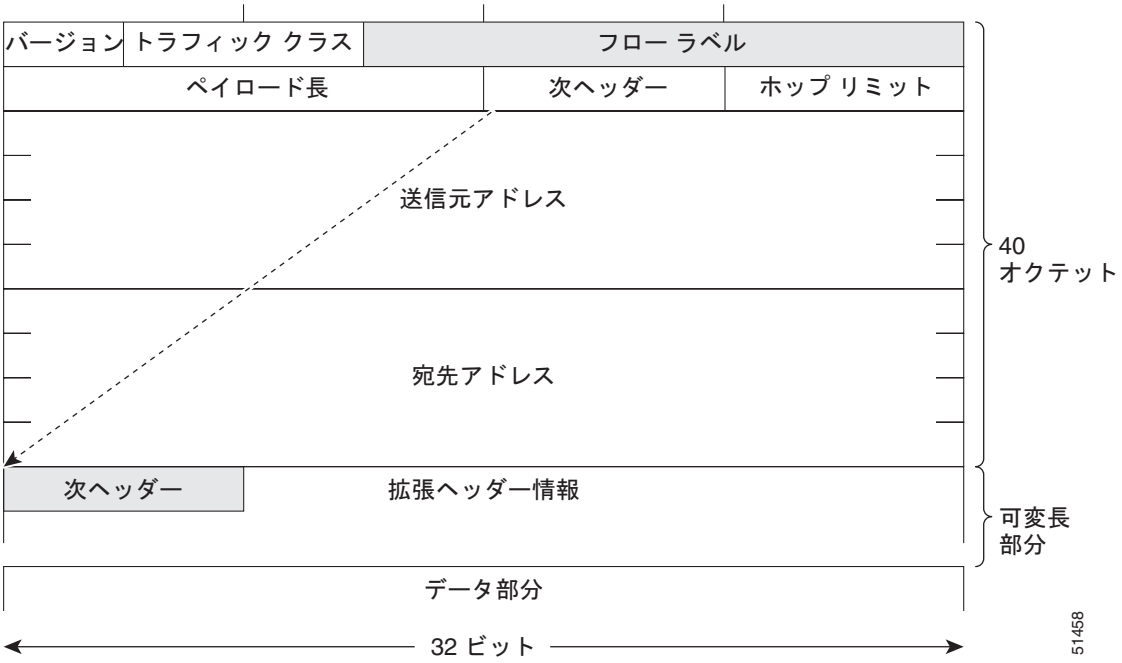
基本 IPv6 パケット ヘッダーには、合計サイズが 40 オクテット（320 ビット）の 8 つのフィールドがあります（図 3-9 を参照）。フラグメンテーションはパケットの送信元により処理され、データリンク レイヤのチェックサムとトランスポート レイヤが使用されます。User Datagram Protocol（UDP）チェックサムにより、内部パケットと基本 IPv6 パケット ヘッダーの整合性がチェックされ、オプション フィールドが 64 ビットに揃えられるため、IPv6 パケットの処理が容易になります。

表 3-2 は、基本 IPv6 パケット ヘッダーのフィールドの一覧です。

表 3-2　基本 IPv6 パケット ヘッダーのフィールド

フィールド	説明
バージョン	IPv4 パケット ヘッダーのバージョン フィールドに該当しますが、IPv4 で示される数字 4 の代わりに、IPv6 では数字 6 が示されます。
トラフィック クラス	IPv4 パケット ヘッダーのサービス タイプ フィールドに該当します。トラフィック クラス フィールドは、差別化されたサービスで使用されるトラフィック クラスのタグをパケットに付けます。
フロー ラベル	IPv6 パケット ヘッダーの新規フィールドです。フロー ラベル フィールドは、ネットワーク レイヤでパケットを差別化する特定のフローのタグを、パケットに付けます。
ペイロード長	IPv4 パケット ヘッダーのパケット長フィールドに該当します。ペイロード長フィールドは、パケットのデータ部分の長さの合計を示します。
次ヘッダー	IPv4 パケット ヘッダーのプロトコル フィールドに該当します。次ヘッダー フィールドの値により、基本 IPv6 ヘッダーの後に続く情報のタイプが決まります。基本 IPv6 ヘッダーの後に続く情報のタイプは、図 3-9 に示すように、TCP パケット、UDP パケット、または拡張ヘッダーなどのトランスポート レイヤ パケットです。
ホップ リミット	IPv4 パケット ヘッダーの生存時間フィールドに該当します。ホップ リミット フィールドは、IPv6 パケットが無効になる前に通過できるルータの最大数を指定します。各ルータを通過するたびに、この値が 1 ずつ減少します。IPv6 ヘッダーにはチェックサムがないため、ルータは、値が減少するたびにチェックサムを再計算する必要がなく、処理リソースの節約となります。
送信元アドレス	IPv4 パケット ヘッダーの送信元アドレス フィールドに該当しますが、IPv4 の 32 ビット送信元アドレスの代わりに、IPv6 では 128 ビット送信元アドレスが含まれます。
宛先アドレス	IPv4 パケット ヘッダーの宛先アドレス フィールドに該当しますが、IPv4 の 32 ビット宛先アドレスの代わりに、IPv6 では 128 ビット宛先アドレスが含まれます。

図 3-9　IPv6 パケット ヘッダーのフォーマット



任意に使用できる拡張ヘッダーおよびパケットのデータ部分は、基本 IPv6 パケット ヘッダーの 8 つのフィールドの後に続きます。拡張ヘッダーがある場合は、各拡張ヘッダーが 64 ビットに揃えられます。IPv6 パケットの拡張ヘッダーの数は決められていません。各拡張ヘッダーは、前のヘッダーの次ヘッダー フィールドに示されます。通常は、最後の拡張ヘッダーに、TCP または UDP などのトランスポート レイヤ プロトコルの次ヘッダー フィールドがあります。図 3-10 は、IPv6 拡張ヘッダのフォーマットを示します。

図 3-10　IPv6 拡張ヘッダーのフォーマット

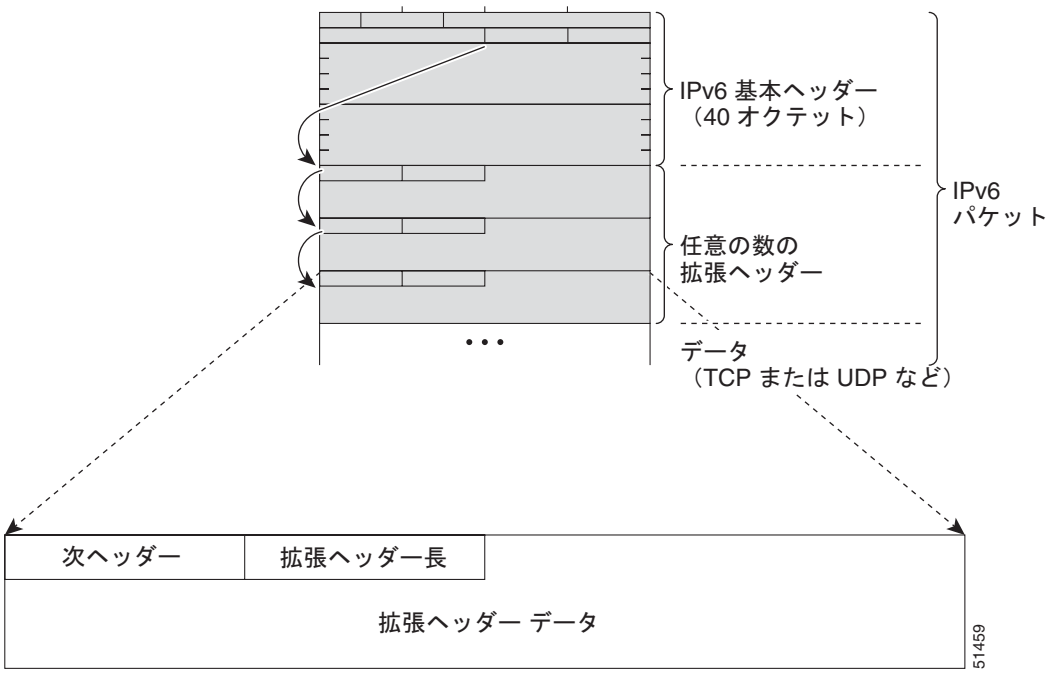


表 3-3 は、拡張ヘッダーのタイプとその次ヘッダー フィールドの値の一覧です。

表 3-3　IPv6 拡張ヘッダーのタイプ

ヘッダーのタイプ	次ヘッダーの値	説明
ホップバイホップ オプション ヘッダー	0	パケットのパス上のすべてのホップで処理されるヘッダー。存在する場合、ホップバイホップ オプション ヘッダーは、つねに基本 IPv6 パケット ヘッダーの直後に続きます。
宛先オプション ヘッダー	6	任意のホップバイホップ オプション ヘッダーの後に続くことのあるヘッダー。このヘッダーは、最終の宛先、およびルーティング ヘッダーで指定された各通過アドレスで処理されます。また、宛先オプション ヘッダーは、任意の Encapsulating Security Payload（ESP）ヘッダーの後に続く場合もあります。この場合の宛先オプション ヘッダーは、最終の宛先でのみ処理されます。
ルーティング ヘッダー	43	送信元のルーティングに使用されるヘッダー
フラグメント ヘッダー	44	送信元が、送信元と宛先の間のパスの Maximum Transmission Unit（MTU; 最大伝送ユニット）より大きいパケットをフラグメント化するときに使用されるヘッダー。フラグメント ヘッダーは、フラグメント化された各パケットで使用されます。
上位層ヘッダー	6（TCP の場合） 17（UDP の場合）	データ転送のためにパケット内で使用されるヘッダー。おもな転送プロトコルは TCP と UDP の 2 つです。

IPv6 の DNS

IPv6 は、DNS 名前 / アドレスおよびアドレス / 名前のルックアップ処理でサポートされる DNS レコード タイプをサポートしています。DNS レコード タイプは IPv6 アドレスをサポートしています。

（注） IPv6 は、IPv6 アドレスから DNS 名への逆マッピングもサポートしています。

表 3-4　IPv6 DNS レコード タイプ

レコード タイプ	説明	フォーマット
AAAA	ホスト名を IPv6 アドレスにマッピングします（IPv4 の A レコードに該当）。	www.abc.test AAAA 3FFE:YYYY:C18:1::2
PTR	IPv6 アドレスをホスト名にマッピングします（IPv4 の PTR レコードに該当）。	2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.8.1.c.0.y.y.y.y.e.f.f.3.ip6.int PTR www.abc.test

IPv6 のパス MTU 探索

IPv4 の場合と同様に、ホストがダイナミックに、データ パス上のすべてのリンクの MTU サイズの差を検出し、それに合わせて調整できるよう、IPv6 でパス MTU 探索を使用できます。ただし、IPv6 では、データ パス上のリンクのパス MTU が小さすぎてパケットを処理できない場合は、パケットの送信元によりフラグメンテーションが処理されます。IPv6 ホストにパケットのフラグメンテーションを処理させると、IPv6 ルータの処理リソースが節約され、IPv6 ネットワークの効率が向上します。

(注)　IPv6 では、最小リンク MTU は 1280 オクテットです。IPv6 リンクには、1500 オクテットの MTU 値の使用をお勧めします。

Cisco Discovery Protocol（CDP）IPv6 アドレスのサポート

隣接情報機能向け CDP IPv6 アドレス サポートを使用して、2 台のシスコ デバイス間で IPv6 アドレス指定情報を転送できます。IPv6 アドレス向け CDP サポートは、ネットワーク管理製品およびトラブルシューティング ツールに IPv6 情報を提供します。

IPv6 の ICMP

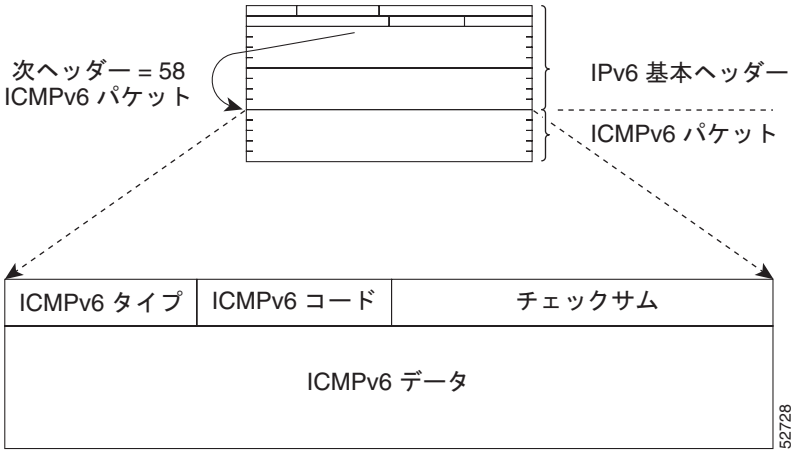
IPv6 で ICMP を使用して、ネットワークの状態に関する情報を提供できます。IPv6 で使用できるバージョンである ICMPv6 は、パケットが正しく処理されない場合にエラーを報告し、ネットワークの状態に関する情報メッセージを送信します。たとえば、パケットが大きすぎて別のネットワークに送信できないために、ルータがパケットを転送できない場合は、ルータにより、送信元のホストに ICMPv6 メッセージが送信されます。さらに、IPv6 の ICMP パケットは IPv6 近隣探索およびパス MTU 探索に使用されます。パス MTU 探索処理により、パケットが確実に、特定のルートでサポートされる最大のサイズで送信されます。

基本 IPv6 パケット ヘッダーの次ヘッダー フィールドの 58 という値は、IPv6 ICMP パケットであることを示します。ICMP パケットは、すべての拡張ヘッダーの後に続く、IPv6 パケット中の最後の情報部分です。IPv6 ICMP パケットでは、ICMPv6 タイプ フィールドと ICMPv6 コード フィールドに、ICMP メッセージ タイプなどの IPv6 ICMP パケット情報が示されます。チェックサム フィールドの値は送信側で計算され、受信側により、ICMP パケット内および IPv6 疑似ヘッダー内のフィールドでチェックされます。

(注)　IPv6 ヘッダーには、チェックサムはありません。ただし、チェックサムは、パケットの誤配信を判定するために、トランスポート レイヤで重要です。計算に IP アドレスが含まれるすべてのチェックサム計算は、新しい 128 ビット アドレスを処理できるよう、IPv6 用に変更する必要があります。チェックサムは、疑似ヘッダーを使用して生成されます。

ICMPv6 データ フィールドには、IP パケット処理に関連するエラー情報または診断情報が含まれます。図 3-11 は、IPv6 ICMP パケット ヘッダのフォーマットを示します。

図 3-11　IPv6 ICMP パケット ヘッダーのフォーマット



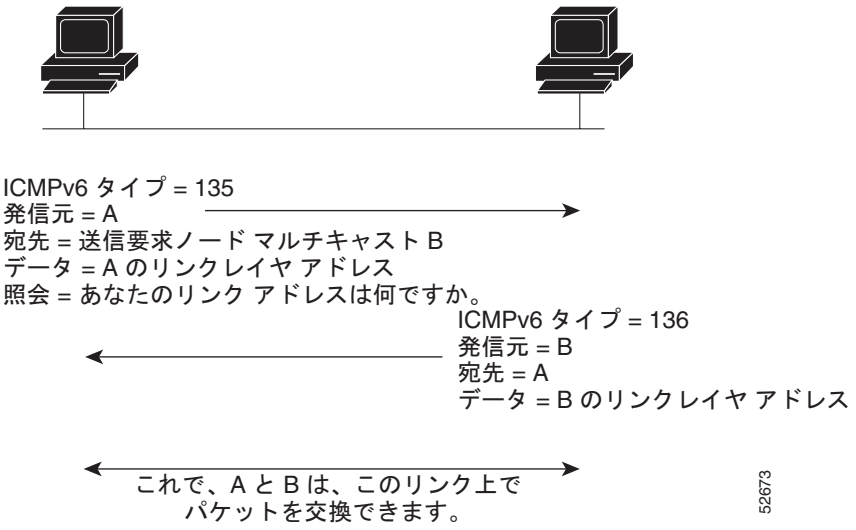
IPv6 近隣探索

IPv6 NDP を使用して、隣接ルータが到達可能かどうかを判定できます。IPv6 ノードは、近隣探索を使用して、同じネットワーク上のノードのアドレス（ローカル リンク）を決定します。そして、ノード自身からのパケットを転送できる隣接ルータを見つけ、その隣接ルータが到達可能かどうかを確認し、リンクレイヤ アドレスの変更を検出します。NDP は ICMP メッセージを使用して、パケットが到達不可能な隣接ルータに送信されたかどうかを検出します。

IPv6 ネイバー送信要求メッセージ

ノードは、同じローカル リンク上の別のノードのリンクレイヤ アドレスを決定するときに、ICMP パケット ヘッダーのタイプ フィールドの値が 135 であるネイバー送信要求メッセージをローカル リンクで送信します（図 3-12 を参照）。送信元アドレスは、ネイバー送信要求メッセージを送信するノードの IPv6 アドレスです。宛先アドレスは、宛先ノードの IPv6 アドレスに対応する送信要求ノード マルチキャスト アドレスです。ネイバー送信要求メッセージには、送信元ノードのリンクレイヤ アドレスも含まれます。

図 3-12　IPv6 近隣探索 — ネイバー送信要求メッセージ



ネイバー送信要求メッセージを受信したあとに、宛先ノードは返信として、ICMP パケット ヘッダーのタイプ フィールドの値が 136 のネイバー アドバタイズメント メッセージをローカル リンクで送信します。送信元アドレスは、ノードの IPv6 アドレス（ネイバー アドバタイズメント メッセージを送信するノード インターフェイスの IPv6 アドレス）です。宛先アドレスは、ネイバー送信要求メッセージを送信したノードの IPv6 アドレスです。データ部分には、ネイバー アドバタイズメント メッセージを送信するノードのリンクレイヤ アドレスが含まれます。

送信元ノードがネイバー アドバタイズメント メッセージを受信すると、送信元ノードと宛先ノードは通信できるようになります。

ネイバー送信要求メッセージにより、ノードがネイバーのリンクレイヤ アドレスを認識したあとに、ネイバーの到達可能性が確認できます。ノードは、ネイバーの到達可能性を確認するときに、ネイバー送信要求メッセージの宛先アドレスを、ネイバーのユニキャスト アドレスとして使用します。

ネイバー アドバタイズメント メッセージは、ローカル リンク上のノードのリンクレイヤ アドレスが変更されたときにも送信されます。変更があったときのネイバー アドバタイズメント メッセージの宛先アドレスは、全ノード マルチキャスト アドレスです。

ネイバー到達不能検出により、ネイバーの障害またはネイバーへの転送パスの障害が特定されます。また、この検出は、ホストと近隣ノード（ホストまたはルータ）の間のすべてのパスで使用されます。ネイバー到達不能検出は、ユニキャスト パケットのみが送信されるネイバーに対してのみ実行され、マルチキャスト パケットが送信されるネイバーに対しては実行されません。

ネイバーは、そのノードから受諾の確認応答（以前にそのノードに送信されたパケットが受信され、処理されたことを示す）が返されると、到達可能とみなされます。受諾の確認応答（TCP などの上位層プロトコルからの）は、接続が順調に進んでいる（宛先に到達しつつある）ことを示します。パケットがピアに到達している場合は、送信元ノードのネクストホップ ネイバーにも到達しています。順調に進んでいることで、ネクストホップ ネイバーが到達可能であることも確認されます。

ローカル リンクにない宛先の場合は、順調に進んでいることで、ファーストホップ ルータが到達可能であることがわかります。上位層プロトコルからの確認応答がない場合、ノードは、ユニキャスト ネイバー送信要求メッセージを使用してネイバーを探し、宛先へのパスがまだ機能しているかどうかを確認します。ネイバーから返信された送信要求ネイバー アドバタイズメント メッセージは、宛先へのパスがまだ機能しているという確認応答です（1 という値が設定された送信要求フラグを持つネイバー アドバタイズメント メッセージは、ネイバー送信要求メッセージへの返信としてのみ送信されます）。非送信要求メッセージが返信された場合は、送信元ノードから宛先ノードまでの片道のパスのみが確認されています。送信要求ネイバー アドバタイズメント メッセージは、往復のパスがいずれも機能していることを示します。

（注） 0 という値が設定された送信要求フラグを持つネイバー アドバタイズメント メッセージは、宛先へのパスがまだ機能していることを示す確認応答とはみなされません。

ネイバー送信要求メッセージは、ユニキャスト IPv6 アドレスをインターフェイスに割り当てる前にそのアドレスが一意であることを確認するために、ステートレス自動設定処理でも使用されます。新規のリンクローカル IPv6 アドレスに対しては、インターフェイスに割り当てられる前に、最初に重複アドレス検出が実行されます（新規アドレスは、重複アドレス検出の実行中は一時的なアドレスのままです）。ノードは、未指定の送信元アドレスと一時的なリンクローカル アドレスがメッセージ本文に含まれるネイバー送信要求メッセージを送信します。そのアドレスがすでに他のノードによって使用されている場合、ノードは、一時的リンクローカル アドレスを含むネイバー アドバタイズメント メッセージを返信します。他のノードが同時に、同じアドレスが一意であることを確認している場合は、そのノードも、ネイバー送信要求メッセージを返信します。ネイバー送信要

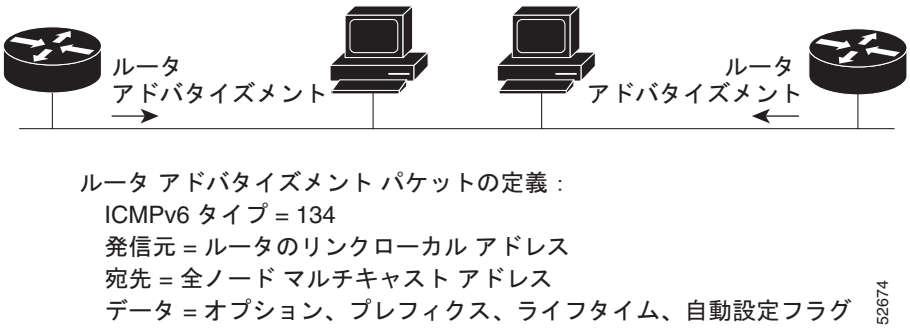
求メッセージの返信としてのネイバー アドバタイズメント メッセージも、同じ一時的アドレスを確認中の他のノードからのネイバー送信要求メッセージも受信しない場合、最初のネイバー送信要求メッセージを送信したノードは、一時的なリンクローカル アドレスが一意であると判断し、そのアドレスをインターフェイスに割り当てます。

IPv6 ルータ アドバタイズメント メッセージ

ルータ アドバタイズメント（RA）メッセージは、ICMP パケット ヘッダーのタイプ フィールドの値が 134 であり、IPv6 ルータの設定済みの各インターフェイスへと定期的に送信されます。ステートレス自動設定が正しく機能するには、RA メッセージでアドバタイズされたプレフィクス長がつねに 64 ビットである必要があります。

RA メッセージは、全ノード マルチキャスト アドレスに送信されます（図 3-13 を参照）。

図 3-13　IPv6 近隣探索 ― RA メッセージ



RA メッセージには通常、次の情報が含まれます。

- ローカル リンク上のノードが、その IPv6 アドレスの自動設定に使用できる 1 つ以上のオンリンク IPv6 プレフィクス
- アドバタイズメントに含まれる各プレフィクスのライフタイム情報
- 完成可能な自動設定のタイプ（ステートレスまたはステートフル）を示すフラグ一式
- デフォルト ルータ情報（アドバタイズメントを送信しているルータをデフォルトとして使用する必要があるかどうか、および、その場合は、ルータがデフォルト ルータとして使用される秒単位の時間）
- ホストが発信するパケットで使用する必要のあるホップ リミットと MTU などの、ホストの詳細情報

RA は、ルータ送信要求メッセージの返信としても送信されます。ルータ送信要求メッセージは、ICMP パケット ヘッダーのタイプ フィールドの値が 133 であり、システム起動時にホストにより送信されます。これにより、ホストは、次回の定期 RA メッセージを待つことなく、ただちに自動設定できます。送信元アドレスは通常、未指定 IPv6 アドレス（0:0:0:0:0:0:0:0）です。ホストでユニキャスト アドレスが設定されている場合は、ルータ送信要求メッセージを送信するインターフェイスのユニキャスト アドレスが、メッセージで送信元アドレスとして使用されます。宛先アドレスは、スコープがリンクである全ルータ マルチキャスト アドレスです。RA がルータ送信要求の返信として送信されるときの RA メッセージの宛先アドレスは、ルータ送信要求メッセージの送信元のユニキャスト アドレスです。

次の RA メッセージ パラメータを設定できます。

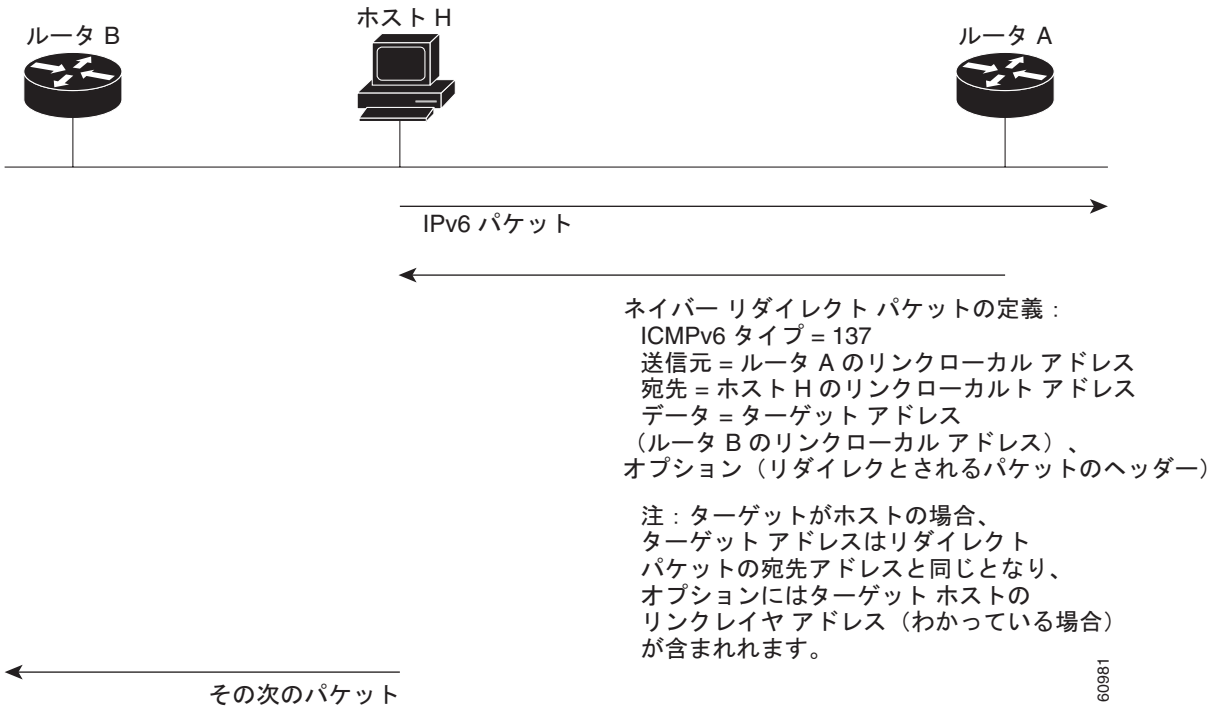
- RA メッセージが定期的に送信される時間の間隔
- デフォルト ルータ（リンクのすべてのノードが使用する）としてのルータの実用性を示すルータのライフタイム値
- リンクで使用されているネットワーク プレフィクス
- （リンクで）ネイバー送信要求メッセージが再送される時間の間隔
- ネイバーが到達可能である（リンク上のすべてのノードが使用できる）とノードが判断するまでの時間

設定されたパラメータは、各インターフェイス固有のパラメータです。RA メッセージ（デフォルト値を含む）の送信は、自動的にイーサネット インターフェイス上でイネーブルになります。他のインターフェイス タイプの場合は、**no ipv6 nd suppress-ra** コマンドを入力して RA メッセージを送信する必要があります。個々のインターフェイスでは、**ipv6 nd suppress-ra** コマンドを入力して、RA メッセージ機能をディセーブルにできます。

IPv6 ネイバー リダイレクト メッセージ

ルータは、ネイバー リダイレクト メッセージを送信して、パス上の宛先へのより適切なファーストホップ ノードをホストに通知します（図 3-14 を参照）。IPv6 ネイバー リダイレクト メッセージは、ICMP パケット ヘッダーのタイプ フィールドの値が 137 となっています。

図 3-14　IPv6 近隣探索 ― ネイバー リダイレクト メッセージ



（注） リダイレクト メッセージの目的のアドレス（最終の宛先）により、確実に隣接ルータのリンクローカル アドレスが特定されるよう、ルータは、その各隣接ルータのリンクローカル アドレスを決定可能である必要があります。スタティック ルーティングの場合は、ルータのリンクローカル アドレスを使用して、ネクストホップ ルータのアドレスを指定する必要があります。ダイナミック ルーティングの場合は、隣接ルータのリンクローカル アドレスを交換するように、すべての IPv6 ルーティング プロトコルを設定する必要があります。

パケットの転送後に、次の条件を満たす場合は、ルータがパケットの送信元にリダイレクト メッセージを送信します。

- パケットの宛先アドレスがマルチキャスト アドレスではない。
- パケットがルータ宛てではなかった。
- パケットが、パケットを受信したインターフェイスから、まもなく送信される。
- ルータが、パケットにより適したファーストホップ ノードはパケットの送信元と同じリンク上にあると決定した。
- パケットの送信元アドレスが、同じリンク上のネイバーのグローバル IPv6 アドレス、またはリンクローカル アドレスである。

仮想化のサポート

IPv6 は、Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）をサポートしています。VRF は Virtual Device Contexts（VDC; 仮想化デバイス コンテキスト）内にあります。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』の 第 13 章「レイヤ 3 仮想化の設定」を参照してください。

IPv6 のライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	IPv6 にはライセンスは不要です。ライセンス パッケージに含まれない機能はいずれも、Cisco NX-OS システム イメージにバンドルされており、無償で提供されます。NX-OS ライセンス方式の詳細説明については、『*Cisco NX-OS Licensing Guide*』を参照してください。

IPv6 の前提条件

IPv6 には、次の前提条件があります。

- IPv6 アドレス指定、IPv6 ヘッダー情報、ICMPv6、IPv6 NDP など、IPv6 の基礎に関する詳しい知識が必要です。
- デバイスをデュアルスタック デバイス（IPv4/IPv6）にする場合は、必ずメモリ / 処理の注意事項に従ってください。

IPv6 の注意事項および制約事項

IPv6 には、次の注意事項、および制約、制限事項があります。

- スイッチは、IPv6 フレームを転送する前にレイヤ 3 パケット情報を確認しないため、IPv6 パケットは、レイヤ 2 LAN スイッチに対して透過的です。IPv6 ホストは、レイヤ 2 LAN スイッチに直接接続できます。
- インターフェイスの同じプレフィクス内に複数の IPv6 グローバル アドレスを設定できます。ただし、インターフェイス上の複数の IPv6 リンクローカル アドレスはサポートされていません。
- RFC 3879 によりサイトローカル アドレスの使用が廃止されたため、RFC 4193 のユニーク ローカル アドレス（UCA）の推奨に従って、プライベート IPv6 アドレスを設定する必要があります。

IPv6 の設定

ここでは、次の内容について説明します。

- IPv6 アドレス指定の設定（p.3-20）
- IPv6 近隣探索の設定（p.3-22）

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

IPv6 アドレス指定の設定

インターフェイスで IPv6 トラフィックを転送するには、インターフェイスで IPv6 アドレスを設定する必要があります。インターフェイスでグローバル IPv6 アドレスを設定すると、リンクローカル アドレスが自動的に設定され、そのインターフェイスで IPv6 が有効となります。

操作の前に

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）。

コマンドの一覧

1. ***config t***
2. ***interface ethernet*** *number*
3. **ipv6 address** *ipv6-address mask* [**secondary**]

または

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

4. **show ipv6 interface**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ethernet** *number*
	
	例：
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

	コマンド	目的

ステップ 3	**ipv6 address** *ipv6-address mask [secondary]*
	or
	ipv6 address *ipv6-address use-link-local-only*
	
	例：
	`switch(config-if)# ipv6 address 2001:0DB8::1/10`
	or
	`switch(config-if)# ipv6 address use-link-local-only`
		インターフェイスに割り当てられた IPv6 アドレスを指定して、インターフェイスで IPv6 処理をイネーブルにします。
		
		ipv6 address コマンドを指定すると、IPv6 アドレスの低位 64 ビットにインターフェイス ID を含むグローバル IPv6 アドレスが設定されます。指定する必要があるのは、このアドレスの 64 ビット ネットワーク プレフィクスだけです。最後の 64 ビットはインターフェイス ID から自動計算されます。
		
		ipv6 address link-local コマンドを指定すると、インターフェイスでリンクローカル アドレスが設定され、インターフェイスで IPv6 がイネーブルになったときに自動設定されるリンクローカル アドレスの代わりに使用されます。
		
		IPv6 アドレスを設定せずに、インターフェイス上で IPv6 処理をイネーブルにします。

ステップ 4	**show ipv6 interface** **例：** `switch(config-if)# show ipv6 interface`	（任意）IPv6 に設定されたインターフェイスを表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、IPv6 アドレスを設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 3/1
switch(config-if)# ipv6 address ?
A:B::C:D/LEN IPv6 prefix format: xxxx:xxxx/ml, xxxx:xxxx::/ml,
xxxx::xx/128
use-link-local-only  Enable IPv6 on interface using only a single link-local
address
switch(config-if)# ipv6 address dc3:dc3::/64 eui64
```

次に、IPv6 インターフェイスを表示する例を示します。

```
switch(config-if)# show ipv6 interface ethernet 3/1
Ethernet3/1, Interface status: protocol-down/link-down/admin-down, iod: 36
    IPv6 address:0dc3:0dc3:0000:0000:0218:baff:fed8:239d
    IPv6 subnet:0dc3:0dc3:0000:0000:0000:0000:0000:0000/64
    IPv6 link-local address:fe80::0218:baff:fed8:239d (default)
    IPv6 multicast routing:disabled
    IPv6 multicast groups locally joined:
        ff02::0001:ffd8:239d  ff02::0002  ff02::0001  ff02::0001:ffd8:239d
    IPv6 multicast (S,G) entries joined:none
    IPv6 MTU:1500 (using link MTU)
    IPv6 RP inbound packet-filtering policy:none
    IPv6 RP outbound packet-filtering policy:none
    IPv6 inbound packet-filtering policy:none
    IPv6 outbound packet-filtering policy:none
    IPv6 interface statistics last reset:never
    IPv6 interface RP-traffic statistics:(forwarded/originated/consumed)
        Unicast packets: 0/0/0
        Unicast bytes: 0/0/0
        Multicast packets: 0/0/0
        Multicast bytes: 0/0/0
```

IPv6 近隣探索の設定

ルータで、IPv6 近隣探索を設定できます。NDP は、IPv6 ノードとルータをイネーブルにして、同じリンク上のネイバーのリンクレイヤ アドレスを決定し、隣接ルータを見つけ、ネイバーの動向を把握します。

操作の前に

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）。最初に、インターフェイスで IPv6 をイネーブルにする必要があります。

コマンドの一覧

1. ***config t***
2. ***interface ethernet*** *number*
3. **ipv6 nd**
4. **show ipv6 nd interface**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ethernet** *number*
	
	例：
	`switch(config)# interface ethernet 2/31`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	**ipv6 nd** **例：** `switch(config-if)# ipv6 nd`	インターフェイスで IPv6 近隣探索をイネーブルにします。
ステップ 4	**show ipv6 nd interface** **例：** `switch(config-if)# show ipv6 nd interface`	（任意）IPv6 近隣探索に設定されたインターフェイスを表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、IPv6 近隣探索の到達可能時間の設定例を示します。

```
switch# config t
switch(config)# interface ethernet 3/1
switch(config-if)# ipv6 nd reachable-time 10
```

次に、IPv6 近隣探索インターフェイスを表示する例を示します。

```
switch(config-if)# show ipv6 nd interface ethernet 3/1
ICMPv6 ND Interfaces for VRF "default"
Ethernet3/1, Interface status: protocol-down/link-down/admin-down
IPv6 address: 0dc3:0dc3:0000:0000:0218:baff:fed8:239d
    ICMPv6 active timers:
        Last Neighbor-Solicitation sent:never
        Last Neighbor-Advertisement sent:never
        Last Router-Advertisement sent:never
        Next Router-Advertisement sent in: 0.000000
    Router-Advertisement parameters:
        Periodic interval:200 to 600 seconds
        Send "Managed Address Configuration" flag:false
        Send "Other Stateful Configuration" flag:false
        Send "Current Hop Limit" field: 64
        Send "MTU" option value: 1500
        Send "Router Lifetime" field:1800 secs
        Send "Reachable Time" field:10 ms
        Send "Retrans Timer" field:0 ms
    Neighbor-Solicitation parameters:
        NS retransmit interval:1000 ms
    ICMPv6 error message parameters:
        Send redirects:false
        Send unreachables:false
```

選択可能なその他の IPv6 近隣探索

次の IPv6 近隣探索コマンドを任意で使用できます。

コマンド	目的
ipv6 nd hop-limit	ルータ アドバタイズメントおよび、ルータにより発信されたすべての IPv6 パケットで使用される最大ホップ数を設定します。
ipv6 nd managed-config-flag	IPv6 ルータ アドバタイズメントに、管理されたアドレス設定フラグを設定します。
ipv6 nd mtu	インターフェイスで送信される IPv6 パケットの MTU サイズを設定します。

コマンド	目的
ipv6 nd ns-interval	インターフェイスで IPv6 ネイバー送信要求メッセージが再送信される時間間隔を設定します。
ipv6 nd other-config-flag	IPv6 ルータ アドバタイズメントに、別のステートフル設定フラグを設定します。
ipv6 nd ra-interval	インターフェイスで IPv6 RA メッセージが送信される時間間隔を設定します。
ipv6 nd ra-lifetime	インターフェイス上の IPv6 RA メッセージのルータのライフタイム値を設定します。
ipv6 nd reachable-time	なんらかの到達可能確認イベントが発生したあとで、リモート IPv6 ノードが到達可能であると判断されるまでの時間を設定します。
ipv6 nd redirects	ICMPv6 リダイレクト メッセージの送信をイネーブルにします。
ipv6 nd retrans-timer	RA のネイバー送信要求メッセージ間のアドバタイズされる時間を設定します。
ipv6 nd suppress-ra	LAN インターフェイス上で IPv6 RA が送信されないようにします。

IPv6 設定の確認

設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show ipv6 interface	IPv6 関連のインターフェイス情報を表示します。
show ipv6 adjacency	隣接関係テーブルを表示します。
show ipv6 icmp	ICMPv6 情報を表示します。
show ipv6 nd	IPv6 近隣探索インターフェイス情報を表示します。
show ipv6 neighbor	IPv6 ネイバー エントリを表示します。

IPv6 設定の例

次に、IPv6 を設定する例を示します。

```
config t
 nterface ethernet 3/1
  ipv6 address dc3:dc3::/64 eui64
   ipv6 nd reachable-time 10
```

デフォルト設定

表 3-5 は、IPv6 パラメータのデフォルト設定の一覧です。

表 3-5　デフォルト IPv6 パラメータ

パラメータ	デフォルト
ND 到達可能時間	0 ミリ秒
ネイバー送信要求再送信間隔	1000 ミリ秒

その他の関連資料

IPv6 の実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.3-26）
- 標準（p.3-26）

関連資料

関連項目	マニュアル名
IPv6 CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0

標準

標準	タイトル
この機能によりサポートされる新規標準や変更された標準はありません。また、この機能による、既存の標準のサポートの変更もありません。	—

CHAPTER 4

DNS の設定

この章では、Domain Name Server（DNS; ドメイン ネーム サーバ）クライアントの設定方法を説明します。

ここでは、次の内容を説明します。

- DNS クライアントについて（p.4-2）
- DNS クライアントのライセンス要件（p.4-4）
- DNS クライアントの前提条件（p.4-4）
- 設定時の注意事項および制約事項（p.4-4）
- DNS クライアントの設定（p.4-5）
- DNS クライアント設定の確認（p.4-9）
- DNS クライアント設定の例（p.4-9）
- DNS クライアント設定の例（p.4-9）
- デフォルト設定（p.4-9）
- その他の関連資料（p.4-10）

DNS クライアントについて

ここでは、次の内容について説明します。

- DNS クライアントの概要（p.4-2）
- ハイ アベイラビリティ（p.4-3）
- 仮想化のサポート（p.4-3）

DNS クライアントの概要

自分で名前の割り当てを管理していないネットワーク内のデバイスとの接続を、ネットワーク デバイスが必要とする場合は、DNS を使用して、ネットワーク間でデバイスを特定する一意のデバイス名を割り当てることができます。DNS は、階層方式を使用して、ネットワーク ノードのホスト名を確立します。これにより、クライアントサーバ方式によるネットワークのセグメントのローカル制御が可能となります。DNS システムは、デバイスのホスト名をそれに関連付けられた IP アドレスに変換して、ネットワーク デバイスを見つけることができます。

インターネット上のドメインは、組織のタイプや場所に基づく一般的なネットワークのグループを表す命名階層ツリーの一部です。ドメイン名は、ピリオド（.）を区切り文字として並べられています。たとえば、シスコは、インターネットでは *com* ドメインで表される営利団体であるため、そのドメイン名は *cisco.com* です。FTP（ファイル転送プロトコル）システムなどの、このドメイン内の特定のホスト名は *ftp.cisco.com* で識別されます。

ネーム サーバ

ネーム サーバはドメイン名の動向を把握し、自身が完全な情報を持っているドメイン ツリーの部分を認識しています。ネーム サーバは、ドメイン ツリーの他の部分の情報を格納している場合もあります。Cisco NX-OS 内の IP アドレスにドメイン名をマッピングするには、最初にホスト名を示し、そのあとにネーム サーバを指定して、DNS サービスをイネーブルにする必要があります。

Cisco NX-OS では、スタティックに IP アドレスをドメイン名にマッピングできます。また、1 つ以上のドメイン ネーム サーバを使用してホスト名の IP アドレスを見つけるよう、Cisco NX-OS を設定することもできます。

DNS の動作

ネーム サーバは、クライアントが DNS サーバに発行した、特定のゾーン内でローカルに定義されたホストの照会を次のように処理します。

- 権限ネーム サーバは、その権限ゾーン内のドメイン名を求める DNS ユーザ照会に、自身のホスト テーブル内にキャッシュされた永久的なエントリを使用して応答します。照会で求められているのが、自身の権限ゾーン内であるが、設定情報が登録されていないドメイン名の場合、権限ネーム サーバは単に、その情報が存在しないと返信します。
- 権限ネーム サーバとして設定されていないネーム サーバは、以前に受信した照会への返信からキャッシュした情報を使用して、DNS ユーザ照会に応答します。ゾーンの権限ネーム サーバとして設定されたルータがない場合は、ローカルに定義されたホストを求める DNS サーバへの照会には、正規の返信は送信されません。

ネーム サーバは、特定のドメインに設定された転送パラメータおよびルックアップ パラメータに従って、DNS 照会に応答します（着信 DNS 照会を転送するか、内部的に生成された DNS 照会を解決します）。

ハイ アベイラビリティ

Cisco NX-OS は、DNS クライアントのステートレス再起動をサポートしています。リブートまたはスーパーバイザ スイッチオーバーのあとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化のサポート

Cisco NX-OS は、同じシステム上で動作する、DNS クライアントの複数インスタンスをサポートしています。各 VDC で DNS クライアントを設定できます。または、VDC 内の各 VRF で、異なる DNS クライアント設定を使用できます。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

DNS クライアントのライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	DNS にはライセンスは不要です。ライセンス パッケージに含まれない機能はいずれも、Cisco NX-OS システム イメージにバンドルされており、無償で提供されます。NX-OS ライセンス方式の詳細説明については、『*Cisco NX-OS Licensing Guide*』を参照してください。

DNS クライアントの前提条件

DNS クライアントには、次の前提条件があります。

- ネットワーク上に DNS ネーム サーバが必要です。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、目的の VDC を入力します（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

DNS クライアントは特定の VRF で設定します。VRF を指定しない場合、Cisco NX-OS はデフォルト VRF を使用します。

DNS クライアントの設定

ここでは、DNS クライアントの設定方法と次の内容について説明します。

- DNS クライアントの設定（p.4-5）
- 仮想化の設定（p.4-7）

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

DNS クライアントの設定

ネットワーク上の DNS サーバを使用するよう、DNS クライアントを設定できます。

操作の前に

ネットワーク上にドメイン ネーム サーバがあることを確認します。

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）。

コマンドの一覧

1. **config** ***t***
2. **ip host** *name address1* [*address2 ... address6*]
3. **ip domain-name** *name* [**all-vrfs use-vrf** *vrf-name*]
4. **ip domain-list** *name* [**all-vrfs use-vrf** *vrf-name*]
5. **ip name-server** *server-address1* [*server-address2 ... server-address6*] [**all-vrfs use-vrf** *vrf-name*]
6. **ip domain lookup**
7. **show hosts**
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**ip host** ***name address1*** [***address2 ... address6***] **例：** `switch(config)# ip host cisco-rtp 192.0.2.1`	ホスト名キャッシュに、6 つまでのスタティック ホスト 名前 / アドレス マッピングを定義します。使用可能なアドレスは、IPv4 アドレスまたは IPv6 アドレスです。

	コマンド	目的
ステップ 3	**ip domain-name** ***name*** [**all-vrfs use-vrf** ***vrf-name***] **例：** `switch(config)# ip domain-name myserver.com`	（任意）Cisco NX-OS が無条件ホスト名を完成するために使用するデフォルト ドメイン ネーム サーバを定義します。または、このドメイン ネーム サーバを、デバイス上のすべての VRF にわたって定義し、このドメイン名を設定した VRF を無効にする VRF を指定することもできます。 Cisco NX-OS は、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にデフォルト ドメイン名をアペンドします。
ステップ 4	**ip domain-list** ***name*** [**all-vrfs use-vrf** ***vrf-name***] **例：** `switch(config)# ip domain-list mycompany.com`	（任意）Cisco NX-OS が無条件ホスト名を完成するために使用できる追加のドメイン ネーム サーバを定義します。または、このドメイン リストを、デバイス上のすべての VRF にわたって定義し、このドメイン リストを設定した VRF を無効にする VRF を指定することもできます。 Cisco NX-OS はドメイン リスト内の各エントリを使用して、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にこのドメイン名をアペンドします。Cisco NX-OS は、一致するものが見つかるまで、ドメイン リストの各エントリにこれを実行します。
ステップ 5	**ip name-server** ***address1*** [***address2 ... address6***] [**all-vrfs use-vrf** ***vrf-name***] **例：** `switch(config)# ip name-server 192.0.2.22`	（任意）最大 6 つのネーム サーバを定義します。使用可能なアドレスは、IPv4 アドレスまたは IPv6 アドレスです。 または、このネーム サーバを、デバイス上のすべての VRF にわたって定義し、このネーム サーバを設定した VRF を無効にする VRF を指定することもできます。
ステップ 6	**ip domain-lookup** **例：** `switch(config)# ip domain-lookup`	（任意）DNS ベースのアドレス変換をイネーブルにします。デフォルトではイネーブルにされています。
ステップ 7	**show hosts** **例：** `switch(config)# show hosts`	（任意）DNS に関する情報を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、デフォルト ドメイン名を設定し、DNS ルックアップをイネーブルにする例を示します。

```
switch# config t
switch(config)# ip domain-name cisco.com 192.0.2.1 all-vrfs use-vrf management
switch(config)# ip domain-lookup
switch(config# copy running-config startup-config
```

仮想化の設定

DNS クライアントを VRF 内で設定できます。vrf コンフィギュレーション モードを使用しない場合は、ご使用の DNS クライアント設定がデフォルト VRF に適用されます。

または、DNS クライアントを設定した VRF 以外の、指定した VRF を使用するよう、DNS クライアントを設定することもできます。たとえば、DNS クライアントを 赤の VRF で設定していても、青の VRF を使用して DNS サーバと通信できます。

操作の前に

ネットワーク上にドメイン ネーム サーバがあることを確認します。

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）。

コマンドの一覧

1. **config** ***t***
2. **vrf context** *vrf-name*
3. **ip domain-name** *name* [**all-vrfs use-vrf** *vrf-name*]
4. **ip domain-list** *name* [**all-vrfs use-vrf** *vrf-name*]
5. **ip name-server** *server-address1* [*server-address2 ... server-address6*] [**all-vrfs use-vrf** *vrf-name*]
6. **ip domain lookup**
7. **show hosts**
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		コンフィギュレーション モードを開始します。

ステップ 1	**vrf context vrf-name**
	
	例：
	`switch(config)# vrf context Red`
	`switch(config-vrf)#`
		VRF を作成し、VRF コンフィギュレーション モードを使用します。

ステップ 2	**ip domain-name *name*** [**all-vrfs use-vrf *vrf-name***] **例：** `switch(config-vrf)# ip domain-name myserver.com`	（任意）Cisco NX-OS が無条件ホスト名を完成するために使用するデフォルト ドメイン ネーム サーバを定義します。または、このドメイン ネーム サーバを、デバイス上のすべての VRF にわたって定義し、このドメイン名を設定した VRF を無効にする VRF を指定することもできます。 Cisco NX-OS は、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にデフォルト ドメイン名をアペンドします。

	コマンド	目的
ステップ 3	**ip domain-list** ***name*** [**all-vrfs use-vrf** ***vrf-name***] **例：** `switch(config-vrf)# ip domain-list mycompany.com`	（任意）Cisco NX-OS が無条件ホスト名を完成するために使用できる追加のドメイン ネーム サーバを定義します。または、このドメイン リストを、デバイス上のすべての VRF にわたって定義し、このドメイン リストを設定した VRF を無効にする VRF を指定することもできます。 Cisco NX-OS はドメイン リスト内の各エントリを使用して、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にこのドメイン名をアペンドします。Cisco NX-OS は、一致するものが見つかるまで、ドメイン リストの各エントリにこれを実行します。
ステップ 4	**ip name-server** ***address1*** [***address2 ... address6***] [**all-vrfs use-vrf** ***vrf-name***] **例：** `switch(config-vrf)# ip name-server 192.0.2.22`	（任意）最大 6 つのネーム サーバを定義します。使用可能なアドレスは、IPv4 アドレスまたは IPv6 アドレスです。 または、このネーム サーバを、デバイス上のすべての VRF にわたって定義し、このネーム サーバを設定した VRF を無効にする VRF を指定することもできます。
ステップ 5	**ip domain-lookup** **例：** `switch(config-vrf)# ip domain-lookup`	（任意）DNS ベースのアドレス変換を、この VRF に対してイネーブルにします。デフォルトではイネーブルにされています。
ステップ 6	**show hosts** **例：** `switch(config)-vrf# show hosts`	（任意）DNS に関する情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-vrf)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、デフォルト ドメイン名を設定し、VRF 内で DNS ルックアップをイネーブルにする例を示します。

```
switch# config t
switch(config)# vrf context Red
switch(config-vrf)# ip domain-name cisco.com 192.0.2.1 all-vrfs use-vrf management
switch(config-vrf)# ip domain-lookup
switch(config-vrf)# copy running-config startup-config
```

DNS クライアント設定の確認

DNS クライアント設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
`show hosts`	DNS に関する情報を表示します。

DNS クライアント設定の例

次に、いくつかの代替ドメイン名を使用したドメイン リストの例を示します。

```
ip domain list csi.com
ip domain list telecomprog.edu
ip domain list merit.edu
```

次に、ホストの名前 / アドレス マッピング処理を設定して、IP DNS ベースの変換を指定する例を示します。この例では、ネーム サーバのアドレスとデフォルト ドメイン名も設定されます。

```
ip domain lookup
ip name-server 192.168.1.111 192.168.1.2
ip domain name cisco.com
```

デフォルト設定

表 4-1 は、DNS クライアント パラメータのデフォルト設定の一覧です。

表 4-1　デフォルト DNS クライアント パラメータ

パラメータ	デフォルト
DNS クライアント	イネーブル

その他の関連資料

DNS クライアントの実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.4-10）
- 標準（p.4-10）

関連資料

関連項目	マニュアル名
DNS クライアント CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

標準

標準	タイトル
この機能によりサポートされる新規標準や変更された標準はありません。また、この機能による、既存の標準のサポートの変更もありません。	—

PART 2

ルーティング

CHAPTER 5

OSPFv2 の設定

この章では、IPv4 ネットワーク向けの OSPFv2（Open Shortest Path First version 2）の設定方法を説明します。

ここでは、次の内容を説明します。

- OSPFv2 について（p.5-2）
- OSPFv2 のライセンス要件（p.5-14）
- OSPFv2 の前提条件（p.5-14）
- 設定時の注意事項および制約事項（p.5-14）
- 基本的 OSPFv2 の設定（p.5-15）
- 高度な OSPFv2 の設定（p.5-24）
- OSPFv2 設定の確認（p.5-42）
- OSPFv2 統計情報の表示（p.5-42）
- OSPFv2 設定の例（p.5-43）
- デフォルト設定（p.5-43）
- その他の関連資料（p.5-44）

OSPFv2 について

OSPFv2 は、IPv4 ネットワーク用 IETF リンクステート プロトコルです（「リンクステート プロトコル」[p.1-9] を参照）。OSPFv2 ルータは、*hello パケット*と呼ばれる特別なメッセージを各 OSPF イネーブル インターフェイスに送信して、他の OSPFv2 隣接ルータを探索します。隣接ルータが発見されると、この 2 台のルータは hello パケットの情報を比較して、両者の設定に互換性のあるかどうかを判定します。これらの隣接ルータは*隣接関係*を確立しようとします。つまり、両者のリンクステート データベースを同期させて、確実に同じ OSPFv2 ルーティング情報を持つようにします。隣接ルータは、各リンクの稼働状態に関する情報、リンクのコスト、およびその他のあらゆる近隣情報を含む*リンクステート アドバタイズメント*（LSA; リンクステート アドバタイズメント）を共有します。これらのルータはその後、受信した LSA をすべての OSPF イネーブル インターフェイスにフラッディングします。これにより、すべての OSPFv2 ルータのリンクステート データベースが最終的に同じになります。すべての OSPFv2 ルータのリンクステート データベースが同じになると、ネットワークは*収束されます*（「収束」[p.1-6] を参照）。その後、各ルータは、ダイクストラの最短パス優先（SPF）アルゴリズムを使用して、自身のルーティング テーブルを構築します。

OSPFv2 ネットワークは、複数のエリアに分割できます。ルータは、ほとんどの LSA を 1 つのエリア内にのみ送信するため、OSPF イネーブル ルータの CPU とメモリの要件が緩やかになります。

OSPFv2 は IPv4 をサポートし、OSPFv3 は IPv6 をサポートしています。詳細については、第 6 章「OSPFv3 の設定」を参照してください。

ここでは、次の内容について説明します。

- hello パケット（p.5-2）
- ネイバー（p.5-3）
- 隣接関係（p.5-3）
- 代表ルータ（p.5-4）
- エリア（p.5-5）
- Link-State Advertisement（p.5-6）
- OSPFv2 とユニキャスト RIB（p.5-7）
- 認証（p.5-8）
- 高度な機能（p.5-8）

hello パケット

OSPFv2 ルータは、すべての OSPF イネーブル インターフェイスに hello パケットを定期的に送信します。ルータがこの hello パケットを送信する頻度は、インターフェイスごとに設定された *hello インターバル*により決定されます。OSPFv2 は、hello パケットを使用して、次のタスクを実行します。

- 近隣探索
- キープアライブ
- 双方向通信
- 代表ルータの選定（「代表ルータ」[p.5-4] を参照）

hello パケットには、リンクの OSPFv2 コスト割り当て、hello 間隔、送信元ルータのオプション機能など、送信元の OSPFv2 インターフェイスとルータに関する情報が含まれます。これらの hello パケットを受信する OSPFv2 インターフェイスは、設定に受信インターフェイスの設定との互換性があるかどうかを判定します。互換性のあるインターフェイスはネイバーとみなされ、ネイバー テーブルに追加されます（「ネイバー」[p.5-3] を参照）。

hello パケットには、送信元インターフェイスが通信したルータのルータ ID のリストも含まれます。受信インターフェイスが、このリストで自身の ID を見つけた場合は、2 つのインターフェイス間で双方向通信が確立されます。

OSPFv2 は、hello パケットをキープアライブ メッセージとして使用して、ネイバーが通信を継続中であるかどうかを判定します。ルータが、設定された *デッド インターバル*（通常は hello 間隔の倍数）で hello パケットを受信しない場合は、そのネイバーがローカル ネイバー テーブルから削除されます。

ネイバー

ネイバーとは見なされるためには、OSPFv2 インターフェイスがリモート インターフェイスとの互換性を持つよう設定されている必要があります。この 2 つの OSPFv2 インターフェイスで、次の基準が一致している必要があります。

- hello 間隔
- デッド間隔
- エリア ID（「エリア」[p.5-5] を参照）
- 認証
- オプション機能

一致する場合は、次の情報がネイバー テーブルに入力されます。

- ネイバー ID ― ネイバーのルータ ID
- 優先度 ― ネイバーの優先度。優先度は、代表ルータの選定（「代表ルータ」[p.5-4] を参照）に使用されます。
- 状態 ― ネイバーから通信があったか、双方向通信の確立処理中であるか、リンクステート情報を共有しているか、または完全な隣接関係が確立されたかを示します。
- デッド タイム ― このネイバーから最後の hello パケットを受信した後に経過した時間を示します。
- IP アドレス ― ネイバーの IP アドレス
- 代表ルータ ― ネイバーが代表ルータ、またはバックアップ代表ルータとして宣言されたかどうかを示します（「代表ルータ」[p.5-4] を参照）。
- ローカル インターフェイス ― このネイバーの hello パケットを受信したローカル インターフェイス

隣接関係

すべてのネイバーが隣接関係を確立するわけではありません。ネットワーク タイプと確立された代表ルータに応じて、完全な隣接関係を確立して、すべてのネイバーと LSA を共有するものと、そうでないものがあります。詳細については、「代表ルータ」（p.5-4）を参照してください。

隣接関係は、OSPF のデータベース説明パケット、リンク状態要求パケット、およびリンク状態更新パケットを使用して確立されます。データベース説明パケットに含まれるのは、ネイバーのリンクステート データベースからの LSA ヘッダーのみです（「リンクステート データベース」[p.5-7] を参照）。ローカル ルータは、これらのヘッダーを自身のリンクステート データベースと比較して、新規の LSA か、更新された LSA かを判定します。ローカル ルータは、新規または更新の情報を必要とする各 LSA について、リンク状態要求パケットを送信します。これに対し、ネイバーはリンク状態更新パケットを返信します。このパケット交換は、両方のルータのリンクステート情報が同じになるまで継続します。

代表ルータ

複数のルータを含むネットワークは、OSPF 特有の状況です。すべてのルータがネットワークで LSA をフラッディングした場合は、同じリンクステート情報が複数の送信元から送信されます。ネットワークのタイプに応じて、OSPFv2 は*代表ルータ*（*DR*）という 1 台のルータを使用して、LSA のフラッディングを制御し、OSPFv2 の残りの部分に対してネットワークを代表する場合があります（「エリア」[p.5-5] を参照）。DR がダウンした場合、OSPFv2 は*バックアップ代表ルータ*（BDR）を選定し、この BDR を使用します。

ネットワーク タイプは次のとおりです。

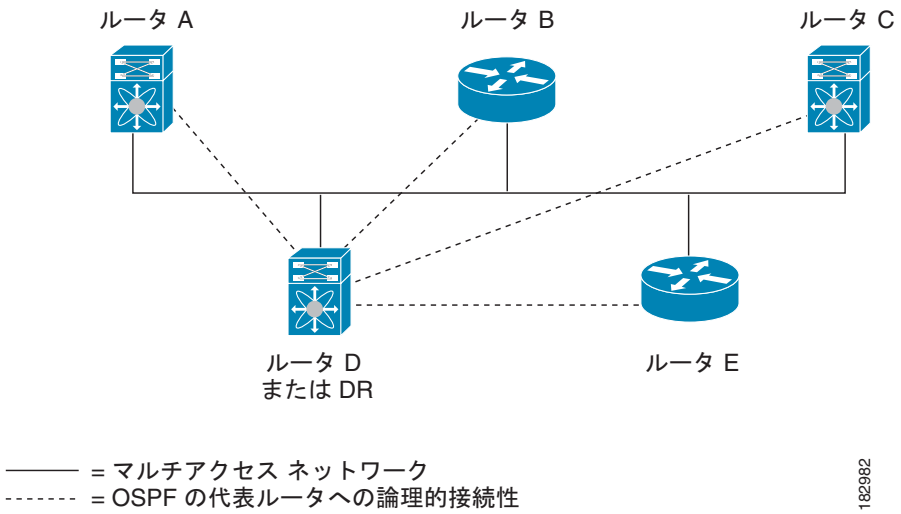
- ポイントツーポイント ― 2 台のルータ間にのみ存在するネットワーク。ポイントツーポイント ネットワーク上の全ネイバーは隣接関係を確立し、DR は存在しません。
- ブロードキャスト ― ブロードキャスト トラフィックが可能なイーサネットなどの共有メディア上で通信できる複数のルータを持つネットワーク。OSPFv2 ルータは DR および BDR を確立し、これらにより、ネットワーク上の LSA フラッディングを制御します。OSPFv2 は、よく知られている IPv4 マルチキャスト アドレス 224.0.0.5 および MAC アドレス 0100.5300.0005 を使用して、ネイバーと通信します。

DR と BDR は、hello パケット内の情報に基づいて選択されます。インターフェイスは hello パケットの送信時に、どれが DR および BDR かわかっている場合は、優先フィールドと、DR および BDR フィールドを設定します。ルータは、hello パケットの DR および BDR フィールドで宣言されたルータと優先フィールドに基づいて、選定手順を実行します。最終的に OSPFv2 は、もっとも大きいルータ ID を DR および BDR として選択します。

他のルータはすべて DR および BDR と隣接関係を確立し、IPv4 マルチキャスト アドレス 224.0.0.6 を使用して、LSA 更新情報を DR と BDR に送信します。図 5-1 は、すべてのルータと DR の間のこの隣接関係を示します。

DR は、ルータ インターフェイスに基づいています。1 つのネットワークの DR であるルータは、別のインターフェイス上の他のネットワークの DR となることはできません。

図 5-1　マルチアクセス ネットワークの DR

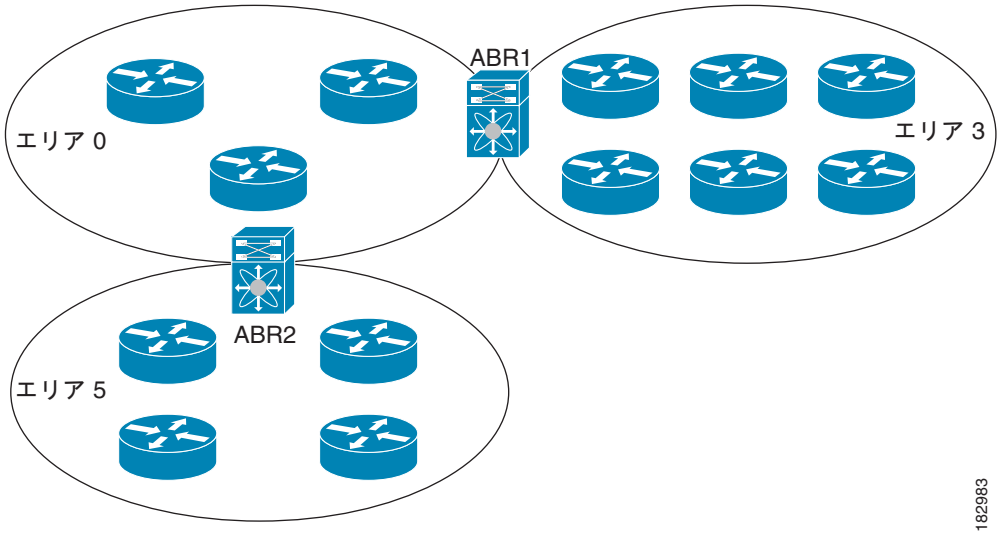


エリア

OSPFv2 ネットワークを複数の*エリア*に分割すると、ルータに要求される OSPFv2 の CPU とメモリに関する要件を制限できます。エリアとは、ルータの論理的な区分で、OSPFv2 ドメイン内にリンクして別のサブドメインを作成します。LSA フラッディングはエリア内でのみ発生し、リンクステート データベースはエリア内のリンクにのみ制限されます。定義されたエリア内のインターフェイスには、エリア ID を割り当てることができます。エリア ID は、10.2.3.1 などの、数字またはドット付き 10 進表記で表現される 32 ビット値です。

OSPFv2 ネットワーク内に複数のエリアを定義する場合は、0 という予約されたエリア ID を持つバックボーン エリアも定義する必要があります。エリアが複数ある場合は、1 台以上のルータが*エリア境界ルータ*（ABR; エリア境界ルータ）となります。ABR は、バックボーン エリアと他の 1 つ以上の定義済みエリアの両方に接続します（図 5-2 を参照）。

図 5-2　OSPFv2 エリア



ABR には、接続するエリアごとに個別のリンクステート データベースがあります。ABR は、接続したエリアの 1 つからバックボーン エリアにネットワーク集約（タイプ 3）LSA（「ルート集約」[p.5-11] を参照）を送信します。バックボーン エリアは、1 つのエリアに関する集約情報を別のエリアに送信します。図 5-2 では、エリア 0 が、エリア 5 に関する集約情報をエリア 3 に送信しています。

OSPFv2 では、Autonomous System Boundary Router（ASBR; 自律システム境界ルータ）という、もう 1 つのルータ タイプも定義されています。このルータは、OSPFv2 エリアを別の自律システムに接続します。自律システムとは、単一の技術的管理エンティティにより制御されるネットワークです。OSPFv2 は、そのルーティング情報を別の自律システムに再配布したり、再配布されたルートを別の自律システムから受信したりできます。詳細については、「高度な機能」（p.5-8）を参照してください。

Link-State Advertisement

OSPFv2 は LSA を使用して、自身のルーティング テーブルを構築します。

ここでは、次の内容について説明します。

- LSA タイプ（p.5-6）
- リンク コスト（p.5-6）
- フラッディングと LSA グループ ペーシング（p.5-7）
- リンクステート データベース（p.5-7）
- 不透明 LSA（p.5-7）

LSA タイプ

表 5-1 は、Cisco NX-OS でサポートされる LSA タイプを示します。

表 5-1　LSA タイプ

タイプ	名前	説明
1	ルータ LSA	すべてのルータが送信する LSA。この LSA には、すべてのリンクの状態とコスト、およびリンク上のすべての OSPFv2 ネイバーの一覧が含まれます。ルータ LSA は SPF 再計算をトリガーします。ルータ LSA はローカル OSPFv2 エリアにフラッディングされます。
2	ネットワーク LSA	DR が送信する LSA。この LSA には、マルチアクセス ネットワーク内のすべてのルータの一覧が含まれます。ネットワーク LSA は SPF 再計算をトリガーします。「代表ルータ」（p.5-4）を参照してください。
3	ネットワーク集約 LSA	ABR が、ローカル エリア内の宛先ごとに外部エリアに送信する LSA。この LSA には、ABR からローカルの宛先へのリンク コストが含まれます。「エリア」（p.5-5）を参照してください。
4	ASBR 集約 LSA	ABR が外部エリアに送信する LSA。この LSA は、リンク コストを ASBR のみにアドバタイズします。「エリア」（p.5-5）を参照してください。
5	AS 外部 LSA	ASBR が生成する LSA。この LSA には、外部 AS 宛先へのリンク コストが含まれます。AS 外部 LSA は、AS 全体にわたってフラッディングされます。「エリア」（p.5-5）を参照してください。
7	NSSA 外部 LSA	ASBR が Not-So-Stubby Area（NSSA）内で生成する LSA。この LSA には、外部 AS 宛先へのリンク コストが含まれます。NSSA 外部 LSA は、ローカル NSSA 内のみでフラッディングされます。「エリア」（p.5-5）を参照してください。
9–11	不透明 LSA	OSPF の拡張に使用される LSA。「不透明 LSA」（p.5-7）を参照してください。

リンク コスト

各 OSPFv2 インターフェイスには、*リンク コスト*が割り当てられます。このコストは任意の数字です。デフォルトでは、Cisco NX-OS が、設定された参照帯域幅をインターフェイス帯域幅で割った値をコストとして割り当てます。デフォルトでは、参照帯域幅は 40 Gbps です。リンク コストは各リンクに対して、LSA 更新情報で伝えられます。

フラッディングと LSA グループ ペーシング

OSPFv2 ルータは、LSA を受信すると、その LSA をすべての OSPF 対応インターフェイスに転送し、OSPFv2 エリアをこの情報でフラッディングします。この LSA フラッディングにより、ネットワーク内のすべてのルータが同じルーティング情報を持つことが保証されます。LSA フラッディングは、OSPFv2 エリアの設定により異なります（「エリア」[p.5-5] を参照）。LSA は、*リンクステート リフレッシュ*時間に基づいて（デフォルトでは 30 分ごとに）フラッディングされます。各 LSA にはそれぞれ、リンクステート リフレッシュ時間が設定されています。

ネットワークの LSA 更新情報のフラッディング レートは、LSA グループ ペーシング機能を使用して制御できます。LSA グループ ペーシングにより、CPU またはバッファの使用率を低下させることができます。この機能により、同様のリンクステート リフレッシュ時間を持つ LSA がグループ化されるため、OSPFv2 で、複数の LSA を 1 つの OSPFv2 更新メッセージにまとめることが可能となります。

デフォルトでは、相互のリンクステート リフレッシュ時間が 4 分以内の LSA が同じグループに入れられます。この値は、大規模なリンクステート データベースでは低く、小規模のデータベースでは高くして、ネットワーク上の OSPFv2 負荷を最適化する必要があります。

リンクステート データベース

各ルータは、OSPFv2 ネットワーク用のリンクステート データベースを維持しています。このデータベースには、接続されたすべての LSA が含まれ、ネットワークを通過するすべてのルートに関する情報が格納されます。OSPFv2 は、この情報を使用して、各宛先への最適なパスを計算し、この最適なパスをルーティング テーブルに入力します。

LSA は、MaxAge と呼ばれる設定済みの時間間隔で 受信された LSA 更新情報がまったくない場合は、リンクステート データベースから削除されます。ルータは、LSA を 30 分ごとに繰り返してフラッディングし、正確なリンクステート情報が期限切れで削除されるのを防ぎます。Cisco NX-OS は、すべての LSA が同時にリフレッシュされるのを防ぐために、LSA グループ機能をサポートしています。詳細については、「フラッディングと LSA グループ ペーシング」（p.5-7）を参照してください。

不透明 LSA

不透明 LSA により、OSPF 機能の拡張が可能となります。不透明 LSA は、標準 LSA ヘッダーと、それに続くアプリケーション固有の情報で構成されます。この情報は、OSPFv2 または他のアプリケーションにより使用される場合があります。OSPFv2 は不透明 LSA を使用して、OSPFv2 グレースフル リスタート機能（「ハイ アベイラビリティとグレースフル リスタート」[p.5-11] を参照）をサポートしています。次のような 3 種類の不透明 LSA タイプが定義されています。

- LSA タイプ 9 — ローカル ネットワークにフラッディングされます。
- LSA タイプ 10 — ローカル エリアにフラッディングされます。
- LSA タイプ 11 — ローカル AS にフラッディングされます。

OSPFv2 とユニキャスト RIB

OSPFv2 は、リンクステート データベースでダイクストラの SPF アルゴリズムを実行します。このアルゴリズムにより、パス上の各リンクのリンク コストの合計に基づいて、各宛先への最適なパスが選択されます。そして、選択された各宛先への最短パスが OSPFv2 ルーティング テーブルに入力されます。OSPFv2 ネットワークが収束すると、このルーティング テーブルはユニキャスト RIB にデータを提供します。OSPFv2 はユニキャスト RIB と通信し、次の動作を行います。

- ルートの追加または削除
- 他のプロトコルからのルートの再配布への対応
- 変更されていない OSPFv2 ルートの削除およびスタブ ルータ アドバタイズメントを行うための収束更新情報の提供（「OSPFv2 スタブ ルータ アドバタイズメント」[p.5-12] を参照）

さらに OSPFv2 は、変更済みダイクストラ アルゴリズムを実行して、集約および外部（タイプ 3、4、5、7）LSA の変更の高速再計算を行います。

認証

OSPFv2 メッセージに認証を設定して、ネットワークでの不正な、または無効なルーティング更新を防止できます。Cisco NX-OS は、次の 2 つの認証方式をサポートしています。

- 簡易パスワード認証
- MD5 認証ダイジェスト

OSPFv2 認証は、OSPFv2 エリアに対して、またはインターフェイスごとに設定できます。

簡易パスワード認証

簡易パスワード認証では、OSPFv2 メッセージの一部として送信された単純なクリア テキストのパスワードを使用します。受信 OSPFv2 ルータが OSPFv2 メッセージを有効なルート更新情報として受け入れるには、同じクリア テキスト パスワードで設定されている必要があります。パスワードがクリア テキストであるため、ネットワーク上のトラフィックを監視できるあらゆるユーザがパスワードを入手できます。

MD5 認証

OSPFv2 メッセージを認証するには、MD5 認証を使用する必要があります。そのためには、ローカル ルータとすべてのリモート OSPFv2 ネイバーが共有するパスワードを設定します。Cisco NX-OS は各 OSPFv2 メッセージに対して、メッセージと暗号化されたパスワードに基づく MD5 一方向メッセージ ダイジェストを作成します。インターフェイスはこのダイジェストを OSPFv2 メッセージとともに送信します。受信する OSPFv2 ネイバーは、同じ暗号化パスワードを使用して、このダイジェストを確認します。メッセージが変更されていない場合はダイジェストの計算が同一であるため、OSPFv2 メッセージは有効とみなされます。

MD5 認証には、ネットワークでのメッセージの再送を防ぐための、各 OSPFv2 メッセージのシーケンス番号が含まれます。

高度な機能

Cisco NX-OS は、ネットワークでの OSPFv2 の可用性やスケーラビリティを向上させる数多くの高度な OSPFv2 機能をサポートしています。ここでは、次の内容について説明します。

- スタブ エリア（p.5-9）
- Not-So-Stubby エリア（p.5-9）
- 仮想リンク（p.5-10）
- ルート再配布（p.5-10）
- ルート集約（p.5-11）
- ハイ アベイラビリティとグレースフル リスタート（p.5-11）
- OSPFv2 スタブ ルータ アドバタイズメント（p.5-12）

- 複数の OSPFv2 インスタンス（p.5-12）
- SPF 最適化（p.5-13）
- 仮想化のサポート（p.5-13）

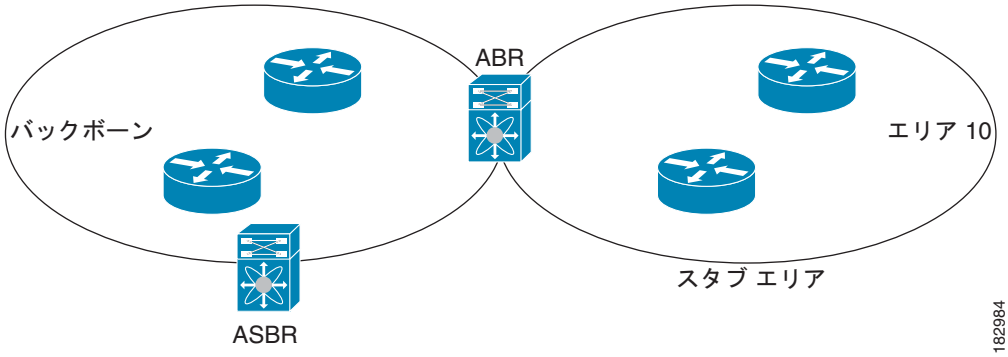
スタブ エリア

エリアを*スタブ エリア*にすると、エリアでフラッディングされる外部ルーティング情報の量を制限できます。スタブ エリアとは、AS 外部（タイプ 5）LSA（「Link-State Advertisement」[p.5-6] を参照）が許可されないエリアです。これらの LSA は通常、外部ルーティング情報を伝播するためにローカル AS 全体でフラッディングされます。スタブ エリアには、次の要件があります。

- スタブ エリア内のすべてのルータはスタブ ルータです。「スタブ ルーティング」（p.1-7）を参照してください。
- スタブ エリアには ASBR ルータは存在しません。
- スタブ エリアには仮想リンクを設定できません。

図 5-3 は、外部 AS に到達するためにエリア 10 内のすべてのルータが ABR を通過する必要のある OSPFv2 AS の例を示します。エリア 10 は、スタブ エリアとして設定できます。

図 5-3　スタブ エリア



スタブ エリアは、外部 AS へのバックボーン エリアを通過する必要のあるすべてのトラフィックにデフォルト ルートを使用します。IPv4 の場合のデフォルト ルートは 0.0.0.0 です。

Not-So-Stubby エリア

Not-So-Stubby Area（*NSSA*）はスタブ エリアに似ていますが、NSSA では、再配布を使用して NSSA 内で AS 外部ルートを使用できる点が異なります。NSSA ASBR はこれらのルートを再配布し、NSSA 外部（タイプ 7）LSA を生成して NSSA 全体でフラッディングします。または、この NSSA 外部 LSA を AS 外部（タイプ 5）LSA に変換するよう、NSSA を他のエリアに接続する ABR を設定することができます。こうすると、ABR は、これらの AS 外部 LSA を OSPFv2 AS 全体にフラッディングします。変換時には、集約とフィルタリングがサポートされます。NSSA 外部 LSA の詳細については、「Link-State Advertisement」（p.5-6）を参照してください。

たとえば、OSPFv2 を使用する中央サイトを、異なるルーティング プロトコルを使用するリモート サイトに接続するときに NSSA を使用すると、管理作業を簡素化できます。リモート サイトへのルートはスタブ エリア内に再配布できないため、NSSA を使用する前に、企業サイトの境界ルータ

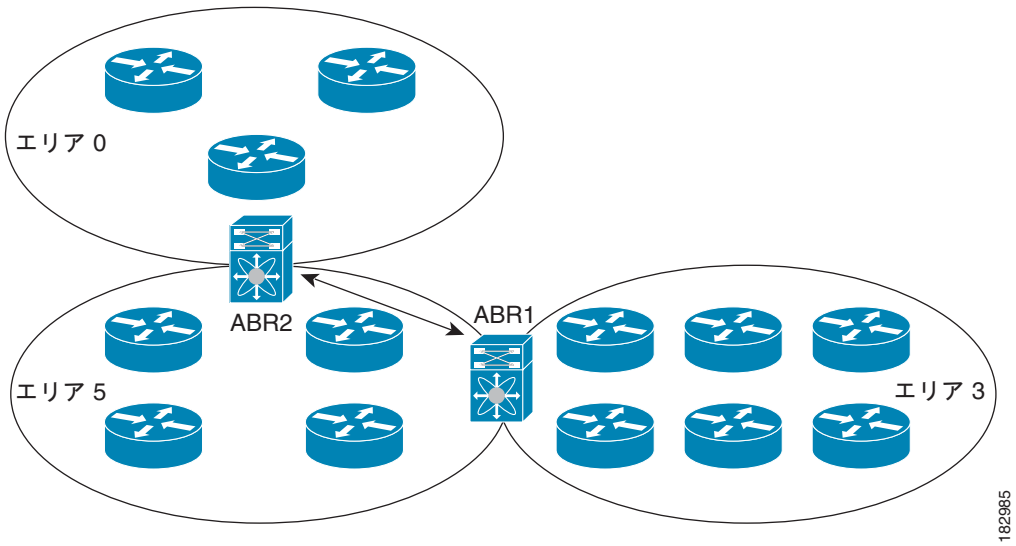
とリモート ルータの間の接続を OSPFv2 スタブ エリアとして実行することはできません。NSSA を使用すると、企業のルータとリモート ルータ間のエリアを NSSA として定義する（「NSSA の設定」[p.5-27] を参照）ことで、OSPFv2 を拡張してリモート接続性をサポートできます。

バックボーン エリア 0 を NSSA にすることはできません。

仮想リンク

仮想リンクを使用すると、物理的に直接接続できない場合に、OSPFv2 エリア ABR をバックボーン エリア ABR に接続できます。図 5-4 は、エリア 3 をエリア 5 経由でバックボーン エリアに接続する仮想リンクを示します。

図 5-4　仮想リンク



また、仮想リンクを使用して、分割エリアから一時的に回復できます。分割エリアは、エリア内のリンクがダウンしたために隔離された一部のエリアで、ここからはバックボーン エリアへの代表 ABR に到達できません。

ルート再配布

OSPFv2 は、ルート再配布を使用して、他のルーティング プロトコルからルートを学習できます。「ルート再配布」（p.1-6）を参照してください。リンク コストをこれらの再配布されたルートに割り当てるか、またはデフォルト リンク コストを再配布されたすべてのルートに割り当てるよう、OSPFv2 を設定します。

ルート再配布では、ルートマップを使用して、再配布する外部ルートを管理します。ルートマップの設定の詳細については、第 14 章「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、これらの外部ルートがローカル OSPFv2 AS でアドバタイズされる前に AS 外部（タイプ 5）LSA および NSSA 外部（タイプ 7）LSA のパラメータを変更できます。

ルート集約

OSPFv2 は、学習したすべてのルートを、すべての OSPF 対応ルータと共有するため、ルート集約を使用して、すべての OSPF 対応ルータにフラッディングされる一意のルートの数を削減した方がよい場合があります。ルート集約により、より具体的な複数のアドレスが、すべての具体的なアドレスを表す 1 つのアドレスに置き換えられるため、ルーティング テーブルが簡素化されます。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。

一般的には、ABR の境界ごとに集約します。集約は 2 つのエリアの間でも設定できますが、バックボーンの方向に集約する方が適切です。こうすると、バックボーンがすべての集約アドレスを受信し、すでに集約されているそれらのアドレスを他のエリアに投入できるためです。集約には、次の 2 タイプがあります。

- エリア間ルート集約
- 外部ルート集約

エリア間ルート集約は ABR 上で設定し、AS 内のエリア間のルートを集約します。集約の利点を生かすには、これらのアドレスを 1 つの範囲内にまとめられるよう、連続するネットワーク番号をエリア内で割り当てる必要があります。

外部ルート集約は、ルート再配布を使用して OSPFv2 に投入される外部ルートに特有のルート集約です。集約する外部の範囲が連続していることを確認する必要があります。異なる 2 台のルータからの重複範囲を集約すると、誤った宛先にパケットが送信される原因となる場合があります。外部ルート集約は、ルートを OSPF に再配布している ASBR で設定してください。

集約アドレスの設定時に Cisco NX-OS は、ルーティング ブラック ホールおよびルート ループを防ぐために、集約アドレスの廃棄ルートを自動的に設定します。

ハイ アベイラビリティとグレースフル リスタート

Cisco NX-OS はハイ アベイラビリティをサポートしています。Cisco NX-OS システムでコールド リブートが発生した場合は、ネットワークがシステムへのトラフィックの転送を停止し、システムをネットワーク トポロジから削除します。このシナリオでは、OSPFv2 でステートレス再起動が発生し、ローカル システム上のネイバーの隣接関係がすべて削除されます。Cisco NX-OS はスタートアップ コンフィギュレーションを適用し、OSPFv2 はネイバーを再探索して、隣接関係を再確立します。

Cisco NX-OS を実行する 2 つのスーパーバイザを持つプラットフォームでは、ステートフル スーパーバイザ スイッチオーバーが発生する場合があります。このスイッチオーバーが発生する前に、OSPFv2 は、OSPFv2 がしばらく使用不可であることを宣言して、グレースフル リスタートを開始します。スイッチオーバー中には、ネットワークでトラフィックの転送が続行され、システムはネットワーク トポロジ内に存在し続けます。

スイッチオーバー後に、Cisco NX-OS は実行コンフィギュレーションを適用し、OSPFv2 は、自身が再び稼働していることをネイバーに通知します。ネイバーは隣接関係の再確立を支援します。

OSPFv2 は、処理に問題が発生した場合は、自動的に再起動します。再起動後に OSPFv2 はグレースフル リスタートを開始するため、プラットフォームがネットワーク トポロジから削除されることはありません。手動で OSPF を再起動した場合は、ステートフル スイッチオーバーと同様のグレースフル リスタートが実行されます。いずれの場合にも実行コンフィギュレーションが適用されます。

グレースフル リスタート、つまり、Nonstop Forwarding（NSF）では、処理の再起動中も OSPFv2 がデータ転送パス上に存在し続けます。OSPFv2 の再起動が必要になると、OSPFv2 は最初に、猶予 LSA と呼ばれるリンクローカル 不透明（タイプ 9）LSA（「不透明 LSA」[p.5-7] を参照）を送信します。この再起動中の OSPFv2 プラットフォームは NSF 対応と呼ばれます。

猶予 LSA には猶予期間が含まれます。猶予期間とは、ネイバー OSPFv2 インターフェイスは再起動中の OSPFv2 インターフェイスからの LSA を待つよう指定された時間です。（通常、OSPFv2 は隣接関係を解消し、ダウンした、または再起動中の OSPFv2 インターフェイスからのすべての LSA を廃棄します。）関与するネイバーは NSF ヘルパーと呼ばれ、再起動中の OSPFv2 インターフェイスが発信するすべての LSA を、このインターフェイスが隣接したままであるかのように維持します。

再起動中の OSPFv2 インターフェイスが稼働を再開すると、ネイバーを再探索して隣接関係を確立し、LSA 更新情報の送信を再開します。この時点で、NSF ヘルパーは、グレースフル リスタートが完了したと認識します。

（注） 再起動中の OSPFv2 インターフェイスが猶予期間内に稼働を再開しない場合、またはネットワークでトポロジが変更された場合は、各 OSPFv2 ネイバーは再起動中の OSPFv2 との隣接関係を解消し、通常の OSPFv2 の再起動として対応します。

（注） グレースフル リスタートでは、OSPFv2 の In-Service Software Upgrade（ISSU; インサービス ソフトウェア アップグレード）のサポートをイネーブルにする必要があります。グレースフル リスタートをディセーブルにした場合は、Cisco NX-OS により、この設定では ISSU をサポートできないという警告が発行されます。

OSPFv2 スタブ ルータ アドバタイズメント

OSPFv2 スタブ ルータ アドバタイズメント機能を使用して、OSPFv2 インターフェイスをスタブ ルータとして機能するよう設定することはできません。この機能は、ネットワークに新規ルータを機能制限付きで導入する場合や、過負荷になっているルータの負荷を制限する場合など、このルータ経由の OSPFv2 トラフィックを制限するときに使用します。また、この機能は、さまざまな管理上またはトラフィック エンジニアリング上の理由により使用される場合もあります。

OSPFv2 スタブ ルータ アドバタイズメントは、OSPFv2 ルータをネットワーク トポロジから削除しませんが、他の OSPFv2 ルータがこのルータを使用して、ネットワークの他の部分にトラフィックをルーティングできないようにします。このルータが宛先のトラフィック、またはこのルータに直接接続されたトラフィックのみが送信されます。

OSPFv2 スタブ ルータ アドバタイズメントは、すべてのスタブ リンク（ローカル ルータに直接接続された）を、ローカル OSPFv2 インターフェイスのコストとしてマークします。すべてのリモート リンクは、最大のコスト（0xFFFF）としてマークされます。

複数の OSPFv2 インスタンス

Cisco NX-OS は、同じノード上で動作する、OSPFv2 プロトコルの複数インスタンスをサポートしています。同じインターフェイス上で複数のインスタンスを設定することはできません。デフォルトでは、すべてのインスタンスが同じシステム ルータ ID を使用します。複数のインスタンスが同じ OSPFv2 AS にある場合は、各インスタンスのルータ ID を手動で設定する必要があります。

SPF 最適化

Cisco NX-OS は、次の方法で SPF アルゴリズムを最適化します。

- ネットワーク（タイプ 2）LSA、ネットワーク集約（タイプ 3）LSA、および AS 外部（タイプ 5）LSA 用部分 SPF ― これらの LSA のいずれかが変更されると、Cisco NX-OS は、全体的な SPF 計算ではなく、高速部分計算を実行します。
- SPF タイマー ― さまざまなタイマーを設定して、SPF 計算を制御できます。これらのタイマーには、後続の SPF 計算の幾何バックオフが含まれます。幾何バックオフにより、複数の SPF 計算による CPU 負荷が制限されます。

仮想化のサポート

OSPFv2 は、Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）をサポートしています。VRF は Virtual Device Contexts（VDC; 仮想化デバイス コンテキスト）内にあります。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。VDC 内に含まれるのは、最大 4 つの OSPFv2 インスタンスです。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』の第 13 章「レイヤ 3 仮想化の設定」を参照してください。

OSPFv2 のライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	OSPFv2 には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細説明およびライセンスの入手と適用の方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

OSPFv2 の前提条件

OSPFv2 には、次の前提条件があります。

- OSPF を設定するための、ルーティングの基礎に関する詳しい知識がある。
- スイッチにログオンしている。
- リモート OSPFv2 ネイバーと通信可能な 1 つ以上の IPv4 用インターフェイスが設定されている。
- Enterprise Services ライセンスがインストールされている。
- OSPFv2 ネットワーク戦略と、ネットワークのプランニングが完成している。たとえば、複数のエリアが必要かどうかを決定する必要があります。
- OSPF 機能がイネーブルにされている（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。
- Advanced Services ライセンスがインストールされ、VDC を設定している場合は、目的の VDC が入力されている（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

OSPFv2 設定時の注意事項および制約事項は次のとおりです。

- VDC 内に含まれるのは、最大 4 つの OSPFv2 インスタンスです。

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

基本的 OSPFv2 の設定

OSPFv2 は、OSPFv2 ネットワークを設計したあとに設定します。

ここでは、次の内容について説明します。

- OSPFv2 機能のイネーブル化（p.5-15）
- OSPFv2 インスタンスの作成（p.5-16）
- OSPFv2 インスタンス上のオプション パラメータの設定（p.5-17）
- OSPFv2 インスタンス上のオプション パラメータの設定（p.5-17）
- OSPFv2 でのネットワークの設定（p.5-18）
- エリアの認証の設定（p.5-20）

OSPFv2 機能のイネーブル化

OSPFv2 を設定するには、その前に OSPFv2 機能をイネーブルにする必要があります。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. ***feature ospf***
3. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature ospf** **例：** `switch(config)# feature ospf`	OSPFv2 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

OSPFv2 機能をディセーブルにし、関連付けられた設定をすべて削除するには、***no feature ospf*** コマンドを使用します。

コマンド	目的
no feature ospf **例：** `switch(config)# no feature ospf`	OSPFv2 機能をディセーブルにして、関連付けられた設定をすべて削除します。

OSPFv2 インスタンスの作成

OSPFv2 設定の最初のステップは OSPFv2 インスタンスの作成です。作成した OSPFv2 インスタンスには、一意のインスタンス タグを割り当てます。インスタンス タグは任意の文字列です。

OSPFv2 インスタンス パラメータの詳細については、「高度な OSPFv2 の設定」（p.5-24）を参照してください。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

show ip ospf *instance-tag* コマンドを使用して、インスタンス タグが使用されていないことを確認します。

OSPFv2 がルータ ID（設定済みのループバック アドレスなど）を入手可能であるか、またはルータ ID オプションを設定する必要があります。

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）。

コマンドの一覧

1. **config t**
2. ***router ospf*** *instance-tag*
3. ***router-id*** *ip-address*
4. ***show ip ospf*** *instance-tag*
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**router-id** *ip-address* **例：** `switch(config-router)# router-id 209.0.2.1`	（任意）OSPFv2 ルータ ID を設定します。この IP アドレスにより、この OSPFv2 インスタンスが識別されます。このアドレスは、システムの設定済みインターフェイス上に存在する必要があります。
ステップ 4	***show ip ospf*** *instance-tag* **例：** `switch(config-router)# show ip ospf 201`	（任意）OSPF 情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

OSPFv2 インスタンスと、関連付けられた設定をすべて削除するには、***no router ospf*** コマンドを使用します。

コマンド	目的
no router ospf *instance-tag* **例：** `switch(config)# no router ospf 201`	OSPF インスタンスと、関連付けられた設定を削除します。

(注)　このコマンドは、インターフェイス モードでは OSPF 設定を削除しません。インターフェイス モードで設定された OSPFv2 コマンドはいずれも、手動で削除する必要があります。

OSPFv2 インスタンス上のオプション パラメータの設定

OSPF のオプション パラメータを設定できます。

OSPFv2 インスタンス パラメータの詳細については、「高度な OSPFv2 の設定」（p.5-24）を参照してください。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

OSPFv2 がルータ ID（設定済みのループバック アドレスなど）を入手可能であるか、またはルータ ID オプションを設定する必要があります。

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）。

詳細な手順

ルータ コンフィギュレーション モードで、次の OSPFv2 用オプション パラメータを設定できます。

コマンド	目的
distance *number* **例：** `switch(config-router)# distance 25`	この OSPFv2 インスタンスの管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 110 です。

log-adjacency-changes [**detail**]
 	
例：
`switch(config-router)#`
`log-adjacency-changes`
	ネイバーの状態が変化するたびに、システム メッセージを生成します。

maximum-paths *path-number* **例：** `switch(config-router)# maximum-paths 4`	ルーティング テーブル内の宛先への同じ OSPFv2 パスの最大数を設定します。このコマンドはロード バランシングに使用されます。有効値の範囲は 1 ～ 16 であり、デフォルトは 8 です。

次に、OSPFv2 インスタンスを作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# copy running-config startup-config
```

OSPFv2 でのネットワークの設定

ルータがこのネットワークへの接続に使用するインターフェイスを介して、OSPFv2 へのネットワークを関連付けることで、このネットワークを設定できます（「ネイバー」[p.5-3] を参照）。すべてのネットワークをデフォルト バックボーン エリア（エリア 0）に追加したり、任意の 10 進数または IP アドレスを使用して新規エリアを作成したりできます。

（注） すべてのエリアは、バックボーン エリアに直接、または仮想リンク経由で接続する必要があります。

（注） インターフェイスに有効な IP アドレスを設定するまでは、OSPF はインターフェイス上でイネーブルにされません。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] 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）。

コマンドの一覧

1. **config t**
2. **interface** *interface-type slot/port*
3. ***ip address*** *ip-prefix/length*
4. ***ip router ospf*** *instance-tag* ***area*** *area-id* [**secondaries none**]
5. ***show ip ospf*** *instance-tag* **interface** *interface-type slot/port*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip address** *ip-prefix/length* **例：** `switch(config-if)# ip address 209.0.2.1/16`	このインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。

	コマンド	目的
ステップ 4	**ip router ospf** *instance-tag* **area** *area-id* [**secondaries none**] **例：** `switch(config-if)# ip router ospf 201 area 0`	OSPFv2 インスタンスおよびエリアにインターフェイスを追加します。
ステップ 5	***show ip ospf*** *instance-tag* **interface** *interface-type slot/port* **例：** `switch(config-if)# show ip ospf 201 interface ethernet 1/2`	（任意）OSPF 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

インターフェイス コンフィギュレーション モードで、省略可能な次の OSPFv2 パラメータを設定できます。

コマンド	目的
ip ospf cost *number* **例：** `switch(config-if)# ip ospf cost 25`	このインターフェイスの OSPFv2 コスト メトリックを設定します。デフォルトでは、参照帯域幅とインターフェイス帯域幅に基づいて、コスト メトリックが計算されます。有効値の範囲は 1 ～ 65535 です。
ip ospf dead-interval *seconds* **例：** `switch(config-if)# ip ospf dead-interval 50`	OSPFv2 デッド間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 です。デフォルトでは、hello 間隔の秒数の 4 倍です。
ip ospf hello-interval *seconds* **例：** `switch(config-if)# ip ospf hello-interval 25`	OSPFv2 hello 間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 秒です。
ip ospf mtu-ignore **例：** `switch(config-if)# ip ospf mtu-ignore`	OSPFv2 で、ネイバーとのあらゆる IP MTU 不一致が無視されるよう設定します。デフォルトでは、ネイバー MTU がローカル インターフェイス MTU が不一致の場合には、隣接関係が確立されません。
ip ospf passive-interface **例：** `switch(config-if)# ip ospf passive-interface`	インターフェイス上でルーティングが更新されないようにします。
ip ospf priority *number* **例：** `switch(config-if)# ip ospf priority 25`	エリアの DR の決定に使用される OSPFv2 優先度を設定します。有効値の範囲は 0 ～ 255 であり、デフォルトは 1 です。「代表ルータ」（p.5-4）を参照してください。
ip ospf shutdown **例：** `switch(config-if)# ip ospf shutdown`	このインターフェイス上の OSPFv2 インスタンスをシャットダウンします。

次に、OSPFv2 インスタンス 201 にネットワーク エリア 10 を追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# ip router ospf 201 area 10
switch(config-if)# copy running-config startup-config
```

インターフェイス設定を確認するには、***show ip ospf interface*** コマンドを使用します。このインターフェイスのネイバーを確認するには、***show ip ospf neighbor*** コマンドを使用します。

エリアの認証の設定

エリア内のすべてのネットワーク、またはエリア内の個々のインターフェイスの認証を設定できます。インターフェイス認証設定を使用すると、エリア認証は無効になります。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

インターフェイス上のすべてのネイバーが、共有認証キーを含め、同じ認証設定を共有することを確認します。

この認証設定のキーチェーンを作成します。詳細については『*Cisco NX-OS Security Configuration Guide*』を参照してください。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***area*** *area-id* ***authentication*** [***message-digest***]
4. ***interface*** *interface-type slot/port*
5. ***ip ospf authentication-key*** [***0*** | ***3***] *key*

   または

   ip ospf message-digest-key *key-id* ***md5*** [***0*** | ***3***] *key*
6. ***show ip ospf*** *instance-tag* **interface** *interface-type slot/port*
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **authentication** [***message-digest***] **例：** `switch(config-router)# area 10 authentication`	エリアの認証モードを設定します。

ステップ 4	***interface*** *interface-type slot/port*
	
	例：
	`switch(config-router)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 5	**ip ospf authentication-key** [***0*** | ***3***] *key* **例：** `switch(config-if)# ip ospf authentication-key 0 mypass`	(任意) このインターフェイスに簡易パスワード認証を設定します。認証が、キーチェーンにもメッセージ ダイジェストにも設定されていない場合は、このコマンドを使用します。0 の場合は、パスワードをクリア テキストで設定します。3 の場合は、パスワードを 3DES 暗号化として設定します。
	ip ospf message-digest-key *key-id* **md5** [***0*** | ***3***] *key* **例：** `switch(config-if)# ip ospf message-digest-key 21 md5 0 mypass`	(任意) このインターフェイスにメッセージ ダイジェスト認証を設定します。認証がメッセージ ダイジェストに設定されている場合は、このコマンドを使用します。key-id の範囲は 1 ～ 255 です。MD5 オプションが 0 の場合はパスワードがクリア テキストで設定され、3 の場合はパス キーが 3DES 暗号化として設定されます。
ステップ 6	***show ip ospf*** *instance-tag* **interface** *interface-type slot/port* **例：** `switch(config-if)# show ip ospf 201 interface ethernet 1/2`	(任意) OSPF 情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	(任意) この設定変更を保存します。

インターフェイスの認証の設定

エリア内の個々のインターフェイスに認証を設定できます。インターフェイス認証設定を使用すると、エリア認証は無効になります。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

インターフェイス上のすべてのネイバーが、共有認証キーを含め、同じ認証設定を共有することを確認します。

この認証設定のキーチェーンを作成します。詳細については『*Cisco NX-OS Security Configuration Guide*』を参照してください。

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）。

コマンドの一覧

1. **config t**
2. ***interface*** *interface-type slot/port*
3. ***ip ospf authentication*** [***key-chain*** *key-id* | ***message-digest*** | ***null***]
4. ***ip ospf authentication-key*** [***0*** | ***3***] *key*

   または

   ip ospf message-digest-key *key-id* ***md5*** [***0*** | ***3***] *key*
5. ***show ip ospf*** *instance-tag* **interface** *interface-type slot/port*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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*** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip ospf authentication** [**key-chain** *key-name* | ***message-digest*** | ***null***] **例：** `switch(config-if)# ip ospf authentication message-digest`	OSPFv2 のインターフェイス認証モードを設定します。これにより、エリアに基づくこのインターフェイスの認証が無効となります。すべてのネイバーが、この認証タイプを共有する必要があります。 キーチェーンの詳細については、『*Cisco NX-OS Security Configuration Guide*』Release 4.0 を参照してください。

	コマンド	目的
ステップ 4	**ip ospf authentication-key** [*0* | *3* | *7*] *key* **例：** switch(config-if)# ip ospf authentication-key 0 mypass	（任意）このインターフェイスに簡易パスワード認証を設定します。認証が、キーチェーンにもメッセージ ダイジェストにも設定されていない場合は、このコマンドを使用します。 オプションは次のとおりです。 • 0 ― パスワードをクリア テキストで設定します。 • 3 ― パス キーを 3DES 暗号化として設定します。 7 ― パス キーを Cisco タイプ 7 暗号化として設定します。
ステップ 5	**ip ospf message-digest-key** *key-id* **md5** [*0* | *3* | *7*] *key* **例：** switch(config-if)# ip ospf message-digest-key 21 md5 0 mypass	（任意）このインターフェイスにメッセージ ダイジェスト認証を設定します。認証がメッセージ ダイジェストに設定されている場合は、このコマンドを使用します。key-id の範囲は 1 ～ 255 です。MD5 オプションは次のとおりです。 • 0 ― パスワードをクリア テキストで設定します。 • 3 ― パス キーを 3DES 暗号化として設定します。 • 7 ― パス キーを Cisco タイプ 7 暗号化として設定します。
ステップ 6	***show ip ospf*** *instance-tag* **interface** *interface-type slot/port* **例：** switch(config-if)# show router ospf 201 interface ethernet 1/2	（任意）OSPF 情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** switch(config)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、インターフェイスに暗号化されていない簡単なパスワードを設定し、イーサネット インターフェイス 1/2 のパスワードを設定する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# ip router ospf 201 area 10
switch(config-if)# ip ospf authentication
switch(config-if)# ip ospf authentication-key 0 mypass
switch(config-if)# copy running-config startup-config
```

高度な OSPFv2 の設定

OSPFv2 は、OSPFv2 ネットワークを設計したあとに設定します。

ここでは、次の内容について説明します。

- 境界ルータのフィルタ リストの設定（p.5-24）
- スタブ エリアの設定（p.5-25）
- Totally Stubby エリアの設定（p.5-26）
- NSSA の設定（p.5-27）
- 仮想リンクの設定（p.5-29）
- 再配布の設定（p.5-31）
- ルート集約の設定（p.5-32）
- スタブ ルート アドバタイズメントの設定（p.5-34）
- デフォルト タイマーの変更（p.5-35）
- グレースフル リスタートの設定（p.5-37）
- OSPFv2 インスタンスの再起動（p.5-39）
- グレースフル リスタートの設定（p.5-37）

境界ルータのフィルタ リストの設定

OSPFv2 ドメインを、関連性のある各ネットワークを含む一連のエリアに分離できます。すべてのエリアは、ABR 経由でバックボーン エリアに接続している必要があります。OSPFv2 ドメインは、*自律システム境界ルータ*（ASBR; 自律システム境界ルータ）を介して、外部ドメインにも接続可能です。「エリア」（p.5-5）を参照してください。

ABR には、省略可能な次の設定パラメータがあります。

- Area range ― エリア間のルート集約を設定します。「ルート集約の設定」（p.5-32）を参照してください。
- Filter list ― ABR 上で、外部エリアから受信したネットワーク集約（タイプ 3）LSA をフィルタリングします。

ASBR もフィルタ リストをサポートしています。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

フィルタ リストが、着信または発信ネットワーク集約（タイプ 3）LSA の IP プレフィクスのフィルタリングに使用するルートマップを作成します。第 14 章「Route Policy Manager の設定」を参照してください。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***area*** *area-id* ***filter-list route-map*** *map-name* ***{in | out}***
4. **show ip ospf policy statistics**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **filter-list route-map** *map-name* {***in*** | ***out***} **例：** `switch(config-router)# area 10 filter-list route-map FilterLSAs in`	ABR 上で着信または発信ネットワーク集約（タイプ 3）LSA をフィルタリングします。
ステップ 4	***show ip ospf policy statistics area*** *id* ***filter-list*** {***in*** | ***out***} **例：** `switch(config-if)# show ip ospf policy statistics area 10 filter-list in`	（任意）OSPF ポリシー情報を表示します。
ステップ 5	**copy running-config startup-config** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、エリア 10 でフィルタ リストを設定する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 filter-list route-map FilterLSAs in
switch(config-router)# copy running-config startup-config
```

スタブ エリアの設定

OSPFv2 ドメインの、外部トラフィックが不要な部分にスタブ エリアを設定できます。スタブ エリアは AS 外部（タイプ 5）LSA をブロックし、不要な、選択したネットワークへの往復のルーティングを制限します。「スタブ エリア」（p.5-9）を参照してください。また、すべての集約ルートがスタブ エリアを経由しないようブロックすることもできます。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

設定されるスタブ エリア内に、仮想リンクと ASBR のいずれも含まれないことを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*

3. ***area*** *area-id* ***stub***
4. ***area*** *area-id* ***default-cost*** *cost*
5. ***show ip ospf*** *instance-tag*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **stub** **例：** `switch(config-router)# area 10 stub`	このエリアをスタブ エリアとして作成します。
ステップ 4	**area** *area-id* **default-cost** *cost* **例：** `switch(config-router)# area 10 default-cost 25`	（任意）このスタブ エリアに送信されるデフォルト集約ルートのコスト メトリックを設定します。有効値の範囲は 0 ～ 16777215 であり、デフォルトは 1 です。
ステップ 5	***show ip ospf*** *instance-tag* **例：** `switch(config-if)# show ip ospf 201`	（任意）OSPF 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、スタブ エリアを作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 stub
switch(config-router)# copy running-config startup-config
```

Totally Stubby エリアの設定

Totally Stubby エリアを作成して、すべての集約ルート更新がスタブ エリアを経由しないようにすることができます。

Totally Stubby エリアを作成するには、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。

コマンド	目的
area *area-id* **stub no-summary** **例：** `switch(config-router)# area 20 stub no-summary`	このエリアを Totally Stubby エリアとして作成します。

NSSA の設定

OSPFv2 ドメインの、ある程度の外部トラフィックが必要な部分に NSSA を設定できます。「Not-So-Stubby エリア」（p.5-9）を参照してください。また、この外部トラフィックを AS 外部（タイプ 5）LSA に変換して、このルーティング情報で OSPFv2 ドメインをフラッディングすることもできます。NSSA は、省略可能な次のパラメータで設定できます。

- No redistribution ― 再配布されたルートが NSSA をバイパスして、OSPFv2 AS 内の他のエリアに再配布されます。このオプションは、NSSA ASBR が ABR も兼ねているときに使用します。
- Default information originate ― 外部 AS へのデフォルト ルートの NSSA 外部（タイプ 7）LSA を生成します。このオプションは、ASBR のルーティング テーブルにデフォルト ルートが含まれる場合に NSSA ASBR 上で使用します。このオプションは、ASBR のルーティング テーブルにデフォルト ルートが含まれるかどうかに関係なく、NSSA ASBR 上で使用できます。
- Route map ― 目的のルートのみが NSSA および他のエリア全体でフラッディングされるよう、外部ルートをフィルタリングします。
- Translate ― NSSA 外のエリア向けに、NSSA 外部 LSA を AS 外部 LSA に変換します。再配布されたルートを OSPFv2 AS 全体でフラッディングするには、このコマンドを NSSA ABR 上で使用します。また、これらの AS 外部 LSA の転送アドレスを無効にすることもできます。このオプションを選択した場合は、転送アドレスが 0.0.0.0 に設定されます。
- No summary ― すべての集約ルートが NSSA でフラッディングされないようにします。このオプションは NSSA ABR 上で使用します。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

設定する NSSA 上に仮想リンクがないことと、この NSSA がバックボーン エリアでないことを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. **area** *area-id* **nssa** [***no-redistribution***] [***default-information-originate*** [***route-map*** *map-name*]] [***no-summary***] [**translate type7** {**always** | **never**} [**suppress-fa**]]
4. ***area*** *area-id* ***default-cost*** *cost*
5. ***show ip ospf*** *instance-tag*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **nssa** [***no-redistribution***] [***default-information-originate***] [***route-map*** *map-name*]] [***no-summary***] [***translate type7*** {***always*** | ***never***} [***suppress-fa***]] **例：** `switch(config-router)# area 10 nssa`	このエリアを NSSA として作成します。
ステップ 4	**area** *area-id* **default-cost** *cost* **例：** `switch(config-router)# area 10 default-cost 25`	（任意）この NSSA に送信されるデフォルト集約ルートのコスト メトリックを設定します。
ステップ 5	***show ip ospf*** *instance-tag* **例：** `switch(config-if)# show ip ospf 201`	（任意）OSPF 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、すべての集約ルート更新をブロックする NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 nssa no-summary
switch(config-router)# copy running-config startup-config
```

次に、デフォルト ルートを生成する NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 nssa default-info-originate
switch(config-router)# copy running-config startup-config
```

次に、外部ルートをフィルタリングし、すべての集約ルート更新をブロックする NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 nssa route-map ExternalFilter no-summary
switch(config-router)# copy running-config startup-config
```

次に、つねに NSSA 外部（タイプ 7）LSA を AS 外部（タイプ 5）LSA に変換する NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 nssa translate type 7 always
switch(config-router)# copy running-config startup-config
```

仮想リンクの設定

仮想リンクは、隔離されたエリアを、中継エリア経由でバックボーン エリアに接続します。「仮想リンク」(p.5-10)を参照してください。仮想リンクには、省略可能な次のパラメータを設定できます。

- Authentication ― 簡単なパスワード認証または MD5 メッセージ ダイジェスト認証、および関連付けられたキーを設定します。
- Dead interval ― ローカル ルータがデッドであることを宣言し、隣接関係を解消する前に、ネイバーが hello パケットを待つ時間を設定します。
- Hello interval ― 連続する hello パケット間の時間間隔を設定します。
- Retransmit interval ― 連続する LSA 間の推定時間間隔を設定します。
- Transmit delay ― LSA をネイバーに送信する推定時間間隔を設定します。

（注） リンクがアクティブになる前に、関与する両方のルータで仮想リンクを設定する必要があります。

スタブ エリアには仮想リンクを追加できません。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] 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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***area*** *area-id* ***virtual-link*** *router-id*
4. ***show ip ospf virtual-link*** [***brief***]
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **virtual-link** *router-id*
	
	例：
	`switch(config-router)# area 10 virtual-link 10.1.2.3`
	`switch(config-router-vlink)#`
		リモート ルータへの仮想リンクの端を作成します。仮想リンクをリモート ルータ上に作成して、リンクを完成させる必要があります。

ステップ 4	***show ip ospf virtual-link*** [***brief***] **例：** `switch(config-router-vlink)# show ip ospf virtual-link`	（任意）OSPF 仮想リンク情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-router-vlink)# copy running-config startup-config`	（任意）この設定変更を保存します。

仮想リンク コンフィギュレーション モードで、省略可能な次のコマンドを設定できます。

コマンドまたはアクション	目的
authentication [**key-chain** *key-id* | ***message-digest*** | ***null***] **例：** `switch(config-router-vlink)# authentication message-digest`	（任意）これにより、エリアに基づくこの仮想リンクの認証が無効となります。
authentication-key [***0*** | ***3***] *key* **例：** `switch(config-router-vlink)# authentication-key 0 mypass`	（任意）この仮想リンクに簡易パスワードを設定します。認証が、キーチェーンにもメッセージ ダイジェストにも設定されていない場合は、このコマンドを使用します。0 の場合は、パスワードをクリア テキストで設定します。3 の場合は、パスワードを 3DES 暗号化として設定します。
dead-interval *seconds* **例：** `switch(config-router-vlink)# dead-interval 50`	（任意）OSPFv2 デッド間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 です。デフォルトでは、hello 間隔の秒数の 4 倍です。
hello-interval *seconds* **例：** `switch(config-router-vlink)# hello-interval 25`	（任意）OSPFv2 hello 間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 秒です。

コマンドまたはアクション	目的

message-digest-key *key-id* **md5** [***0*** | ***3***] *key*
 	
例：
`switch(config-router-vlink)#`
`message-digest-key 21 md5 0 mypass`
	（任意）この仮想リンクにメッセージ ダイジェスト認証を設定します。認証がメッセージ ダイジェストに設定されている場合は、このコマンドを使用します。0 の場合は、パスワードをクリア テキストで設定します。3 の場合は、パス キーを 3DES 暗号化として設定します。

retransmit-interval *seconds* **例：** `switch(config-router-vlink)# retransmit-interval 50`	（任意）OSPFv2 再送間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 5 です。
transmit-delay *seconds* **例：** `switch(config-router-vlink)# transmit-delay 2`	（任意）OSPFv2 送信遅延を秒単位で設定します。有効値の範囲は 1 ～ 450 であり、デフォルトは 1 です。

次に、2 つの ABR 間に簡単な仮想リンクを作成する例を示します。

ABR 1（ルータ ID 27.0.0.55）の設定は、次のとおりです。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 virtual-link 10.1.2.3
switch(config-router)# copy running-config startup-config
```

ABR 2（ルータ ID 10.1.2.3）の設定は、次のとおりです。

```
switch# config t
switch(config)# router ospf 101
switch(config-router)# area 10 virtual-link 27.0.0.55
switch(config-router)# copy running-config startup-config
```

再配布の設定

他のルーティング プロトコルから学習したルートを、ASBR 経由で OSPFv2 AS に再配布できます。

OSPF でのルート再配布には、省略可能な次のパラメータを設定できます。

- Default information originate ― 外部 AS へのデフォルト ルートの AS 外部（タイプ 5）LSA を生成します。
- Default metric ― すべての再配布ルートに同じコスト メトリックを設定します。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

再配布で使用する、必要なルートマップを作成します。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***redistribute*** {***bgp*** *id* | ***direct*** | ***eigrp*** *id* | ***isis*** *id* | ***ospf*** *id* | ***rip*** *id* | ***static***} ***route-map*** *map-name*
4. ***default-information originate*** [***always***] [***route-map*** *map-name*]

5. ***default-metric*** *cost*
6. ***exit***
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 ospf** *instance-tag* **例：** switch(config)# router ospf 201 switch(config-router)#	新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。
ステップ 3	***redistribute*** {***bgp*** *id* | ***direct*** | ***eigrp*** *id* | ***isis*** *id* | ***ospf*** *id* | ***rip*** *id* | ***static***} ***route-map*** *map-name* **例：** switch(config-router)# redistribute bgp route-map FilterExternalBGP	設定したルートマップ経由で、選択したプロトコルを OSPF に再配布します。
ステップ 4	***default-information originate*** [***always***] [***route-map*** map-name] **例：** switch(config-router)# default-information-originate route-map DefaultRouteFilter	この OSPF ドメインにデフォルト ルートを作成します。always を使用すると、つねにデフォルトを作成します。route-map を使用すると、条件付きで（RIB にデフォルト ルートがある場合）、ルートマップが true を返す場合にデフォルト ルートを生成します。（再配布の概念に追加）
ステップ 5	**default-metric** *cost* **例：** switch(config-router)# default-metric 25	再配布されたルートのコスト メトリックを設定します。
ステップ 6	***copy running-config startup-config*** **例：** switch(config-router)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、Border Gateway Protocol（BGP）を OSPF に再配布する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# redistribute bgp route-map FilterExternalBGP
switch(config-router)# copy running-config startup-config
```

ルート集約の設定

集約されたアドレス範囲を設定して、エリア間ルートのルート集約を設定できます。また、ASBR 上のこれらのルートの集約アドレスを設定して、外部の再配布されたルートのルート集約を設定することもできます。「ルート集約」（p.5-11）を参照してください。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」 [p.5-15] 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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***area*** *area-id* ***range*** *ip-prefix/length* [***no-advertise***]

   または
4. ***summary-address*** *ip-prefix/length* [***no-advertise*** | ***tag*** *tag-id*]
5. ***show ip ospf summary-address***
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	***area*** *area-id* ***range*** *ip-prefix/length* [***no-advertise***] **例：** `switch(config-router)# area 10 range 10.3.0.0/16`	一定の範囲のアドレスの集約アドレスを ABR 上に作成します。この集約アドレスをネットワーク集約（タイプ 3）LSA にアドバタイズしないようにすることもできます。
ステップ 4	***summary-address*** *ip-prefix/length* [**no-advertise** | ***tag*** tag] **例：** `switch(config-router)# summary-address 10.5.0.0/16 tag 2`	一定の範囲のアドレスの集約アドレスを ABR 上に作成します。ルートマップによる再配布で使用できるよう、この集約アドレスにタグを割り当てることもできます。
ステップ 5	***show ip ospf summary-address*** **例：** `switch(config-router)# show ip ospf summary-address`	（任意）OSPF 集約アドレスに関する情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ABR 上のエリア間の集約アドレスを作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# area 10 range 10.3.0.0/16
switch(config-router)# copy running-config startup-config
```

次に、ABR 上の集約アドレスを作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# summary-address 10.5.0.0/16
switch(config-router)# copy running-config startup-config
```

スタブ ルート アドバタイズメントの設定

短期間だけ、このルータ経由の OSPFv2 トラフィックを制限する場合は、スタブ ルート アドバタイズメントを使用します。「OSPFv2 スタブ ルータ アドバタイズメント」（p.5-12）を参照してください。

スタブ ルート アドバタイズメントは、省略可能な次のパラメータで設定できます。

- On startup ― 指定した宣言期間だけ、スタブ ルート アドバタイズメントを送信します。
- Wait for BGP ― BGP が収束するまで、スタブ ルート アドバタイズメントを送信します。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] 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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***max-metric router-lsa*** [***on-startup*** [*announce-time*] [***wait-for bgp*** *tag*]]
4. ***copy running-config startup-config***

（注） ルータの実行コンフィギュレーションがグレースフル　シャットダウンを行うよう設定されている場合は、その実行コンフィギュレーションを保存しないでください。保存すると、ルータが、リロード後に最大メトリックをアドバタイズし続けることになります。

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 ospf** *instance-tag* **例：** switch(config)# router ospf 201 switch(config-router)#	新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。
ステップ 3	***max-metric router-lsa*** [***on-startup*** [*announce-time*] [***wait-for bgp*** *tag*]] **例：** switch(config-router)# max-metric router-lsa	OSPFv2 スタブ ルート アドバタイズメントを設定します。on-start-up を使用すると、アドバタイズメントの最初の起動時またはシステムの起動時にアドバタイズします。bgp のインスタンスの起動を待ち、概念に追加します。
ステップ 4	***copy running-config startup-config*** **例：** switch(config-router)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、起動時にスタブ ルータ アドバタイズメント機能を、デフォルトの 600 秒間イネーブルにする例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# max-metric router-lsa on-startup
switch(config-router)# copy running-config startup-config
```

デフォルト タイマーの変更

OSPFv2 には、プロトコル メッセージの動作および SPF 計算を制御する数多くのタイマーが含まれます。OSPFv2 には、省略可能な次のタイマー パラメータが含まれます。

- LSA arrival time ― ネイバーから着信する LSA 間で許容される最小間隔を設定します。この時間より短時間で到着する LSA はドロップされます。
- Pacing LSAs ― LSA が集められてグループ化され、リフレッシュされて、チェックサムが計算される間隔、つまり期限切れとなる間隔を設定します。このタイマーは、LSA 更新が実行される頻度を制御し、LSA 更新メッセージで送信される LSA 更新の数を制御します（「フラッディングと LSA グループ ペーシング」[p.5-7] を参照）。
- Throttle LSAs ― LSA 生成のレート制限を設定します。このタイマーは、トポロジが変更されない場合に LSA が生成される頻度を制御します。
- Throttle SPF calculation ― SPF 計算の実行頻度を制御します。

インターフェイス レベルでは、次のタイマーも制御できます。

- Retransmit interval ― 連続する LSA 間の推定時間間隔を設定します。
- Transmit delay ― LSA をネイバーに送信する推定時間を設定します。

hello 間隔とデッド タイマーに関する情報の詳細については、「OSPFv2 でのネットワークの設定」（p.5-18）を参照してください。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」 [p.5-15] 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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***timers lsa-arrival*** *msec*
4. ***timers lsa-group-pacing*** *seconds*
5. ***timers throttle lsa*** {***network*** *hold-interval* | ***router*** *hold-interval*}
6. ***timers throttle spf*** *delay-time hold-time*
7. ***interface*** *type slot/port*
8. ***ip ospf hello-interval*** *seconds*
9. ***ip ospf dead-interval*** *seconds*
10. ***ip ospf retransmit-interval*** *seconds*
11. ***ip ospf transmit-delay*** *seconds*
12. ***show ip ospf***
13. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**timers lsa-arrival** *msec* **例：** `switch(config-router)# timers lsa-arrival 2000`	LSA 到着時間をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 1000 ミリ秒です。
ステップ 4	**timers lsa-group-pacing** *seconds* **例：** `switch(config-router)# timers lsa-group-pacing 2000`	LSA がグループ化される間隔を秒で設定します。有効値の範囲は 1 ～ 1800 であり、デフォルトは 240 秒です。
ステップ 5	**timers throttle lsa** {**network** *delay-interval* | **router** *hold-interval*} **例：** `switch(config-router)# timers throttle lsa network 3000`	LSA 生成のレート制限をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 5000 ミリ秒です。

	コマンド	目的
ステップ 6	**timers throttle spf** *delay-time hold-time max-wait* **例：** `switch(config-router)# timers throttle spf 3000 2000 4000`	SPF 最適パス スケジュール初期遅延時間と、各 SPF 最適パス計算間の最小保留時間（秒単位）を設定します。有効値の範囲は 1 ～ 600000 です。デフォルトは、遅延時間なし、および保留時間 5000 ミリ秒です。

ステップ 7	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 8	**ip ospf hello-interval** *seconds* **例：** `switch(config-if)# ip ospf retransmit-interval 30`	このインターフェイスの hello 間隔を設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 です。
ステップ 9	**ip ospf dead-interval** *seconds* **例：** `switch(config-if)# ip ospf retransmit-interval 30`	このインターフェイスのデッド間隔を設定します。有効値の範囲は 1 ～ 65535 です。
ステップ 10	**ip ospf retransmit-interval** *seconds* **例：** `switch(config-if)# ip ospf retransmit-interval 30`	このインターフェイスから送信される各 LSA 間の推定時間間隔を設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 5 です。

ステップ 11	**ip ospf transmit-delay** *seconds*
	例：
	`switch(config-if)# ip ospf transmit-delay 600`
	`switch(config-if)#`
		LSA をネイバーに送信する推定時間間隔を秒で設定します。有効値の範囲は 1 ～ 450 であり、デフォルトは 1 です。

ステップ 12	***show ip ospf*** **例：** `switch(config-if)# show ip ospf`	（任意）OSPF に関する情報を表示します。
ステップ 13	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、lsa-group-pacing オプションで LSA フラッディングを制御する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# timers lsa-group-pacing 300
switch(config-router)# copy running-config startup-config
```

グレースフル リスタートの設定

グレースフル リスタートは、デフォルトでイネーブルにされています。OSPFv2 インスタンスのグレースフル リスタートには、省略可能な次のパラメータを設定できます。

- Grace period ― グレースフル リスタートの開始後に、ネイバーが隣接関係を解消するまでに待つ時間を設定します。
- Helper mode disabled ― ローカル OSPFv2 インスタンスのヘルパー モードをディセーブルにします。OSPFv2 は、ネイバーのグレースフル リスタートには関与しません。

- Planned graceful restart only ― 予定された再起動の場合にのみグレースフル リスタートがサポートされるよう、OSPFv2 を設定します。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] を参照）。

すべてのネイバーで、一致した省略可能なパラメータ一式とともにグレースフル リスタートが設定されていることを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospf** *instance-tag*
3. ***graceful-restart***
4. ***graceful-restart grace-period*** *seconds*
5. ***graceful-restart helper-disable***
6. ***graceful-restart planned-only***
7. ***show ip ospf*** *instance-tag*
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	
	例：
	`switch(config)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**graceful-restart** **例：** `switch(config-router)# graceful-restart`	グレースフル リスタートをイネーブルにします。グレースフル リスタートは、デフォルトでイネーブルにされています。
ステップ 4	**graceful-restart grace-period** *seconds* **例：** `switch(config-router)# graceful-restart grace-period 120`	（任意）猶予期間を秒で設定します。有効値の範囲は 5 ～ 1800 であり、デフォルトは 60 秒です。
ステップ 5	**graceful-restart helper-disable** **例：** `switch(config-router)# graceful-restart helper-disable`	（任意）ヘルパー モードをディセーブルにします。デフォルトではイネーブルにされています。

	コマンド	目的
ステップ 6	**graceful-restart planned-only** **例：** `switch(config-router)# graceful-restart planned-only`	（任意）予定された再起動時にのみグレースフル リスタートを設定します。
ステップ 7	***show ip ospf*** *instance-tag* **例：** `switch(config-if)# show ip ospf 201`	（任意）OSPF 情報を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ディセーブルにされているグレースフル リスタートをイネーブルにし、猶予期間を 120 秒に設定する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# graceful-restart
switch(config-router)# graceful-restart grace-period 120
switch(config-router)# copy running-config startup-config
```

OSPFv2 インスタンスの再起動

OSPv2 インスタンスを再起動できます。再起動すると、インスタンスのすべてのネイバーが解消されます。

OSPFv2 インスタンスを再起動して、関連付けられたすべてのネイバーを削除するには、次のコマンドを使用します。

コマンド	目的
restart ospf instance-tag **例：** `switch(config)# restart ospf 201`	OSPFv2 インスタンスを再起動して、すべてのネイバーを削除します。

仮想化による OSPFv2 の設定

各 VDC で複数の OSPFv2 インスタンスを設定できます。各 VDC 内に複数の VRF を作成して、各 VRF で同じまたは複数の OSPFv2 インスタンスを使用することもできます。VRF には OSPFv2 インターフェイスを割り当てます。

（注） インターフェイスの VRF を設定したあとに、インターフェイスの他のすべてのパラメータを設定します。インターフェイスの VRF を設定すると、そのインターフェイスのすべての設定が削除されます。

操作の前に

VDC を作成します。

OSPF 機能がイネーブルにされていることを確認します（「OSPFv2 機能のイネーブル化」[p.5-15] 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）。

コマンドの一覧

1. **config t**
2. ***vrf* context** *vrf_name*
3. exit
4. **router ospf** *instance-tag*
5. ***vrf*** *vrf-name*
6. <optional parameters configured>
7. ***interface*** *interface-type slot/port*
8. ***vrf member*** *vrf-name*
9. ***ip-address*** *ip-prefix/length*
10. ***router ospf*** *instance-tag* ***area*** *area-id*
11. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 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	**router ospf** *instance-tag*
	
	例：
	`switch(config-vrf)# router ospf 201`
	`switch(config-router)#`
		新規 OSPFv2 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 4	**vrf** *vrf-name*
	
	例：
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		VRF 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	**maximum-paths** *paths* **例：** `switch(config-router-vrf)# maximum-paths 4`	（任意）この VRF のルーティング テーブル内の宛先への、同じ OSPFv2 パスの最大数を設定します。ロード バランシングに使用されます。

	コマンド	目的

ステップ 6	**interface** *interface-type slot/port*
	
	例：
	`switch(config-router-vrf)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 7	**vrf member** *vrf-name* **例：** `switch(config-if)# vrf member RemoteOfficeVRF`	このインターフェイスを VRF に追加します。
ステップ 8	**ip address** *ip-prefix/length* **例：** `switch(config-if)# ip address 209.0.2.1/16`	このインターフェイスの IP アドレスを設定します。この手順は、このインターフェイスを VRF に割り当てたあとに行う必要があります。
ステップ 9	**ip router ospf** *instance-tag* **area** *area-id* **例：** `switch(config-if)# ip router ospf 201 area 0`	設定した OSPFv2 インスタンスおよびエリアに、このインターフェイスを割り当てます。
ステップ 10	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、VRF を作成して、その VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context NewVRF
switch(config)# router ospf 201
switch(config)# interface ethernet 1/2
switch(config-if)# vrf member NewVRF
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# ip router ospf 201 area 0
switch(config)# copy running-config startup-config
```

OSPFv2 設定の確認

OSPFv2 設定を確認するには、次のコマンドを使用します。

コマンド	目的
show ip ospf	OSPFv2 設定を表示します。
show ip ospf border-routers [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 境界ルータ設定を表示します。
show ip ospf database [**vrf** {***vrf-name*** | **all** | **default** | **management**}]	OSPFv2 リンクステート データベースの要約を表示します。
show ip ospf interface *number* [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 インターフェイス設定を表示します。
show ip ospf lsa-content-changed-list *interface-type number* [**vrf** {*vrf-name* | **all** | **default** | **management**}]	変更された OSPFv2 LSA を表示します。
show ip ospf neighbors [*neighbor-id***] [detail]** [*interface-type number*] [**vrf** {*vrf-name* | **all** | **default** | **management**}] [**summary**]	OSPFv2 ネイバーの一覧を表示します。
show ip ospf request-list *neighbor-id* [*interface-type number*] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 リンクステート要求の一覧を表示します。
show ip ospf retransmission-list *neighbor-id* [*interface-type number*] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 リンクステート再送の一覧を表示します。
show ip ospf route [*ospf-route*] [**summary**] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	内部 OSPFv2 ルートを表示します。
show ip ospf summary-address [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 集約アドレスに関する情報を表示します。
show ip ospf virtual-links [**brief**] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 仮想リンクに関する情報を表示します。
show ip ospf vrf {*vrf-name* | **all** | **default** | **management**}	VRF ベースの OSPFv2 設定に関する情報を表示します。
show running-configuration ospf	現在実行中の OSPFv2 設定を表示します。

OSPFv2 統計情報の表示

OSPFv2 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 ip ospf memory	OSPFv2 メモリ使用状況の統計情報を表示します。
show ip ospf policy statistics area *area-id* **filter-list** {**in** | **out**} [**vrf** {*vrf-name* | **all** | **default** | **management**}]	エリアの OSPFv2 ルート ポリシー統計情報を表示します。
show ip ospf policy statistics redistribute {**bgp** *id*| **direct** | **eigrp** *id* | **isis** *id* | **ospf** *id* | **rip** *id* | **static**} **vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 ルート ポリシー統計情報を表示します。
show ip ospf statistics [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 イベント カウンタを表示します。
show ip ospf traffic [*interface-type number*] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv2 パケット カウンタを表示します。

OSPFv2 設定の例

次に、OSPFv2 を設定する例を示します。

```
feature ospf
router ospf 201
 router-id 290.0.2.1

interface ethernet 1/2
 ip router ospf 201 area 10
 ip ospf authentication
 ip ospf authentication-key 0 mypass
```

デフォルト設定

表 5-2 は、OSPFv2 パラメータのデフォルト設定の一覧です。

表 5-2　デフォルト OSPFv2 パラメータ

パラメータ	デフォルト
Hello interval	10 秒
Dead interval	40 秒
Graceful restart grace period	60 秒
Graceful restart notify period	15 秒
OSPFv2 feature	ディセーブル
Stub router advertisement announce time	600 秒
Reference bandwidth for link cost calculation	40 Gbps
LSA minimal arrival time	1000 ミリ秒
LSA group pacing	240 秒
SPF calculation initial delay time	0 ミリ秒
SPF calculation hold time	5000 ミリ秒
SPF calculation initial delay time	0 ミリ秒

その他の関連資料

OSPF の実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.5-44）
- MIB（p.5-44）

関連資料

関連項目	マニュアル名
OSPFv2 CLI コマンド	『*Cisco NX-OS Unicast Routing Command Line Reference*』
VDC	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』
IPv6 ネットワーク向け OSPFv3	第 6 章「OSPFv3 の設定」
ルートマップ	第 14 章「Route Policy Manager の設定」

MIB

MIB	MIB のリンク
• OSPF-MIB • OSPF-TRAP-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CHAPTER 6

OSPFv3 の設定

この章では、IPv6 ネットワーク向けの OSPFv3（Open Shortest Path First version 3）の設定方法を説明します。

ここでは、次の内容を説明します。

- OSPFv3 について（p.6-2）
- OSPFv3 のライセンス要件（p.6-13）
- OSPFv3 の前提条件（p.6-13）
- 設定時の注意事項および制約事項（p.6-13）
- 基本的 OSPFv3 の設定（p.6-14）
- 高度な OSPFv3 の設定（p.6-20）
- OSPFv3 設定の確認（p.6-37）
- OSPFv3 統計情報の表示（p.6-37）
- OSPFv3 設定の例（p.6-38）
- 関連資料（p.6-38）
- デフォルト設定（p.6-38）
- その他の関連資料（p.6-39）

OSPFv3 について

OSPFv3 は、IETF リンクステート プロトコル（「概要」[p.1-1] を参照）です。OSPFv3 ルータは、*hello パケット*と呼ばれる特別なメッセージを各 OSPF イネーブル インターフェイスに送信して、他の OSPFv3 隣接ルータを探索します。隣接ルータが発見されると、この 2 台のルータは hello パケットの情報を比較して、両者の設定に互換性のあるかどうかを判定します。これらの隣接ルータは*隣接関係*を確立しようとします。つまり、両者のリンクステート データベースを同期させて、確実に同じ OSPFv3 ルーティング情報を持つようにします。隣接ルータは、各リンクの稼働状態に関する情報、リンクのコスト、およびその他のあらゆる近隣情報を含む*リンクステート アドバタイズメント*（LSA; リンクステート アドバタイズメント）を共有します。これらのルータはその後、受信した LSA をすべての OSPF イネーブル インターフェイスにフラッディングします。これにより、すべての OSPFv3 ルータのリンクステート データベースが最終的に同じになります。すべての OSPFv3 ルータのリンクステート データベースが同じになると、ネットワークは*収束されます*（「収束」[p.1-6] を参照）。その後、各ルータは、ダイクストラの最短パス優先（SPF）アルゴリズムを使用して、自身のルーティング テーブルを構築します。

OSPFv3 ネットワークは、複数のエリアに分割できます。ルータは、ほとんどの LSA を 1 つのエリア内にのみ送信するため、OSPF イネーブル ルータの CPU とメモリの要件が緩やかになります。

OSPFv3 は IPv6 をサポートしています。IPv4 向けの OSPF の詳細については、第 5 章「OSPFv2 の設定」を参照してください。

ここでは、次の内容について説明します。

- OSPFv3 と OSPFv2 の比較（p.6-2）
- hello パケット（p.6-3）
- ネイバー（p.6-3）
- 隣接関係（p.6-4）
- 代表ルータ（p.6-4）
- エリア（p.6-5）
- Link-State Advertisement（p.6-6）
- OSPFv3 と IPv6 ユニキャスト RIB（p.6-8）
- 認証（p.6-8）
- アドレス ファミリのサポート（p.6-8）
- 高度な機能（p.6-9）

OSPFv3 と OSPFv2 の比較

OSPFv3 プロトコルの大半は OSPFv2 と同じです。OSPFv3 については RFC 2740 で説明されています。

OSPFv3 プロトコルと OSPFv2 プロトコルの重要な相違点は、次のとおりです。

- OSPFv2 を拡張した OSPFv3 では、IPv6 ルーティング プレフィクスとサイズの大きい IPv6 アドレスのサポートを提供しています。
- OSPFv3 の LSA は、アドレスとマスクではなく、プレフィクスとプレフィクス長として表現されます。
- ルータ ID とエリア ID は 32 ビット数で、IPv6 アドレスとは無関係です。
- OSPFv3 では、近隣探索およびその他の機能にリンクローカル IPv6 アドレスを使用します。
- OSPFv3 は、IPv6 を使用して認証を行います。
- OSPFv3 では、LSA タイプが再定義されています。

hello パケット

OSPFv3 ルータは、すべての OSPF イネーブル インターフェイスに hello パケットを定期的に送信します。ルータがこの hello パケットを送信する頻度は、インターフェイスごとに設定された *hello インターバル*により決定されます。OSPFv3 は、hello パケットを使用して、次のタスクを実行します。

- 近隣探索
- キープアライブ
- 双方向通信
- 代表ルータの選定（「代表ルータ」[p.6-4] を参照）

hello パケットには、リンクの OSPFv3 コスト割り当て、hello 間隔、送信元ルータのオプション機能など、送信元の OSPFv3 インターフェイスとルータに関する情報が含まれます。これらの hello パケットを受信する OSPFv3 インターフェイスは、設定に受信インターフェイスの設定との互換性があるかどうかを判定します。互換性のあるインターフェイスはネイバーとみなされ、ネイバー テーブルに追加されます（「ネイバー」[p.6-3] を参照）。

hello パケットには、送信元インターフェイスが通信したルータのルータ ID のリストも含まれます。受信インターフェイスが、このリストで自身の ID を見つけた場合は、2 つのインターフェイス間で双方向通信が確立されます。

OSPFv3 は、hello パケットをキープアライブ メッセージとして使用して、ネイバーが通信を継続中であるかどうかを判定します。ルータが、設定された*デッド インターバル*（通常は hello 間隔の倍数）で hello パケットを受信しない場合は、そのネイバーがローカル ネイバー テーブルから削除されます。

ネイバー

ネイバーとは見なされるためには、OSPFv3 インターフェイスがリモート インターフェイスとの互換性を持つよう設定されている必要があります。この 2 つの OSPFv3 インターフェイスで、次の基準が一致している必要があります。

- hello 間隔
- デッド間隔
- エリア ID（「エリア」[p.6-5] を参照）
- 認証
- オプション機能

一致する場合は、次の情報がネイバー テーブルに入力されます。

- ネイバー ID ― 隣接ルータのルータ ID
- 優先度 ― 隣接ルータの優先度。優先度は、代表ルータの選定（「代表ルータ」[p.6-4] を参照）に使用されます。
- 状態 ― ネイバーから通信があったか、双方向通信の確立処理中であるか、リンクステート情報を共有しているか、または完全な隣接関係が確立されたかを示します。
- デッド タイム ― このネイバーから最後の hello パケットを受信したあとに経過した時間を示します。
- リンクローカル IPv6 アドレス ― ネイバーのリンクローカル IPv6 アドレス
- 代表ルータ ― ネイバーが代表ルータ、またはバックアップ代表ルータとして宣言されたかどうかを示します（「代表ルータ」[p.6-4] を参照）。
- ローカル インターフェイス ― このネイバーの hello パケットを受信したローカル インターフェイス

最初の hello パケットが新規ネイバーから受信されると、そのネイバーは、init 状態のネイバー テーブルに入力されます。いったん双方向通信が確立されると、隣接状態は双方向となります。2 つのインターフェイスが互いのリンクステート データベースを交換するため、次に ExStart および交換状態となります。これらがすべて完了すると、ネイバーは完全な状態へと移行し、これが完全な隣接関係となります。ネイバーは、デッド間隔で hello パケットをまったく送信しない場合は、ダウン状態に移行し、隣接とはみなされなくなります。

隣接関係

すべてのネイバーが隣接関係を確立するわけではありません。ネットワーク タイプと確立された代表ルータに応じて、完全な隣接関係を確立して、すべてのネイバーと LSA を共有するものと、そうでないものがあります。詳細については、「代表ルータ」（p.6-4）を参照してください。

隣接関係は、OSPFv3 のデータベース説明パケット、リンク状態要求パケット、およびリンク状態更新パケットを使用して確立されます。データベース説明パケットに含まれるのは、ネイバーのリンクステート データベースからの LSA ヘッダーのみです（「リンクステート データベース」[p.6-8]を参照）。ローカル ルータは、これらのヘッダーを自身のリンクステート データベースと比較して、新規の LSA か、更新された LSA かを判定します。ローカル ルータは、新規または更新の情報を必要とする各 LSA について、リンク状態要求パケットを送信します。これに対し、ネイバーはリンク状態更新パケットを返信します。このパケット交換は、両方のルータのリンクステート情報が同じになるまで継続します。

代表ルータ

複数のルータを含むネットワークは、OSPFv3 特有の状況です。すべてのルータがネットワークで LSA をフラッディングした場合は、同じリンクステート情報が複数の送信元から送信されます。ネットワークのタイプに応じて、OSPFv3 は*代表ルータ*（*DR*）という 1 台のルータを使用して、LSA のフラッディングを制御し、OSPFv3 の残りの部分に対してネットワークを代表する場合があります（「エリア」[p.6-5] を参照）。DR がダウンした場合、OSPFv3 は*バックアップ代表ルータ*（BDR）を選定し、この BDR を使用します。

ネットワーク タイプは次のとおりです。

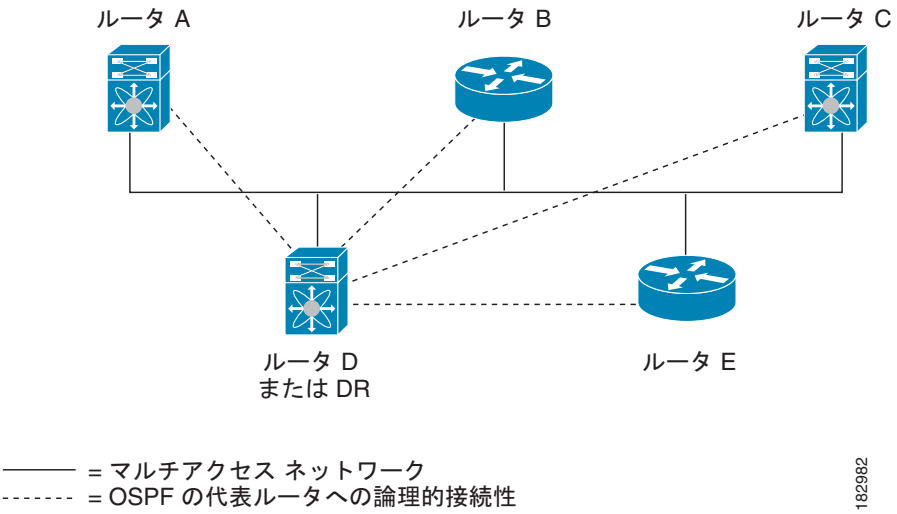
- ポイントツーポイント ― 2 台のルータ間にのみ存在するネットワーク。ポイントツーポイント ネットワーク上の全ネイバーは隣接関係を確立し、DR は存在しません。
- ブロードキャスト ― ブロードキャスト トラフィックが可能なイーサネットなどの共有メディア上で通信できる複数のルータを持つネットワーク。OSPFv3 ルータは DR および BDR を確立し、これらにより、ネットワーク上の LSA フラッディングを制御します。OSPFv3 は、よく知られている IPv6 マルチキャスト アドレス FF02::5 および MAC アドレス 0100.5300.0005 を使用して、ネイバーと通信します。

DR と BDR は、hello パケット内の情報に基づいて選択されます。インターフェイスは hello パケットの送信時に、どれが DR および BDR かわかっている場合は、優先フィールドと、DR および BDR フィールドを設定します。ルータは、Hallo パケットの DR および BDR フィールドで宣言されたルータと優先フィールドに基づいて、選定手順を実行します。最終的に OSPFv3 は、もっとも大きいルータ ID を DR および BDR として選択します。

他のルータはすべて DR および BDR と隣接関係を確立し、IPv6 マルチキャスト アドレス FF02::6 を使用して、LSA 更新情報を DR と BDR に送信します。図 6-1 は、すべてのルータと DR の間のこの隣接関係を示します。

DR は、ルータ インターフェイスに基づいています。1 つのネットワークの DR であるルータは、別のインターフェイス上の他のネットワークの DR となることはできません。

図 6-1　マルチアクセス ネットワークの DR

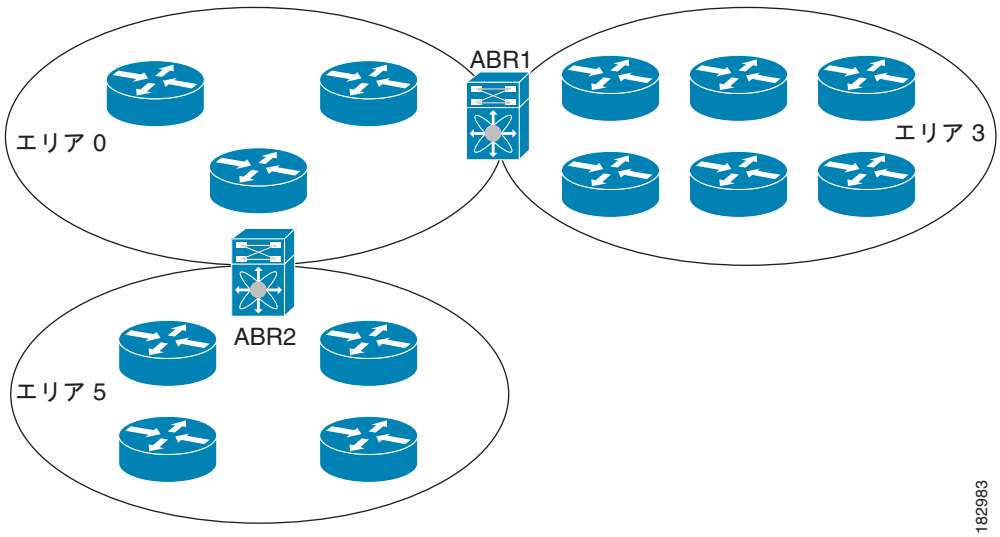


エリア

OSPFv3 ネットワークを複数の*エリア*に分割すると、ルータに要求される OSPFv3 の CPU とメモリに関する要件を制限できます。エリアとは、ルータの論理的な区分で、OSPFv3 ドメイン内にリンクして別のサブドメインを作成します。LSA フラッディングはエリア内でのみ発生し、リンクステート データベースはエリア内のリンクにのみ制限されます。定義されたエリア内のインターフェイスには、エリア ID を割り当てることができます。エリア ID は、10.2.3.1 などの、数字またはドット付き 10 進表記で表現される 32 ビット値です。

OSPFv3 ネットワーク内に複数のエリアを定義する場合は、0 という予約されたエリア ID を持つバックボーン エリアも定義する必要があります。エリアが複数ある場合は、1 台以上のルータが Area Border Router（ABR; エリア境界ルータ）となります。ABR は、バックボーン エリアと他の 1 つ以上の定義済みエリアの両方に接続します（図 6-2 を参照）。

図 6-2　OSPFv3 エリア



ABR には、接続するエリアごとに個別のリンクステート データベースがあります。ABR は、接続したエリアの 1 つからバックボーン エリアにエリア間プレフィクス（タイプ 3）LSA（「ルート集約」[p.6-11] を参照）を送信します。バックボーン エリアは、1 つのエリアに関する集約情報を別のエリアに送信します。図 6-2 では、エリア 0 が、エリア 5 に関する集約情報をエリア 3 に送信しています。

OSPFv3 では、Autonomous System Boundary Router（ASBR; 自律システム境界ルータ）という、もう 1 つのルータ タイプも定義されています。このルータは、OSPFv3 エリアを別の Autonomous System（AS; 自律システム）に接続します。自律システムとは、単一の技術的管理エンティティにより制御されるネットワークです。OSPFv3 は、そのルーティング情報を別の AS に再配布したり、再配布されたルートを別の AS から受信したりできます。詳細については、「高度な機能」（p.6-9）を参照してください。

Link-State Advertisement

OSPFv3 は LSA を使用して、自身のルーティング テーブルを構築します。

ここでは、次の内容について説明します。

- LSA タイプ（p.6-6）
- リンク コスト（p.6-7）
- フラッディングと LSA グループ ペーシング（p.6-7）
- リンクステート データベース（p.6-8）

LSA タイプ

表 6-1 は、Cisco NX-OS でサポートされる LSA タイプを示します。

表 6-1　LSA タイプ

タイプ	名前	説明
1	ルータ LSA	すべてのルータが送信する LSA。この LSA には、すべてのリンクの状態とコストが含まれます。プレフィクス情報は含まれません。ルータ LSA は SPF 再計算をトリガーします。ルータ LSA はローカル OSPFv3 エリアにフラッディングされます。
2	ネットワーク LSA	DR が送信する LSA。この LSA には、マルチアクセス ネットワーク内のすべてのルータの一覧が含まれます。プレフィクス情報は含まれません。ネットワーク LSA は SPF 再計算をトリガーします。「代表ルータ」（p.6-4）を参照してください。
3	エリア間プレフィクス LSA	ABR が、ローカル エリア内の宛先ごとに外部エリアに送信する LSA。この LSA には、ABR からローカルの宛先へのリンク コストが含まれます。「エリア」（p.6-5）を参照してください。
4	エリア間ルータ LSA	ABR が外部エリアに送信する LSA。この LSA は、リンク コストを ASBR のみにアドバタイズします。「エリア」（p.6-5）を参照してください。
5	AS 外部 LSA	ASBR が生成する LSA。この LSA には、外部 AS 宛先へのリンク コストが含まれます。AS 外部 LSA は、AS 全体にわたってフラッディングされます。「エリア」（p.6-5）を参照してください。

表 6-1　LSA タイプ（続き）

タイプ	名前	説明
7	タイプ 7 LSA	ASBR が NSSA 内で生成する LSA。この LSA には、外部 AS 宛先へのリンク コストが含まれます。タイプ 7 LSA は、ローカル NSSA 内のみでフラッディングされます。「エリア」（p.6-5）を参照してください。
8	リンク LSA	すべてのルータが、リンクローカル フラッディング スコープを使用して送信する LSA（「フラッディングと LSA グループ ペーシング」[p.6-7] を参照）。この LSA には、このリンクのリンクローカル アドレスと IPv6 アドレスが含まれます。
9	エリア内プレフィクス LSA	すべてのルータが送信する LSA。この LSA には、プレフィクスまたはリンク状態へのあらゆる変更が含まれます。エリア内プレフィクス LSA はローカル OSPFv3 エリアにフラッディングされます。この LSA は SPF 再計算をトリガーしません。
11	猶予 LSA	再起動されるルータが、リンクローカル フラッディング スコープを使用して送信する LSA。この LSA は、OSPFv3 のグレースフル リスタートに使用されます。「ハイ アベイラビリティとグレースフル リスタート」（p.6-11）を参照してください。

リンク コスト

各 OSPFv3 インターフェイスには、*リンク コスト*が割り当てられます。このコストは任意の数字です。デフォルトでは、Cisco NX-OS が、設定された参照帯域幅をインターフェイス帯域幅で割った値をコストとして割り当てます。デフォルトでは、参照帯域幅は 40 Gbps です。リンク コストは各リンクに対して、LSA 更新情報で伝えられます。

フラッディングと LSA グループ ペーシング

OSPFv3 は、LSA タイプに応じて、ネットワークのさまざまな部分に LSA 更新をフラッディングします。OSPFv3 は、次のフラッディング スコープを使用します

- リンクローカル ― LSA は、ローカル リンクでのみフラッディングされ、それより先にはフラッディングされません。リンク LSA および猶予 LSA に使用されます。
- エリアローカル ― LSA は、単一の OSPF エリア全体にのみフラッディングされます。ルータ LSA、ネットワーク LSA、エリア間プレフィクス LSAs、エリア間ルータ LSA、およびエリア内プレフィクス LSA に使用されます。
- AS スコープ ― LSA は、ルーティング ドメイン全体にフラッディングされます。AS 外部 LSA に使用されます。

LSA フラッディングにより、ネットワーク内のすべてのルータが同じルーティング情報を持つことが保証されます。LSA フラッディングは、OSPFv3 エリアの設定により異なります（「エリア」[p.6-5] を参照）。LSA は、*リンクステート リフレッシュ*時間に基づいて（デフォルトでは 30 分ごとに）フラッディングされます。各 LSA にはそれぞれ、リンクステート リフレッシュ時間が設定されています。

ネットワークの LSA 更新情報のフラッディング レートは、LSA グループ ペーシング機能を使用して制御できます。LSA グループ ペーシングにより、CPU またはバッファの使用率を低下させることができます。この機能により、同様のリンクステート リフレッシュ時間を持つ LSA がグループ化されるため、OSPFv3 で、複数の LSA を 1 つの OSPFv3 更新メッセージにまとめることが可能となります。

デフォルトでは、相互のリンクステート リフレッシュ時間が 4 分以内の LSA が同じグループに入れられます。この値は、大規模なリンクステート データベースでは低く、小規模のデータベースでは高くして、ネットワーク上の OSPFv3 負荷を最適化する必要があります。

リンクステート データベース

各ルータは、OSPFv3 ネットワーク用のリンクステート データベースを維持しています。このデータベースには、接続されたすべての LSA が含まれ、ネットワークを通過するすべてのルートに関する情報が格納されます。OSPFv3 は、この情報を使用して、各宛先への最適なパスを計算し、この最適なパスをルーティング テーブルに入力します。

LSA は、MaxAge と呼ばれる設定済みの時間間隔で 受信された LSA 更新情報がまったくない場合は、リンクステート データベースから削除されます。ルータは、LSA を 30 分ごとに繰り返してフラッディングし、正確なリンクステート情報が期限切れで削除されるのを防ぎます。Cisco NX-OS は、すべての LSA が同時にリフレッシュされるのを防ぐために、LSA グループ機能をサポートしています。詳細については、「フラッディングと LSA グループ ペーシング」（p.6-7）を参照してください。

OSPFv3 と IPv6 ユニキャスト RIB

OSPFv3 は、リンクステート データベースでダイクストラの SPF アルゴリズムを実行します。このアルゴリズムにより、パス上の各リンクのリンク コストの合計に基づいて、各宛先への最適なパスが選択されます。そして、選択された各宛先への最短パスが OSPFv3 ルーティング テーブルに入力されます。OSPFv3 ネットワークが収束すると、このルーティング テーブルは IPv6 ユニキャスト RIB にデータを提供します。OSPFv3 は IPv6 ユニキャスト RIB と通信し、次の動作を行います。

- ルートの追加または削除
- 他のプロトコルからのルートの再配布への対応
- 変更されていない OSPFv3 ルートの削除およびスタブ ルータ アドバタイズメントを行うための収束更新情報の提供（「複数の OSPFv3 インスタンス」[p.6-12] を参照）

さらに OSPFv3 は、変更済みダイクストラ アルゴリズムを実行して、エリア間プレフィクス、エリア間ルータ、AS 外部、タイプ 7、およびエリア内プレフィクス（タイプ 3、4、5、7、8）の各 LSA の変更の高速再計算を行います。

認証

OSPFv3 は、OSPFv3 パケットの整合性、認証、および機密性を保証するために IPv6 認証ヘッダーと IPv6 Encapsulating Security Payload を必要とします（第 3 章「IPv6 の設定」を参照）。

アドレス ファミリのサポート

Cisco NX-OS は、ユニキャスト IPv6 やマルチキャスト IPv6 などの複数のアドレス ファミリをサポートしています。*アドレス ファミリ*に特有の OSPFv3 機能は、次のとおりです。

- デフォルト ルート
- ルート集約
- ルート再配布
- 境界ルータのフィルタ リスト
- SPF 最適化

これらの機能の設定時に IPv6 ユニキャスト アドレス ファミリ コンフィギュレーション モードにするには、***address-family ipv6 unicast*** コマンドを使用します。

高度な機能

Cisco NX-OS は、ネットワークでの OSPFv3 の可用性やスケーラビリティを向上させる数多くの高度な OSPFv3 機能をサポートしています。

ここでは、次の内容について説明します。

- スタブ エリア（p.6-9）
- Not-So-Stubby エリア（p.6-10）
- 仮想リンク（p.6-10）
- ルート再配布（p.6-11）
- ルート集約（p.6-11）
- ハイ アベイラビリティとグレースフル リスタート（p.6-11）
- 複数の OSPFv3 インスタンス（p.6-12）
- SPF 最適化（p.6-12）
- 仮想化のサポート（p.6-13）

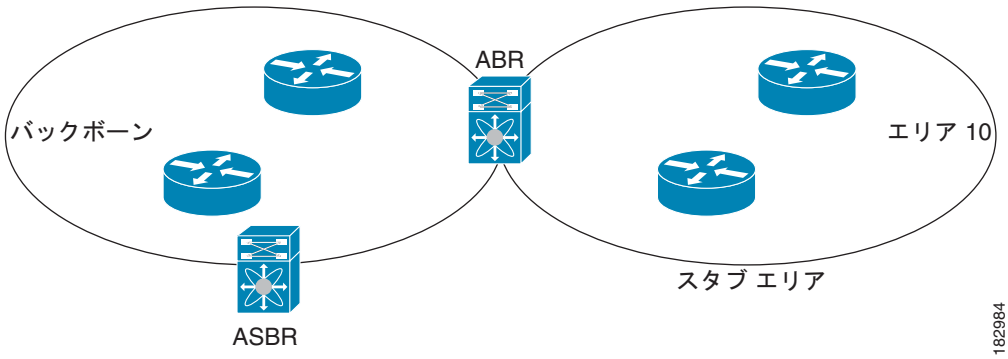
スタブ エリア

エリアを*スタブ エリア*にすると、エリアでフラッディングされる外部ルーティング情報の量を制限できます。スタブ エリアとは、AS 外部（タイプ 5）LSA（「Link-State Advertisement」[p.6-6] を参照）が許可されないエリアです。これらの LSA は通常、外部ルーティング情報を伝播するためにローカル AS 全体でフラッディングされます。スタブ エリアには、次の要件があります。

- スタブ エリア内のすべてのルータはスタブ ルータです。「スタブ ルーティング」（p.1-7）を参照してください。
- スタブ エリアには ASBR ルータは存在しません。
- スタブ エリアには仮想リンクを設定できません。

図 6-3 は、外部 AS に到達するためにエリア 10 内のすべてのルータが ABR を通過する必要のある OSPFv3 AS の例を示します。エリア 10 は、スタブ エリアとして設定できます。

図 6-3　スタブ エリア



スタブ エリアは、外部 AS へのバックボーン エリアを通過する必要のあるすべてのトラフィックにデフォルト ルートを使用します。デフォルト ルートは、プレフィクス長が IPv6 向けに 0 に設定されたエリア間プレフィクス LSA です。

Not-So-Stubby エリア

Not-So-Stubby Area（*NSSA*）はスタブ エリアに似ていますが、NSSA では、再配布を使用して NSSA 内で AS 外部ルートを使用できる点が異なります。NSSA ASBR はこれらのルートを再配布し、タイプ 7 LSA を生成して NSSA 全体にフラッディングします。または、このタイプ 7 LSA を AS 外部（タイプ 5）LSA に変換するよう、NSSA を他のエリアに接続する ABR を設定することができます。こうすると、ABR は、これらの AS 外部 LSA を OSPFv3 AS 全体にフラッディングします。変換時には、集約とフィルタリングがサポートされます。タイプ 7 LSA の詳細については、「Link-State Advertisement」（p.6-6）を参照してください。

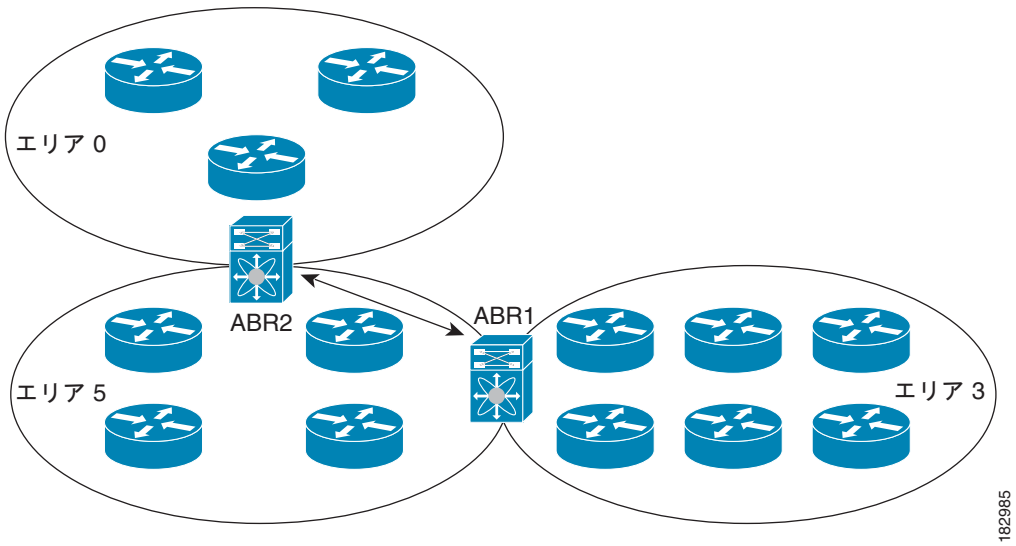
たとえば、OSPFv3 を使用する中央サイトを、異なるルーティング プロトコルを使用するリモート サイトに接続するときに NSSA を使用すると、管理作業を簡素化できます。リモート サイトへのルートはスタブ エリア内に再配布できないため、NSSA を使用する前に、企業サイトの境界ルータとリモート ルータの間の接続を OSPFv3 スタブ エリアとして実行することはできません。2 つのルーティング プロトコルを維持する必要があります。NSSA を使用すると、企業のルータとリモート ルータ間のエリアを NSSA として定義する（「NSSA の設定」[p.6-23] を参照）ことで、OSPFv3 を拡張してリモート接続性をサポートできます。

バックボーン エリア 0 を NSSA にすることはできません。

仮想リンク

仮想リンクを使用すると、物理的に直接接続できない場合に、OSPFv3 エリア ABR をバックボーン エリア ABR に接続できます。図 6-4 は、エリア 3 をエリア 5 経由でバックボーン エリアに接続する仮想リンクを示します。

図 6-4　仮想リンク



また、仮想リンクを使用して、分割エリアから一時的に回復できます。分割エリアは、エリア内のリンクがダウンしたために隔離された一部のエリアで、ここからはバックボーン エリアへの代表 ABR に到達できません。

ルート再配布

OSPFv3 は、ルート再配布を使用して、他のルーティング プロトコルからルートを学習できます。「ルート再配布」（p.1-6）を参照してください。リンク コストをこれらの再配布されたルートに割り当てるか、またはデフォルト リンク コストを再配布されたすべてのに割り当てるよう、OSPFv3 を設定します。

ルート再配布では、ルートマップを使用して、再配布する外部ルートを管理します。ルートマップの設定の詳細については、第 14 章「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、これらの外部ルートがローカル OSPFv3 AS でアドバタイズされる前に AS 外部（タイプ 5）LSA およびタイプ 7 LSA のパラメータを変更できます。

ルート集約

OSPFv3 は、学習したすべてのルートを、すべての OSPF 対応ルータと共有するため、ルート集約を使用して、すべての OSPF 対応ルータにフラッディングされる一意のルートの数を削減した方がよい場合があります。ルート集約により、より具体的な複数のアドレスが、すべての具体的なアドレスを表す 1 つのアドレスに置き換えられるため、ルーティング テーブルが簡素化されます。たとえば、2010:11:22:0:1000::1 と 2010:11:22:0:2000:679:1 を 1 つの集約アドレス 2010:11:22::/32 に置き換えることができます。

一般的には、ABR の境界ごとに集約します。集約は 2 つのエリアの間でも設定できますが、バックボーンの方向に集約する方が適切です。こうすると、バックボーンがすべての集約アドレスを受信し、すでに集約されているそれらのアドレスを他のエリアに投入できるためです。集約には、次の 2 タイプがあります。

- エリア間ルート集約
- 外部ルート集約

エリア間ルート集約は ABR 上で設定し、AS 内のエリア間のルートを集約します。集約の利点を生かすには、これらのアドレスを 1 つの範囲内にまとめられるよう、連続するネットワーク番号をエリア内で割り当てる必要があります。

外部ルート集約は、ルート再配布を使用して OSPFv3 に投入される外部ルートに特有のルート集約です。集約する外部の範囲が連続していることを確認する必要があります。異なる 2 台のルータからの重複範囲を集約すると、誤った宛先にパケットが送信される原因となる場合があります。外部ルート集約は、ルートを OSPF に再配布している ASBR で設定してください。

集約アドレスの設定時に Cisco NX-OS は、ルーティング ブラック ホールおよびルート ループを防ぐために、集約アドレスの廃棄ルートを自動的に設定します。

ハイ アベイラビリティとグレースフル リスタート

Cisco NX-OS はハイ アベイラビリティをサポートしています。Cisco NX-OS システムでコールド リブートが発生した場合は、ネットワークがシステムへのトラフィックの転送を停止し、システムをネットワーク トポロジから削除します。このシナリオでは、OSPFv3 でステートレス再起動が発生し、ローカル システム上のネイバーの隣接関係がすべて削除されます。Cisco NX-OS はスタートアップ コンフィギュレーションを適用し、OSPFv3 はネイバーを再探索して、隣接関係を再確立します。

OSPFv3 は、処理に問題が発生した場合は、自動的に再起動します。再起動後に OSPFv3 はグレースフル リスタートを開始するため、プラットフォームがネットワーク トポロジから削除されることはありません。手動で OSPF を再起動した場合は、ステートフル スイッチオーバーと同様のグレースフル リスタートが実行されます。いずれの場合にも実行コンフィギュレーションが適用されます。

グレースフル リスタート、つまり、Nonstop Forwarding（NSF）では、処理の再起動中も OSPFv3 がデータ転送パス上に存在し続けます。OSPFv3 は再起動が必要になると、最初にリンクローカル猶予（タイプ 11）LSA を送信します。この再起動中の OSPFv3 プラットフォームは NSF 対応と呼ばれます。

猶予 LSA には猶予期間が含まれます。猶予期間とは、ネイバー OSPFv3 インターフェイスは再起動中の OSPFv3 インターフェイスからの LSA を待つよう指定された時間です。（通常、OSPFv3 は隣接関係を解消し、ダウンした、または再起動中の OSPFv3 インターフェイスからのすべての LSA を廃棄します。）関与するネイバーは NSF ヘルパーと呼ばれ、再起動中の OSPFv3 インターフェイスが発信するすべての LSA を、このインターフェイスが隣接したままであるかのように維持します。

再起動中の OSPFv3 インターフェイスが稼働を再開すると、ネイバーを再探索して隣接関係を確立し、LSA 更新情報の送信を再開します。この時点で、NSF ヘルパーは、グレースフル リスタートが完了したと認識します。

（注） 再起動中の OSPFv3 インターフェイスが猶予期間内に稼働を再開しない場合、またはネットワークでトポロジが変更された場合は、各 OSPFv3 ネイバーは再起動中の OSPFv3 との隣接関係を解消し、通常の OSPFv3 の再起動として対応します。

（注） グレースフル リスタートでは、OSPFv3 の In-Service Software Upgrade（ISSU; インサービス ソフトウェア アップグレード）のサポートをイネーブルにする必要があります。グレースフル リスタートをディセーブルにした場合は、Cisco NX-OS により、この設定では ISSU をサポートできないという警告が発行されます。

複数の OSPFv3 インスタンス

Cisco NX-OS は、OSPFv3 プロトコルの複数インスタンスをサポートしています。デフォルトでは、すべてのインスタンスが同じシステム ルータ ID を使用します。複数のインスタンスが同じ OSPFv3 AS にある場合は、各インスタンスのルータ ID を手動で設定する必要があります。

OSPFv3 ヘッダーには、特定の OSPFv3 インスタンスの OSPFv3 パケットを識別するためのインスタンス ID フィールドが含まれます。この OSPv3 インスタンスを割り当てることができます。インターフェイスは、パケット ヘッダーの OSPFv3 インスタンス ID が一致しない OSPFv3 パケットをすべてドロップします。

Cisco NX-OS では、インターフェイス上に 1 つの OSPFv3 インスタンスのみが許可されます。

SPF 最適化

Cisco NX-OS は、次の方法で SPF アルゴリズムを最適化します。

- ネットワーク（タイプ 2）LSA、エリア間プレフィクス（タイプ 3）LSA、および AS 外部（タイプ 5）LSA 用部分 SPF — これらの LSA のいずれかが変更されると、Cisco NX-OS は、全体的な SPF 計算ではなく、高速部分計算を実行します。
- SPF タイマー — さまざまなタイマーを設定して、SPF 計算を制御できます。これらのタイマーには、後続の SPF 計算の幾何バックオフが含まれます。幾何バックオフにより、複数の SPF 計算による CPU 負荷が制限されます。

仮想化のサポート

OSPFv3 は、Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）をサポートしています。VRF は Virtual Device Contexts（VDC; 仮想化デバイス コンテキスト）内にあります。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』の第 13 章「レイヤ 3 仮想化の設定」を参照してください。

OSPFv3 のライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	OSPFv3 には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細説明およびライセンスの入手と適用の方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

OSPFv3 の前提条件

OSPFv3 には、次の前提条件があります。

- OSPFv3 を設定するための、ルーティングの基礎に関する詳しい知識がある。
- スイッチにログオンしている。
- リモート OSPFv3 ネイバーと通信可能な 1 つ以上の IPv6 用インターフェイスが設定されている。
- Enterprise Services ライセンスがインストールされている。
- OSPFv3 ネットワーク戦略と、ネットワークのプランニングが完成している。たとえば、複数のエリアが必要かどうかを決定する必要があります。
- OSPF 機能がイネーブルにされている（「OSPFv3 機能のイネーブル化」[p.6-14] を参照）。
- Advanced Services ライセンスがインストールされ、VDC を設定している場合は、目的の VDC が入力されている（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。
- IPv6 アドレス指定および基本設定に関する詳しい知識がある。IPv6 ルーティングおよびアドレス指定の詳細については、第 3 章「IPv6 の設定」を参照してください。

設定時の注意事項および制約事項

OSPFv3 設定時の注意事項および制約事項は次のとおりです。

- VDC 内に含まれるのは、最大 4 つの OSPFv3 インスタンスです。

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

基本的 OSPFv3 の設定

OSPFv3 は、OSPFv3 ネットワークを設計したあとに設定します。

ここでは、次の内容について説明します。

- OSPFv3 機能のイネーブル化（p.6-14）
- OSPFv3 インスタンスの作成（p.6-15）
- OSPFv3 でのネットワークの設定（p.6-17）

OSPFv3 機能のイネーブル化

OSPFv3 を設定するには、その前に OSPFv3 機能をイネーブルにする必要があります。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. ***feature ospfv3***
3. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature ospfv3** **例：** `switch(config)# feature ospfv3`	OSPFv3 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

OSPFv3 機能をディセーブルにし、関連付けられた設定をすべて削除するには、***no feature ospfv3*** コマンドを使用します。

コマンド	目的
no feature ospfv3 **例：** `switch(config)# no feature ospfv3`	OSPFv3 機能をディセーブルにして、関連付けられた設定をすべて削除します。

OSPFv3 インスタンスの作成

OSPFv3 設定の最初のステップは、インスタンス、つまり OSPFv3 インスタンスの作成です。作成した OSPFv3 インスタンスには、一意のインスタンス タグを割り当てます。インスタンス タグは任意の文字列です。各 OSPFv3 インスタンスには、省略可能な次のパラメータも設定できます。

- Router ID ― この OSPFv3 インスタンスのルータ ID を設定します。このパラメータを使用しない場合は、ルータ ID 選択アルゴリズムが使用されます。「ルータ ID」（p.1-5）を参照してください。
- Administrative distance ― ルーティング情報の送信元の信頼性をランク付けします。「管理ディスタンス」（p.1-7）を参照してください。
- Log adjacency changes ― OSPFv3 ネイバーの状態が変化するたびにシステム メッセージを作成します。
- Maximum paths ― OSPFv3 が、特定の宛先についてルーティング テーブルにインストールする同等パスの最大数を設定します。このパラメータは、複数パス間のロード バランシングに使用します。
- Reference bandwidth ― ネットワークの算出 OSPFv3 コスト メトリックを制御します。算出コストは、参照帯域幅をインターフェイス帯域幅で割った値です。算出コストは、ネットワークが OSPFv3 インスタンスに追加されるときにリンク コストを割り当てると、無効にすることができます。「OSPFv3 でのネットワークの設定」（p.6-17）を参照してください。

OSPFv3 インスタンス パラメータの詳細については、「高度な OSPFv3 の設定」（p.6-20）を参照してください。

操作の前に

OSPFv3 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14]を参照）。

使用する予定の OSPFv3 インスタンス タグが、このルータ上では使用されていないことを確認します。

インスタンス タグが使用されていないことを確認するには、***show ospfv3*** *instance-tag* コマンドを使用します。

OSPFv3 がルータ ID（設定済みのループバック アドレスなど）を入手可能であるか、またはルータ ID オプションを設定する必要があります。

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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***router-id*** *ip-address*
4. ***show ipv6 ospfv3*** *instance-tag*
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**router-id** *id* **例：** `switch(config-router)# router-id 209.0.2.1`	（任意）OSPFv3 ルータ ID を設定します。このドット付き 10 進表記の ID で、この OSPFv3 インスタンスが識別されます。この ID は、システムの設定済みインターフェイス上に存在する必要があります。
ステップ 4	***show ipv6 ospfv3*** *instance-tag* **例：** `switch(config-router)# show ipv6 ospfv3 201`	（任意）OSPFv3 情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

OSPFv3 インスタンスと、関連付けられた設定をすべて削除するには、***no router ospfv3*** コマンドを使用します。

コマンド	目的
no router ospfv3 *instance-tag* **例：** `switch(config)# no router ospfv3 201`	OSPFv3 インスタンスと、関連付けられたすべての設定を削除します。

（注） このコマンドは、インターフェイス モードでは OSPF 設定を削除しません。インターフェイス モードで設定された OSPFv3 コマンドはいずれも、手動で削除する必要があります。

ルータ コンフィギュレーション モードで、次の OSPFv3 用オプション パラメータを設定できます。

コマンド	目的
log-adjacency-changes [**detail**] **例：** `switch(config-router)# log-adjacency-changes`	ネイバーの状態が変化するたびに、システム メッセージを生成します。

address-family ipv6 unicast コマンドを使用すると、アドレス ファミリ コンフィギュレーション モードで、省略可能な次の OSPFv3 用パラメータを設定できます。

コマンド	目的
distance *number* **例：** `switch(config-router-af)# distance 25`	この OSPFv3 インスタンスの管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 110 です。
maximum-paths *paths* **例：** `switch(config-router-af)# maximum-paths 4`	ルーティング　テーブル内の宛先への同じ OSPFv3 パスの最大数を設定します。有効値の範囲は 1 ～ 16 であり、デフォルトは 8 です。ロード バランシングに使用されます。

次に、OSPFv3 インスタンスを作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# copy running-config startup-config
```

OSPFv3 でのネットワークの設定

ルータがこのネットワークへの接続に使用するインターフェイスを介して、OSPFv3 へのネットワークを関連付けることで、このネットワークを設定できます（「ネイバー」[p.6-3] を参照）。すべてのネットワークをデフォルト バックボーン エリア（エリア 0）に追加したり、任意の 10 進数または IP アドレスを使用して新規エリアを作成したりできます。

（注） すべてのエリアは、バックボーン エリアに直接、または仮想リンク経由で接続する必要があります。

（注） インターフェイスの有効な IPv6 アドレスを設定するまでは、インターフェイス上で OSPFv3 がイネーブルになりません。

操作の前に

OSPFv3 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14] 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）。

コマンドの一覧

1. **config t**
2. **interface** *interface-type slot/port*
3. ***ipv6 address*** *ipv6-prefix/length*
4. ***ipv6 router ospfv3*** *instance-tag* ***area*** *area-id* [**secondaries none**]
5. ***show ipv6 ospfv3*** *instance-tag* **interface** *interface-type slot/port*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ipv6 address** *ipv6-prefix/length* **例：** `switch(config-if)# ipv6 address 2001:0DB8::1/48`	このインターフェイスに IPv6 アドレスを割り当てます。
ステップ 4	**ipv6** ***router*** **ospfv3** *instance-tag* **area** *area-id* [**secondaries none**] **例：** `switch(config-if)# ipv6 router ospfv3 201 area 0`	OSPFv3 インスタンスおよびエリアにインターフェイスを追加します。
ステップ 5	***show ipv6 ospfv3*** *instance-tag* **interface** *interface-type slot/port* **例：** `switch(config-if)# show ipv6 ospfv3 201 interface ethernet 1/2`	（任意）OSPFv3 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

インターフェイス コンフィギュレーション モードで、省略可能な次の OSPFv3 パラメータを設定できます。

コマンド	目的
ospfv3 cost *number* **例：** `switch(config-if)# ospfv3 cost 25`	このインターフェイスの OSPFv3 コスト メトリックを設定します。デフォルトでは、参照帯域幅とインターフェイス帯域幅に基づいて、コスト メトリックが計算されます。有効値の範囲は 1 ～ 65535 です。
ospfv3 dead-interval *seconds* **例：** `switch(config-if)# ospfv3 dead-interval 50`	OSPFv3 デッド間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 です。デフォルトでは、hello 間隔の秒数の 4 倍です。
ospfv3 hello-interval *seconds* **例：** `switch(config-if)# ospfv3 hello-interval 25`	OSPFv3 hello 間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 秒です。
ospfv3 instance *instance* **例：** `switch(config-if)# ospfv3 instance 25`	OSPFv3 インスタンス ID を設定します。有効値の範囲は 0 ～ 255 であり、デフォルトは 0 です。インスタンス ID のスコープはリンクローカルです。

コマンド	目的
ospfv3 mtu-ignore **例：** `switch(config-if)# ospfv3 mtu-ignore`	OSPFv3 で、ネイバーとのあらゆる IP Maximum Transmission Unit（MTU; 最大伝送ユニット）不一致が無視されるよう設定します。デフォルトでは、ネイバー MTU がローカル インターフェイス MTU が不一致の場合には、隣接関係が確立されません。
ospfv3 network {**broadcast** | **point-point**} **例：** `switch(config-if)# ospfv3 network broadcast`	OSPFv3 ネットワーク タイプを設定します。
ospfv3 passive-interface **例：** `switch(config-if)# ospfv3 passive-interface`	インターフェイス上でルーティングが更新されないようにします。
ospfv3 priority *number* **例：** `switch(config-if)# ospfv3 priority 25`	エリアの DR の決定に使用される OSPFv3 優先度を設定します。有効値の範囲は 0 ～ 255 であり、デフォルトは 1 です。「代表ルータ」（p.6-4）を参照してください。
ospfv3 shutdown **例：** `switch(config-if)# ospfv3 shutdown`	このインターフェイス上の OSPFv3 インスタンスをシャットダウンします。

次に、OSPFv3 インスタンス 201 にネットワーク エリア 10 を追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ipv6 address 2001:0DB8::1/48
switch(config-if)# ipv6 ospfv3 201 area 10
switch(config-if)# copy running-config startup-config
```

高度な OSPFv3 の設定

OSPFv3 は、OSPFv3 ネットワークを設計したあとに設定します。

ここでは、次の内容について説明します。

- 境界ルータのフィルタ リストの設定（p.6-20）
- スタブ エリアの設定（p.6-21）
- Totally Stubby エリアの設定（p.6-23）
- NSSA の設定（p.6-23）
- 仮想リンクの設定（p.6-25）
- 再配布の設定（p.6-27）
- ルート集約の設定（p.6-28）
- デフォルト タイマーの変更（p.6-30）
- グレースフル リスタートの設定（p.6-32）
- OSPFv3 インスタンスの再起動（p.6-34）
- 仮想化による OSPFv3 の設定（p.6-34）

境界ルータのフィルタ リストの設定

OSPFv3 ドメインを、関連性のある各ネットワークを含む一連のエリアに分離できます。すべてのエリアは、ABR 経由でバックボーン エリアに接続している必要があります。OSPFv3 ドメインは、ASBR を介して、外部ドメインにも接続可能です。「エリア」（p.6-5）を参照してください。

ABR には、省略可能な次の設定パラメータがあります。

- Area range — エリア間のルート集約を設定します。「ルート集約の設定」（p.6-28）を参照してください。
- Filter list — ABR 上で、外部エリアから受信したエリア間プレフィクス（タイプ 3）LSA をフィルタリングします。

ASBR もフィルタ リストをサポートしています。

操作の前に

フィルタ リストが、着信または発信エリア間プレフィクス（タイプ 3）LSA の IP プレフィクスのフィルタリングに使用するルートマップを作成します。第 14 章「Route Policy Manager の設定」を参照してください。

OSPFv3 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14]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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***address-family ipv6 unicast***
4. ***area*** *area-id* ***filter-list route-map*** *map-name* ***{in | out}***
5. ***show ipv6 ospfv3 policy statistics area*** *id* ***filter-list*** {***in*** | ***out***}
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**address-family ipv6 unicast**
	
	例：
	`switch(config-router)# address-family ipv6 unicast`
	`switch(config-router-af)#`
		IPv6 ユニキャスト アドレス ファミリ モードを開始します。

ステップ 4	**area** *area-id* **filter-list route-map** *map-name* {***in*** | ***out***} **例：** `switch(config-router-af)# area 10 filter-list route-map FilterLSAs in`	ABR 上で着信または発信エリア間プレフィクス（タイプ 3）LSA をフィルタリングします。
ステップ 5	***show ipv6 ospfv3 policy statistics area*** *id* ***filter-list*** {***in*** | ***out***} **例：** `switch(config-if)# show ipv6 ospfv3 policy statistics area 10 filter-list in`	（任意）OSPFv3 ポリシー情報を表示します。
ステップ 6	**copy running-config startup-config** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ディセーブルにされているグレースフル リスタートをイネーブルにする例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# area 10 filter-list route-map FilterLSAs in
switch(config-router-af)# copy running-config startup-config
```

スタブ エリアの設定

OSPFv3 ドメインの、外部トラフィックが不要な部分にスタブ エリアを設定できます。スタブ エリアは AS 外部（タイプ 5）LSA をブロックし、不要な、選択したネットワークへの往復のルーティングを制限します。「スタブ エリア」（p.6-9）を参照してください。また、すべての集約ルートがスタブ エリアを経由しないようブロックすることもできます。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14] を参照）。

設定されるスタブ エリア内に、仮想リンクと ASBR のいずれも含まれないことを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***area*** *area-id* ***stub***
4. ***address-family ipv6 unicast***
5. ***area*** *area-id* ***default-cost*** *cost*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **stub** **例：** `switch(config-router)# area 10 stub`	このエリアをスタブ エリアとして作成します。

ステップ 4	**address-family ipv6 unicast**
	
	例：
	`switch(config-router)# address-family ipv6 unicast`
	`switch(config-router-af)#`
		（任意）IPv6 ユニキャスト アドレス ファミリ モードを開始します。

ステップ 5	**area** *area-id* **default-cost** *cost* **例：** `switch(config-router-af)# area 10 default-cost 25`	（任意）このスタブ エリアに送信されるデフォルト集約ルートのコスト メトリックを設定します。有効値の範囲は 0 ～ 16777215 です。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、すべての集約ルート更新をブロックするスタブ エリアを作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 stub no-summary
switch(config-router)# copy running-config startup-config
```

Totally Stubby エリアの設定

Totally Stubby エリアを作成して、すべての集約ルート更新がスタブ エリアを経由しないようにすることができます。

Totally Stubby エリアを作成するには、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。

コマンド	目的
area *area-id* **stub no-summary** **例：** `switch(config-router)# area 20 stub no-summary`	このエリアを Totally Stubby エリアとして作成します。

NSSA の設定

OSPFv3 ドメインの、ある程度の外部トラフィックが必要な部分に NSSA を設定できます。「Not-So-Stubby エリア」（p.6-10）を参照してください。また、この外部トラフィックを AS 外部（タイプ 5）LSA に変換して、このルーティング情報で OSPFv3 ドメインをフラッディングすることもできます。NSSA は、省略可能な次のパラメータで設定できます。

- No redistribution ― 再配布されたルートが NSSA をバイパスして、OSPFv3 AS 内の他のエリアに再配布されます。このオプションは、NSSA ASBR が ABR も兼ねているときに使用します。
- Default information originate ― 外部 AS へのデフォルト ルートのタイプ 7 LSA を生成します。このオプションは、ASBR のルーティング テーブルにデフォルト ルートが含まれる場合に NSSA ASBR 上で使用します。このオプションは、ASBR のルーティング テーブルにデフォルト ルートが含まれるかどうかに関係なく、NSSA ASBR 上で使用できます。
- Route map ― 目的のルートのみが NSSA および他のエリア全体でフラッディングされるよう、外部ルートをフィルタリングします。
- Translate ― NSSA 外のエリア向けに、タイプ 7 LSA を AS 外部 LSA に変換します。再配布されたルートを OSPFv3 AS 全体でフラッディングするには、このコマンドを NSSA ABR 上で使用します。また、これらの AS 外部 LSA の転送アドレスを無効にすることもできます。
- No summary ― すべての集約ルートが NSSA でフラッディングされないようにします。このオプションは NSSA ABR 上で使用します。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14]を参照）。

設定する NSSA 上に仮想リンクがないことと、この NSSA がバックボーン エリアでないことを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. **area** *area-id* **nssa** [***no-redistribution***] [***default-information-originate*** [***route-map*** *map-name*]] [***no-summary***] [**translate type7** {**always** | **never**} [**suppress-fa**]]
4. ***address-family ipv6 unicast***
5. ***area*** *area-id* ***default-cost*** *cost*
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **nssa** [***no-redistribution***] [***default-information-originate***] [***route-map*** *map-name*]] [***no-summary***] [***translate type7*** {***always*** | ***never***} [**suppress-fa**]] **例：** `switch(config-router)# area 10 nssa`	このエリアを NSSA として作成します。

ステップ 4	**address-family ipv6 unicast**
	
	例：
	`switch(config-router)# address-family ipv6 unicast`
	`switch(config-router-af)#`
		（任意）IPv6 ユニキャスト アドレス ファミリ モードを開始します。

ステップ 5	**area** *area-id* **default-cost** *cost* **例：** `switch(config-router-af)# area 10 default-cost 25`	（任意）この NSSA に送信されるデフォルト集約ルートのコスト メトリックを設定します。有効値の範囲は 0 ～ 16777215 です。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、すべての集約ルート更新をブロックする NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 nssa no-summary
switch(config-router)# copy running-config startup-config
```

次に、デフォルト ルートを生成する NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 nssa default-info-originate
switch(config-router)# copy running-config startup-config
```

次に、外部ルートをフィルタリングし、すべての集約ルート更新をブロックする NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 nssa route-map ExternalFilter no-summary
switch(config-router)# copy running-config startup-config
```

次に、つねにタイプ 7 LSA を AS 外部（タイプ 5）LSA に変換する NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 nssa translate type 7 always
switch(config-router)# copy running-config startup-config
```

次に、すべての集約ルート更新をブロックする NSSA を作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 nssa no-summary
switch(config-router)# copy running-config startup-config
```

仮想リンクの設定

仮想リンクは、隔離されたエリアを、中継エリア経由でバックボーン エリアに接続します。「仮想リンク」（p.6-10）を参照してください。仮想リンクには、省略可能な次のパラメータを設定できます。

- Authentication ― 簡単なパスワード認証または MD5 メッセージ ダイジェスト認証、および関連付けられたキーを設定します。
- Dead interval ― ローカル ルータがデッドであることを宣言し、隣接関係を解消する前に、ネイバーが hello パケットを待つ時間を設定します。
- Hello interval ― 連続する hello パケット間の時間間隔を設定します。
- Retransmit interval ― 連続する LSA 間の推定時間間隔を設定します。
- Transmit delay ― LSA をネイバーに送信する推定時間間隔を設定します。

（注）　リンクがアクティブになる前に、関与する両方のルータで仮想リンクを設定する必要があります。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14] 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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***area*** *area-id* ***virtual-link*** *router-id*
4. ***show ipv6 ospfv3 virtual-link*** [***brief***]
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**area** *area-id* **virtual-link** *router-id*
	
	例：
	`switch(config-router)# area 10 virtual-link 2001:0DB8::1`
	`switch(config-router-vlink)#`
		リモート ルータへの仮想リンクの端を作成します。仮想リンクをリモート ルータ上に作成して、リンクを完成させる必要があります。

ステップ 4	***show ipv6 ospfv3 virtual-link*** [***brief***] **例：** `switch(config-if)# show ipv6 ospfv3 virtual-link`	（任意）OSPFv3 仮想リンク情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

仮想リンク コンフィギュレーション モードで、省略可能な次のコマンドを設定できます。

コマンドまたはアクション	目的
dead-interval *seconds* **例：** `switch(config-router-vlink)# dead-interval 50`	（任意）OSPFv3 デッド間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 です。デフォルトでは、hello 間隔の秒数の 4 倍です。
hello-interval *seconds* **例：** `switch(config-router-vlink)# hello-interval 25`	（任意）OSPFv3 hello 間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 秒です。
retransmit-interval *seconds* **例：** `switch(config-router-vlink)# retransmit-interval 50`	（任意）OSPFv3 再送間隔を秒単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 5 です。
transmit-delay *seconds* **例：** `switch(config-router-vlink)# transmit-delay 2`	（任意）OSPFv3 送信遅延を秒単位で設定します。有効値の範囲は 1 ～ 450 であり、デフォルトは 1 です。

次に、2 つの ABR 間に簡単な仮想リンクを作成する例を示します。

ABR 1（ルータ ID 2001:0DB8::1）の設定は、次のとおりです。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# area 10 virtual-link 2001:0DB8::10
switch(config-router)# copy running-config startup-config
```

ABR 2（ルータ ID 2001:0DB8::10）の設定は、次のとおりです。

```
switch# config t
switch(config)# router ospf 101
switch(config-router)# area 10 virtual-link 2001:0DB8::1
switch(config-router)# copy running-config startup-config
```

再配布の設定

他のルーティング プロトコルから学習したルートを、ASBR 経由で OSPFv3 AS に再配布できます。

OSPF でのルート再配布には、省略可能な次のパラメータを設定できます。

- Default information originate ― 外部 AS へのデフォルト ルートの AS 外部（タイプ 5）LSA を生成します。
- Default metric ― すべての再配布ルートに同じコスト メトリックを設定します。

操作の前に

再配布で使用する、必要なルートマップを作成します。

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14] 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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***address-family ipv6 unicast***
4. ***redistribute*** {***bgp*** *id* | ***direct*** | ***isis*** *id* | ***rip*** *id* | ***static***} ***route-map*** *map-name*
5. ***default-information originate*** [***always***] [***route-map*** *map-name*]
6. ***default-metric*** *cost*
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 ospfv3** *instance-tag* **例：** switch(config)# router ospfv3 201 switch(config-router)#	新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。
ステップ 3	**address-family ipv6 unicast** **例：** switch(config-router)# address-family ipv6 unicast switch(config-router-af)#	IPv6 ユニキャスト アドレス ファミリ モードを開始します。
ステップ 4	***redistribute*** {***bgp*** *id* | ***direct*** | ***isis*** *id* | ***rip*** *id* | ***static***} ***route-map*** *map-name* **例：** switch(config-router-af)# redistribute bgp route-map FilterExternalBGP	設定したルートマップ経由で、選択したプロトコルを OSPFv3 に再配布します。
ステップ 5	***default-information originate*** [***always***] [***route-map*** map-name] **例：** switch(config-router-af)# default-information-originate route-map DefaultRouteFilter	この OSPFv3 ドメインにデフォルト ルートを作成します。
ステップ 6	**default-metric** *cost* **例：** switch(config-router-af)# default-metric 25	再配布されたルートのコスト メトリックを設定します。有効値の範囲は 1 ～ 16777214 です。
ステップ 7	***copy running-config startup-config*** **例：** switch(config-router)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、Border Gateway Protocol（BGP）を OSPFv3 に再配布する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router)# redistribute bgp route-map FilterExternalBGP
switch(config-router)# copy running-config startup-config
```

ルート集約の設定

集約されたアドレス範囲を設定して、エリア間ルートのルート集約を設定できます。また、ASBR 上のこれらのルートの集約アドレスを設定して、外部の再配布されたルートのルート集約を設定することもできます。「ルート集約」（p.6-11）を参照してください。

操作の前に

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」 [p.6-14] 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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***address-family ipv6 unicast***
4. ***area*** *area-id* ***range*** *ipv6-prefix/length* [***no-advertise***]

   または
5. ***summary-address*** *ipv6-prefix/length* [***no-advertise***] [***tag*** *tag*]
6. ***show ipv6 ospfv3 summary-address***
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**address-family ipv6 unicast**
	
	例：
	`switch(config-router)# address-family ipv6 unicast`
	`switch(config-router-af)#`
		IPv6 ユニキャスト アドレス ファミリ モードを開始します。

ステップ 4	***area*** *area-id* ***range*** *ipv6-prefix/length* [***no-advertise***] **例：** `switch(config-router-af)# area 10 range 2001:0DB8::/48 advertise`	*アドレス範囲に対する集約アドレスを ABR 上に作成します。*この集約アドレスをエリア間プレフィクス（タイプ 3）LSA でアドバタイズすることもできます。
ステップ 5	***summary-address*** *ipv6-prefix/length* [***no-advertise***][***tag*** *tag*] **例：** `switch(config-router-af)# summary-address 2001:0DB8::/48 tag 2`	一定の範囲のアドレスの集約アドレスを ABR 上に作成します。ルートマップによる再配布で使用できるよう、この集約アドレスにタグを割り当てることもできます。
ステップ 6	***show ipv6 ospfv3 summary-address*** **例：** `switch(config-router)# show ipv6 ospfv3 summary-address`	（任意）OSPFv3 集約アドレスに関する情報を表示します。

	コマンド	目的
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ABR 上のエリア間の集約アドレスを作成する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router)# area 10 range 2001:0DB8::/48
switch(config-router)# copy running-config startup-config
```

次に、ABR 上の集約アドレスを作成する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# address-family ipv6 unicast
switch(config-router)# summary-address 2001:0DB8::/48
switch(config-router)# copy running-config startup-config
```

デフォルト タイマーの変更

OSPFv3 には、プロトコル メッセージの動作および SPF 計算を制御する数多くのタイマーが含まれます。OSPFv3 には、省略可能な次のタイマー パラメータが含まれます。

- LSA arrival time ― ネイバーから着信する LSA 間で許容される最小間隔を設定します。この時間より短時間で到着する LSA はドロップされます。
- Pacing LSAs ― LSA が集められてグループ化され、リフレッシュされて、チェックサムが計算される間隔、つまり期限切れとなる間隔を設定します。このタイマーは、LSA 更新が実行される頻度を制御し、LSA 更新メッセージで送信される LSA 更新の数を制御します（「フラッディングと LSA グループ ペーシング」[p.6-7] を参照）。
- Throttle LSAs ― LSA 生成のレート制限を設定します。このタイマーは、トポロジが変更されない場合に LSA が生成される頻度を制御します。
- Throttle SPF calculation ― SPF 計算の実行頻度を制御します。

インターフェイス レベルでは、次のタイマーも制御できます。

- Retransmit interval ― 連続する LSA 間の推定時間間隔を設定します。
- Transmit delay ― LSA をネイバーに送信する推定時間間隔を設定します。

hello 間隔とデッド タイマーに関する情報の詳細については、「OSPFv3 でのネットワークの設定」（p.6-17）を参照してください。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***timers lsa-arrival*** *msec*

4. ***timers lsa-group-pacing*** *seconds*
5. ***timers throttle lsa intra-area-prefix*** *hold-interval*
6. ***timers throttle lsa link*** *hold-interval*
7. ***timers throttle lsa network*** *hold-interval*
8. ***timers throttle lsa router*** *hold-interval*
9. ***address-family ipv6 unicast***
10. ***timers throttle spf*** *delay-time hold-time*
11. ***interface*** *type slot/port*
12. ***ospfv3 retransmit-interval*** *seconds*
13. ***ospfv3 transmit-delay*** *seconds*
14. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**timers lsa-arrival** *msec* **例：** `switch(config-router)# timers lsa-arrival 2000`	LSA 到着時間をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 1000 ミリ秒です。
ステップ 4	**timers lsa-group-pacing** *seconds* **例：** `switch(config-router)# timers lsa-group-pacing 2000`	LSA がグループ化される間隔を秒で設定します。有効値の範囲は 1 ～ 1800 であり、デフォルトは 240 秒です。
ステップ 5	**timers throttle lsa intra-area-prefix** *hold-interval}* **例：** `switch(config-router)# timers throttle lsa intra-area-prefix 3000`	エリア間プレフィクス LSA 生成のレート制限をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 5000 ミリ秒です。
ステップ 6	**timers throttle lsa link** *hold-interval}* **例：** `switch(config-router)# timers throttle lsa intra-area-prefix 3000`	リンク LSA 生成のレート制限をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 5000 ミリ秒です。
ステップ 7	**timers throttle lsa network** *delay-interval* **例：** `switch(config-router)# timers throttle lsa network 3000`	LSA 生成のレート制限をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 5000 ミリ秒です。
ステップ 8	**timers throttle lsa router** *hold-interval}* **例：** `switch(config-router)# timers throttle lsa network 3000`	LSA 生成のレート制限をミリ秒で設定します。有効値の範囲は 10 ～ 600000 であり、デフォルトは 5000 ミリ秒です。

	コマンド	目的

ステップ 9	**address-family ipv6 unicast**
	
	例：
	`switch(config-router)# address-family ipv6 unicast`
	`switch(config-router-af)#`
		IPv6 ユニキャスト アドレス ファミリ モードを開始します。

ステップ 10	**timers throttle spf** *delay-time hold-time* **例：** `switch(config-router)# timers throttle spf 3000 2000`	SPF 最適パス スケジュール初期遅延時間と、各 SPF 最適パス計算間の最小保留時間（秒単位）を設定します。有効値の範囲は 1 ～ 600000 です。デフォルトは、遅延時間なし、および保留時間 5000 ミリ秒です。

ステップ 11	**interface** *type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 12	**ospfv3 retransmit-interval** *seconds* **例：** `switch(config-if)# ospfv3 retransmit-interval 30`	このインターフェイスから送信される各 LSA 間の推定時間間隔を設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 5 です。

ステップ 13	**ospfv3 transmit-delay** *seconds*
	
	例：
	`switch(config-if)# ospfv3 transmit-delay 600`
	`switch(config-if)#`
		LSA をネイバーに送信する推定時間間隔を秒で設定します。有効値の範囲は 1 ～ 450 であり、デフォルトは 1 です。

ステップ 14	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、lsa-group-pacing オプションで LSA フラッディングを制御する例を示します。

```
switch# config t
switch(config)# router ospf 201
switch(config-router)# timers lsa-group-pacing 300
switch(config-router)# copy running-config startup-config
```

グレースフル リスタートの設定

グレースフル リスタートは、デフォルトでイネーブルにされています。OSPFv3 インスタンスのグレースフル リスタートには、省略可能な次のパラメータを設定できます。

- Grace period ― グレースフル リスタートの開始後に、ネイバーが隣接関係を解消するまでに待つ時間を設定します。
- Helper mode disabled ― ローカル OSPFv3 インスタンスのヘルパー モードをディセーブルにします。OSPFv3 は、ネイバーのグレースフル リスタートには関与しません。
- Planned graceful restart only ― 予定された再起動の場合にのみグレースフル リスタートがサポートされるよう、OSPFv3 を設定します。

操作の前に

OSPFv3 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」[p.6-14] を参照）。

すべてのネイバーで、一致した省略可能なパラメータ一式とともにグレースフル リスタートが設定されていることを確認します。

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）。

コマンドの一覧

1. **config t**
2. **router ospfv3** *instance-tag*
3. ***graceful-restart***
4. ***graceful-restart grace-period*** *seconds*
5. ***graceful-restart helper-disable***
6. ***graceful-restart planned-only***
7. ***show ipv6 ospfv3*** *instance-tag*
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 ospfv3** *instance-tag*
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 3	**graceful-restart** **例：** `switch(config-router)# graceful-restart`	グレースフル リスタートをイネーブルにします。グレースフル リスタートは、デフォルトでイネーブルにされています。
ステップ 4	**graceful-restart grace-period** *seconds* **例：** `switch(config-router)# graceful-restart grace-period 120`	猶予期間を秒で設定します。有効値の範囲は 5 ～ 1800 であり、デフォルトは 60 秒です。
ステップ 5	**graceful-restart helper-disable** **例：** `switch(config-router)# graceful-restart helper-disable`	ヘルパー モードをディセーブルにします。デフォルトではイネーブルにされています。
ステップ 6	**graceful-restart planned-only** **例：** `switch(config-router)# graceful-restart planned-only`	予定された再起動時にのみグレースフル リスタートを設定します。
ステップ 7	***show ipv6 ospfv3*** *instance-tag* **例：** `switch(config-if)# show ipv6 ospfv3 201`	（任意）OSPFv3 情報を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、ディセーブルにされているグレースフル リスタートをイネーブルにし、猶予期間を 120 秒に設定する例を示します。

```
switch# config t
switch(config)# router ospfv3 201
switch(config-router)# graceful-restart
switch(config-router)# graceful-restart grace-period 120
switch(config-router)# copy running-config startup-config
```

OSPFv3 インスタンスの再起動

OSPv3 インスタンスを再起動できます。再起動すると、インスタンスのすべてのネイバーが解消されます。

OSPFv3 インスタンスを再起動して、関連付けられたすべてのネイバーを削除するには、次のコマンドを使用します。

コマンド	目的
restart ospfv3 instance-tag **例：** switch(config)# restart ospfv3 201	OSPFv3 インスタンスを再起動して、すべてのネイバーを削除します。

仮想化による OSPFv3 の設定

各 VDC で複数の OSPFv3 インスタンスを設定できます。各 VDC 内に複数の VRF を作成して、各 VRF で同じまたは複数の OSPFv3 インスタンスを使用することもできます。VRF には OSPFv3 インターフェイスを割り当てます。

（注） インターフェイスの VRF を設定したあとに、インターフェイスの他のすべてのパラメータを設定します。インターフェイスの VRF を設定すると、そのインターフェイスのすべての設定が削除されます。

操作の前に

VDC を作成します。

OSPF 機能がイネーブルにされていることを確認します（「OSPFv3 機能のイネーブル化」 [p.6-14] 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）。

コマンドの一覧

1. **config t**
2. ***vrf* context** *vrf_name*
3. **router ospfv3** *instance-tag*
4. ***vrf*** *vrf-name*
5. <optional parameters configured>
6. ***interface*** *type slot/port*

7. ***vrf member*** *vrf-name*
8. ***ipv6 address*** *ipv6-prefix/length*
9. ***ipv6 ospfv3*** *instance-tag* ***area*** *area-id*
10. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 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	**router ospfv3** *instance-tag*
	例：
	`switch(config)# router ospfv3 201`
	`switch(config-router)#`
		新規 OSPFv3 インスタンスを作成して、設定済みのインスタンス タグを割り当てます。

ステップ 4	**vrf** *vrf-name*
	例：
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		VRF 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	**maximum-paths** *paths* **例：** `switch(config-router-vrf)# maximum-paths 4`	（任意）この VRF のルーティング テーブル内の宛先への、同じ OSPFv3 パスの最大数を設定します。ロード バランシングに使用されます。

ステップ 6	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 7	**vrf member** *vrf-name* **例：** `switch(config-if)# vrf member RemoteOfficeVRF`	このインターフェイスを VRF に追加します。
ステップ 8	**ipv6 address** *ipv6-prefix/length* **例：** `switch(config-if)# ipv6 address 2001:0DB8::1/48`	このインターフェイスの IP アドレスを設定します。この手順は、このインターフェイスを VRF に割り当てたあとに行う必要があります。
ステップ 9	**ipv6 ospfv3** *instance-tag* **area** *area-id* **例：** `switch(config-if)# ipv6 ospfv3 201 area 0`	設定した OSPFv3 インスタンスおよびエリアに、このインターフェイスを割り当てます。
ステップ 10	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、VRF を作成して、その VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context NewVRF
switch(config-vrf)# exit
switch(config)# router ospfv3 201
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# vrf member NewVRF
switch(config-if)# ipv6 address 2001:0DB8::1/48
switch(config-if)# ipv6 ospfv3 201 area 0
switch(config-if)# copy running-config startup-config
```

OSPFv3 設定の確認

OSPFv3 設定情報を確認するには、次のコマンドのいずれかを使用します。

コマンド	目的
show ipv6 ospfv3	OSPFv3 設定を表示します。
show ipv6 ospfv3 border-routers	ABR および ASBR への内部 OSPF ルーティング テーブル エントリを表示します。
show ipv6 ospfv3 database	特定のルータの OSPFv3 データベースに関係する情報の一覧を表示します。
show ipv6 ospfv3 interface *type number* [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv3 インターフェイス設定を表示します。
show ipv6 ospfv3 neighbors	ネイバー情報を表示します。***clear ospfv3 neighbors*** コマンドを使用すると、すべてのネイバーとの隣接関係を削除できます。
show ipv6 ospfv3 request-list	ルータから要求されている LSA の一覧を表示します。
show ipv6 ospfv3 retransmission-list	再送を待っている LSA の一覧を表示します。
show ipv6 ospfv3 summary-address	OSPFv3 インスタンスで設定されている、すべての集約アドレス再配布情報の一覧を表示します。
show running-configuration ospfv3	現在実行中の OSPFv3 設定を表示します。

OSPFv3 統計情報の表示

OSPFv3 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 ipv6 ospfv3 memory	OSPFv3 メモリ使用状況の統計情報を表示します。
show ipv6 ospfv3 policy statistics area *area-id* **filter-list** {**in** | **out**} [**vrf** {*vrf-name* | **all** | **default** | **management**}]	エリアの OSPFv3 ルート ポリシー統計情報を表示します。
show ipv6 ospfv3 policy statistics redistribute {**bgp** *id*| **direct** | **isis** *id* | **rip** *id* | **static**} **vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv3 ルート ポリシー統計を表示します。
show ipv6 ospfv3 statistics [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv3 イベント カウンタを表示します。
show ipv6 ospfv3 traffic [*interface-type number*] [**vrf** {*vrf-name* | **all** | **default** | **management**}]	OSPFv3 パケット カウンタを表示します。

OSPFv3 設定の例

次に、OSPFv3 を設定する例を示します。

```
feature ospfv3
router ospfv3 201
 router-id 290.0.2.1

interface ethernet 1/2
 ipv6 address 2001:0DB8::1/48
 ipv6 ospfv3 201 area 10
```

関連資料

次の項目には、OSPF に関する詳細情報が含まれています。

- 第 5 章「OSPFv2 の設定」
- 第 14 章「Route Policy Manager の設定」

デフォルト設定

表 6-2 は、OSPFv3 パラメータのデフォルト設定の一覧です。

表 6-2　デフォルト OSPFv3 パラメータ

パラメータ	デフォルト
Hello interval	10 秒
Dead interval	40 秒
Graceful restart grace period	60 秒
Graceful restart notify period	15 秒
OSPFv3 feature	ディセーブル
Stub router advertisement announce time	600 秒
Reference bandwidth for link cost calculation	40 Gbps
LSA minimal arrival time	1000 ミリ秒
LSA group pacing	240 秒
SPF calculation initial delay time	0 ミリ秒
SPF calculation hold time	5000 ミリ秒
SPF calculation initial delay time	0 ミリ秒

その他の関連資料

OSPF の実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.6-39）
- MIB（p.6-39）

関連資料

関連項目	マニュアル名
OSPFv3 CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

MIB

MIB	MIB のリンク
• OSPF-MIB • OSPF-TRAP-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CHAPTER 7

EIGRP の設定

この章では、Enhanced Interior Gateway Routing Protocol（*EIGRP*）を設定する方法について説明します。

ここでは、次の内容を説明します。

- EIGRP について（p.7-2）
- EIGRP のライセンス要件（p.7-9）
- EIGRP の前提条件（p.7-9）
- 設定時の注意事項および制約事項（p.7-9）
- 基本的 EIGRP の設定（p.7-10）
- 高度な EIGRP の設定（p.7-14）
- 高度な EIGRP の設定（p.7-14）
- EIGRP 統計情報の表示（p.7-26）
- EIGRP 設定の例（p.7-26）
- デフォルト設定（p.7-27）
- その他の関連資料（p.7-28）

EIGRP について

EIGRP は、リンクステート プロトコルの機能にディスタンス ベクトル プロトコルの利点を組み合わせたプロトコルです。EIGRP は、定期的に Hello メッセージを送信して近隣探索を行います。EIGRP は、新規ネイバーを検出すると、すべてのローカル EIGRP ルートおよびルート メトリックに対する一回限りの更新を送信します。受信側の EIGRP ルータは、受信したメトリックと、その新規ネイバーにローカルで割り当てられたリンクのコストに基づいて、ルート ディスタンスを計算します。この最初の全面的なルーティング テーブルの更新後は、ルート変更の影響を受けるネイバーにのみ、差分更新が EIGRP により送信されます。この処理により、収束にかかる時間が短縮され、EIGRP が使用する帯域幅が最小限になります。

ここでは、次の内容について説明します。

- EIGRP のコンポーネント（p.7-2）
- EIGRP ルート更新（p.7-3）
- 高度な EIGRP（p.7-4）

EIGRP のコンポーネント

EIGRP には、次の基本コンポーネントがあります。

- Reliable Transport Protocol（p.7-2）
- 近隣探索および近隣回復（p.7-2）
- DUAL（p.7-3）

Reliable Transport Protocol

*リライアブル トランスポート プロトコル*により、すべてのネイバーへの EIGRP パケットの配信が保証されます。（「近隣探索および近隣回復」[p.7-2] を参照）。Reliable Transport Protocol は、マルチキャスト パケットとユニキャスト パケットの混合伝送をサポートしています。高信頼性転送では、未確認パケットが保留されているときにも、マルチキャスト パケットの迅速な送信が可能です。この方式により、さまざまな速度のリンクでも短い収束時間が維持されるようになります。マルチキャスト パケットとユニキャスト パケットの送信を制御するデフォルト タイマーの変更の詳細については、「高度な EIGRP の設定」（p.7-14）を参照してください。

Reliable Transport Protocol には、次のメッセージ タイプが含まれます。

- Hello ― 近隣探索および近隣回復に使用されます。EIGRP はデフォルトでは、定期的なマルチキャスト Hello メッセージをローカル ネットワーク上に、設定された *hello インターバル*で送信します。デフォルトの hello 間隔は 5 秒です。
- 確認 ― 更新、照会、返信を確実に受信したことを確認します。
- 更新 ― ルーティング情報が変更されると、その影響を受けるネイバーに送信されます。更新には、ルートの宛先、アドレス マスク、および遅延や帯域幅などのルート メトリックが含まれます。更新情報は EIGRP トポロジ テーブルに格納されます。
- 照会および返信 ― 必要に応じて、EIGRP が使用する DUAL の一部として送信されます。

近隣探索および近隣回復

EIGRP は、Reliable Transport Protocol からの Hello メッセージを使用して、直接接続されたネットワーク上の隣接 EIGRP ルータを探索します。EIGRP により、ネイバー テーブルにネイバーが追加されます。ネイバー テーブルの情報には、ネイバーのアドレス、ネイバーが認識されたインターフェイス、および、ネイバーが到達不能であると宣言するまでの EIGRP の待機時間である*ホールド タイム*が含まれます。デフォルトの保留時間は、hello 間隔の 3 倍または 15 秒です。

EIGRP は、ローカル EIGRP ルーティング情報を共有するために、一連の更新メッセージを新規ネイバーに送信します。このルート情報は EIGRP トポロジ テーブルに格納されます。このように EIGRP ルート情報全体を最初に送信したあとは、ルーティングが変更されたときにのみ、EIGRP により更新メッセージが送信されます。これらの更新メッセージは新情報または更新情報のみを含んでおり、変更の影響を受けるネイバーにのみ送信されます。「EIGRP ルート更新」（p.7-3）を参照してください。

EIGRP はネイバーへのキープアライブとして、Hello メッセージも使用します。Hello メッセージを受信している限り、Cisco DC-OS は、ネイバーがダウンせずに機能していると判定します。

DUAL

Diffusing アップデート アルゴリズム（DUAL）により、トポロジ テーブルの宛先ネットワークに基づいてルーティング情報が計算されます。トポロジ テーブルには、次の情報が含まれます。

- IP アドレス / マスク — この宛先のネットワーク アドレスおよびネットワーク マスク
- サクセサ — すべての*フィージブル サクセッサ*または、現在の*フィージブル ディスタンス*より短いディスタンスをアドバタイズするネイバーの IP アドレスおよびローカル インターフェイス接続
- Feasibility Distance（FD; フィージビリティ ディスタンス）— 計算された、宛先までの最短ディスタンス。FD は、ネイバーからアドバタイズされたディスタンスの合計に、そのネイバーまでのリンクのコストを加えた値です。

DUAL は、ディスタンス メトリックを使用して、ループが発生しない効率的なパスを選択します。DUAL はルートを選択し、フィージブル サクセサに基づいてユニキャスト Routing Information Base（RIB）に挿入します。トポロジが変更されると、DUAL は、トポロジ テーブルでフィージブル サクセサを探します。フィージブル サクセサが見つかった場合、DUAL は、最短のフィージブル ディスタンスを持つフィージブル サクセサを選択して、それをユニキャスト RIB に挿入します。これにより、再計算が不要となります。

フィージブル サクセサが存在しないが、宛先をアドバタイズするネイバーが存在する場合は、DUAL がパッシブ状態からアクティブ状態へと移行し、新しいサクセサまたは宛先へのネクストホップ ルータを決定する再計算をトリガーします。ルートの再計算に必要な時間は収束時間に影響します。EIGRP は照会メッセージをすべてのネイバーに送信し、フィージブル サクセサを探します。フィージブル サクセサを持つネイバーは、その情報を含む返信メッセージを送信します。フィージブル サクセサを持たないネイバーは、DUAL の再計算をトリガーします。

EIGRP ルート更新

トポロジが変更されると、EIGRP は、変更されたルーティング情報のみを含む更新メッセージを、影響を受けるネイバーに送信します。更新メッセージには、新規の、または更新されたネットワーク宛先へのディスタンス情報が含まれます。

EIGRP でのディスタンス情報は、帯域幅、遅延、負荷使用状況、リンクの信頼性などの使用可能なルート メトリックの組み合わせとして表現されます。各メトリックには重みが関連付けられており、これにより、メトリックがディスタンスの計算に含まれるかどうかが決定します。このメトリックの重みは設定するこができます。特性を微調整して最適なパスを完成することもできますが、設定可能なメトリックの大部分でデフォルト設定を使用することをお勧めします。

ここでは、次の内容について説明します。

- 内部ルート メトリック（p.7-4）
- 外部ルート メトリック（p.7-4）
- EIGRP とユニキャスト RIB（p.7-4）

内部ルート メトリック

内部ルートとは、同じ EIGRP Autonomous System（AS; 自律システム）内のネイバー間のルートです。これらのルートには、次のメトリックがあります。

- ネクストホップ — ネクストホップ ルータの IP アドレス
- 遅延 — 宛先ネットワークへのルートを形成するインターフェイス上で設定された遅延の合計。10 マイクロ秒単位で設定されます。
- 帯域幅 — 宛先へのルートの一部であるインターフェイスで設定された最小帯域幅から計算されます。

（注） デフォルト帯域幅の値の使用をお勧めします。この帯域幅パラメータは EIGRP でも使用されます。

- MTU — 宛先へのルート上の最大伝送単位の最小値
- ホップ カウント — 宛先までにルートが通過するホップまたはルータの数。このメトリックは、DUAL 計算で直接には使用されません。
- 信頼性 — 宛先までのリンクの信頼性を示します。
- 負荷 — 宛先までのリンク上のトラフィックの量を示します。

デフォルトで EIGRP は、帯域幅と遅延のメトリックを使用して、宛先までのディスタンスを計算します。計算に他のメトリックが含まれるよう、メトリックの重みを変更できます。

外部ルート メトリック

外部ルートとは、異なる EIGRP AS にあるネイバー間のルートです。これらのルートには、次のメトリックがあります。

- ネクストホップ — ネクストホップ ルータの IP アドレス
- ルータ ID — このルートを EIGRP に再配布したルータのルータ ID
- AS 番号 — 宛先の AS の番号
- プロトコル ID — 宛先へのルートを学習したルーティング プロトコルを表すコード
- タグ — ルートマップで使用可能な任意のタグ
- メトリック — 外部ルーティング プロトコルの、このルートのルート メトリック

EIGRP とユニキャスト RIB

EIGRP は、学習したルートをすべて、EIGRP トポロジ テーブルとユニキャスト RIB に追加します。トポロジが変更されると、EIGRP は、これらのルートを使用してフィージブル サクセサを探します。EIGRP は、他のルーティング プロトコルから EIGRP に再配布されたあらゆるルートの変更についてのユニキャスト RIB からの通知も待ち受けます。

高度な EIGRP

EIGRP の高度な機能を使用して、EIGRP の設定を最適化できます。

ここでは、次の内容について説明します。

- 認証（p.7-5）
- スタブ ルータ（p.7-5）

- ルート集約（p.7-5）
- ルート再配布（p.7-6）
- ロード バランシング（p.7-6）
- スプリット ホライズン（p.7-6）
- 仮想化のサポート（p.7-7）
- グレースフル リスタートとハイ アベイラビリティ（p.7-7）

認証

EIGRP メッセージに認証を設定して、ネットワークでの不正な、または無効なルーティング更新を防止できます。Cisco NX-OS は MD5 認証ダイジェストをサポートしています。

認証キーのキーチェーン管理を使用して、インターフェイスごとに EIGRP 認証を設定できます。キーチェーン管理を使用すると、MD5 認証ダイジェストが使用する認証キーへの変更を管理できます。キーチェーン作成の詳細については、『*Cisco NX-OS Security Configuration Guide*』を参照してください。

MD5 認証を行うには、ローカル ルータとすべてのリモート EIGRP ネイバーが共有するパスワードを設定します。EIGRP メッセージが作成されると、Cisco NX-OS は、そのメッセージ自体と暗号化されたパスワードに基づいて MD5 一方向メッセージ ダイジェストを作成し、このダイジェストを EIGRP メッセージとともに送信します。受信する EIGRP ネイバーは、同じ暗号化パスワードを使用して、このダイジェストを確認します。メッセージが変更されていない場合は計算が同一であるため、EIGRP メッセージは有効と見なされます。

MD5 認証には各 EIGRP メッセージのシーケンス番号も含まれており、これにより、ネットワークでのメッセージの再送が防止されます。

スタブ ルータ

EIGRP スタブ ルーティング機能を使用して、ネットワークの安定性を向上させ、リソースの使用を削減し、スタブ ルータ設定を簡素化することができます。スタブ ルータは、リモート ルータ経由で EIGRP ネットワークに接続します。「スタブ ルーティング」（p.1-7）を参照してください。

EIGRP スタブ ルーティングを使用すると、EIGRP を使用するよう分散ルータとリモート ルータを設定し、リモート ルータのみをスタブとして設定する必要があります。EIGRP スタブ ルーティングで、分散ルータでの集約が自動的にイネーブルになるわけではありません。ほとんどの場合、分散ルータでの集約の設定が必要です。

EIGRP スタブ ルーティングを使用しない場合は、分散ルータからリモート ルータに送信されたルートがフィルタリングまたは集約された後でも、問題が発生することがあります。たとえば、ルートが企業ネットワーク内のどこかで失われた場合に、EIGRP が分散ルータに照会を送信することがあります。分散ルータは、ルートが集約されている場合でも、リモート ルータに照会を送信することがあります。分散ルータとリモート ルータの間の WAN リンク上の通信に問題が発生した場合は、EIGRP がアクティブ状態のままとなり、ネットワークの他の場所が不安定となる場合があります。EIGRP スタブ ルーティングを使用すると、リモート ルータに照会が送信されなくなります。

ルート集約

指定したインターフェイスに集約アドレスを設定できます。ルート集約により、多くのより具体的なアドレスが、すべての具体的なアドレスを表す 1 つのアドレスに置き換えられるため、ルーティング テーブルが簡素化されます。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。

より具体的なアドレスがルーティング テーブルにある場合、EIGRP は、より具体的なルートの最小メトリックに等しいメトリックを持つインターフェイスからの集約アドレスをアドバタイズします。

（注）　Cisco NX-OS は、自動ルート集約をサポートしていません。

ルート再配布

EIGRP を使用して、スタティック ルート、他の EIGRP AS から学習したルート、または他のプロトコルからのルートを再配布できます。再配布を含むルート ポリシーを設定して、どのルートが EIGRP に渡されるかを制御します。ルート ポリシーを使用すると、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。第 14 章「Route Policy Manager の設定」を参照してください。

インポートされた EIGRP へのすべてのルートに使用されるデフォルト メトリックも設定できます。

ロード バランシング

ロード バランシングを使用すると、ルータが、宛先アドレスから同じディスタンスのすべてのルータ ネットワーク ポートにトラフィックを配布することが可能になります。ロード バランシングにより、ネットワーク セグメントの使用率が向上し、それによってネットワーク帯域幅の効率も向上します。

Cisco NX-OS は、EIGRP ルーティング テーブルおよびユニキャスト RIB 中の 16 までの等コスト パスを使用する Equal Cost Multiple Path（ECMP; 等コスト マルチパス）機能をサポートしています。これらのパスの一部または全部に対してトラフィックのロード バランシングを行うよう、EIGRP を設定できます。

スプリット ホライズン

スプリット ホライズンを使用して、EIGRP が、ルートを伝えたインターフェイスからそのルートをアドバタイズしないようにすることができます。

スプリット ホライズンは、EIGRP 更新パケットおよび EIGRP 照会パケットの送信を制御する方式です。インターフェイスでスプリット ホライズンをイネーブルにすると、Cisco NX-OS は、このインターフェイスから学習された宛先への更新パケットも照会パケットも送信しません。この方法で更新パケットと照会パケットを制御すると、ルーティング ループが発生する可能性が低下します。

ポイズン リバースによるスプリット ホライズンにより、EIGRP は、EIGRP がルートを学習したインターフェイス経由で、そのルートを到達不能としてアドバタイズするよう設定されます。

EIGRP は、次のシナリオでスプリット ホライズン、またはポイズン リバースによるスプリット ホライズンを使用します。

- スタートアップ モードで、2 台のルータ間で初めてトポロジ テーブルを交換する。
- トポロジ テーブルの変更をアドバタイズする。
- 照会メッセージを送信する。

デフォルトでは、スプリット ホライズン機能がすべてのインターフェイスでイネーブルになっています。

仮想化のサポート

Cisco NX-OS は、同じシステム上で動作する、EIGRP プロトコルの複数インスタンスをサポートしています。EIGRP は、Virtual Routing and Forwarding Instance（VRF; 仮想ルーティング / 転送インスタンス）をサポートしています。VRF は Virtual Device Contexts（VDC; 仮想化デバイス コンテキスト）内にあります。デフォルトでは、特に別の VDC および VRF を設定しない限り、Cisco NX-OS によりデフォルト VDC およびデフォルト VRF が使用されます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

デフォルトでは、すべてのインスタンスが同じシステム ルータ ID を使用します。複数のインスタンスが同じ EIGRP AS にある場合は、各インスタンスのルータ ID を手動で設定する必要があります。

グレースフル リスタートとハイ アベイラビリティ

Cisco NX-OS は、EIGRP の Nonstop Forwarding（NSF）とグレースフル リスタートをサポートしています。

EIGRP の NSF を使用すると、フェールオーバー後に EIGRP ルーティング プロトコル情報が復元される間に、データ パケットを FIB 内の既存のルートで転送できます。NSF では、ピア ネットワーキング デバイスにルーティング フラップが発生しません。フェールオーバー中は、スタンバイ スーパーバイザがアクティブになる間に、データ トラフィックがインテリジェント モジュール経由で転送されます。

Cisco NX-OS システムでコールド リブートが発生した場合、ネットワークはシステムにトラフィックを転送せず、システムをネットワーク トポロジから削除します。このシナリオでは、EIGRP でステートレス リスタートが行われ、すべてのネイバーは削除されます。Cisco NX-OS はスタートアップ コンフィギュレーションを適用し、EIGRP は再度近隣探索を行い、EIGRP ルーティング情報全体を再度共有します。

Cisco NX-OS を実行するデュアル スーパーバイザ プラットフォームでは、ステートフル スーパーバイザ スイッチオーバーが発生する場合があります。このスイッチオーバーが発生する前に、EIGRP はグレースフル リスタートを使用して、EIGRP がしばらく使用不可であることを宣言します。スイッチオーバー中に EIGRP は NSF を使用して、FIB 中の情報に基づいてトラフィックの転送を続行します。また、システムはネットワーク トポロジから排除されません。

グレースフル リスタート対応のルータは、Hello メッセージを使用して、そのネイバーにグレースフル リスタート動作が開始されたことを通知します。グレースフル リスタート認識ルータがグレースフル リスタート対応のネイバーから、グレースフル リスタート動作が進行中であるという通知を受信すると、両方のルータはただちにそのトポロジ テーブルを交換します。グレースフル リスタート認識ルータは、その後、次の動作を行って再起動中のルータを支援します。

- ルータは EIGRP Hello 保留タイマーを時間切れにして、設定された Hello メッセージの時間間隔を短縮します。これにより、グレースフル リスタート認識ルータが再起動中のルータにすばやく返信できるようになり、再起動中のルータが再度近隣探索を行って、トポロジを再構築するために必要な時間が短縮されます。
- ルータは、ルート保留タイマーを開始します。このタイマーで、グレースフル リスタート認識ルータが、再起動中のルータのために既知のルートを保留する時間の長さが設定されます。デフォルトの時間は 240 秒です。
- ルータはピア　リストに隣接ルータが再起動中であることを記し、隣接関係を維持します。また、再起動中の隣接ルータが、グレースフル リスタート認識ルータから送信されるトポロジ テーブルを受信可能であるという合図をするか、ルート保留タイマーが切れるまで、ルートを保留します。グレースフル リスタート認識ルータでルート保留タイマーが切れた場合は、グレースフル リスタート認識ルータは保留ルートを廃棄し、再起動中のルータをネットワークに追加された新規ルータとして取り扱い、隣接関係を再度確立します。

スイッチオーバー後に、Cisco NX-OS は実行コンフィギュレーションを適用し、EIGRP は、自身が再び稼働していることをネイバーに通知します。

（注） グレースフル リスタートでは、EIGRP の In-Service Software Upgrade（ISSU; インサービス ソフトウェア アップグレード）のサポートをイネーブルにする必要があります。グレースフル リスタートをディセーブルにした場合は、Cisco NX-OS により、この設定では ISSU をサポートできないという警告が発行されます。

EIGRP のライセンス要件

次の表は、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	EIGRP には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細説明およびライセンスの入手と適用の方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

EIGRP の前提条件

EIGRP には、次の前提条件があります。

- EIGRP 機能がイネーブルにされている（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、目的の VDC を入力します（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

EIGRP 設定時の注意事項および制約事項は次のとおりです。

- 他のプロトコル、接続されたルータ、またはスタティック ルートからの再配布には、メトリック設定（デフォルト メトリック設定オプションまたはルート ポリシーによる）が必要です（第 14 章「Route Policy Manager の設定」を参照）。
- グレースフル スタートについては、NSF 認識ルータが動作中であり、ネットワークで完全に収束している場合にのみ、このルータが NSF 対応ルータのグレースフル リスタート動作を支援できます。
- グレースフル リスタートについては、グレースフル リスタートに関与する隣接デバイスが NSF 認識または NSF 対応である必要があります。
- Cisco NX-OS EIGRP には Cisco IOS の EIGRP との互換性があります。
- 妥当な理由がない限り、メトリックの重みを変更しないでください。メトリックの重みを変更した場合は、同じ AS 内のすべての EIGRP ルータに、それを適用する必要があります。
- 大規模ネットワークの場合は、スタブの使用を検討してください。
- EIGRP ベクトル メトリックは維持されないため、異なる EIGRP AS 間での再配布は避けてください。
- **no ip next-hop-self** コマンドでは、ネクストホップへの到達可能性は保証されていません。
- **ip passive-interface eigrp** コマンドを使用すると、ネイバーが形成されなくなります。
- Cisco NX-OS は IGRP も、IGRP および EIGRP クラウドの接続もサポートしていません。
- 自動集約は、デフォルトではイネーブルにされていません。
- Cisco NX-OS は IP のみをサポートしています。

（注）　Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

基本的 EIGRP の設定

ここでは、次の内容について説明します。

- EIGRP 機能のイネーブル化（p.7-10）
- EIGRP インスタンスの作成（p.7-11）
- EIGRP インスタンスの再起動（p.7-13）
- EIGRP インスタンスのディセーブル化（p.7-13）
- インターフェイスでの EIGRP のディセーブル化（p.7-13）

EIGRP 機能のイネーブル化

EIGRP を設定するには、その前に EIGRP 機能をイネーブルにする必要があります。

操作の前に

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）。

コマンドの一覧

1. **config t**
2. ***feature eigrp***
3. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature eigrp** **例：** `switch(config)# feature eigrp`	EIGRP 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

EIGRP 機能をディセーブルにし、関連付けられた設定をすべて削除するには、***no feature eigrp*** コマンドを使用します。

コマンド	目的
no feature eigrp **例：** `switch(config)# no feature eigrp`	EIGRP 機能をディセーブルにして、関連付けられたコンフィギュレーションをすべて削除します。

EIGRP インスタンスの作成

EIGRP インスタンスを作成して、そのインスタンスにインターフェイスを関連付けることができます。この EIGRP プロセスに一意の AS 番号を割り当てます（「自律システム」[p.1-5] を参照）。ルート再配布をイネーブルにしていない限り、外部 AS からルートがアドバタイズされることも、受信されることもありません。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。

EIGRP がルータ ID（設定済みのループバック アドレスなど）を入手可能であるか、またはルータ ID オプションを設定する必要があります。

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）。

コマンドの一覧

1. **config t**
2. **router eigrp** *as-number*
3. <configure optional parameters>
4. ***interface*** *interface-type slot/port*
5. ***ip router eigrp*** *as-number*
6. ***show ip eigrp interfaces***
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 eigrp** *as-number*
	
	例：
	`switch(config)# router eigrp 201`
	`switch(config-router)#`
		新しい EIGRP プロセスを作成し、それに AS 番号を設定します。

ステップ 3	**eigrp router-id** *ip-address* **例：** `switch(config-router)# eigrp router-id 192.0.2.1`	（任意）EIGRP ルータ ID を設定します。この IP アドレスにより、この EIGRP インスタンスが識別されます。このアドレスは、システムの設定済みインターフェイス上に存在する必要があります。
	router-id *ip-address* **例：** `switch(config-router)# router-id 192.0.2.1`	（任意）EIGRP ルータ ID を設定します。このコマンドは、***eigrp router-id*** コマンドと同じです。

	コマンド	目的
ステップ 4	**eigrp log-neighbor-changes** **例：** `switch(config-router)# eigrp log-neighbor-changes`	（任意）ネイバーの状態が変化するたびに、システム メッセージを生成します。このコマンドは、デフォルトでイネーブルにされています。
ステップ 5	**eigrp log-neighbor-warnings** [*seconds*] **例：** `switch(config-router)# eigrp log-neighbor-warnings`	（任意）ネイバー警告が発生するたびに、システム メッセージを生成します。警告メッセージの時間間隔を、1 ～ 65535 の秒数で設定できます。デフォルト値は 10 秒です。このコマンドは、デフォルトでイネーブルにされています。

ステップ 6	**interface** *interface-type slot/port*
	
	例：
	`switch(config-router)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 7	**ip router eigrp** *as-number* **例：** `switch(config-if)# ip router eigrp 201`	このインターフェイスを、設定された EIGRP プロセスに関連付けます。
ステップ 8	**show ip eigrp interfaces** **例：** `switch(config-if)# show ip eigrp interfaces`	EIGRP インターフェイスに関する情報を表示します。
ステップ 9	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

EIGRP プロセスと、関連付けられた設定を削除するには、***no router eigrp*** コマンドを使用します。

コマンド	目的
no router eigrp *as-number* **例：** `switch(config)# no router eigrp 201`	EIGRP プロセスと、関連付けられたすべての設定を削除します。

（注） インターフェイス モードで設定された EIGRP コマンドもすべて、手動で削除する必要があります。

次に、EIGRP プロセスを作成し、EIGRP のインターフェイスを設定する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config)# interface ethernet 1/2
switch(config-if)# ip router eigrp 201
switch(config-if)# no shutdown
switch(config-if)# copy running-config startup-config
```

その他の EIGRP パラメータの詳細については、「高度な EIGRP の設定」（p.7-14）を参照してください。

EIGRP インスタンスの再起動

EIGRP インスタンスを再起動できます。再起動すると、インスタンスのすべてのネイバーが解消されます。

EIGRP インスタンスを再起動して、関連付けられたすべてのネイバーを削除するには、次のコマンドを使用します。

コマンド	目的
restart eigrp instance-tag **例：** switch(config)# restart eigrp 201	EIGRP インスタンスを再起動して、すべてのネイバーを削除します。

EIGRP インスタンスのディセーブル化

EIGRP インスタンス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。

コマンド	目的
switch(config-router)# **shutdown** **例：** switch(config-router)# shutdown	この EIGRP インスタンスをディセーブルにします。

インターフェイスでの EIGRP のディセーブル化

インターフェイスで EIGRP をディセーブルにするには、インターフェイス 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。

コマンド	目的
switch(config-if)# **ip eigrp as-numver shutdown** **例：** switch(config-router)# ip eigrp 201 shutdown	このインターフェイスで EIGRP をディセーブルにします。

高度な EIGRP の設定

ここでは、次の内容について説明します。

- EIGRP での認証の設定（p.7-14）
- EIGRP スタブ ルーティングの設定（p.7-16）
- EIGRP の集約アドレスの設定（p.7-16）
- EIGRP へのルート再配布（p.7-17）
- EIGRP でのロード バランシングの設定（p.7-18）
- EIGRP のグレースフル リスタートの設定（p.7-19）
- hello パケットの間隔と保留時間の調整（p.7-21）
- スプリット ホライズンのディセーブル化（p.7-21）
- EIGRP の調整（p.7-22）
- EIGRP の仮想化の設定（p.7-24）

EIGRP での認証の設定

EIGRP のネイバー間での認証を設定できます。「認証」（p.7-5）を参照してください。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。

EIGRP プロセスのすべてのネイバーが、共有認証キーを含め、同じ認証設定を共有することを確認します。

この認証設定のキーチェーンを作成します。詳細については『*Cisco NX-OS Security Configuration Guide*』を参照してください。

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）。

コマンドの一覧

1. **config t**
2. **router eigrp** *as-number*
3. ***interface*** *interface-type slot/port*
4. ***ip router eigrp*** *as-number*
5. **ip authentication *key-chain eigrp*** *as-number key-chain*
6. ***ip authentication mode eigrp*** *as-number* ***md5***
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 eigrp** *as-number*
	
	例：
	`switch(config)# router eigrp 201`
	`switch(config-router)#`
		新しい EIGRP プロセスを作成し、それに AS 番号を設定します。

ステップ 3	**interface** *interface-type slot/port*
	
	例：
	`switch(config)interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**ip router eigrp** *as-number* **例：** `switch(config-if)# ip router eigrp 201`	このインターフェイスを、設定された EIGRP プロセスに関連付けます。
ステップ 5	**ip authentication key-chain *eigrp*** *as-number key-chain* **例：** `switch(config-if)# ip authentication key-chain eigrp 201 routeKeys`	このインターフェイスの EIGRP プロセスにキーチェーンを関連付けます。
ステップ 6	**ip authentication mode *eigrp*** *as-number* ***md5*** **例：** `switch(config-if)# ip authentication mode eigrp 201 md5`	このインターフェイスの MD5 メッセージ ダイジェスト認証モードを設定します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、EIGRP の MD5 メッセージ ダイジェスト認証をイーサネット インターフェイス 1/2 上で設定する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# ip router eigrp 201
switch(config-if)# ip authentication key-chain eigrp 201 routeKeys
switch(config-if)# ip authentication mode eigrp 201 md5
switch(config-if)# copy running-config startup-config
```

EIGRP スタブ ルーティングの設定

ルータで EIGRP スタブ ルーティングを設定するには、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。

コマンド	目的
switch(config-router)# **eigrp stub** [**leak-map** *map-name* | **receive-only** | **redistributed** [**direct**]] **例：** switch(config-router)# eigrp stub redistributed	リモート ルータを EIGRP スタブ ルータとして設定します。

次に、直接接続され、再配布されるルートをアドバタイズするスタブ ルータを設定する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config-router)# eigrp stub direct redistributed
switch(config-router)# copy running-config startup-config
```

ルータがスタブ ルータとして設定されていることを確認するには、**show ip eigrp neighbor detail** コマンドを使用します。出力の最後の行は、リモート ルータまたはスポーク ルータのスタブ ステータスを示します。次に、**show ip eigrp neighbor detail** コマンドの出力例を示します。

```
Router# show ip eigrp neighbor detail
IP-EIGRP neighbors for process 201
H   Address                 Interface   Hold Uptime   SRTT   RTO  Q  Seq Type
                                        (sec)         (ms)       Cnt Num
0   10.1.1.2                Se3/1         11 00:00:59    1  4500  0  7
   Version 12.1/1.2, Retrans: 2, Retries: 0
   Stub Peer Advertising ( CONNECTED SUMMARY ) Routes
```

EIGRP の集約アドレスの設定

指定したインターフェイスに集約アドレスを設定できます。より具体的なアドレスがルーティング テーブルにある場合、EIGRP は、より具体的なすべてのルートの最小メトリックに等しいメトリックを持つインターフェイスからの集約アドレスをアドバタイズします。「ルート集約」（p.7-5）を参照してください。

集約アドレスを設定するには、インターフェイス 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。

コマンド	目的
switch(config-if)# **ip summary-address eigrp** *as-number ip-prefix/length* [*distance* | **leak-map** *map-name*] **例：** switch(config-if)# ip summary-address eigrp 201 209.0.2.0/8	集約アドレスを、IP アドレスとネットワーク マスク、または IP プレフィクス / 長さとして設定します。また、この集約アドレスの管理ディスタンスを設定することもできます。集約アドレスのデフォルト管理ディスタンスは 5 です。

次に、EIGRP によりネットワーク 209.0.2.0 がイーサネット 1/2 のみに集約される用にする例を示します。

```
switch(config)# interface ethernet 1/2
switch(config-if)# ip summary-address eigrp 201 209.0.2.0 255.255.255.0
```

EIGRP へのルート再配布

他のルーティング プロトコルから EIGRP にルートを再配布できます。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。

他のプロトコルから再配布されるルートには、メトリック（デフォルト メトリック設定オプションまたはルート ポリシーによる）を設定する必要があります。

ルートマップを作成して、EIGRP に再配布されるルートのタイプを管理する必要があります。第 14 章「Route Policy Manager の設定」を参照してください。

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）。

コマンドの一覧

1. **config t**
2. ***router eigrp*** *as-number*
3. **redistribute** {{***bgp*** | ***eigrp*** | ***isis*** | ***ospf*** | ***rip***} *as-number* | ***direct*** | ***static***} ***route-map*** *name*
4. ***default-metric*** *bandwidth delay reliability loading mtu*
5. ***show ip eigrp policy statistics redistribute***
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 eigrp** *as-number*
	
	例：
	`switch(config)# router eigrp 201`
	`switch(config-router)#`
		新しい EIGRP プロセスを作成し、それに AS 番号を設定します。

ステップ 3	**redistribute** {{***bgp*** | ***eigrp*** | ***isis*** | ***ospf*** | ***rip***} *as-number* | ***direct*** | ***static***} ***route-map*** *name* **例：** `switch(config-router)# redistribute bgp 100 route-map BGPFilter`	1 つのルーティング ドメインから EIGRP にルートを注入します。

	コマンド	目的
ステップ 4	***default-metric*** *bandwidth delay reliability loading mtu* **例：** `switch(config-router)# default-metric 500000 30 200 1 1500`	ルート再配布で学習したルートに割り当てられるメトリックを設定します。デフォルト値は次のとおりです。 • bandwidth ― 100000 kbps • delay ― 100（10 マイクロ秒単位） • reliability ― 255 • loading ― 1 • MTU ― 1500
ステップ 5	***show ip eigrp policy statistics redistribute*** **例：** `switch(config-router)# show ip eigrp policy statistics redistribute bgp`	EIGRP ポリシー統計に関する情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、BGP を EIGRP に再配布する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config-router)# redistribute bgp 100 route-map BGPFilter
switch(config-router)# default-metric 500000 30 200 1 1500
switch(config-router)# copy running-config startup-config
```

EIGRP でのロード バランシングの設定

EIGRP でのロード バランシングを設定できます。最大パス オプションを使用して、ECMP ルートの数を設定できます。「EIGRP でのロード バランシングの設定」（p.7-18）を参照してください。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] 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）。

コマンドの一覧

1. **config t**
2. ***router eigrp*** *as-number*
3. **maximum-paths** *num-paths*
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 eigrp** *as-number* **例：** switch(config)# router eigrp 201 switch(config-router)#	新しい EIGRP プロセスを作成し、それに AS 番号を設定します。
ステップ 3	**maximum-paths** *num-paths* **例：** switch(config-router)# maximum-paths 5	EIGRP がルーティング テーブルに受け入れる等コスト パスの数を設定します。有効値の範囲は 1 ～ 16 であり、デフォルトは 16 です。
ステップ 4	***copy running-config startup-config*** **例：** switch(config-router)# copy running-config startup-config	（任意）この設定変更を保存します。

次に、6 つまでの等コストパスによる、EIGRP の等コスト ロード バランシングを設定する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config-router)# maximum-paths 6
switch(config-router)# copy running-config startup-config
```

EIGRP のグレースフル リスタートの設定

EIGRP のグレースフル リスタートまたは NSF を設定できます。「グレースフル リスタートとハイアベイラビリティ」（p.7-7）を参照してください。

（注） この機能を使用するには、NSF とグレースフル リスタートをイネーブルにする必要があります。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。

NSF 認識ルータが動作中であり、ネットワークで完全に収束している場合にのみ、このルータが NSF 対応ルータのグレースフル リスタート動作を支援できます。

グレースフル リスタートに関与する隣接デバイスが NSF 認識または NSF 対応である必要があります。

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）。

コマンドの一覧

1. **config t**
2. ***router eigrp*** *as-number*
3. **nsf**
4. **eigrp graceful-restart**
5. **timers nsf converge** *seconds*
6. **timers nsf route-hold** *seconds*
7. **timers nsf signal** *seconds*
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 eigrp** *as-number*
	
	例：
	`switch(config)# router eigrp 201`
	`switch(config-router)#`
		新しい EIGRP プロセスを作成し、それに AS 番号を設定します。

ステップ 3	**nsf** **例：** `switch(config-router)# nsf`	NSF をイネーブルにします。この機能は、デフォルトでイネーブルにされています。
ステップ 4	**eigrp graceful-restart** **例：** `switch(config-router)# graceful-restart`	グレースフル リスタートをイネーブルにします。この機能は、デフォルトでイネーブルにされています。
ステップ 5	**timers nsf converge** *seconds* **例：** `switch(config-router)# timers nsf converge 100`	スイッチオーバー後に収束するまでの制限時間を設定します。有効値の範囲は 60 ～ 180 秒です。デフォルト値は 120 です。
ステップ 6	**timers nsf route-hold** *seconds* **例：** `switch(config-router)# timers nsf route-hold 200`	グレースフル リスタート認識ピアから学習したルートの保留時間を設定します。有効値の範囲は 20 ～ 300 秒です。デフォルト値は 240 です。
ステップ 7	**timers nsf signal** *seconds* **例：** `switch(config-router)# timers nsf signal 15`	グレースフル リスタートの信号を送信する時間制限を設定します。有効値の範囲は 10 ～ 30 秒です。デフォルトでは 20 秒です。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、デフォルト タイマー値を使用してグレースフル リスタートを設定する例を示します。

```
switch# config t
switch(config)# router eigrp 201
switch(config-router)# nsf
switch(config-router)# graceful-restart
switch(config-router)# copy running-config startup-config
```

hello パケットの間隔と保留時間の調整

各 Hello メッセージの間隔と保留時間を調整できます。

デフォルトでは、5 秒ごとに Hello メッセージが送信されます。保留時間は Hello メッセージでアドバタイズされ、送信者が有効であると見なすまでの時間をネイバーに示します。デフォルトの保留時間は、hello 間隔の 3 倍または 15 秒です。

hello パケットの間隔を変更するには、インターフェイス 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。

コマンド	目的
`switch(config-if)# ip hello-interval eigrp` *as-number seconds*	EIGRP ルーティング処理の hello 間隔を設定します。有効値の範囲は 1 ～ 65535 秒であり、デフォルトは 5 です。

非常に混雑した大規模ネットワークでは、一部のルータが、デフォルト保留時間内にネイバーから hello パケットを受信できない可能性があります。この場合は、保留時間を増やすことをお勧めします。

保留時間を変更するには、インターフェイス 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。

コマンド	目的
`switch(config-if)# ip hold-time eigrp` *autonomous-system-number seconds*	EIGRP ルーティング処理の保留時間を設定します。有効値の範囲は 1 ～ 65535 です。

タイマー設定を確認するには、***show ip eigrp interface detail*** コマンドを使用します。

スプリット ホライズンのディセーブル化

スプリット ホライズンを使用して、ルート情報がルータにより、その情報の送信元インターフェイスの外部にアドバタイズされないようにすることができます。通常はスプリット ホライズンにより、特にリンクに障害がある場合に、複数のルーティング デバイス間での通信が最適化されます。

デフォルトでは、スプリット ホライズンがすべてのインターフェイスでイネーブルにされています。

スプリット ホライズンをディセーブルにするには、インターフェイス 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。

コマンド	目的
`switch(config-if)# no ip split-horizon eigrp` *as-number*	スプリット ホライズンをディセーブルにします。

EIGRP の調整

省略可能なパラメータを設定して、EIGRP をネットワークに合わせて調整できます。

ルータ コンフィギュレーション モードで、省略可能な次のパラメータを設定できます。

コマンド	目的
default-information originate [**always** | **route-map** *map-name*] **例：** `switch(config-router)# default-information originate always`	プレフィクス 0.0.0.0/0 を持つデフォルト ルートを発信するか、受け入れます。ルートマップが提供されると、ルートマップが true 状態となっている場合にのみデフォルト ルートが発信されます。
distance *internal external* **例：** `switch(config-router)# distance 25 100`	この EIGRP プロセスの管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 です。内部の値で、同じ AS 内で学習したルートのディスタンスが設定されます（デフォルト値は 90 です）。外部の値で、外部 AS から学習したルートのディスタンスが設定されます（デフォルト値は 170 です）。
metric max-hops *hop-count* **例：** `switch(config-router)# metric max-hops 70`	アドバタイズされるルートに許容される最大ホップ数を設定します。ホップ数がこの最大値を超えるルートは、到達不能としてアドバタイズされます。有効値の範囲は 1 ～ 255 であり、デフォルトは 100 です。
metric weights *tos k1 k2 k3 k4 k5* **例：** `switch(config-router)# metric weights 0 1 3 0 1 0`	EIGRP メトリックまたは K 値を調整します。EIGRP は次の式を使用して、ネットワークへの合計メトリックを決定します。 metric = [k1*bandwidth + (k2*bandwidth)/(256 – load) + k3*delay] * [k5/(reliability + k4)] デフォルト値と有効値範囲は、次のとおりです。 • TOS ― 有効値の範囲は 0 ～ 8 です。 • k1 ― 1。有効値の範囲は 1 ～ 255 です。 • k2 ― 0。有効値の範囲は 0 ～ 255 です。 • k3 ― 1。有効値の範囲は 1 ～ 255 です。 • k4 ― 0。有効値の範囲は 0 ～ 255 です。 • k5 ― 0。有効値の範囲は 0 ～ 255 です。
timers active-time {*time-limit* | **disabled**} **例：** `switch(config-router)# timers active-time 200.`	（照会の送信後に）ルートがアクティブ（SIA）状態のままとなっていることを宣言するまでに、ルータが待機する時間を分単位で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 3 です。

インターフェイス コンフィギュレーション モードで、省略可能な次のパラメータを設定できます。

コマンド	目的
ip bandwidth eigrp *as-number bandwidth*	インターフェイス上の EIGRP の帯域幅メトリックを設定します。有効値の範囲は 1 ～ 10000000 kbps です。
ip bandwidth-percent eigrp *as-number percent*	EIGRP がインターフェイス上で使用する可能性のある帯域幅のパーセンテージを設定します。デフォルトでは 50% です。
no ip delay eigrp *as-number delay*	インターフェイス上の EIGRP の遅延メトリックを設定します。有効値の範囲は 1 ～ 16777215（単位は 10 マイクロ秒）です。
ip distribute-list eigrp *as-number* {**prefix-list** *name*| **route-map** name} {**in** | **out**}	このインターフェイス上の EIGRP のルータ フィルタリング ポリシーを設定します。
no ip next-hop-self eigrp *as-number*	このインターフェイスのアドレスではなく、受信したネクストホップ アドレスを使用するよう、EIGRP を設定します。デフォルトでは、このインターフェイスの IP アドレスをネクストホップ アドレスに使用します。
ip offset-list eigrp *as-number* {**prefix-list** *name*| **route-map** name} {**in** | **out**} *offset*	EIGRP が学習したルートに、着信および発信メトリックへのオフセットを追加します。
ip passive-interface eigrp *as-number*	EIGRP インターフェイス上でルーティングが更新されないようにします。

EIGRP の仮想化の設定

各 VDC で複数の EIGRP プロセスを設定できます。各 VDC 内に複数の VRF を作成して、各 VRF で同じまたは複数の EIGRP プロセスを使用することもできます。VRF にはインターフェイスを割り当てます。

（注） インターフェイスの VRF を設定したあとに、インターフェイスの他のすべてのパラメータを設定します。インターフェイスの VRF を設定すると、そのインターフェイスの他の設定がすべて削除されます。

操作の前に

EIGRP 機能がイネーブルにされていることを確認します（「EIGRP 機能のイネーブル化」[p.7-10] を参照）。

VDC と VRF を作成します。

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）。

コマンドの一覧

1. **config t**
2. ***vrf context*** *vrf-name*
3. ***router eigrp*** *as-number*
4. ***interface ethernet*** *slot/port*
5. ***vrf member*** *vrf-id*
6. ***ip-address*** *ip-prefix/length*
7. ***ip router eigrp*** *as-number*
8. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 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	**router eigrp** *as-number*
	
	例：
	`switch(config)# router eigrp 201`
	`switch(config-router)#`
		新しい EIGRP プロセスを作成し、それに AS 番号を設定します。

	コマンド	目的

ステップ 4	**interface ethernet** *slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 5	**vrf member** *vrf-name* **例：** `switch(config-if)# vrf member RemoteOfficeVRF`	このインターフェイスを VRF に追加します。
ステップ 6	**ip router eigrp** *as-number* **例：** `switch(config-if)# ip router eigrp 201`	このインターフェイスを EIGRP プロセスに追加します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

次に、VRF を作成して、その VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context NewVRF
switch(config-vrf)# router eigrp 201
switch(config-router)# interface ethernet 1/2
switch(config-if)# ip router eigrp 201
switch(config-if)# vrf NewVRF
switch(config-if)# copy running-config startup-config
```

EIGRP 設定の確認

EIGRP 設定を確認するには、次のコマンドを使用します。

コマンド	目的
show ip eigrp *as-number*	EIGRP 設定に関する情報を表示します。
show ip eigrp [*as-number*] **interfaces** [*type number*] [detail]	インターフェイス上の EIGRP 設定に関する情報を表示します。
show ip eigrp *as-number* **neighbors** [*type number*] [**detail**]	EIGRP ネイバーに関する情報を表示します。
show ip eigrp [*as-number*] **route** [*ip-prefix/length*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**vrf** *vrf-name*]	EIGRP ルートに関する情報を表示します。
show ip eigrp [*as-number*] **topology** [*ip-prefix/length*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**vrf** *vrf-name*]	EIGRP トポロジに関する情報を表示します。
show running-configuration eigrp	現在実行中の EIGRP 設定を表示します。

EIGRP ネイバー設定を確認するには、**show ip eigrp neighbors** コマンドを使用します。

EIGRP 統計情報の表示

EIGRP 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 ip eigrp [*as-number*] **accounting** [**vrf** *vrf-name*]	EIGRP の課金統計情報を表示します。
show ip eigrp [*as-number*] **policy statistics redistribute**	EIGRP の再配布統計情報を表示します。
show ip eigrp [*as-number*] **traffic** [**vrf** *vrf-name*]	EIGRP のトラフィック統計情報を表示します。

EIGRP 設定の例

次に、EIGRP を設定する例を示します。

```
feature eigrp
interface ethernet 1/2
 ip address 209.0.2.55/24
 ip router eigrp 201
  no shutdown
router eigrp 201
  router-id 209.0.2.1
```

関連資料

ルートマップの詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

デフォルト設定

表 7-1 は、IP パラメータのデフォルト設定の一覧です。

表 7-1　デフォルト IP パラメータ

パラメータ	デフォルト
Administrative distance	• Internal routes ― 90 • External routes ― 170
Bandwidth percent	50%
Default metric for redistributed routes	• bandwidth ― 100000 kbps • delay ― 100（10 マイクロ秒単位） • reliability ― 255 • loading ― 1 • MTU ― 1500
EIGRP feature	ディセーブル
Hello interval	5 秒
Hold time	15 秒
Maximum equal cost paths	16
Metric weights	1 0 1 0 0
Next-hop address advertised	ローカル インターフェイスの IP アドレス
NSF convergence time	120
NSF route-hold time	240
NSF signal time	20
Redistribution	ディセーブル
Split horizon	イネーブル

その他の関連資料

EIGRP の実装に関する詳細情報については、次のページを参照してください。

- 関連資料（p.7-28）

関連資料

関連項目	マニュアル名
EIGRP CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』
http://www.cisco.com/warp/public/103/1.html	『*Introduction to EIGRP Tech Note*』
http://www.cisco.com/en/US/tech/tk365/technologies_q_and_a_item09186a008012dac4.shtml	EIGRP Frequently Asked Questions

CHAPTER 8

IS-IS の設定

この章では、統合 IS-IS の設定方法について説明します。

ここでは、次の内容を説明します。

- IS-IS 情報（p.8-2）
- IS-IS のライセンス要件（p.8-7）
- IS-IS の前提条件（p.8-7）
- 設定時の注意事項および制約事項（p.8-7）
- IS-IS の設定（p.8-8）
- IS-IS の設定確認（p.8-26）
- IS-IS 統計情報の表示（p.8-27）
- IS-IS の設定例（p.8-27）
- 関連項目（p.8-28）
- デフォルト設定（p.8-28）
- デフォルト設定（p.8-28）
- その他の関連資料（p.8-29）

IS-IS 情報

IS-IS は、ISO（国際標準化機構）/IEC（国際電気標準化会議）10589 に基づく IGP です。Cisco NX-OS は、IPv4（インターネット プロトコル バージョン 4）および IPv6 をサポートします。IS-IS はネットワーク トポロジの変化を検出し、他のネットワーク ノードへのループフリー ルートを計算できる、ダイナミック リンクステート ルーティング プロトコルです。各ルータは、ネットワークの状態を記述するリンクステート データベースを維持し、設定された各リンクにパケットを送信してネイバーを検出します。IS-IS はネットワークを介して各ネイバーにリンクステート情報をフラッディングします。ルータもすべての既存ネイバーを通じて、リンクステート データベースのアドバタイズメントおよびアップデートを送信します。

ここでは、次の内容について説明します。

- IS-IS の概要（p.8-2）
- IS-IS 認証（p.8-4）
- メッシュ グループ（p.8-4）
- 過負荷ビット（p.8-4）
- ルート集約（p.8-5）
- ルートの再配布（p.8-5）
- ロード バランシング（p.8-5）
- 仮想化サポート（p.8-5）
- ハイ アベイラビリティおよびグレースフル リスタート（p.8-6）
- 複数の IS-IS インスタンス（p.8-6）

IS-IS の概要

IS-IS は設定されている各インターフェイスに *hello パケット*を送信し、IS-IS ネイバー ルータを検出します。hello パケットには認証、エリア、サポート対象プロトコルなど、受信側インターフェイスが発信側インターフェイスとの互換性を判別するために使用する情報が含まれます。互換インターフェイスは隣接関係を形成し、リンクステート アップデート メッセージ（LSP）を使用して、リンクステート データベースのルーティング情報をアップデートします。ルータはデフォルトで、10 分間隔で定期的に LSP リフレッシュを送信し、LSP は 20 分間（LSP ライフタイム）リンクステート データベースに残ります。LSP ライフタイムが終了するまでにルータが LSP リフレッシュを受信しなかった場合、ルータはデータベースから LSP を削除します。

LSP インターバルは、LSP ライフタイムより短くする必要があります。そうしないと、リフレッシュ前に LSP がタイムアウトします。

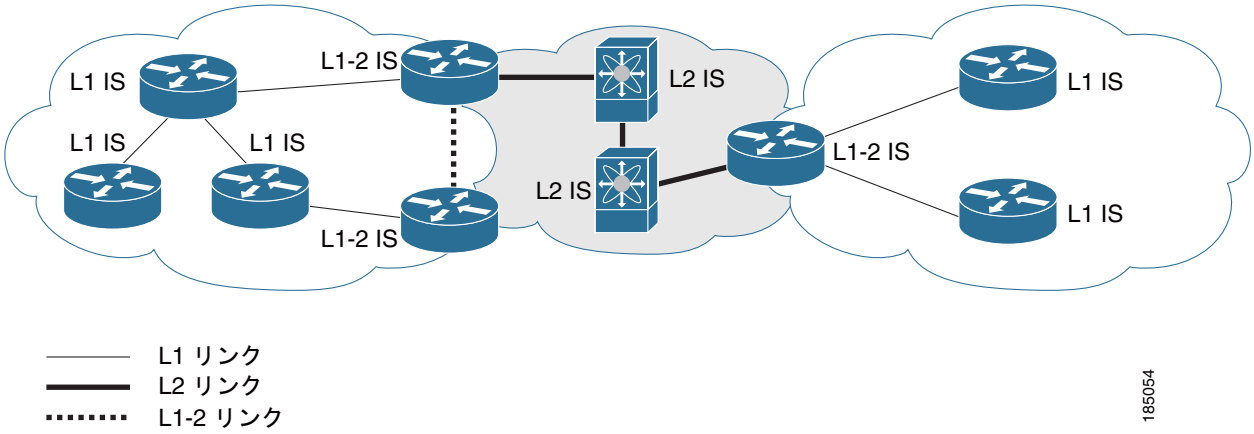
IS-IS エリア

IS-IS ネットワークは、ネットワーク内のすべてのルータが含まれるシングル エリアとして設計することも、バックボーンまたは レベル 2 エリアに接続する複数のエリアとして設計することもできます。非バックボーン エリアのルータは レベル 1 ルータで、ローカル エリア内で隣接関係を確立します（エリア内ルーティング）。レベル 2 エリアのルータは、他のレベル 2 ルータと隣接関係を確立し、レベル 1 エリア間のルーティングを実行します（エリア間ルーティング）。1 つのルータにレベル 1 エリアとレベル 2 エリアの両方を設定できます。これらのレベル 1/ レベル 2 ルータは、エリア境界ルータとして動作し、ローカル エリアからレベル 2 バックボーン エリアに情報をルーティングします（図 8-1 を参照）。

レベル 1 エリア内のルータは、そのエリア内の他のすべてのルータに対する到達方法を認識します。エリア間では、ルータはレベル 2 エリアに接続するエリア境界ルータへの到達方法を認識します。レベル 2 ルータは、他のエリア境界ルータおよび他のレベル 2 ルータへの到達方法を認識します。レベル 1/ レベル 2 ルータは 2 つのエリアの境界にまたがり、レベル 2 バックボーン エリアとの間で双方向にトラフィックをルーティングします。

Cisco NX-OS の IS-IS インスタンスは、レベル 1 またはレベル 2 エリアを 1 つだけサポートするか、またはそれぞれのエリアを 1 つずつサポートします。デフォルトでは、すべての IS-IS インスタンスが自動的にレベル 1 およびレベル 2 ルーティングをサポートします。

図 8-1　エリアに分割された IS-IS ネットワーク



ASBR（自律システム境界ルータ）は、IS-IS AS（自律システム）全体に外部宛先をアドバタイズします。外部ルートは、他のプロトコルから IS-IS に再配布されたルートです。

NET およびシステム ID

IS-IS インスタンスごとに NET が関連付けられています。NET は、その IS-IS インスタンスをエリア内で一意に特定する IS-IS システム ID とエリア ID からなります。たとえば、NET が 47.0004.004d.0001.0001.0c11.1111.00 の場合、システム ID は 0000.0c11.1111.00、エリア ID は 47.0004.004d.0001 です。

DIS

IS-IS はブロードキャスト ネットワーク内で DIS（designated intermediate system; 代表中継システム）を使用し、各ルータがブロードキャスト ネットワーク上の他のすべてのルータと不要なリンクを形成することがないようにします。IS-IS ルータは DIS に LSP を送信し、DIS がブロードキャスト ネットワークのあらゆるリンクステート情報を管理します。エリア内で DIS を選択するために IS-IS に使用させる IS-IS プライオリティをユーザ側で設定できます。

（注） ポイントツーポイント ネットワークでは DIS は不要です。

IS-IS 認証

隣接関係および LSP 交換を制御するために、認証を設定できます。ネイバーになろうとするルータは、設定されている認証レベルの同じパスワードを交換する必要があります。パスワードが無効なルータは、IS-IS によってブロックされます。IS-IS 認証はグローバルに設定することも、レベル 1、レベル 2、またはレベル 1/ レベル 2 両方のルーティングに対応する個々のインターフェイスに設定することもできます。

IS-IS がサポートする認証方式は、次のとおりです。

- クリア テキスト ― 交換するすべてのパケットで、クリアテキストの 128 ビット パスワードが伝送されます。
- MD5 ダイジェスト ― 交換するすべてのパケットで、128 ビット鍵に基づくメッセージ ダイジェストが伝送されます。

パッシブ アタック対策上、IS-IS が MD5 秘密鍵をクリアテキストとしてネットワークに流すことはありません。また、リプレイ アタックから保護するために、IS-IS は各パケットにシーケンス番号を組み込みます。

hello および LSP 認証用のキーチェーンも使用できます。キーチェーン管理については、『*Cisco Cisco NX-OS Security Configuration Guide*』を参照してください。

メッシュ グループ

*メッシュ グループ*は、一連のインターフェイスであり、それらのインターフェイスを介して到達可能なすべてのルータは、他の各ルータとの間に 1 つ以上のリンクがあります。ネットワークから 1 つまたは複数のルータを切り離さないと、多数のリンクが失敗する可能性があります。

通常のフラッディングでは、新しい LSP を受信したインターフェイスは、その LSP をルータ上の他のすべてのインターフェイスにフラッディングします。メッシュ グループを使用する場合、メッシュ グループに含まれているインターフェイスは新しい LSP を受信しても、メッシュ グループ内の他のインターフェイスには、新しい LSP をフラッディングしません。

（注） 特定のメッシュ ネットワーク トポロジーで、ネットワークのスケーラビリティを向上させるために、LSP を制限しなければならない場合があります。LSP フラッディングを制限すると、ネットワークの信頼性も下がります（障害発生時）。したがって、メッシュ グループはどうしても必要な場合に限り、慎重にネットワークを設計したうえで使用することを推奨します。

ルータ間のパラレル リンクに、ブロック モードでメッシュ グループを設定することもできます。このモードでは、各ルータがそれぞれリンクステート情報を最初に交換すると、それ以後はメッシュ グループのそのインターフェイスですべての LSP がブロックされます。

過負荷ビット

IS-IS は過負荷ビットを使用して、トラフィックの転送にはローカル ルータを使用しないが、引き続き、そのローカル ルータ宛てのトラフィックをルーティングすることを他のルータに指示します。

過負荷ビットを使用する状況は、次のとおりです。

- ルータがクリティカル条件下にある。
- ネットワークに対して通常手順でルータの追加および除去を行う。
- その他（管理上またはトラフィック エンジニアリング上）の理由。たとえば、BGP コンバージェンス待ちなどです。

ルート集約

サマリー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 を 1 つのサマリー アドレス 10.1.0.0/16 に置き換えることができます。

IS-IS はルーティング テーブルに含まれている固有性の強いルートが多いほど、固有性の強いルートの最小メトリックと同じメトリックを指定して、サマリー アドレスをアドバタイズします。

（注）　Cisco NX-OS は、自動ルート集約をサポートしません。

ルートの再配布

IS-IS を使用すると、スタティック ルート、他の IS-IS AS が学習したルート、または他のプロトコルからのルートを再配布できます。再配布を設定するには、ルート ポリシーを使用して、IS-IS に渡すルートを制御します。ルート ポリシーでは、宛先、送信元プロトコル、ルート タイプ、ルート タグなどのアトリビュートに基づいて、ルートをフィルタリングできます。詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

IS-IS ルーティング ドメインにルートを再配布しても、デフォルトでは Cisco NX-OS がそのつど、IS-IS ルーティング ドメインにデフォルト ルートを再配布することはありません。IS-IS で *デフォルト ルート*を発生させ、ルート ポリシーでそのルートを制御できます。

IS-IS にインポートされたすべてのルートに使用する、デフォルトのメトリックも設定できます。

ロード バランシング

ロード バランシングを使用すると、ルータによって、宛先アドレスから同じ距離にあるすべてのルータ ネットワーク ポートにトラフィックが分散されます。ロード バランシングは、ネットワーク セグメントの使用率を向上させ、有効ネットワーク帯域幅を増加させます。

Cisco NX-OS は、ECMP（等コスト マルチパス）機能をサポートします。IS-IS ルート テーブルおよびユニキャスト RIB の等コスト パスは最大 16 です。これらのパスの一部または全部でトラフィックのロード バランシングが行われるように、IS-IS を設定できます。

仮想化サポート

Cisco NX-OS は、同一システム上で動作する複数の IS-IS プロトコル インスタンスをサポートします。IS-IS は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。VDC で設定できる IS-IS インスタンスは、最大 4 つです。

特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

ハイ アベイラビリティおよびグレースフル リスタート

Cisco NX-OS システムでコールド リブートが発生した場合、ネットワークはシステムにトラフィックを転送しないで、ネットワーク トポロジからシステムを削除します。この状況では、IS-IS はステートレス リスタートになり、Cisco NX-OS がすべてのネイバーを削除します。Cisco NX-OS によってスタートアップ コンフィギュレーションが適用され、IS-IS はネイバーを再発見して、再びすべての IS-IS ルーティング情報を共有します。

Cisco NX-OS はハイ アベイラビリティをサポートします。Cisco NX-OS システムでコールド リブートが発生した場合、ネットワークはシステムへのトラフィック転送を中止し、ネットワーク トポロジからシステムを削除します。この状況では、IS-IS はステートレス リスタートになり、ローカル システムからすべてのネイバーを削除します。Cisco NX-OS によってスタートアップ コンフィギュレーションが適用され、IS-IS はネイバーを再発見して、再びすべての IS-IS ルーティング情報を共有します。

Cisco NX-OS を実行するスーパーバイザが 2 つあるプラットフォームでは、ステートフル スーパーバイザ スイッチオーバーが行われます。IS-IS はスイッチオーバーの発生前に、IS-IS がしばらく使用できなくなることを告知することによって、グレースフル リスタートを開始します。スイッチオーバーの間、ネットワークはトラフィック転送を続け、ネットワーク トポロジにシステムを残します。

スイッチオーバー後、Cisco NX-OS によって実行コンフィギュレーションが適用され、IS-IS は再び動作可能になったことをネイバーに通知します。

この間に問題が発生した場合は、IS-IS が自動的にリスタートします。リスタート後、IS-IS はプラットフォームがネットワーク トポロジから排除されないように、グレースフル リスタートを開始します。IS-IS を手動でリスタートした場合は、グレースフル リスタートが実行され、ステートフル スイッチオーバーと同様の結果が得られます。どちらの場合も、実行コンフィギュレーションが適用されます。

（注） IS-IS の ISSU（インサービス ソフトウェア アップグレード）をサポートするには、グレースフル リスタートをイネーブルにする必要があります。グレースフル リスタートをディセーブルにすると、この設定では ISSU をサポートできないことを伝える警告が Cisco NX-OS から出されます。

複数の IS-IS インスタンス

Cisco NX-OS は、同一システム上で動作する最大 4 つの IS-IS プロトコル インスタンスをサポートします。同一インターフェイス上で複数のインスタンスを設定することはできません。すべてのインスタンスで同じシステム ルータ ID を使用します。

IS-IS のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	IS-IS には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細、ライセンスを取得して適用する方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

IS-IS の前提条件

IS-IS の前提条件は、次のとおりです。

- IS-IS 機能をイネーブルにする必要があります（「IS-IS 機能のイネーブル化」[p.8-9] を参照）。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

IS-IS 設定時の注意事項および制約事項は、次のとおりです。

- VDC ごとに最大 4 つの IS-IS インスタンスを設定できます。

IS-IS の設定

IS-IS を設定する手順は、次のとおりです。

ステップ 1　IS-IS 機能をイネーブルにします（「IS-IS 機能のイネーブル化」[p.8-9] を参照）。

ステップ 2　IS-IS インスタンスを作成します（「IS-IS インスタンスの作成」[p.8-10] を参照）。

ステップ 3　IS-IS　インスタンスにインターフェイスを追加します（「インターフェイス上での　IS-IS　の設定」[p.8-12] を参照）。

ステップ 4　認証、メッシュ グループ、ダイナミック ホスト交換などのオプション機能を設定します。

ここでは、次の内容について説明します。

- IS-IS コンフィギュレーション モード（p.8-8）
- IS-IS 機能のイネーブル化（p.8-9）
- IS-IS インスタンスの作成（p.8-10）
- インターフェイス上での IS-IS の設定（p.8-12）
- エリアでの IS-IS 認証の設定（p.8-14）
- インターフェイス上での IS-IS 認証の設定（p.8-15）
- メッシュ グループの設定（p.8-16）
- DIS の設定（p.8-16）
- ダイナミック ホスト交換の設定（p.8-17）
- 過負荷ビットの設定（p.8-17）
- サマリー アドレスの設定（p.8-17）
- 再配布の設定（p.8-18）
- 仮想化の設定（p.8-21）
- IS-IS の調整（p.8-24）

（注）　Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

IS-IS コンフィギュレーション モード

ここでは各コンフィギュレーション モードの開始方法について説明します。各モードから **?** コマンドを入力すると、そのモードで使用できるコマンドが表示されます。

ここでは、次の内容について説明します。

- ルータ コンフィギュレーション モード（p.8-9）
- 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（p.8-9）

ルータ コンフィギュレーション モード

ルータ コンフィギュレーション モードを開始する例を示します。

```
switch#: conf t
switch(config)# router isis isp
switch(config-router)#
```

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

ルータ アドレス ファミリ コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router isis isp
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)#
```

IS-IS 機能のイネーブル化

IS-IS を設定する前に、IS-IS 機能をイネーブルにする必要があります。

準備作業

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）。

手順概要

1. **config t**
2. ***feature isis***
3. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature isis** **例：** `switch(config)# feature isis`	IS-IS 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

IS-IS 機能をディセーブルにして、関連するすべての設定を削除する場合は、***no feature isis*** コマンドを使用します。

コマンド	目的
no feature isis **例：** `switch(config)# no feature isis`	IS-IS 機能をディセーブルにして、関連するすべての設定を削除します。

IS-IS インスタンスの作成

IS-IS インスタンスを作成し、そのインスタンスのエリア レベルを設定できます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. **router is-is** *instance-tag*
3. ***net*** *network-entity-title*
4. ***is-type*** {***level-1*** | ***level-2*** | ***level-1-2***}
5. ***show isis*** [***vrf*** *vrf-name*] ***process***
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 isis** *instance-tag*
	
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい IS-IS インスタンスを作成します。

ステップ 3	**net** *network-entity-title*
	
	例：
	`switch(config-router)# net`
	`47.0004.004d.0001.0001.0c11.1111.00`
		この IS-IS インスタンスに対応する NET を設定します。

ステップ 4	**is-type** {***level-1*** | ***level-2*** | ***level-1-2***} **例：** `switch(config-router)# is-type level-2`	（任意）この IS-IS インスタンスのエリア レベルを設定します。デフォルトは level-1-2 です。

	コマンド	目的
ステップ 5	**show isis** [**vrf** *vrf-name*] **process** **例：** `switch(config)# show isis process`	（任意）すべての IS-IS インスタンスについて、IS-IS 要約情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

IS-IS インスタンスおよび関連するすべての設定を削除する場合は、***no router isis*** コマンドを使用します。

コマンド	目的
no router isis *instance-tag* **例：** `switch(config)# no router isis Enterprise`	IS-IS インスタンスおよび関連するすべての設定を削除します。

（注） IS-IS インスタンスに関するすべての設定を完全に削除するには、インターフェイス モードで設定した IS-IS コマンドも削除する必要があります。

IS-IS には次のオプション パラメータを設定できます。

コマンド	目的
distance *value* **例：** `switch(config-router)# distance 30`	IS-IS の管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 115 です。「管理ディスタンス」（p.1-7）を参照してください。
log-adjacency-changes **例：** `switch(config-router)# log-adjacency-changes`	IS-IS ネイバーのステートが変化するたびに、システム メッセージを送信します。
lsp-mtu *size* **例：** `switch(config-router)# lsp-mtu 600`	この IS-IS インスタンスにおける LSP の MTU を設定します。有効値の範囲は 128 ～ 4352 バイトです。デフォルト値は 1492 です。
maximum-paths *number* **例：** `switch(config-router)# maximum-paths 6`	IS-IS がルート テーブルで維持する等コスト パスの最大数を設定します。有効値の範囲は 1 ～ 16 であり、デフォルトは 4 です。
reference-bandwidth *bandwidth-value* {**Mbps** | **Gbps**} **例：** `switch(config-router)# reference-bandwidth 100 Gbps`	IS-IS コスト メトリックの計算に使用する、デフォルトの基準帯域幅を設定します。有効値の範囲は 1 ～ 4000 Gbps です。デフォルト値は 40 Gbps です。

レベル 2 エリアで IS-IS インスタンスを作成する例を示します。

```
switch# config t
switch(config)# router isis Enterprise
switch(config-router)# net 47.0004.004d.0001.0001.0c11.1111.00
switch(config-router)# is-type level 2
switch(config-router)# copy running-config startup-config
```

ネイバーの統計情報を消去し、隣接関係を削除する場合は、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。

コマンド	目的
clear isis [*instance-tag*] **adjacency** [***** | *system-id* | *interface*] **例：** `switch(config-if)# clear isis adjacency *`	ネイバーの統計情報を消去し、この IS-IS インスタンスの隣接関係を削除します。

IS-IS インスタンスの再起動

IS-IS インスタンスは再起動が可能です。再起動すると、インスタンスのすべてのネイバーが消去されます。

IS-IS インスタンスを再起動し、関連付けられたすべてのネイバーを削除するには、次のコマンドを使用します。

コマンド	目的
restart isis instance-tag **例：** `switch(config)# restart isis Enterprise`	IS-IS インスタンスを再起動し、すべてのネイバーを削除します。

インターフェイス上での IS-IS の設定

IS-IS インスタンスにインターフェイスを追加できます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. ***ip router isis*** *instance-tag*

   または

   ipv6 router isis *instance-tag*
4. ***show isis*** [***vrf*** *vrf-name*] [*instance-tag*] ***interface*** [*interface-type slot/port*]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip router isis** *instance-tag* **例：** `switch(config-if)# ip router isis Enterprise`	この IPv4 インターフェイスを IS-IS インスタンスに関連付けます。
	ipv6 router isis *instance-tag* **例：** `switch(config-if)# ipv6 router isis Enterprise`	この IPv6 インターフェイスを IS-IS インスタンスに関連付けます。
ステップ 4	***show isis*** [***vrf*** *vrf-name*] [*instance-tag*] ***interface*** [*interface-type slot/port*] **例：** `switch(config)# show isis Enterprise ethernet 1/2`	（任意）インターフェイスの IS-IS 情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

インターフェイス モードでは、IS-IS に次のオプション パラメータを設定できます。

コマンド	目的
isis circuit-type {**level-1** | **level-2** | **level-1-2**} **例：** `switch(config-if)# isis circuit-type level-2`	このインターフェイスが関与する隣接関係のタイプを設定します。このコマンドを使用するのは、レベル 1 とレベル 2 の両方のエリアにルータが関係する場合だけです。
isis metric *value* {**level-1** | **level-2**} **例：** `switch(config-if)# isis metric 30`	このインターフェイスの IS-IS メトリックを設定します。有効値の範囲は 1 ～ 16777214 であり、デフォルトは 10 です。
isis passive {**level-1** | **level-2** | **level-1-2**} **例：** `switch(config-if)# isis passive level-2`	インターフェイスが隣接関係を形成しないようにしながら、なおかつ、インターフェイスに関連付けられたプレフィクスをアドバタイズするようにします。

IS-IS インスタンスにインターフェイス ethernet 1/2 を追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ip router isis Enterprise
switch(config-if)# copy running-config startup-config
```

エリアでの IS-IS 認証の設定

エリアで LSP を認証するように IS-IS を設定できます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. ***router isis*** *instance-tag*
3. ***authentication-type*** {***cleartext*** | ***md5***} {***level-1*** | ***level-2***}
4. ***authentication-key keychain*** *key* {***level-1*** | ***level-2***}
5. ***authentication-check*** {***level-1*** | ***level-2***}
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 isis** *instance-tag*
	
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい IS-IS インスタンスを作成します。

ステップ 3	***authentication-type*** {***cleartext*** | ***md5***} {***level-1*** | ***level-2***} **例：** `switch(config-router)# authentication-type cleartext level-2`	クリアテキストまたは MD5 認証ダイジェストとして、レベル 1 またはレベル 2 エリアに使用する認証方式を設定します。
ステップ 4	**authentication-key keychain** *key* {***level-1*** | ***level-2***} **例：** `switch(config-router)# authentication-key ISISKey level-2`	IS-IS エリアレベル認証に使用する認証鍵を設定します。
ステップ 5	***authentication-check*** {***level-1*** | ***level-2***} **例：** `switch(config-router)# authentication-check level-2`	（任意）受信パケットの認証パラメータチェックをイネーブルにします。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

IS-IS インスタンスにクリアテキスト認証を設定する例を示します。

```
switch# config t
switch(config)# router isis Enterprise
switch(config-router)# authentication-type cleartext level-2
switch(config-router)# authentication-key keychain ISISKey level-2
switch(config-router)# copy running-config startup-config
```

インターフェイス上での IS-IS 認証の設定

インターフェイス上で hello パケットを認証するように IS-IS を設定できます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. ***interface*** *interface-type slot/port*
3. ***isis authentication-type*** {***cleartext*** | ***md5***} [***level-1*** | ***level-2***]
4. ***isis authentication-key keychain*** *key* [***level-1*** | ***level-2***]
5. ***isis authentication-check*** [***level-1*** | ***level-2***]
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	***isis authentication-type*** {***cleartext*** | ***md5***} [***level-1*** | ***level-2***] **例：** `switch(config-if)# isis authentication-type cleartext level-2`	クリアテキストまたは MD5 認証ダイジェストとして、このインターフェイスにおける IS-IS 認証タイプを設定します。
ステップ 4	**isis authentication-key keychain** *key* [***level-1*** | ***level-2***] **例：** `switch(config-if)# isis authentication-key ISISKey level-2`	このインターフェイス上で IS-IS に使用する認証鍵を設定します。

	コマンド	目的
ステップ 5	***isis authentication-check*** {***level-1*** | ***level-2***} **例：** `switch(config-if)# isis authentication-check`	（任意）受信パケットの認証パラメータ チェックをイネーブルにします。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

IS-IS インスタンスにクリアテキスト認証を設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# isis authentication-type cleartext level-2
switch(config-if)# isis authentication-key keychain ISISKey
switch(config-if)# copy running-config startup-config
```

メッシュ グループの設定

メッシュ グループにインターフェイスを追加することによって、そのメッシュ グループ内のインターフェイスに対する LSP フラッディング量を制限できます。任意で、メッシュ グループ内のインターフェイスに対して、すべての LSP フラッディングをブロックすることもできます。

メッシュ グループにインターフェイスを追加するには、インターフェイス 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。

コマンド	目的
isis mesh-group {***blocked*** | *mesh-id*} **例：** `switch(config-if)# isis mesh-group 1`	メッシュ グループにこのインターフェイスを追加します。有効値の範囲は 1 ～ 4294967295 です。

DIS の設定

インターフェイス プライオリティを設定することによって、ルータがマルチアクセス ネットワークの DIS（代表中継システム）になるように設定できます。

DIS を設定するには、インターフェイス 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。

コマンド	目的
isis priority *number* **例：** `switch(config-if)# isis priority 100`	DIS 選択のためのプライオリティを設定します。有効値の範囲は 0 ～ 127 です。デフォルトは 64 です。

ダイナミック ホスト交換の設定

ダイナミック ホスト交換を使用することによって、システム ID とルータのホスト名がマッピングされるように IS-IS を設定できます。

ダイナミック ホスト交換を設定するには、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。

コマンド	目的
hostname dynamic **例：** `switch(config-router)# hostname dynamic`	ダイナミック ホスト交換をイネーブルにします。

過負荷ビットの設定

最短パス優先（SPF）を計算するときの中間ホップとしてこのルータを使用しないことを他のルータに伝えるように、ルータを設定できます。任意で、起動時に BGP が収束するまで、一時的に過負荷ビットを設定することもできます。

過負荷ビットを設定する以外に、レベル 1 またはレベル 2 トラフィックに関して、LSP からの特定タイプの IP プレフィクス アドバタイズメントを抑制することが必要な場合もあります。

過負荷ビットを設定するには、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。

コマンド	目的
set-overload-bit {***always*** | ***on-startup*** {*seconds* | ***wait-for bgp*** *as-number*}} [***suppress*** [***interlevel*** | ***external***]] **例：** `switch(config-router)# set-overload-bit on-startup 30`	IS-IS に過負荷ビットを設定します。*seconds* の範囲は 5 ～ 86400 です。

サマリー アドレスの設定

ルーティング テーブルでサマリー アドレスによって表される集約アドレスを作成できます。1 つのサマリー アドレスに、特定のレベルの複数のアドレス グループを含めることができます。Cisco NX-OS は固有性の強いすべてのルートのうち、最小メトリックをアドバタイズします。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. ***router isis*** *instance-tag*
3. ***address-family*** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
4. ***summary-address*** {*ip-prefix/mask-len* | *ipv6-prefix/mask-len*} {**level-1** | ***level-2*** | ***level-1-2***}

5. ***show isis*** [***vrf*** *vrf-name*] ***ip summary-address*** *ip-prefix* [***longer-prefixes***]
6. ***show isis*** [***vrf*** *vrf-name*] ***ipv6 summary-address*** *ipv6-prefix* [***longer-prefixes***]
7. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 isis** *instance-tag*
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい IS-IS インスタンスを作成します。

ステップ 3	***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 モードを開始します。

ステップ 4	***summary-address*** {*ip-prefix/mask-len* | *ipv6-prefix/mask-len*} {**level-1** | ***level-2*** | ***level-1-2***} **例：** `switch(config-router-af)# summary-address 192.0.2.0/24 level-2`	IPv4 または IPv6 アドレスに対応する、ISIS エリア用のサマリー アドレスを設定します。
ステップ 5	***show isis*** [***vrf*** *vrf-name*] ***ip summary-address*** *ip-prefix* [***longer-prefixes***]] **例：** `switch(config-if)# show isis ip summary-address`	（任意）IS-IS IPv4 サマリー アドレス情報を表示します。
ステップ 6	***show isis*** [***vrf*** *vrf-name*] ***ipv6 summary-address*** *ipv6-prefix* [***longer-prefixes***]] **例：** `switch(config-if)# show isis ipv6 summary-address`	（任意）IS-IS IPv6 サマリー アドレス情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config--if)# copy running-config startup-config`	（任意）この設定変更を保存します。

IS-IS に IPv4 ユニキャスト サマリー アドレスを設定する例を示します。

```
switch# config t
switch(config)# router isis Enterprise
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# summary-address 192.0.2.0/24 level-2
switch(config-router-af)# copy running-config startup-config
```

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、IS-IS ネットワークを通じてその情報を再配布するように、IS-IS 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。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] 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）。

手順概要

1. **config t**
2. ***router isis*** *instance-tag*
3. ***address-family*** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
4. ***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | ***isis*** *id* | ***ospf*** *id* | ***ospfv3*** *id* | ***rip*** *id* | ***static*** } ***route-map*** *map-name*
5. ***default-information originate*** [***always***] [***route-map*** *map-name*]
6. ***distribute*** {***level-1*** | ***level-2***} ***into*** {***level-1*** | ***level-2***} {***route-map*** *route-map* | ***all***}
7. ***show isis*** [***vrf*** *vrf-name*] {***ip*** | ***ipv6***} ***route*** {*ip-prefix* | *ip6-prefix*}[***detail*** | ***longer-prefixes*** [***summary*** | ***detail***]]
8. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 isis** *instance-tag*
	
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい IS-IS インスタンスを作成します。

ステップ 3	***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 モードを開始します。

ステップ 4	***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | ***isis*** *id* | ***ospf*** *id* | ***ospfv3*** *id* | ***rip*** *id* | ***static*** | ***direct***} ***route-map*** *map-name* **例：** `switch(config-router-af)# redistribute eigrp 201 route-map ISISmap`	他のプロトコルからのルートを IS-IS に再配布します。ルート マップの詳細については、「ルート マップの設定」（p.14-9）を参照してください。
ステップ 5	***default-information originate*** [***always***] [***route-map*** *map-name*] **例：** `switch(config-router-af)# default-information originate always`	（任意）IS-IS へのデフォルト ルートを作成します。
ステップ 6	***distribute*** {***level-1*** | ***level-2***} ***into*** {***level-1*** | ***level-2***} {***route-map*** *route-map* | ***all***} **例：** `switch(config-router-af)# distribute level-1 into level-2 all`	（任意）一方の IS-IS レベルから他方の IS-IS レベルへ、ルートを再配布します。

	コマンド	目的
ステップ 7	***show isis*** [***vrf*** *vrf-name*] {***ip*** | ***ipv6***} ***route*** {*ip-prefix ip6-prefix*}[***detail*** | ***longer-prefixes*** [***summary*** | ***detail***]] **例：** `switch(config-router-af)# show isis ip route`	（任意）IS-IS のルートを表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-router-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

EIGRP を IS-IS に再配布する例を示します。

```
switch# config t
switch(config)# router isis Enterprise
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# redistribute eigrp 201 route-map ISISmap
switch(config-router-af)# copy running-config startup-config
```

グレースフル リスタートの設定

IS-IS にグレースフル リスタートを設定できます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] を参照）。

VDC および VRF を作成します。

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）。

手順概要

1. **config t**
2. ***router isis*** *instance-tag*
3. ***graceful-restart***
4. ***graceful-restart*** **t3 manual** *time*
5. ***show running-config isis***
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 isis** *instance-tag*
	
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		名前を設定して、新しい IS-IS プロセスを作成します。

ステップ 3	***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。イネーブルがデフォルトです。
ステップ 4	***graceful-restart t3 manual*** *time* **例：** `switch(config-router)# graceful-restart t3 manual 300`	グレースフル リスタート T3 タイマーを設定します。有効値の範囲は 30 ～ 65535 秒です。デフォルト値は 60 です。
ステップ 5	***show running-config isis*** **例：** `switch(config-router)# show running-config isis`	（任意）IS-IS の設定を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config-router)# copy running-config startup-config`	（任意）この設定変更を保存します。

グレースフル リスタートをイネーブルにする例を示します。

```
switch# config t
switch(config)# router isis Enterprise
switch(config-router)# graceful-restart
switch(config-router)# copy running-config startup-config
```

仮想化の設定

VDC ごとに複数の IS-IS インスタンスを設定できます。各 VDC 内で複数の VRF を作成することもできます。また、各 VRF で同じ IS-IS インスタンスを使用することも、複数の IS-IS インスタンスを使用することも可能です。VRF に IS-IS インターフェイスを割り当てます。

設定した VRF に NET を設定する必要があります。

（注） インターフェイスに VRF を設定したあとで、その他のインターフェイス パラメータを設定します。インターフェイスに VRF を設定すると、そのインターフェイスに関するすべての設定が削除されます。

準備作業

IS-IS 機能がイネーブルになっていることを確認します（「IS-IS 機能のイネーブル化」[p.8-9] を参照）。

VDC を作成します。

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）。

手順概要

1. **config t**
2. ***vrf* context** *vrf_name*
3. **exit**
4. **router isis** *instance-tag*
5. ***vrf*** *vrf_name*
6. **net** *network-entity-title*
7. オプション パラメータの設定
8. ***interface*** *type slot/port*
9. ***vrf member*** *vrf-name*
10. ***ip address*** *ip-prefix/length*
11. ***ip router isis*** *instance-tag*

    または

    ipv6 router isis *instance-tag*
12. ***show isis*** [***vrf*** *vrf-name*] [*instance-tag*] ***interface*** [*interface-type slot/port*]
13. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 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	**router isis** *instance-tag*
	
	例：
	`switch(config)# router isis Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい IS-IS インスタンスを作成します。

ステップ 4	**vrf** *vrf-name*
	
	例：
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		（任意）VRF コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 5	**net** *network-entity-title* **例：** switch(config-router-vrf)# net 47.0004.004d.0001.0001.0c11.1111.00	この IS-IS インスタンスに対応する NET を設定します。
ステップ 6	**exit** **例：** switch(config-router)# exit switch(config-router)#	ルータ VRF 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	***address-family*** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***} **例：** switch(config-router)# address-family ipv4 unicast switch(config-router-af)#	（任意）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 8	***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | ***isis*** *id* | ***ospf*** *id* | ***ospfv3*** *id* | ***rip*** *id* | ***static*** | ***direct***} ***route-map*** *map-name* **例：** switch(config-router-af)# redistribute eigrp 201 route-map ISISmap	（任意）他のプロトコルからのルートを IS-IS に再配布します。ルート マップの詳細については、「ルート マップの設定」（p.14-9）を参照してください。
ステップ 9	**interface ethernet** *slot/port* **例：** switch(config)# interface ethernet 1/2 switch(config-if)#	インターフェイス コンフィギュレーション モードを開始します。
ステップ 10	**vrf member** *vrf-name* **例：** switch(config-if)# vrf member RemoteOfficeVRF	VRF にこのインターフェイスを追加します。
ステップ 11	**ip address** *ip-prefix/length* **例：** switch(config-if)# ip address 209.0.2.1/16	このインターフェイスの IP アドレスを設定します。この作業は、VRF にこのインターフェイスを割り当てたあとで行う必要があります。
ステップ 12	**ip router isis** *instance-tag* **例：** switch(config-if)# ip router isis Enterprise	この IPv4 インターフェイスを IS-IS インスタンスに関連付けます。
	ipv6 router isis *instance-tag* **例：** switch(config-if)# ipv6 router isis Enterprise	この IPv6 インターフェイスを IS-IS インスタンスに関連付けます。
ステップ 13	***show isis*** [***vrf*** *vrf-name*] [*instance-tag*] ***interface*** [*interface-type slot/port*] **例：** switch(config-if)# show isis Enterprise ethernet 1/2	（任意）VRF に含まれるインターフェイスの IS-IS 情報を表示します。
ステップ 14	***copy running-config startup-config*** **例：** switch(config-if)# copy running-config startup-config	（任意）この設定変更を保存します。

VRF を作成し、VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context NewVRF
switch(config-vrf)# exit
switch(config)# router isis Enterprise
switch(config-router)# vrf NewVRF
switch(config-router-vrf)# net 47.0004.004d.0001.0001.0c11.1111.00
switch(config-router-vrf)# interface ethernet 1/2
switch(config-if)# vrf membmer NewVRF
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# ip router isis Enterprise
switch(config-if)# copy running-config startup-config
```

IS-IS の調整

ネットワーク要件に合わせて IS-IS を調整できます。

IS-IS 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
lsp-gen-interval [**level-1** | **level-2**] *lsp-max-wait* [*lsp-initial-wait lsp-second-wait*] **例：** `switch(config-router)# lsp-gen-interval level-1 500 500 500`	LSP 発生に関する IS-IS スロットルを設定します。オプション パラメータは次のとおりです。 • lsp-max-wait ― トリガーから LSP 発生までの最大待ち時間。有効値の範囲は 500 ～ 65535 ミリ秒です。 • lsp-initial-wait ― トリガーから LSP 発生までの初期待ち時間。有効値の範囲は 50 ～ 65535 ミリ秒です。 • lsp-second-wait ― バックオフ時の LSP スロットルに使用する第 2 待ち時間。有効値の範囲は 50 ～ 65535 ミリ秒です。
max-lsp-lifetime *lifetime* **例：** `switch(config-router)# max-lsp-lifetime 500`	LSP の最大ライフタイムを秒数で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 1200 です。
spf-interval [**level-1** | **level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*] **例：** `switch(config-router)# spf-interval level-2 500 500 500`	LSA 到着までのインターバルを設定します。オプション パラメータは次のとおりです。 • lsp-max-wait ― トリガーから SPF 計算までの最大待ち時間。有効値の範囲は 500 ～ 65535 ミリ秒です。 • lsp-initial-wait ― トリガーから SPF 計算までの初期待ち時間。有効値の範囲は 50 ～ 65535 ミリ秒です。 • lsp-second-wait ― バックオフ時の SPF 計算に使用する第 2 待ち時間。有効値の範囲は 50 ～ 65535 ミリ秒です。
shutdown **例：** `switch(config-router)# shutdown`	設定を削除しないで、この IS-IS インスタンスをシャットダウンします。
wide-metric-only **例：** `switch(config-router)# wide-metric-only`	ワイド メトリックをアドバタイズするように、IS-IS を設定します。

ルータ アドレス コンフィギュレーション モードで次のオプション コマンドを使用できます。

コマンド	目的
adjacency-check **例：** `switch(config-router-af)# adjacency-check`	隣接関係チェックを実行し、IS-IS インスタンスが同じアドレス ファミリをサポートするリモート IS-IS エンティティに限って隣接関係を形成していることを確認します。イネーブルがデフォルトです。

IS-IS を調整するには、インターフェイス 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
isis hello-interval *seconds* [***level-1*** | ***level-2***] **例：** `switch(config-if)# isis hello-interval 20`	IS-IS に hello インターバルを秒数で設定します。有効値の範囲は 1 ～ 65535 であり、デフォルトは 10 です。
isis hello-multiplier *num* [***level-1*** | ***level-2***] **例：** `switch(config-if)# isis hello-multiplier 20`	ルータが隣接関係を破棄するまでに、ネイバーが見逃さなければならない IS-IS hello パケットの数を指定します。有効値の範囲は 3 ～ 1000 であり、デフォルトは 3 です。
isis hello-padding **例：** `switch(config-if)# isis hello-padding`	フル MTU に hello パケットを埋め込みます。イネーブルがデフォルトです。
isis lsp-interval *milliseconds* **例：** `switch(config-if)# isis lsp-interval 20`	フラッディング時にこのインターフェイスで LSP が送信される間隔をミリ秒数で設定します。有効値の範囲は 10 ～ 65535 であり、デフォルトは 33 です。

IS-IS の設定確認

IS-IS の設定を確認するには、次のコマンドを使用します。

コマンド	目的
show isis [**vrf** *vrf-name*] **adjacency** [*interface*] [**detail** | **summary**]	IS-IS の隣接関係を表示します。これらの統計情報を消去するには、**clear isis adjacency** コマンドを使用します。
show isis [**vrf** *vrf-name*] **database** [**level-1** | **level-2**] [**detail** | **summary**] [*LSP ID*] {[**ip prefix** *ip-prefix*] | [**ipv6 prefix** *ipv6-prefix*] | [**router-id** *router-id*] | [**adjacency** *node-id*]}	IS-IS LSP データベースを表示します。
show isis [**vrf** *vrf-name*] *hostname*	ダイナミック ホスト交換情報を表示します。
show isis [**vrf** *vrf-name*] [*instance-tag*] **interface** [*interface*]	IS-IS インターフェイス情報を表示します。
show isis [**vrf** *vrf-name*] **mesh-group**	メッシュ グループ情報を表示します。
show isis [**vrf** *vrf-name*] **process**	IS-IS 情報を表示します。
show isis [**vrf** *vrf-name*] [**ipv6] route** [ip-prefix | *ipv6-prefix] [detail* | **longer-prefixes** [**summary** | **detail**]]	IS-IS ルート テーブルを表示します。
show isis [**vrf** *vrf-name*] **spf-log** [**detail**]	IS-IS SPF 計算の統計情報を表示します。
show isis [**vrf** *vrf-name*]　[**ipv6**] **summary-address**s [*ip-prefix* | *ipv6-prefix* [*longer-prefixes*]]	IS-IS のサマリー アドレス情報を表示します。
show running-configuration isis	現在の実行中の IS-IS 設定を表示します。

各コマンド出力のフィールドの詳細については、『*Cisco NX-OS Command Reference*』を参照してください。

IS-IS 統計情報の表示

IS-IS 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 isis [**vrf** *vrf-name*] [instance-tag] **adjacency** [*interface*] [**system-ID**]	IS-IS 隣接関係の統計情報を表示します。
show isis [**vrf** *vrf-name*] [*instance-tag*] **statistics** [*interface*]	IS-IS インターフェイスの統計情報を表示します。
show isis [**vrf** *vrf-name*] [**ip** | **ipv4**] **route-map statistics redistribute** {**bgp** *id* | **direct** | **eigrp** *id* | **isis** *id* | **ospf** *id* | **rip** *id* | **static**}	IS-IS 再配布の統計情報を表示します。
show isis [**vrf** *vrf-name*] **route-map statistics distribute** {**level-1** | **level-2**} **into** {**level-1** | **level-2**}}	レベル間で配布されたルートに関する、IS-IS 配布統計情報を表示します。

IS-IS 設定の統計情報を消去するには、次のいずれかの作業を行います。

コマンド	目的
clear isis [**vrf** *vrf-name*] [instance-tag] **adjacency** [*interface*] [**system-ID**]	IS-IS 隣接関係の統計情報を消去します。
clear isis [**vrf** *vrf-name*] [*instance-tag*] **statistics** [*interface*]	IS-IS インターフェイスの統計情報を消去します。
clear isis [**vrf** *vrf-name*] [**ip** | **ipv4**] **route-map statistics redistribute** {**bgp** *id* | **direct** | **eigrp** *id* | **isis** *id* | **ospf** *id* | **rip** *id* | **static**}	IS-IS 再配布の統計情報を消去します。
clear isis [**vrf** *vrf-name*] **route-map statistics distribute** {**level-1** | **level-2**} **into** {**level-1** | **level-2**}}	レベル間で配布されたルートに関する、IS-IS 配布統計情報を消去します。

IS-IS の設定例

［手順詳細の表で使用したコマンド例と整合する設定例を作成］

IS-IS の設定例を示します。

```
router isis Enterprise
 is-type level-1
 net 49.0001.0000.0000.0003.00
 graceful-restart

interface Loopback0
 passive
 address-family ipv4 unicast

interface POS0/3/0/1
 hello-password text encrypted 05080F1C2243
 address-family ipv4 unicast
```

関連項目

ルート マップの詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

デフォルト設定

表 8-1 に、IS-IS パラメータのデフォルト設定を示します。

表 8-1　デフォルトの IS-IS パラメータ

パラメータ	デフォルト
管理ディスタンス	115
エリア レベル	level-1-2
DIS プライオリティ	64
グレースフル リスタート	イネーブル
hello 回数	3
hello 埋め込み	イネーブル
hello タイム	10 秒
IS-IS 機能	ディセーブル
LSP インターバル	33
LSP MTU	1492
LSP 最大ライフタイム	1200 秒
最大パス数	4
メトリック	10
基準帯域幅	40 Gbps

その他の関連資料

IS-IS 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。

- 関連資料（p.8-29）
- 規格（p.8-29）

関連資料

関連項目	マニュアル名
IS-IS CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 9

ベーシック BGP の設定

この章では、Cisco NX-OS デバイス上で BGP（ボーダー ゲートウェイ プロトコル）を設定する方法について説明します。

ここでは、次の内容を説明します。

- ベーシック BGP の概要（p.9-2）
- ベーシック BGP のライセンス要件（p.9-7）
- BGP の前提条件（p.9-7）
- BGP に関する注意事項および制限事項（p.9-7）
- CLI コンフィギュレーション モード（p.9-8）
- ベーシック BGP の設定（p.9-10）
- ベーシック BGP の設定確認（p.9-15）
- BGP 統計情報の表示（p.9-16）
- ベーシック BGP の設定例（p.9-16）
- 関連項目（p.9-16）
- 関連情報（p.9-17）
- デフォルト設定（p.9-17）
- その他の関連資料（p.9-18）

ベーシック BGP の概要

Cisco NX-OS は BGP バージョン 4 をサポートします。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 対応デバイスとの間で TCP セッションを確立するための、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。

BGP ではパスベクトル ルーティング アルゴリズムを使用して、BGP 対応ネットワーク デバイスまたは *BGP スピーカ*間でルーティング情報を交換します。各 BGP スピーカはこの情報を使用して、特定の宛先までのパスを判別し、なおかつルーティング ループを伴うパスを検出して回避します。ルーティング情報には、宛先の実際のルート プレフィクス、宛先に対する AS（自律システム）のパス、およびその他のパス アトリビュートが含まれます。

BGP はデフォルトで、宛先ホストまたはネットワークへのベスト パスとして、1 つだけパスを選択します。各パスは、BGP ベストパス分析で使用される well-known 必須、well-known ディスクレショナリ、およびオプションのトランジット アトリビュートを伝送します。BGP ポリシーを設定し、これらのアトリビュートの一部を変更することによって、BGP パス選択を制御できます。詳細については、「ルート ポリシーおよび BGP セッションのリセット」（p.10-3）を参照してください。

BGP は、ロード バランシングまたは ECMP（等コスト マルチパス）もサポートします。詳細については、「ロード シェアリングおよびマルチパス」（p.10-7）を参照してください。

ネットワークにベーシック BGP を導入して設定するには、次の概念を理解しておく必要があります。

- BGP AS（p.9-2）
- 管理ディスタンス（p.9-2）
- BGP ピア（p.9-3）
- BGP ルータ ID（p.9-3）
- BGP パス選択（p.9-3）
- BGP およびユニキャスト RIB（p.9-6）
- BGP の仮想化（p.9-6）

BGP AS

AS（*自律システム*）は、単一の技術管理エンティティによって制御されるネットワークです。「自律システム」（p.1-5）を参照してください。AS は 1 つまたは複数の IGP および整合性のある一連のルーティング ポリシーを使用して、ルーティング ドメインを形成します。BGP は 16 ビットおよび 32 ビットの AS 番号をサポートします。

個々の BGP AS は external BGP（eBGP; 外部 BGP）ピアリング セッションを通じて、ルーティング情報をダイナミックに交換します。同じ AS 内の BGP スピーカは、internal BGP（iBGP; 内部 BGP）を通じて、ルーティング情報を交換できます。

管理ディスタンス

*管理ディスタンス*は、ルーティング情報源の信頼性に関する格付けです。「管理ディスタンス」（p.1-7）を参照してください。

BGP はデフォルトで、表 9-1 の管理ディスタンスを使用します。

表 9-1　デフォルトの BGP 管理ディスタンス

ディスタンス	デフォルト値	機能
外部	20	eBGP から学習したルートに適用
内部	200	iBGP から学習したルートに適用
ローカル	200	ルータを起点とするルートに適用

（注） 管理ディスタンスが BGP パス選択アルゴリズムに影響を与えることはありませんが、BGP で学習されたルートが IP ルーティング テーブルに組み込まれるかどうかを左右します。

BGP ピア

BGP スピーカが別の BGP スピーカを自動的に検出することはありません。ユーザ側で BGP スピーカ間の関係を設定する必要があります。*BGP ピア*は、もう 1 つの BGP スピーカとの間にアクティブな TCP 接続が存在する BGP スピーカです。

BGP は TCP ポート 179 を使用して、ピアとの TCP セッションを作成します。ピア間で TCP 接続が確立されると、各 BGP ピアは最初に相手と、それぞれのすべてのルートを交換し、BGP ルーティング テーブルを完成させます。初期交換以後、BGP ピアはネットワーク トポロジが変化したとき、またはルーティング ポリシーが変更されたときに、差分アップデートだけを送信します。このようなアップデートからアップデートまでの非アクティブ期間中に、ピアは*キープアライブ*という特殊なメッセージを交換します。*ホールド タイム*は、次の BGP アップデートまたはキープアライブ メッセージを受信するまでに経過することが許容される、最大時間限度です。

BGP ルータ ID

ピア間で BGP セッションを確立するには、BGP に*ルータ ID* を与える必要があります。ルータ ID は BGP セッションの確立時に、OPEN メッセージで BGP ピアに送信されます。BGP ルータ ID は 32 ビット値であり、IPv4 アドレスで表すことがよくあります。ルータ ID はユーザ側で設定できます。ルータ ID はデフォルトで、Cisco NX-OS によってルータのループバック インターフェイスの IPv4 アドレスに設定されます。ルータ上でループバック インターフェイスが設定されていない場合は、ルータ上の物理インターフェイスに設定されている最大の IPv4 アドレスが BGP ルータ ID を表すものとして、ソフトウェアによって選択されます。

BGP にルータ ID が設定されていない場合、BGP ピアとのピアリング セッションを確立することはできません。

BGP パス選択

BGP は複数の送信元から、同じルートのアドバタイズメントを受信する可能性があります。BGP はベスト パスとして、パスを 1 つだけ選択します。BGP はパスを選択すると、そのパスを IP ルーティング テーブルに格納し、ピアにパスを伝達します。

所定のネットワークでパスが追加または削除されるたびに、ベストパス アルゴリズムが実行されます。ベストパス アルゴリズムは、ユーザが BGP 設定を変更した場合にも実行されます。BGP は所定のネットワークで使用できる一連の有効パスの中から、最適なパスを選択します。

Cisco NX-OS は 3 つの部分に分けて、BGP ベストパス アルゴリズムを実行します。

- パート 1 ― 2 つのパスを比較し、どちらが適切かを判別します（「パス ペアの比較」[p.9-4] を参照）。

- パート 2 ― すべてのパスを繰り返し、全体として最適なパスを選択するためにパスを比較する順序を決定します（「比較順序」[p.9-5] を参照）。
- パート 3 ― 新しいベスト パスを使用するに足るだけの差が新旧のベスト パスにあるかどうかを判別します（「ベスト パス変更の抑制」[p.9-6] を参照）。

（注） 重要なのは、パート 2 で決定される比較順序です。A、B、C という 3 つのパスがあるとします。A と B を比較した場合、Cisco NX-OS は A を選択します。B と C を比較した場合、Cisco NX-OS は B を選択します。しかし、A と C を比較した場合、Cisco NX-OS は A を選択しません。これは一部の BGP メトリックが同じ隣接 AS からのパスだけに適用され、すべてのパスにわたっては適用されないからです。

パス選択には、BGP AS パス アトリビュートが使用されます。AS パス アトリビュートには、アドバタイズされたパスでたどる AS 番号のリストが含まれます。BGP AS を AS の集合または連合に細分化する場合は、AS パスにローカル定義の AS を指定した連合セグメントが含まれます。

パス ペアの比較

次に、Cisco NX-OS が 2 つのパスを比較して、より適切なパスを判別する基本的なステップについて説明します。

1. 比較する有効なパスを選択します（たとえば、到達不能なネクストホップがあるパスは無効です）。
2. 重み値が最大のパスを選択します。
3. ローカル プリファレンスが最大のパスを選択します。
4. パスの一方がローカル起点の場合、そのパスを選択します。
5. AS パス が短い方のパスを選択します。

（注） AS パス長を計算するときに、Cisco NX-OS は連合セグメントを無視し、AS セットを 1 として数えます。詳細については、「AS 連合」（p.10-5）を参照してください。

6. オリジンが低い方のパスを選択します。IGP は EGP より低いとみなされます。
7. multi exit discriminator（MED）が小さい方のパスを選択します。

   このステップが実行されるされないを左右する、一連のオプションを選択できます。Cisco NX-OS が MED を比較するのは、通常、同じ AS のピアから両方のパスを受け取った場合です。それ以外の場合、Cisco NX-OS は MED の比較を省略します。

   パスのピア AS に関係なく、ベストパス アルゴリズムの MED 比較が必ず実行されるように、Cisco NX-OS を設定することもできます。詳細については、「ベストパス アルゴリズムの調整」（p.10-8）を参照してください。この設定を行わなかった場合、MED 比較が実行されるかどうかは次のように、比較する 2 つのパスの AS パス アトリビュートによって決まります。

   a. パスに AS パス または AS_SET から始まる AS パス がない場合、パスは内部であり、他の内部パスに対して MED が比較されます。
   b. AS パスが AS_SEQUENCE から始まる場合、ピア AS がシーケンスで最初の AS 番号になり、同じピア AS を持つ他のパスに対して MED が比較されます。
   c. AS パス に連合セグメントだけが含まれている場合、または連合セグメントで始まり、AS_SET が続いている場合、パスは内部であり、他の内部パスに対して MED が比較されます。

d. AS パスが連合セグメントで始まり、AS_SEQUENCE が続いている場合、ピア AS が AS_SEQUENCE で最初の AS 番号になり、同じピア AS を持つ他のパスに対して MED が比較されます。

(注) Cisco NX-OS がパスの指定された MED アトリビュートを受信しなかった場合、欠落 MED が使用可能な最大値になるように、ユーザがベストパス アルゴリズムを設定していない限り、Cisco NX-OS は MED を 0 とみなします。詳細については、「ベストパス アルゴリズムの調整」（p.10-8）を参照してください。

e. 非決定性の MED 比較機能がイネーブルの場合、ベスト パス アルゴリズムでは IOS スタイルの MED 比較が使用されます。詳細については、「ベストパス アルゴリズムの調整」（p.10-8）を参照してください。

8. 一方のパスが内部ピアから、他方のパスが外部ピアからの場合、外部ピアからのパスが選択されます。
9. ネクストホップ アドレスへの IGP メトリックが異なるパスの場合、IGP メトリックが小さい方のパスを選択されます。
10. 最後に実行したベストパス アルゴリズムによって選択されたパスが使用されます。

ステップ 1 ～ 9 のすべてのパス パラメータが同じ場合、ルータ ID を比較するようにベスト パス アルゴリズムを設定できます。詳細については、「ベストパス アルゴリズムの調整」（p.10-8）を参照してください。パスに送信元アトリビュートが含まれている場合、Cisco NX-OS はそのアトリビュートをルータ ID として使用して比較します。発信もとアトリビュートが含まれていない場合、Cisco NX-OS はパスを送信したピアの ルータ ID を使用します。パス間でルータ ID が異なる場合、Cisco NX-OS はルータ ID が小さい方のパスを選択します。

(注) 送信元をルータ ID として使用する場合は、2 つのパスに同じルータ ID を与えることができます。また、同じピア ルータとの 2 つの BGP セッションが可能です。したがって、同じルータ ID で 2 つのパスを受信できます。

11. クラスタ長が短い方のパスが選択されます。クラスタ リスト アトリビュートの指定されたパスを受け取らなかった場合、クラスタ長は 0 です。
12. IP アドレスが小さい方のピアから受信したパスが選択されます。ローカル発生のパス（再配布のパスなど）は、ピア IP アドレスが 0 になります。

(注) ステップ 9 以降が同じパスは、マルチパスを設定している場合、マルチパスに使用できます。詳細については、「ロード シェアリングおよびマルチパス」（p.10-7）を参照してください。

比較順序

BGP ベストパス アルゴリズム実行の 2 番めの部分で、パスの比較順序が決定されます。Cisco NX-OS は次のように、比較順序を決定します。

1. 各グループ内ではすべてのパスにわたって MED を比較できるような形で、パスをグループに分けます。「パス ペアの比較」（p.9-4）と同じ規則を使用して、2 つのパス間で MED を比較できるかどうかが決定されます。この比較では通常、ネイバー AS ごとに 1 つずつグループができます。***bgp bestpath med always*** コマンドを設定した場合、すべてのパスが含まれたグループが 1 つだけ存在することになります。

2. 常に最適な方を維持しながら、グループのすべてのパスを反復することによって、各グループのベスト パスが決定されます。各パスをそれまでの一時的なベスト パスと比較し、それまでのベスト パスより適切な場合は、そのパスが新しく一時的なベスト パスになり、グループの次のパスとの比較が行われます。
3. ステップ 2 の各グループで選択されたベスト パスからなる、パス セットが形成されます。このパス セットでもステップ 2 と同様、それぞれの比較を繰り返すことによって、全体としてのベスト パスが選択されます。

ベスト パス変更の抑制

アルゴリズム実行の次の部分は、新しいベスト パスを使用するかどうかの決定です。ベストパス選択アルゴリズムが任意になるまで、新しいベスト パスがまったく同じ場合、ルータは引き続き既存のベスト パスを使用できます（ルータ ID が同じ場合）。引き続き既存のベスト パスを使用することによって、ネットワークにおけるルート変更を回避できます。

抑制をオフにするには、ルータ ID を比較するようにベスト パス アルゴリズムを設定します。詳細については、「ベストパス アルゴリズムの調整」（p.10-8）を参照してください。この機能を設定すると、新しいベスト パスがつねに既存のベスト パスより優先されます。

この機能を設定しないと、次の条件が発生した場合に、ベスト パス変更を抑制できません。

- 既存のベスト パスが無効になった。
- 既存または新しいベスト パスを内部（または連合）ピアから受信したか、またはローカルに発生した（再配布などによって）。
- 同じピアからパスを受信した（パスのルータ ID が同じ）。
- パス間で重み値、ローカル プリファレンス、オリジン、またはネクストホップ アドレスに対する IGP メトリックが異なっている。
- パス間で MED が異なっている。

BGP およびユニキャスト RIB

BGP はユニキャスト RIB（ルーティング情報ベース）と通信して、ユニキャスト ルーティング テーブルに IPv4 ルートを格納します。ベスト パスの選択後、ベスト パスの変更をルーティング テーブルに反省させる必要があると BGP が判別した場合、BGP はユニキャスト RIB にルート アップデートを送信します。

BGP はユニキャスト RIB における BGP ルートの変更に関して、ルート通知を受け取ります。さらに、再配布をサポートする他のプロトコル ルートに関するルート通知を受け取ります。

BGP はネクストホップの変更に関する通知も、ユニキャスト RIB から受け取ります。BGP はこれらの通知を使用して、ネクストホップ アドレスへの到達可能性および IGP メトリックを追跡します。

ユニキャスト RIB でネクストホップ到達可能性または IGP メトリックが変更されるたびに、BGP は影響を受けるルートについて、ベスト パス再計算を開始させます。

BGP は IPv6 ユニキャスト RIB と通信し、IPv6 ルートについて、これらの動作を実行します。

BGP の仮想化

BGP は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

ベーシック BGP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	BGP には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細、ライセンスを取得して適用する方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

BGP の前提条件

BGP の前提条件は、次のとおりです。

- BGP 機能をイネーブルにする必要があります（「BGP 機能のイネーブル化」[p.9-10] 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。
- 再帰ネクストホップ解決に対応できる IGP を 1 つ以上設定する必要があります。
- BGP セッションを確立するネイバー環境で、アドレス ファミリを設定する必要があります。

BGP に関する注意事項および制限事項

BGP を設定するときには、次の注意事項および制約事項に従ってください。

- 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、BGP/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 ポリシーを指定します。
- VRF 内で BGP ルータ ID を定義します。
- 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、BGP セッション フラップが発生する可能性があります。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。
- VRF を設定する場合は、Advanced Services ライセンスをインストールし、所定の VRF を開始してください（第 13 章「レイヤ 3 仮想化の設定」を参照）。

CLI コンフィギュレーション モード

ここでは BGP に対応する各 CLI コンフィギュレーション モードの開始方法について説明します。各モードから **?** コマンドを入力すると、そのモードで使用できるコマンドが表示されます。

グローバル コンフィギュレーション モード

グローバル コンフィギュレーション モードは、BGP プロセスを作成したり、AS 連合、ルート ダンプニングなどの拡張機能を設定したりする場合に使用します。第 10 章「拡張 BGP の設定」を参照してください。ルータ コンフィギュレーション モードを開始する例を示します。

```
switch# configuration
switch(config)# router bgp 5
switch(config-router)#
```

BGP は VRF（仮想ルーティングおよびフォワーディング）をサポートします。ネットワークで VRF を使用する場合は、適切な VRF 内で BGP を設定してください。詳細については、「仮想化の設定」（p.10-33）を参照してください。

VRF コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 7
switch(config-router)# vrf context vrf_A
switch(config-router-vrf)#
```

アドレス ファミリ コンフィギュレーション モード

任意で、BGP がサポートするアドレス ファミリを設定できます。アドレス ファミリ用の機能を設定する場合は、ルータ コンフィギュレーション モードで ***address-family*** コマンドを使用します。ネイバーに対応する特定のアドレス ファミリを設定する場合は、ネイバー コンフィギュレーション モードで ***address-family*** コマンドを使用します。

ルート再配布、アドレス集約、ロード バランシング、およびその他の拡張機能を使用する場合は、アドレス ファミリを設定する必要があります。ルータ コンフィギュレーション モードからアドレス ファミリ コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 5
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)#
```

VRF を使用している場合に、VRF アドレス ファミリ コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 7
switch(config-router)# vrf context vrf_A
switch(config-router-vrf)# address-family ipv4 unicast
switch(config-router-vrf-af)#
```

ネイバー コンフィギュレーション モード

Cisco NX-OS には、BGP ピアを設定するためのネイバー コンフィギュレーション モードがあります。ネイバー コンフィギュレーション モードを使用して、ピアのあらゆるパラメータを設定します。ネイバー コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 5
switch(config-router)# neighbor 192.0.2.1
switch(config-router-neighbor)#
```

VRF ネイバー コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 7
switch(config-router)# vrf context vrf_A
switch(config-router-vrf)# neighbor 192.0.2.1
switch(config-router-vrf-neighbor)#
```

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

アドレス ファミリ固有のネイバー設定を入力し、ネイバーのアドレス ファミリをイネーブルにするには、ネイバー コンフィギュレーション サブモード内のアドレス ファミリ コンフィギュレーション サブモードを使用できます。このモードは、所定のネイバーに認められるプレフィクス数の制限、EBGP のプライベート AS 番号の削除といった拡張機能に使用します。ネイバー アドレス ファミリ コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 5
switch(config-router# neighbor 192.0.2.1
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)#
```

VRF ネイバー アドレス ファミリ コンフィギュレーション モードを開始する例を示します。

```
switch(config)# router bgp 7
switch(config-router)# vrf context vrf_A
switch(config-router-vrf)# neighbor 209.165.201.1
switch(config-router-vrf-neighbor)# address-family ipv4 unicast
switch(config-router-vrf-neighbor-af)#
```

ベーシック BGP の設定

ベーシック BGP を設定するには、BGP をイネーブルにして、BGP ピアを設定する必要があります。ベーシック BGP ネットワークの設定は、いくつかの必須作業と多数の任意の作業からなります。BGP ルーティング プロセスおよび BGP ピアの設定は必須です。

ここでは、次の内容について説明します。

- BGP 機能のイネーブル化（p.9-10）
- BGP インスタンスの作成（p.9-11）
- BGP インスタンスの再起動（p.9-12）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

BGP 機能のイネーブル化

BGP を設定するには、BGP 機能をイネーブルにしておく必要があります。

準備作業

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）。

手順概要

1. **config t**
2. ***feature bgp***
3. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature bgp** **例：** `switch(config)# feature bgp`	BGP 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

BGP 機能をディセーブルにして、関連するすべての設定を削除する場合は、***no feature bgp*** コマンドを使用します。

コマンド	目的
no feature bgp **例：** `switch(config)# no feature bgp`	BGP 機能をディセーブルにして、関連するすべての設定を削除します。

BGP インスタンスの作成

BGP インスタンスを作成し、BGP インスタンスにルータ ID を割り当てることができます。「BGP ルータ ID」（p.9-3）を参照してください。Cisco NX-OS は、2 バイトまたは 4 バイトの AS（自律システム）番号をサポートします。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] を参照）。

BGP はルータ ID（設定済みループバック アドレスなど）を取得できなければなりません。

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）。

手順概要

1. **config t**
2. **router bgp** *autonomous-system-number*
3. ***router-id*** *ip-address*
4. **address-family** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
5. ***network*** *ip-prefix* [***route-map*** *map-name*]
6. **show bgp all**
7. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 40000`
	`switch(config-router)#`
		BGP をイネーブルにして、ローカル BGP スピーカに AS 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。形式は < 上位 16 ビット 10 進数 >.< 下位 16 ビット 10 進数 > です。

ステップ 3	**router-id** *ip-address* **例：** `switch(config-router)# router-id 192.0.2.255`	（任意）BGP ルータ ID を設定します。この IP アドレスによって、この BGP スピーカを特定します。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。

	コマンド	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)#`
		（任意）IPv4 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。このコマンドによって、すべての 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	***network*** *ip-prefix* [***route-map*** *map-name*] **例：** `switch(config-router-af)# network 192.0.2.0`	（任意）この AS にローカルとしてネットワークを指定し、BGP ルーティング テーブルに追加します。 エクステリア プロトコルの場合、network コマンドでアドバタイズするネットワークを制御します。インテリア プロトコルでは、network コマンドを使用して、アップデートの送信先を決定します。
ステップ 6	**show bgp all** **例：** `switch(config-router-af)# show bgp all`	（任意）すべての BGP アドレス ファミリに関する情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-router-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

BGP プロセスおよび関連するすべての設定を削除する場合は、***no router bgp*** コマンドを使用します。

コマンド	目的
no router bgp *autonomous-system-number* **例：** `switch(config)# no router bgp 201`	BGP プロセスおよび関連する設定を削除します。

IPv4 ユニキャスト アドレス ファミリを指定して BGP をイネーブルに設定し、アドバタイズするネットワークを 1 つ追加する例を示します。

```
switch# config t
switch(config)# router bgp 40000
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# network 192.0.2.0
switch(config-router-af)# copy running-config startup-config
```

BGP インスタンスの再起動

BGP インスタンスの再起動が可能です。再起動すると、インスタンスのすべてのピア セッションが消去されます。

BGP インスタンスを再起動し、関連付けられたすべてのピアを削除するには、次のコマンドを使用します。

コマンド	目的
restart bgp instance-tag **例：** switch(config)# restart bgp 201	BGP インスタンスを再起動し、すべてのピアリング セッションをリセットまたは再確立します。

BGP ピアの設定

BGP プロセス内で BGP ピアを設定できます。BGP ピアごとに、関連付けられたキープアライブ タイマーとホールド タイマーがあります。これらのタイマーは、グローバルに設定することも、BGP ピアごとに設定することもできます。ピア設定はグローバル設定を上書きします。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

手順概要

1. **config t**
2. **router bgp** *autonomous-system-number*
3. **neighbor** *ip-address* **remote-as** *as-number*
4. ***description*** *text*
5. ***timers*** *keepalive-time hold-time*
6. **address-family** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
7. **show bgp** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***} **neighbors**
8. **copy running-config startup-config**

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 40000 switch(config-router)#	BGP をイネーブルにして、ローカル BGP スピーカに AS 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 209.165.201.1 remote-as 45000 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	目的

ステップ 4	***description*** *text*
	
	例：
	`switch(config-router-neighbor)# description Peer Router B`
	`switch(config-router-neighbor)#`
		（任意）ネイバーの説明を追加します。最大 80 文字の英数字ストリングを使用できます。

ステップ 5	**timers** *keepalive-time hold-time* **例：** `switch(config-router-neighbor)# timers 30 90`	（任意）ネイバーのキープアライブおよびホールド タイムを表す BGP タイマー値を追加します。有効値の範囲は 0 ～ 3600 秒です。デフォルトは、キープアライブ タイムで 60 秒、ホールド タイムで 180 秒です。
ステップ 6	**shutdown** **例：** `switch(config-router-neighbor)# shutdown`	（任意）この BGP ネイバーを管理目的でシャットダウンします。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	**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 8	**show bgp** {***ipv4*** | ***ipv6***}{***unicast*** | ***multicast***} **neighbors** **例：** `switch(config-router-neighbor-af)# show bgp ipv4 unicast neighbors`	（任意）GBP ピアの情報を表示します。
ステップ 9	***copy running-config startup-config*** **例：** `switch(config-router-neighbor-af) copy running-config startup-config`	（任意）この設定変更を保存します。

BGP ピアの設定例を示します。

```
switch# config t
switch(config)# router bgp 40000
switch(config-router)# neighbor 192.0.2.1 remote-as 45000
switch(config-router-neighbor)# description Peer Router B
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# copy running-config startup-config
```

ベーシック BGP の設定確認

BGP の設定を確認するには、次のコマンドを使用します。

コマンド	目的
show bgp [**vrf** *vrf-name*] **all** [**summary**]	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 [**vrf** *vrf-name*] **convergence**	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community** {**regexp** | [**community**] [**no-advertise**] [**no-export**] [**no-export-subconfed**]}	BGP コミュニティと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community-list** *list-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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**damp-params** | **dampened-paths**}	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **history-paths**	BGP ルート ヒストリ パスを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **filter-list** *list-name*	BGP フィルタ リストの情報を表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **neighbors** [*ip-address* | *ipv6-prefix*]	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**nexthop** | **nexthop-database**}	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **policy** *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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **prefix-list** *list-name*	プレフィクス リストと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **received-paths**	ソフト再構成用に保存されている BGP パスを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **regexp** *expression*	AS_path 正規表現と一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **route-map** *map-name*	ルート マップと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] **peer-policy** *name*	BGP ピア ポリシー情報を表示します。
show bgp [**vrf** *vrf-name*] **peer-session** *name*	BGP ピア セッション情報を表示します。
show bgp [**vrf** *vrf-name*] **peer-template** *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 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 統計情報の表示

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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **flap-statistics**	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消去するには、**clear bgp flap-statistics** コマンドを使用します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **neighbors** [*ip-address* | *ipv6-prefix*]	BGP ピアの統計情報を表示します。これらの統計情報を消去するには、**clear bgp neighbors** コマンドを使用します。
show bgp [**vrf** *vrf-name*] **sessions**	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消去するには、**clear bgp sessions** コマンドを使用します。

ベーシック BGP の設定例

ベーシック BGP の設定例を示します。

```
feature bgp
router bgp 40000
  address-family ipv4 multicast
    network 192.0.2.0/24
    network 209.165.201.0/27
  address-family ipv4 unicast
    network 192.0.2.0/24
    network 209.165.201.0/27
   address-family ipv6 multicast
    network 2001::0DB8::/64
    network 2001::0DB8:0:1::/64
  address-family ipv6 unicast
     network 2001:0DB8::/64
    network 2001:0DB8:0:1::/64
  neighbor 2001:ODB8:0:1::55 remote-as 30
    address-family ipv6 multicast
    address-family ipv6 unicast
  neighbor 209.165.201.1 remote-as 45000
    address-family ipv4 multicast
    address-family ipv4 unicast
```

関連項目

BGP の関連項目は、次のとおりです。

- 第 14 章「Route Policy Manager の設定」

関連情報

次の機能の詳細について、第 10 章「拡張 BGP の設定」を参照してください。

- ピア テンプレート
- ルートの再配布
- ルート マップ

デフォルト設定

表 9-2 に、BGP パラメータのデフォルト設定を示します。

表 9-2　デフォルトの BGP パラメータ

パラメータ	デフォルト
BGP 機能	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

その他の関連資料

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。

- 関連資料（p.9-18）
- MIB（p.9-18）

関連資料

関連項目	マニュアル名
BGP CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

MIB

MIB	MIB リンク
BGP4-MIB CISCO-BGP4-MIB	MIB を検索してダウンロードするには、次の URL にアクセスしてください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

技術サポート

説明	リンク
TAC のホーム ページには、製品リンク、テクノロジー、ソリューション、テクニカル ティップス、ツールを含め、30,000 ページに及ぶ検索可能な技術コンテンツがあります。Cisco.com の登録ユーザは、このページからログインして、さらに多くのコンテンツを利用できます。	http://www.cisco.com/public/support/tac/home.shtml

CHAPTER 10

拡張 BGP の設定

この章では、BGP（ボーダー ゲートウェイ プロトコル）の拡張機能を設定する方法について説明します。

ここでは、次の内容を説明します。

- 拡張 BGP の概要（p.10-2）
- 拡張 BGP のライセンス要件（p.10-11）
- BGP の前提条件（p.10-11）
- BGP に関する注意事項および制限事項（p.10-11）
- 拡張 BGP の設定（p.10-12）
- 拡張 BGP の設定確認（p.10-35）
- BGP 統計情報の表示（p.10-36）
- 関連項目（p.10-36）
- デフォルト設定（p.10-36）
- デフォルト設定（p.10-36）
- その他の関連資料（p.10-37）

拡張 BGP の概要

BGP は、組織または AS（自律システム）間のループフリー ルーティングを提供する、ドメイン間ルーティング 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の、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 external BGP（eBGP; 外部 BGP）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、internal BGP（iBGP; 内部 BGP）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。

ここでは、次の内容について説明します。

- ピア テンプレート（p.10-2）
- 認証（p.10-3）
- ルート ポリシーおよび BGP セッションのリセット（p.10-3）
- eBGP（p.10-4）
- iBGP（p.10-4）
- 機能ネゴシエーション（p.10-6）
- AS 連合（p.10-5）
- ルータ リフレクタ（p.10-5）
- ルート ダンプニング（p.10-6）
- ロード シェアリングおよびマルチパス（p.10-7）
- ルート集約（p.10-7）
- ルートの再配布（p.10-8）
- BGP の調整（p.10-8）
- マルチプロトコル BGP（p.10-8）
- グレースフル リスタートおよびハイ アベイラビリティ（p.10-9）
- ピア テンプレート（p.10-2）

ピア テンプレート

BGP ピア テンプレートを使用すると、共通のコンフィギュレーション ブロックを作成し、類似している BGP ピア間で再利用できます。各ブロックで、ピアに継承させる一連のアトリビュートを定義します。継承したアトリビュート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 AS 番号、セッション タイマーといった BGP セッション アトリビュートを定義します。peer-session テンプレートは、別の peer-session テンプレートからアトリビュートを継承することもできます（ローカル定義のアトリビュートによって、継承した peer-session アトリビュートは上書きされます）。
- *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） BGP ピア間で MD5 パスワードを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クス、AS_path アトリビュートなど、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 アトリビュートを変更することもできます。ルート ポリシーの詳細については、第 15 章「ポリシーベース ルーティングの設定」を参照してください。

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 ピアリング セッションのリセット方法として、次の 3 種類をサポートします。

- 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 はピアに、プレフィクスの発信ルート リフレッシュを自動的に送信します。
- 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、およびその他の機能にルート マップを使用します。ルート マップの詳細については、第 14 章「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。

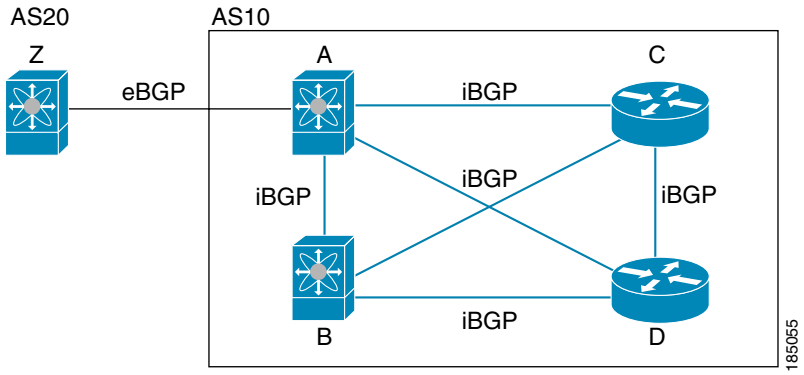
eBGP ピアリング セッションの確立には、ループバック インターフェイスを使用します。ループバック インターフェイスは、インターフェイス フラップが発生する可能性が小さいからです。インターフェイス フラップが発生するのは、障害またはメンテナンスが原因で、インターフェイスが管理上アップまたはダウンになったときです。マルチホップ、高速外部フェールオーバー、および一般的な TTL（存続可能時間）セキュリティ メカニズム サポートについては、「eBGP の設定」（p.10-22）を参照してください。

iBGP

iBGP を使用すると、同じ AS 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 AS 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 10-1 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 10-1　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。

(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。

ここでは、次の内容について説明します。

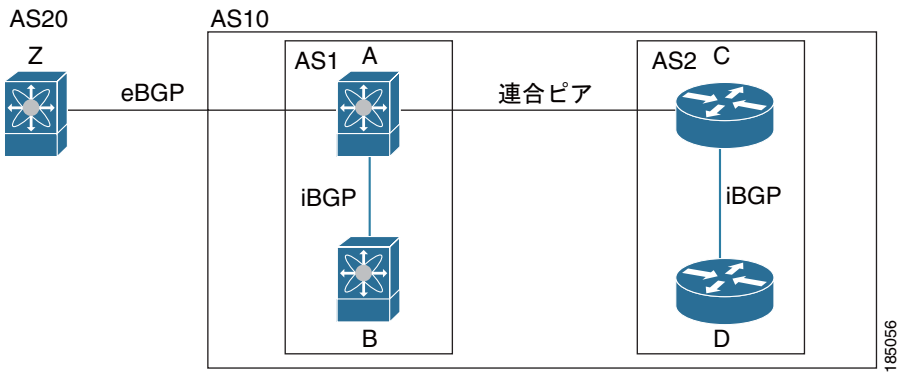
- AS 連合（p.10-5）
- ルータ リフレクタ（p.10-5）

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。AS を複数のサブ AS 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 AS 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 10-2 に、図 10-1 の BGP ネットワークを 2 つのサブ AS に分割し、1 つの連合にしたものを示します。

図 10-2　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、図 10-1 のフルメッシュ AS に比べて、リンク数を少なくできます。

ルータ リフレクタ

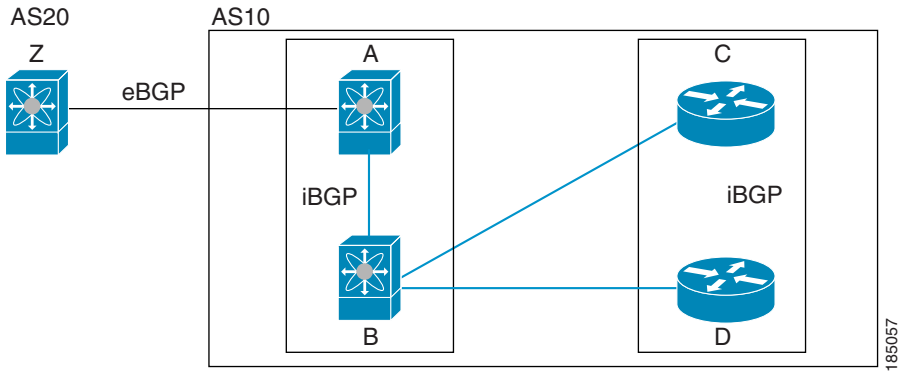
ルータ リフレクタ構成を使用することによって、iBGP メッシュを緩和することもできます。ルータ リフレクタは学習したルートをネイバーに渡し、すべての iBGP ピアをフルメッシュにしなくてもすむようにします。

図 10-1 に、メッシュの iBGP スピーカを 4 つ使用する（ルータ A、B、C、および D）、単純な iBGP 構成を示します。ルータ 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。

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 10-3 では、ルータ B がルータ リフレクタです。ルータ A からアドバタイズされたルートを受信したルート リフレクタは、そのルートをルータ C および D にアドバタイズ（リフレクション）します。ルータ A からルータ C および D の両方にアドバタイズする必要はなくなります。

図 10-3　ルータ リフレクタ



ルータ リフレクタおよびクライアント ピアは、クラスタを形成します。ルータ 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ことを保証するために、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、ピアがサポートする BGP 拡張機能について学習できます。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 ルートの伝播を最小限に抑える 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。

3 つの BGP AS からなるネットワークの場合について考えてみます。それぞれ AS1、AS2、および AS3 です。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 AS 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、再利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。再利用限度が 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、次のアトリビュートが同じ場合に、等コスト パスとみなされます。

- 重み値
- 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。

（注） 異なる AS 連合から受け取ったパスは、外部 AS_path 値およびその他のアトリビュートが同じ場合に、等コスト パスとみなされます。

（注） iBGP マルチパスに関してルータ リフレクタを設定すると、ルータ リフレクタがピアに選択されたベスト パスを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。

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。

アドバタイズするルートが少なくなるように、BGP ルート テーブルでは集約プレフィクスを使用します。

（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 はサマリー廃棄の管理ディスタンスを 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。

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。再配布を指定してルート ポリシーを設定し、BGP に渡されるルートを制御します。ルート ポリシーでは、宛先、送信元プロトコル、ルート タイプ、ルート タグなどのアトリビュートに基づいて、ルートをフィルタリングできます。詳細については、第 14 章「Route Policy Manager の設定」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。

ここでは、次の内容について説明します。

- BGP タイマー（p.10-8）
- ベストパス アルゴリズムの調整（p.10-8）

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、アルゴリズムでの MED アトリビュートおよびルータ ID の扱い方を変更することを含め、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。

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（MBGP）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 ルート用のルート セットを 1 つ伝送できます。

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 コンフィギュレーション モードを使用します。

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。

MGMP を使用するマルチキャストの設定例については、『*Cisco Cisco NX-OS Multicast Configuration Guide*』を参照してください。

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 の無停止フォワーディングおよびグレースフル リスタートをサポートします。

BGP の無停止フォワーディングを使用すると、FIB（転送情報ベース）の既知のルートでデータ パケットを転送し、なおかつフェールオーバー後に BGP ルーティング プロトコル情報を復元できます。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にトラフィックを転送しないで、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 はスタートアップ コンフィギュレーションを適用し、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。

デュアル スーパーバイザ構成の Cisco NX-OS 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実行前に、グレースフル リスタートが始まり、BGP がしばらく使用できなくなることが BGP によって告知されます。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。リスタートされたルータは、ピアからのルートを「stale」（古い）として明示します。

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。

スイッチオーバー後、Cisco NX-OS によって実行コンフィギュレーションが適用され、BGP は再び動作可能になったことをネイバーに通知します。

ISSU

BGP の ISSU（インサービス ソフトウェア アップグレード）をサポートするには、グレースフル リスタートをイネーブルにする必要があります。

BGP ではピア ホールド タイマーを使用して、非アクティブになったピアのセッションを切断し、応答を中止します。ISSU プロセスの一部として、スイッチオーバー中に BGP コントロール パケットが受信または送信されないことがあり、ピアがキープアライブ メッセージ損失を認識する可能性があります。ただし、ホールド タイムをスイッチオーバー タイムより長くしておくと、ピアがローカル ルータとのセッションを切断することはありません。

スイッチオーバーが発生すると、ピアはローカル ルータ上の新しいアクティブ TCP から TCP 接続リセットを受信します。グレースフル リスタートがイネーブルになっている場合、ピアはルータが再起動したという表示としてリセットを扱い、グレースフル リスタート ヘルパー手順を開始します。

システム スイッチオーバー タイム（約 15 秒）に満たないホールド タイムが設定されている場合、Cisco NX-OS は ISSU を保証できません。

BGP による ISSU サポートは、次のとおりです。

- グレースフル リスタートをディセーブルにすると、この設定では ISSU をサポートできないことを伝える警告が Cisco NX-OS から出されます。
- システム スイッチオーバー タイムに満たないホールド タイムを設定すると、Cisco NX-OS から同様の警告が出されます。ピアが短縮ホールド タイムのネゴシエーションを行うと、Cisco NX-OS がメッセージを記録します。
- スイッチオーバーの前に、Cisco NX-OS が BGP ISSU 関連のコールバック ルーチンを実行した場合は、BGP がすべてのアクティブ ピアについて、グレースフル リスタート ステータスおよびホールド タイムの両方を確認します。グレースフル リスタートがディセーブルになっている場合、またはアクティブ ピアのホールド タイムがシステム スイッチオーバー タイムより短い場合は、Cisco NX-OS が該当する警告を発行し、スイッチオーバーの強制についてはユーザの裁量にまかせます。
- ISSU　がサポートされない場合、スイッチオーバーを強制実行できますが、この強制スイッチオーバー中の転送は維持されません。

仮想化サポート

Cisco NX-OS は、同一システム上で動作する複数の BGP プロトコル インスタンスをサポートします。BGP は、virtual device context（仮想デバイス コンテキスト ; VDC）内に存在する VRF（仮想ルーティングおよびフォワーディング）をサポートします。VDC で設定できる BGP インスタンスは 1 つですが、システム上では複数の VDC を使用できます。

特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

拡張 BGP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	BGP には Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細、ライセンスを取得して適用する方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

BGP の前提条件

BGP の前提条件は、次のとおりです。

- BGP 機能をイネーブルにする必要があります（「BGP 機能のイネーブル化」[p.9-10] 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（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。

BGP に関する注意事項および制限事項

BGP に関する注意事項および制約事項は、次のとおりです。

- 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。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

拡張 BGP の設定

ここでは、拡張 BGP の設定方法について説明します。内容は、次のとおりです。

- BGP セッション テンプレートの設定（p.10-12）
- BGP peer-policy テンプレートの設定（p.10-15）
- BGP ピア テンプレートの設定（p.10-17）
- プレフィクス ピアリングの設定（p.10-19）
- BGP 認証の設定（p.10-20）
- BGP セッションのリセット（p.10-20）
- AS 連合の設定（p.10-23）
- 機能ネゴシエーションのディセーブル化（p.10-21）
- eBGP の設定（p.10-22）
- AS 連合の設定（p.10-23）
- ルータ リフレクタの設定（p.10-23）
- ルート ダンプニングの設定（p.10-25）
- ロード シェアリングおよび ECMP の設定（p.10-26）
- 最大プレフィクス数の設定（p.10-26）
- ダイナミック機能の設定（p.10-27）
- 集約アドレスの設定（p.10-27）
- ルートの再配布の設定（p.10-27）
- BGP の調整（p.10-29）
- グレースフル リスタートの設定（p.10-31）
- 仮想化の設定（p.10-33）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

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、そのあとで 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 アトリビュートを設定できます。

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第 2 テンプレートを設定し、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アトリビュートは、ネイバーが BGP テンプレートから継承したアトリビュートより優先されます。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

（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時を明示的に上書きできます。アトリビュート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。

手順概要

1. **config t**
2. **router bgp** *autonomous-system-number*
3. **template peer-session** *template-name*
4. セッション テンプレートに適切なアトリビュートを追加します。
5. **exit**
6. **neighbor** *ip-address* **remote-as** *as-number*
7. **inherit peer-session** *template-name*
8. 適切なネイバー アトリビュートを追加します。
9. **show bgp peer-session** *template-name*
10. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 45000`
	`switch(config-router)#`
		BGP をイネーブルにして、ローカル BGP スピーカに AS 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	**password** *number password* **例：** `switch(config-router-stmp)# password 0 test`	（任意）ネイバーにクリアテキスト 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。

	コマンド	目的
ステップ 5	**timers** *keepalive hold* **例：** `switch(config-router-stmp)# timers 30 90`	(任意)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 40000`
	`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	***description*** *text*
	
	例：
	`switch(config-router-neighbor)# description Peer Router A`
	`switch(config-router-neighbor)`
		(任意）ネイバーの説明を追加します。

ステップ 10	***show bgp peer-session*** *template-name* **例：** `switch(config-router-neighbor)# show bgp peer-session BaseSession`	(任意）peer-policy テンプレートを表示します。
ステップ 11	***copy running-config startup-config*** **例：** `switch(config-router-neighbor)# copy running-config startup-config`	(任意）この設定変更を保存します。

適用されたテンプレートを確認するには、**show bgp neighbor** コマンドを使用します。テンプレートで使用できるあらゆるコマンドの詳細については、『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0 を参照してください。

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# config t
switch(config)# router bgp 45000
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 40000
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)# 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アトリビュートを定義できます。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アトリビュートは、ネイバーが BGP テンプレートから継承したアトリビュートより優先されます。

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 ファミリ固有のアトリビュートを設定できます。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

（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時を明示的に上書きできます。アトリビュート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。

手順概要

1. **config t**
2. **router bgp** *autonomous-system-number*
3. **template peer-policy** *template-name*
4. ポリシー テンプレートに適切なアトリビュートを追加します。
5. **exit**
6. **neighbor** *ip-address* **remote-as** *as-number*
7. **address-family** {***ipv4*** | ***ipv6}*** ***{multicast*** | **unicast**}
8. **inherit peer-policy** *template-name preference*
9. **show bgp peer-policy** *template-name*
10. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 45000`
	`switch(config-router)#`
		BGP をイネーブルにして、ローカル BGP スピーカに AS 番号を割り当てます。

	コマンド	目的

ステップ 3	**template peer-policy** *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	**advertise-active-only** **例：** `switch(config-router-ptmp)# advertise-active-only`	（任意）アクティブ ルートだけをピアにアドバタイズします。
ステップ 5	**maximum-prefix** *number* **例：** `switch(config-router-ptmp)# maximum-prefix 20`	（任意）このピアに認めるプレフィ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 40000`
	`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)#`
		IPv4 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 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	***show bgp peer-policy*** *template-name* **例：** `switch(config-router-neighbor-af)# show bgp peer-policy BasePolicy`	（任意）peer-policy テンプレートを表示します。
ステップ 11	***copy running-config startup-config*** **例：** `switch(config-router-neighbor-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

適用されたテンプレートを確認するには、**show bgp neighbor** コマンドを使用します。テンプレートで使用できるあらゆるコマンドの詳細については、『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0 を参照してください。

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。

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# config t
switch(config)# router bgp 40000
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 45000
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 ピア 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 アトリビュートとポリシー アトリビュート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アトリビュートは、ネイバーが BGP テンプレートから継承したアトリビュートより優先されます。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、最大プレフィクス数、ネクストホップ セルフ、タイマーなど、セッション アトリビュートおよびアドレス ファミリ アトリビュートをサポートします。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

（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時を明示的に上書きできます。アトリビュート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。

手順概要

1. **config t**
2. **router bgp** *autonomous-system-number*
3. **template peer** *template-name*
4. ピア テンプレートに適切なアトリビュートを追加します。
5. **exit**
6. **neighbor** *ip-address*
7. **inherit peer** *template-name*
8. 適切なネイバー アトリビュートを追加します。
9. **show bgp peer-template** *template-name*
10. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 45000`	BGP モードを開始し、ローカル BGP スピーカに AS 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`
	`switch(config-router-neighbor)#`
		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	***inherit peer-session***n *template-name* **例：** `switch(config-router-neighbor)# inherit peer-session BaseSession`	（任意）peer テンプレートで peer-session テンプレートを継承します。

ステップ 5	**address-family** {***ipv4*** | ***ipv6***}{***multicast*** | ***unicast***}
	
	例：
	`switch(config-router-neighbor)# address-family ipv4 unicast`
	`switch(config-router-neighbor-af)#`
		（任意）IPv4 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 モードを設定します。

ステップ 6	***inherit peer-policy*** *template-name preference* **例：** `switch(config-router-neighbor-af)# inherit peer-policy BasePolicy 1`	（任意）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、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 7	**exit**
	
	例：
	`switch(config-router-neighbor-af)# exit`
	`switch(config-router-neighbor)#`
		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	**timers** *keepalive hold* **例：** `switch(config-router-neighbor)# timers 45 100`	（任意）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`
	`switch(config-router)#`
		BGP peer 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 10	**neighbor** *ip-address* **remote-as** *as-number*
	
	例：
	`switch(config-router)# neighbor 192.168.1.2 remote-as 40000`
	`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 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。

	コマンド	目的
ステップ 12	**timers** *keepalive hold* **例：** `switch(config-router-neighbor)# timers 60 120`	（任意）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	***show bgp peer-template*** *template-name* **例：** `switch(config-router-neighbor-af)# show bgp peer-template BasePeer`	（任意）peer テンプレートを表示します。
ステップ 14	***copy running-config startup-config*** **例：** `switch(config-router-neighbor-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

適用されたテンプレートを確認するには、**show bgp neighbor** コマンドを使用します。テンプレートで使用できるあらゆるコマンドの詳細については、『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0 を参照してください。

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。

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# config t
switch(config)# router bgp 45000
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 40000
switch(config-router-neighbor)# inherit peer BasePeer
switch(config-router-neighbor)# copy running-config startup-config
```

プレフィクス ピアリングの設定

BGP では IPv4 および IPv6 の両方のプレフィクスを使用して、ピア セットを定義できます。これにより、コンフィギュレーションに各ネイバーを追加する必要がなくなるので、テンプレートを使用する以上に設定を簡素化できます。

プレフィクス ピアリングを定義する場合は、プレフィ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かぎり、BGP はプレフィクスおよび AS から接続するピアを受け付けます。

プレフィクス 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るので、ネットワークの安定性が向上します。prefix-peer-time-out のデフォルト設定は 30 秒です。

BGP プレフィクス ピアリング タイムアウト値を設定するには、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。

コマンド	目的
timers prefix-peer-timeout *value* **例：** `switch(config-router-neighbor)# timers prefix-peer-timeout 120`	プレフィクス ピアリングのタイムアウト値を設定します。有効値の範囲は 0 ～ 1200 秒です。デフォルト値は 30 です。

ピアの最大数を設定するには、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。

コマンド	目的
maximum-peers *value* **例：** `switch(config-router-neighbor)# timers prefix-peer-timeout 120`	このプレフィクス ピアリングの最大ピア数を設定します。有効値の範囲は 1 ～ 1000 です。

最大 10 のピアを受け付けるプレフィクス ピアリングの設定例を示します。

```
switch(config)# router bgp 1
switch(config-router)# timers prefix-peer-timeout 120
switch(config-router)# neighbor 10.100.200.0/24 remote-as 1
switch(config-router-neighbor)# maximum-peers 10
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)#
```

所定のプレフィクス 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するには、**show ip bgp neighbor** コマンドを使用します。

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 アップデートを認証するように BGP を設定できます。

MD5 認証を使用するように 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。

コマンド	目的
password [**0** | **3** | **7**] *string* **例：** `switch(config-router-neighbor)# password BGPpassword`	MGP ネイバー セッションの MD5 パスワードを設定します。

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 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。

コマンド	目的
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。

BGP ネイバー セッションをリセットするには、任意のモードで次のコマンドを使用します。

コマンド	目的
clear bgp {***ip*** | ***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。この方法は、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。この場合、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 アドレスを変更するには、ネイバー アドレス ファミリ コンフィギュレーション モードで次のパラメータを設定します。

コマンド	目的
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。
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。

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。

コマンド	目的
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。

eBGP の設定

ここでは、次の内容について説明します。

- eBGP シングルホップ チェックのディセーブル化（p.10-22）
- eBGP マルチホップの設定（p.10-22）
- 高速外部フェールオーバーのディセーブル化（p.10-23）
- AS 連合の設定（p.10-23）

eBGP シングルホップ チェックのディセーブル化

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ディセーブルにするように、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。

コマンド	目的
disable-connected-check **例：** `switch(config-router-neighbor)# soft-reconfiguration inbound`	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。

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 TTL（存続可能時間）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。

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。

コマンド	目的
ebgp-multihop *ttl-value* **例：** `switch(config-router-neighbor)# ebgp-multihop 5`	eBGP マルチホップの eBGP TTL を設定します。有効値の範囲は 2 ～ 255 です。このコマンドを設定したあとで、BGP セッションを手動でリセットする必要があります。

高速外部フェールオーバーのディセーブル化

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 が高速外部フェールオーバーを開始します。この高速外部フェ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。

高速外部フェ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。

コマンド	目的
no fast-external-failover **例：** `switch(config-router)# no fast-external-failover`	eBGP ピアの高速外部フェールオーバーをディセーブルにします。イネーブルがデフォルトです。

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 AS グループは、AS 番号として連合 ID を持つ、1 つの AS 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 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。

コマンド	目的
confederation identifier *as-number* **例：** `switch(config-router)# confederation identifier 4000`	AS 連合を表す連合 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。

AS 連合に所属する AS を設定するには、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。

コマンド	目的
bgp confederation peers *as-number* [*as-number2...*] **例：** `switch(config-router)# bgp confederation peers 5 33 44`	連合に所属する AS のリストを指定します。このコマンドによって、BGP ネイバー セッションの自動通知およびセッション リセットが開始されます。

ルータ リフレクタの設定

ルータ 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タ リフレクタ クライアントとして、iBGP ピアを設定できます。ルータ 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、ルータ リフレクタが 1 つ存在します。このような状況では、ルータ 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、複数のルータ 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タ 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、ルータ リフレクタが同じクラスタ内のルータ 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。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

手順概要

1. **config t**
2. **router bgp** *as-number*
3. **cluster-id** *cluster-id*
4. **address-family** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
5. ***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. **show bgp** {**ip** | **ipv6**} {**unicast** | **multicast**} *as-number*
11. **copy running-config startup-config**

手順詳細

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 45000`
	`switch(config-router)#`
		BGP モードを開始し、ローカル BGP スピーカに AS 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タ 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	**client-to-client reflection** **例：** `switch(config-router-af)# client-to-client reflection`	（任意）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です。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 6	**exit**
	
	例：
	`switch(config-router-neighbor)# 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 40000`
	`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 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 ルータ リフレクタとしてスイッチを設定し、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	***show bgp*** {***ip*** | ***ipv6***} {***unicast*** | ***multicast***} ***neighbors*** **例：** `switch(config-router-neighbor-af)# show bgp ip unicast neighbors`	（任意）BGP ピアを表示します。
ステップ 11	***copy running-config startup-config*** **例：** `switch(config-router-neighbor-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

ルータ 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 45000
switch(config-router)# neighbor 192.0.2.10 remote-as 40000
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
```

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。

コマンド	目的
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 ― 範囲は 1 ～ 45 • reuse-limit ― 範囲は 1 ～ 20000 • suppress-limit ― 範囲は 1 ～ 20000 • max-suppress-time ― 範囲は 1 ～ 255

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。

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。

コマンド	目的
maximum-paths [***ibgp***] *maxpaths* **例：** `switch(config-router-af)# maximum-paths 12`	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。有効値の範囲は 1 ～ 16 です。

最大プレフィクス数の設定

BGP が BGP ピアから受け取ることのできるプレフィクスの最大数を設定できます。任意で、プレフィ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。

コマンド	目的
maximum-prefix *maximum* [*threshold*] [***restart*** *time* | **warming-only**] **例：** `switch(config-router-neighbor-af)# maximum-paths 12`	ピアからのプレフィ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* ― 範囲は 1 ～ 300000 • *threshold* ― 範囲は 1 ～ 100%。デフォルト値は 75% です。 • *time* ― 範囲は 1 ～ 65635 分 このコマンドによって、プレフィ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。

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。

コマンド	目的
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。 デフォルトでディセーブルです。

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。

コマンド	目的
aggregate-address *ip-prefix/length* [***as-set***] [***summary-only***] [***advertise-map*** *map-name*] [***attribute-map*** *map-name*] [*suppress-**map** map-name*] **例：** `switch(config-router-af)# aggregate-address 192.0.2.0/8 as-set`	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パスに含まれるすべての要素からなる、AS セットです。 • ***as-set*** キーワードで、AS セット パス情報および関係するパスに基づくコミュニティ情報が生成されます。 • ***summary-only*** キーワードによって、アップデートから固有性の強いルートがすべてフィルタリングされます。 • ***advertise-map*** キーワードおよび引数では、選択されたルートからアトリビュート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、集約からアトリビュート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。

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。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

手順概要

1. **config t**
2. **router bgp** *as-number*
3. ***address-family*** {***ipv4*** | ***ipv6***} {***unicast*** | ***multicast***}
4. ***redistribute*** {***direct*** | ***eigrp*** *as* | ***isis*** *id* | ***ospf*** *id* | ***ospfv3*** *id* | ***rip*** *id* | ***static route-map*** *map-name*
5. ***default-metric*** *value*
6. ***exit***
7. ***copy running-config startup-config***

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 45000 switch(config-router)#	BGP モードを開始し、ローカル BGP スピーカに AS 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 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 モードを開始します。
ステップ 4	***redistribute*** {***direct*** | ***eigrp*** *as* | ***isis*** *id* | ***ospf*** *id* | ***ospfv3*** *id* | ***rip*** *id* | ***static*** | ***direct***} ***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。ルート マップの詳細については、「ルート マップの設定」（p.14-9）を参照してください。
ステップ 5	***default-metric*** *value* **例：** switch(config-router-af)# default-metric 33	（任意）BGP へのデフォルト ルートを作成します。
ステップ 6	***copy running-config startup-config*** **例：** switch(config-router-af)# copy running-config startup-config	（任意）この設定変更を保存します。

EIGRP を BGP に再配布する例を示します。

```
switch# config t
switch(config)# router bgpEnterprise
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
```

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*** | ***compare-routerid*** | ***med*** {***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 AS からのパスの MED を比較します。 • ***compare-routerid*** ― 同一の eBGP パスのルータ ID を比較します。 • ***med missing-as-worst*** ― 脱落 MED を最上位 MED として扱います。 • ***med non-deterministic*** ― 同じ AS からのパス間で、必ずしも最適な MED パスを選択しません。
enforce-first-as **例：** `switch(config-router)# enforce-first-as`	ネイバー AS を eBGP の AS_path アトリビュートで指定する最初の AS 番号にします。
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。
router-id *id* **例：** `switch(config-router)# router-id 209.165.20.1`	この BGP スピーカのルータ ID を手動で設定します。
timers [***bestpath-delay*** *delay* | ***bgp*** *keepalive holdtime* | ***prefix-peer-timeout*** *timeout*] **例：** `switch(config-router)# timers bgp 90 270`	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。 • *delay* ― 再起動後の初期ベスト パス タイムアウト値。有効値の範囲は 0 ～ 3600 秒です。デフォルト値は 300 です。 • keepalive ― BGP セッション キープアライブ タイム。有効値の範囲は 0 ～ 3600 秒です。デフォルト値は 60 です。 • holdtime ― BGP セッション ホールド タイム。有効値の範囲は 0 ～ 3600 秒です。デフォルト値は 180 です。 • timeout ― プレフィクス ピア タイムアウト値。有効値の範囲は 0 ～ 1200 秒です。デフォルト値は 30 です。 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 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
distance *ebgp-distance ibgp distance local-distance* **例：** `switch(config-router-af)# distance 20 100 200`	BGP の管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは次のとおりです。 • eBGP ディスタンス ― 20 • iBGP ディスタンス ― 200 • ローカル ディスタンス ― 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管理ディスタンスです。

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。
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。
remove-private-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。
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。

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
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。
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。
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 アトリビュートを送信します。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。

グレースフル リスタートの設定

BGP のグレースフル リスタートを設定し、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。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] を参照）。

VDC および VRF を作成します。

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）。

手順概要

1. **config t**
2. ***router bgp*** *as-number*
3. ***graceful-restart***
4. ***graceful-restart*** [***restart-time*** *time* | ***stalepath-time*** *time*]
5. ***graceful-restart-helper***
6. ***show running-config bgp***
7. ***copy running-config startup-config***

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 201 switch(config-router)#	AS 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 3	***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。イネーブルがデフォルトです。 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 4	***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。有効値の範囲は 1 ～ 3600 秒です。デフォルトは 120 です。 • 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。有効値の範囲は 1 ～ 3600 秒です。デフォルトは 300 です。 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-helper*** **例：** switch(config-router)# graceful-restart-helper	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、グレースフル リスタートをディセーブルにしていながら、グレースフル リスタート ヘルパー機能はイネーブルにする必要がある場合に使用します。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 6	***show running-config bgp*** **例：** switch(config-router)# show running-config bgp	（任意）BGP の設定を表示します。
ステップ 7	***copy running-config startup-config*** **例：** switch(config-router)# copy running-config startup-config	（任意）この設定変更を保存します。

グレースフル リスタートをイネーブルにする例を示します。

```
switch# config t
switch(config)# router bgp 201
switch(config-router)# graceful-restart
switch(config-router)# copy running-config startup-config
```

仮想化の設定

各 VDC で 1 つずつ BGP プロセスを設定できます。各 VDC 内で複数の 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。

準備作業

BGP 機能がイネーブルになっていることを確認します（「BGP 機能のイネーブル化」[p.9-10] 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）。

手順概要

1. **config t**
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. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 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	***exit***
	
	例：
	`switch(config-vrf)# exit`
	`switch(config)#`
		VRF コンフィギュレーション モードを終了します。

ステップ 4	**router bgp** *as-number*
	
	例：
	`switch(config)# router bgp 201`
	`switch(config-router)#`
		AS 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	目的

ステップ 5	**vrf** *vrf-name*
	
	例：
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		ルータ VRF コンフィギュレーション 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。

ステップ 6	**neighbor** *ip-address* **remote-as** *as-number*
	
	例：
	`switch(config-router-vrf)# neighbor 209.165.201.1 remote-as 45000`
	`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	***copy running-config startup-config*** **例：** `switch(config-router-vrf-neighbor)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRF を作成し、VRF でルータ ID を設定する例を示します。

```
switch# config t
switch(config)# vrf context NewVRF
switch(config-vrf)# exit
switch(config)# router bgp 201
switch(config-router)# vrf NewVRF
switch(config-router-vrf)# neighbor 209.165.201.1 remote-as 45000
switch(config-router-vrf-neighbor)# copy running-config startup-config
```

拡張 BGP の設定確認

BGP の設定を確認するには、次のコマンドを使用します。

コマンド	目的
show bgp [**vrf** *vrf-name*] **all** [**summary**]	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 [**vrf** *vrf-name*] **convergence**	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community** {**regexp** | [**community**] [**no-advertise**] [**no-export**] [**no-export-subconfed**]}	BGP コミュニティと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **community-list** *list-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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**damp-params** | **dampened-paths**}	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **history-paths**	BGP ルート ヒストリ パスを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **filter-list** *list-name*	BGP フィルタ リストの情報を表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **neighbors** [*ip-address* | *ipv6-prefix*]	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] {**nexthop** | **nexthop-database**}	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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **policy** *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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **prefix-list** *list-name*	プレフィクス リストと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **received-paths**	ソフト再構成用に保管されている BGP パスを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **regexp** *expression*	AS_path 正規表現と一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **route-map** *map-name*	ルート マップと一致する BGP ルートを表示します。
show bgp [**vrf** *vrf-name*] **peer-policy** *name*	BGP ピア ポリシー情報を表示します。
show bgp [**vrf** *vrf-name*] **peer-session** *name*	BGP ピア セッション情報を表示します。
show bgp [**vrf** *vrf-name*] **peer-template** *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 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 統計情報の表示

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 [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **flap-statistics**	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消去するには、**clear bgp flap-statistics** コマンドを使用します。
show bgp [**vrf** *vrf-name*] {**ip** | **ipv6**} {**unicast** | **multicast**} [*ip-address* | *ipv6-prefix*] **neighbors** [*ip-address* | *ipv6-prefix*]	BGP ピアの統計情報を表示します。これらの統計情報を消去するには、**clear bgp neighbors** コマンドを使用します。
show bgp [**vrf** *vrf-name*] **sessions**	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消去するには、**clear bgp sessions** コマンドを使用します。

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。

- 第 9 章「ベーシック BGP の設定」
- 第 14 章「Route Policy Manager の設定」

デフォルト設定

表 10-1 に、BGP パラメータのデフォルト設定を示します。

表 10-1　デフォルトの BGP パラメータ

パラメータ	デフォルト
BGP 機能	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

その他の関連資料

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。

- 関連資料（p.10-37）
- MIB（p.10-37）

関連資料

関連項目	マニュアル名
BGP CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

MIB

MIB	MIB リンク
BGP4-MIB CISCO-BGP4-MIB	MIB を検索してダウンロードするには、次の URL にアクセスしてください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CHAPTER 11

RIP の設定

この章では、RIP の設定方法について説明します。

ここでは、次の内容を説明します。

- RIP 情報（p.11-2）
- RIP のライセンス要件（p.11-5）
- RIP の前提条件（p.11-5）
- 設定時の注意事項および制約事項（p.11-5）
- RIP の設定（p.11-6）
- RIP の設定確認（p.11-17）
- RIP 統計情報の表示（p.11-17）
- RIP の設定例（p.11-18）
- 関連項目（p.11-18）
- 関連情報（p.11-18）
- デフォルト設定（p.11-18）
- その他の関連資料（p.11-19）

RIP 情報

ここでは、次の内容について説明します。

- RIP の概要（p.11-2）
- RIPv2 の認証（p.11-2）
- スプリット ホライズン（p.11-3）
- ルート フィルタリング（p.11-3）
- ルート集約（p.11-3）
- ルートの再配布（p.11-4）
- ロード バランシング（p.11-4）
- ハイ アベイラビリティ（p.11-4）
- 仮想化サポート（p.11-4）

RIP の概要

RIP は UDP（ユーザ データグラム プロトコル）データ パケットを使用して、小規模なインターネットワークでルーティング情報を交換します。RIPv2 は IPv4 をサポートし、RIPng は IPv6 をサポートします。RIP next generation（RIPng）が IPv6 の認証を使用するのに対して、RIPv2 は RIPv2 プロトコルがサポートするオプションの認証機能を使用します（「RIPv2 の認証」[p.11-2] を参照）。

（注） この章では特に指示がないかぎり、RIP は RIPv2 と RIPng の両方を表します。

RIP では次の 2 種類のメッセージを使用します。

- 要求 ― マルチキャスト アドレスに送信され、他の RIP 対応ルータにルート アップデートを要求します。
- 応答 ― デフォルトでは 30 秒間隔で送信されます（「RIP の設定確認」[p.11-17] を参照）。ルータも、要求メッセージの受信後に応答メッセージを送信します。応答メッセージには、RIP ルート テーブル全体が含まれます。RIP ルーティング テーブルが 1 つの応答パケットに収まらない場合、RIP は 1 つの要求に対して複数の応答パケットを送信します。

RIP はルーティング メトリックとして、*ホップ カウント*を使用します。ホップ カウントは、パケットが宛先に到達するまでに、通過できるルータの数です。直接接続されたネットワークのメトリックは 1 です。到達不能なネットワークのメトリックは 16 です。RIP はこのようにメトリックの範囲が小さいので、大規模なネットワークに適したルーティング プロトコルではありません。

RIPv2 の認証

ネットワークで不正または無効なルーティング アップデートが行われないように、RIP メッセージに認証を設定できます。Cisco NX-OS は、単純なパスワードまたは MD5 認証ダイジェストをサポートします。

認証鍵のキーチェーン管理を使用することによって、インターフェイスごとに RIP 認証を設定できます。キーチェーン管理によって、MD5 認証ダイジェストまたは単純テキスト パスワード認証で使用される認証鍵の変更を制御できます。キーチェーン作成の詳細については、『*Cisco NX-OS Security Configuration Guide*』を参照してください。

MD5 認証ダイジェストを使用するには、ローカル ルータとすべてのリモート RIP ネイバーで共有されるパスワードを設定します。Cisco NX-OS は、メッセージそのものと暗号化されたパスワードに基づいて、MD5 一方向メッセージ ダイジェストを作成し、RIP メッセージ（要求または応答）とともにこのダイジェストを送信します。受信側の RIP ネイバーは、同じ暗号パスワードを使用して、ダイジェストを検証します。メッセージが変更されていない場合は、計算が一致し、RIP メッセージは有効とみなされます。

MD5 認証ダイジェストの場合はさらに、ネットワークでメッセージが再送されないように、各 RIP メッセージにシーケンス番号が組み込まれます。

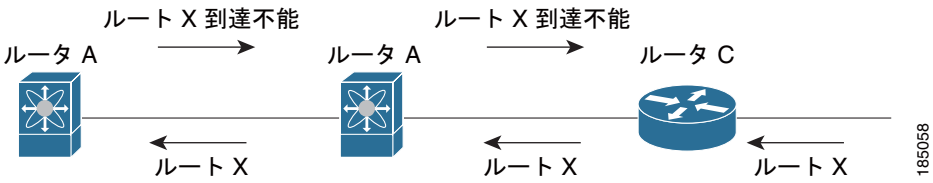
スプリット ホライズン

スプリット ホライズンを使用すると、ルートを学習したインターフェイスからは、RIP がルートをアドバタイズしないようにできます。

スプリット ホライズンは、RIP アップデートおよびクエリー パケットの送信を制御する方法です。インターフェイス上でスプリット ホライズンがイネーブルの場合、Cisco NX-OS はそのインターフェイスから学習した宛先にはアップデート パケットを送信しません。この方法でアップデート パケットを制御すると、ルーティング ループの発生する可能性が小さくなります。

ポイズン リバースを指定してスプリット ホライズンを使用すると、ルートを学習したインターフェイス経由では到達不能であると RIP が学習したルートをアドバタイズするように、インターフェイスを設定できます。図 11-1 に、ポイズン リバースをイネーブルにしてスプリット ホライズンを指定した、RIP ネットワークの例を示します。

図 11-1　スプリット ホライズン ポイズン リバースを指定した RIP



ルータ C はルート X について学習し、そのルートを ルータ B にアドバタイズします。ルータ B は次に、ルート X をルータ A にアドバタイズしますが、ルータ C には、ルート X 到達不能アップデートを戻します。

スプリット ホライズンはデフォルトで、すべてのインターフェイスでイネーブルです。

ルート フィルタリング

RIP 対応インターフェイス上でルート ポリシーを設定すると、RIP アップデートをフィルタリングできます。Cisco NX-OS は、ルート ポリシーで許可されたルートだけを使用して、ルート テーブルをアップデートします。

ルート集約

指定したインターフェイスに、複数のサマリー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 を 1 つのサマリー アドレス 10.1.0.0/16 に置き換えることができます。

RIP はルーティング テーブルに含まれている固有性の強いルートが多いほど、固有性の強いルートの最大メトリックと同じメトリックのインターフェイスからのサマリー アドレスをアドバタイズします。

（注）　Cisco NX-OS は、自動ルート集約をサポートしません。

ルートの再配布

RIP を使用すると、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できます。再配布を設定するには、ルート ポリシーを使用して、RIP に渡すルートを制御します。ルート ポリシーでは、宛先、送信元プロトコル、ルート タイプ、ルート タグなどのアトリビュートに基づいて、ルートをフィルタリングできます。詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

RIP ルーティング ドメインにルートを再配布しても、デフォルトでは Cisco NX-OS がそのつど、RIP ルーティング ドメインにデフォルト ルートを再配布することはありません。RIP へのデフォルト ルートを発生させ、ルート ポリシーでそのルートを制御できます。

RIP にインポートされたすべてのルートに使用する、デフォルトのメトリックも設定できます。

ロード バランシング

ロード バランシングを使用すると、ルータによって、宛先アドレスから同じ距離にあるすべてのルータ ネットワーク ポートにトラフィックが分散されます。ロード バランシングは、ネットワーク セグメントの使用率を向上させ、有効ネットワーク帯域幅を増加させます。

Cisco NX-OS は、ECMP（等コスト マルチパス）機能をサポートします。RIP ルート テーブルおよびユニキャスト RIB の等コスト パスは最大 16 です。これらのパスの一部または全部でトラフィックのロード バランシングが行われるように、RIP を設定できます。

ハイ アベイラビリティ

Cisco NX-OS は、RIP のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバー後に、Cisco NX-OS が実行コンフィギュレーションを適用し、RIP がただちに要求パケットを送信して、ルーティング テーブルに再入力します。

仮想化サポート

Cisco NX-OS は、同一システム上で動作する複数の RIP プロトコル インスタンスをサポートします。RIP は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。

VDC で設定できる RIP インスタンスは、最大 4 つです。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

RIP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	RIP にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

RIP の前提条件

RIP の前提条件は、次のとおりです。

- RIP 機能をイネーブルにする必要があります（「RIP 機能のイネーブル化」[p.11-6] を参照）。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

RIP に関する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 は、RIPv1 をサポートしません。RIPv1 パケットを受信した Cisco NX-OS は、メッセージを記録してパケットを廃棄します。
- Cisco NX-OS は、RIPv1 ルータとの隣接関係を確立しません。

RIP の設定

ここでは、次の内容について説明します。

- RIP 機能のイネーブル化（p.11-6）
- RIP インスタンスの作成（p.11-7）
- インターフェイス上での RIP の設定（p.11-9）
- 受動インターフェイスの設定（p.11-11）
- ルート集約の設定（p.11-11）
- ルートの再配布の設定（p.11-12）
- 仮想化の設定（p.11-13）
- RIP の調整（p.11-16）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

RIP 機能のイネーブル化

RIP を設定するには、RIP 機能をイネーブルにしておく必要があります。

準備作業

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）。

手順概要

1. **config t**
2. ***feature rip***
3. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature rip** **例：** `switch(config)# feature rip`	RIP 機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

RIP 機能をディセーブルにして、関連するすべての設定を削除する場合は、***no feature rip*** コマンドを使用します。

コマンド	目的
no feature rip **例：** `switch(config)# no feature rip`	RIP 機能をディセーブルにして、関連するすべての設定を削除します。

RIP インスタンスの作成

RIP インスタンスを作成し、そのインターフェイス用のアドレス ファミリを設定できます。

準備作業

RIP 機能がイネーブルになっていることを確認します（「RIP 機能のイネーブル化」[p.11-6] 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）。

手順概要

1. **config t**
2. **router rip** *instance-tag*
3. ***address-family*** {***ip*** | ***ipv6***} ***unicast***
4. ***show*** {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] [***vrf*** *vrf-name*]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 rip** *instance-tag*
	
	例：
	`switch(config)# router RIP Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい RIP インスタンスを作成します。

ステップ 3	**address-family** {***ipv4*** | ***ipv6***} **unicast**
	
	例：
	`switch(config-router)# address-family ipv4 unicast`
	`switch(config-router-af)#`
		この RIP インスタンスのアドレス ファミリを設定し、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 4	**show** {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] [**vrf** *vrf-name*] **例：** `switch(config-router-af)# show ip rip`	（任意）すべての RIP インスタンスについて、RIP 要約情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-router-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

RIP インスタンスおよび関連するすべての設定を削除する場合は、***no router rip*** コマンドを使用します。

コマンド	目的
no router rip *instance-tag* **例：** `switch(config)# no router rip Enterprise`	RIP インスタンスおよび関連するすべての設定を削除します。

(注)　インターフェイス モードで設定した RIP コマンドを削除することも必要です。

アドレス ファミリ コンフィギュレーション モードでは、RIP に次のオプション パラメータを設定できます。

コマンド	目的
distance *value* **例：** `switch(config-router-af)# distance 30`	RIP の管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 120 です。「管理ディスタンス」（p.1-7）を参照してください。
maximum-paths *number* **例：** `switch(config-router-af)# maximum-paths 6`	RIP がルート テーブルで維持する等コスト パスの最大数を設定します。有効値の範囲は 1 ～ 16 であり、デフォルトは 16 です。

IPv4 に対応する RIP インスタンスを作成し、ロード バランシングのための等コスト パス数を設定する例を示します。

```
switch# config t
switch(config)# router rip Enterprise
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# max-paths 10
switch(config-router-af)# copy running-config startup-config
```

RIP インスタンスの再起動

RIP インスタンスの再起動が可能です。再起動すると、インスタンスのすべてのネイバーが消去されます。

RIP インスタンスを再起動し、関連付けられたすべてのネイバーを削除するには、次のコマンドを使用します。

コマンド	目的
restart rip instance-tag **例：** `switch(config)# restart rip Enterprise`	RIP インスタンスを再起動し、すべてのネイバーを削除します。

インターフェイス上での RIP の設定

RIP インスタンスにインターフェイスを追加できます。

準備作業

RIP 機能がイネーブルになっていることを確認します（「RIP 機能のイネーブル化」[p.11-6] を参照）。

RIP を設定する前に、必要に応じて有効な VDC を開始します。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. {***ip*** | ***ipv6***} ***router rip*** *instance-tag*
4. ***show***{***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***interface*** [*interface-type slot/port*] [***vrf*** *vrf-name*] [***detail***]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	{***ip*** | ***ipv6***} **router rip** *instance-tag* **例：** `switch(config-if)# ip router rip Enterprise`	このインターフェイスを RIP インスタンスに関連付けます。
ステップ 4	***show*** {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***interface*** [*interface-type slot/port*] [***vrf*** *vrf-name*] [***detail***] **例：** `switch(config-if)# show ip rip Enterprise tethernet 1/2`	（任意）インターフェイスの RIP 情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

RIP インスタンスにインターフェイス ethernet 1/2 を追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ip router rip Enterprise
switch(config)# copy running-config startup-config
```

RIP 認証の設定

インターフェイス上で RIP パケットの認証を設定できます。

準備作業

RIP 機能がイネーブルになっていることを確認します（「RIP 機能のイネーブル化」[p.11-6] 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）。

認証をイネーブルにする前に、必要に応じてキーチェーンを設定します。キーチェーン実装の詳細については、『*Cisco NX-OS Security Configuration Guide*』を参照してください。

手順概要

1. **config t**
2. ***interface*** *interface-type slot/port*
3. ***ip rip authentication mode*****{*****text*** **|** ***md5*****}**
4. ***ip rip authentication keychain*** *key*
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	***ip rip authentication mode*** **{*****text*** **|** ***md5*****}** **例：** `switch(config-if)# ip rip authentication mode md5`	クリアテキストまたは MD5 認証ダイジェストとして、このインターフェイスにおける RIP 認証タイプを設定します。
ステップ 4	**ip rip authentication keychain** *key* **例：** `switch(config-if)# ip rip authentication keychain RIPKey`	このインターフェイス上で RIP に使用する認証鍵を設定します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

キーチェーンを作成し、RIP インターフェイス上で MD5 認証を設定する例を示します。

```
switch# config t
switch(config)# key chain RIPKey
switch(config)# key-string myrip
switch(config)# accept-lifetime 00:00:00 Jan 01 2000 infinite
switch(config)# send-lifetime 00:00:00 Jan 01 2000 infinite
switch(config)# interface ethernet 1/2
switch(config-if)# ip rip authentication mode md5
switch(config-if)# ip rip authentication keychain RIPKey
switch(config-if)# copy running-config startup-config
```

受動インターフェイスの設定

インターフェイスを受動モードに設定することによって、ルートを受信するが、ルート アップデートの送信は行わないように RIP インターフェイスを設定できます。

受動モードで RIP インターフェイスを設定するには、インターフェイス 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。

コマンド	目的
{**ip** | **ipv6**} **rip passive-interface** **例：** `switch(config-if)# ip rip passive-interface`	インターフェイスを受動モードに設定します。

ポイズン リバースを指定した スプリット ホライズンの設定

ポイズン リバースをイネーブルにすることによって、ルートを学習したインターフェイス経由では到達不能であると RIP が学習したルートをアドバタイズするように、インターフェイスを設定できます。

インターフェイス上で、ポイズン リバースを指定してスプリット ホライズンを設定するには、インターフェイス 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。

コマンド	目的
{**ip** | **ipv6**} **rip poison-reverse** **例：** `switch(config-if)# ip rip poison-reverse`	ポイズン リバースを指定してスプリット ホライズンをイネーブルにします。ポイズン リバースを指定したスプリット ホライズンは、デフォルトでディセーブルです。

ルート集約の設定

ルーティング テーブルでサマリー アドレスによって代表される集約アドレスを作成できます。Cisco NX-OS は、固有性の強いすべてのルートの中でメトリックが最小のサマリー アドレス メトリックをアドバタイズします。

インターフェイス上でサマリー アドレスを設定するには、インターフェイス 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。

コマンド	目的
{**ip** | **ipv6**} **rip summary-address** {*ip-prefix/mask-len* | *ipv6-prefix/mask-len*} **例：** `switch(config-if)# ip router rip summary-address 192.0.2.0/24`	IPv4 または IPv6 アドレスに対応する、RIP 用のサマリー アドレスを設定します。

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、RIP ネットワークを通じてその情報を再配布するように、RIP を設定できます。再配布されたルートを任意で、デフォルト ルートとして割り当てることができます。

準備作業

RIP 機能がイネーブルになっていることを確認します（「RIP 機能のイネーブル化」[p.11-6] を参照）。

RIP を設定する前に、必要に応じて有効な VDC を開始します。

再配布を設定する前に、ルート マップを設定します。ルート マップ設定の詳細については、「ルート マップの設定」（p.14-9）を参照してください。

手順概要

1. **config t**
2. ***router rip*** *instance-tag*
3. ***address-family*** {***ipv4*** | ***ipv6***} ***unicast***
4. ***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | **isis** *tag* | ***rip*** *tag* | {***ospf*** *tag* | ***ospfv3*** *tag* } | ***static***} ***route-map*** *map-name*
5. ***default-information originate*** [***always***] [***route-map*** *map-name*]
6. ***default-metric*** *value*
7. ***show***{***ip*** | ***ipv6}*** ***rip route*** [{*ip-prefix* | *ip6-prefix*}[***longer-prefixes*** | ***shorter-prefixes]]*** [***vrf*** *vrf-name*] [***summary***]
8. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 rip** *instance-tag*
	
	例：
	`switch(config)# router rip Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい RIP インスタンスを作成します。

	コマンド	目的

ステップ 3	***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 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	***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* |***isis*** *tag* | ***rip*** *tag* | {***ospf*** *tag* | ***ospfv3*** *tag*} ***static***} ***route-map*** *map-name* **例：** `switch(config-router-af)# redistribute eigrp 201 route-map RIPmap`	他のプロトコルからのルートを RIP に再配布します。ルート マップの詳細については、「ルート マップの設定」（p.14-9）を参照してください。
ステップ 5	***default-information originate*** [***always***] [***route-map*** *map-name*] **例：** `switch(config-router-af)# default-information originate always`	（任意）RIP へのデフォルト ルートを作成し、任意でルート マップで制御します。
ステップ 6	***default-metric*** *value* **例：** `switch(config-router-af)# distribute level-1 into level-2 all`	（任意）再配布されたすべてのルートにデフォルト メトリックを設定します。有効値の範囲は 1 ～ 15 であり、デフォルトは 1 です。
ステップ 7	***show*** {***ip*** | ***ipv6***} ***rip route*** [{*ip-prefix* | *ip6-prefix*} [***longer-prefixes*** | ***shorter-prefixes***] [***vrf*** *vrf-name*][***summary***] **例：** `switch(config-router-af)# show ip rip route`	（任意）RIP のルートを表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-router-af)# copy running-config startup-config`	（任意）この設定変更を保存します。

EIGRP を RIP に再配布する例を示します。

```
switch# config t
switch(config)# router rip Enterprise
switch(config-router)# address-family ipv4 unicast
switch(config-router-af)# redistribute eigrp 201 route-map RIPmap
switch(config-router-af)# copy running-config startup-config
```

仮想化の設定

VDC ごとに複数の RIP インスタンスを設定できます。各 VDC 内で複数の VRF を作成することもできます。また、各 VRF で同じ RIP インスタンスを使用することも、複数の RIP インスタンスを使用することも可能です。VRF に RIP インターフェイスを割り当てます。

（注） インターフェイスに VRF を設定したあとで、その他のインターフェイス パラメータを設定します。インターフェイスに VRF を設定すると、そのインターフェイスに関するすべての設定が削除されます。

準備作業

RIP 機能がイネーブルになっていることを確認します（「RIP 機能のイネーブル化」[p.11-6] を参照）。

VDC を作成します。

手順概要

1. **config t**
2. ***vrf* context** *vrf_name*
3. **exit**
4. **router rip** *instance-tag*
5. ***vrf*** *vrf-name*
6. **address-family {ipv4 | ipv6} unicast**
7. オプション パラメータの設定
8. ***interface ethernet*** *slot/port*
9. ***vrf member*** *vrf-name*
10. ***ip-address*** *ip-prefix/length*
11. {**ip** | **ipv6**} ***router rip*** *instance-tag*
12. ***show*** {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***interface*** [*interface-type slot/port*] [***vrf*** *vrf-name*]
13. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *vrf-name*
	
	例：
	`switch(config)# vrf RemoteOfficeVRF`
	`switch(config-vrf)#`
		新しい VRF を作成します。

ステップ 3	**exit**
	
	例：
	`switch(config-vrf)# exit`
	`switch(config)#`
		VRF コンフィギュレーション モードを終了します。

ステップ 4	**router rip** *instance-tag*
	
	例：
	`switch(config)# router rip Enterprise`
	`switch(config-router)#`
		instance tag を設定して、新しい RIP インスタンスを作成します。

ステップ 5	**vrf context** *vrf-name*
	
	例：
	`switch(config)# vrf context RemoteOfficeVRF`
	`switch(config-vrf)#`
		新しい VRF を作成し、VRF コンフィギュレーション モードを開始します。

	コマンド	目的

ステップ 6	***address-family*** {***ipv4*** | ***ipv6***} ***unicast***
	
	例：
	`switch(config-router-vrf)# address-family ipv4 unicast`
	`switch(config-router-vrf-af)#`
		（任意）この RIP インスタンスの VRF アドレス ファミリを設定します。

ステップ 7	***redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | **isis** *tag* | ***rip*** *tag* | {***ospf*** *tag* | ***ospfv3*** *tag*} ***static***} ***route-map*** *map-name* **例：** `switch(config-router-vrf-af)# redistribute eigrp 201 route-map RIPmap`	（任意）他のプロトコルからのルートを RIP に再配布します。ルート マップの詳細については、「ルート　マップの設定」（p.14-9）を参照してください。

ステップ 8	**interface ethernet** *slot/port*
	
	例：
	`switch(config-router-vrf-af)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 9	**vrf member** *vrf-name* **例：** `switch(config-if)# vrf member RemoteOfficeVRF`	VRF にこのインターフェイスを追加します。
ステップ 10	**ip address** *ip-prefix/length* **例：** `switch(config-if)# ip address 209.0.2.1/16`	このインターフェイスの IP アドレスを設定します。この作業は、VRF にこのインターフェイスを割り当てたあとで行う必要があります。
ステップ 11	{***ip*** | ***ipv6***} **router rip** *instance-tag* **例：** `switch(config-if)# ip router rip Enterprise`	このインターフェイスを RIP インスタンスに関連付けます。
ステップ 12	***show*** {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***interface*** [*interface-type slot/port*] [***vrf*** *vrf-name*] **例：** `switch(config-if)# show ip rip Enterprise ethernet 1/2`	（任意）VRF に含まれるインターフェイスの RIP 情報を表示します。
ステップ 13	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRF を作成し、VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context RemoteOfficeVRF
switch(config-vrf)# exit
switch(config)# router rip Enterprise
switch(config-router)# vrf RemoteOfficeVRF
switch(config-router-vrf)# address-family ipv4 unicast
switch(config-router-vrf-af)# redistribute eigrp 201 route-map RIPmap
switch(config-router-vrf-af)# interface ethernet 1/2
switch(config-if)# vrf member RemoteOfficeVRF
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# ip router rip Enterprise
switch(config-if)# copy running-config startup-config
```

RIP の調整

ネットワーク要件に合わせて RIP を調整できます。RIP では複数のタイマーを使用して、ルーティング アップデート間隔、ルートが無効になるまでの時間の長さ、およびその他のパラメータを決定します。これらのタイマーを調整すると、インターネットワークのニーズに適合するように、ルーティング プロトコルのパフォーマンスを調整できます。

（注） ネットワーク上のすべての RIP 対応ルータで、RIP タイマーに同じ値を設定する必要があります。

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

コマンド	目的
timers basic *update timeout holddown garbage-collection* **例：** `switch(config-router-af)# timers basic 40 120 120 100`	RIP タイマーを秒数で設定します。パラメータは次のとおりです。 • update ― 範囲は 5 ～任意の正の整数。デフォルト値は 30 です。 • timeout ― ルートの無効を宣言するまでに、Cisco NX-OS が待機する時間。タイムアウト インターバルが終了するまでに、このルートのアップデート情報を Cisco NX-OS が受信しなかった場合、Cisco NX-OS はルートの無効を宣言します。有効値の範囲は 1 ～任意の正の整数です。デフォルト値は 180 です。 • holddown ― 無効ルートに関するよりよいルート情報を Cisco NX-OS が無視する時間。有効値の範囲は 0 ～任意の正の整数です。デフォルト値は 180 です。 • garbage-collection ― Cisco NX-OS がルートを無効として表示してから、Cisco NX-OS がそのルートをルーティング テーブルから削除するまでの時間。有効値の範囲は 1 ～任意の正の整数です。デフォルト値は 120 です。

RIP を調整するには、インターフェイス 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
{***ip*** | ***ipv6***} ***rip metric-offset*** *value* **例：** `switch(config-if)# ip rip metric-offset 10`	このインターフェイスで受信する各ルータのメトリックに値を追加します。有効値の範囲は 1 ～ 15 であり、デフォルトは 1 です。
{***ip*** | ***ipv6***} ***rip route-filter {prefix-list*** *list-name* ***| route-map*** *map-name*| [***in*** | ***out***] **例：** `switch(config-if)# ip rip route-filter route-map InputMap in`	着信または発信 RIP アップデートをフィルタリングするための、ルート マップを指定します。

RIP の設定確認

RIP の設定を確認するには、次のコマンドを使用します。

コマンド	目的
show {***ip*** | ***ipv6***} ***rip*** **instance** [*instance-tag*] [***vrf*** *vrf-name*]	RIP インスタンスの状態を表示します。
show {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***interface*** *slot/port* ***detail*** [***vrf*** *vrf-name*]	インターフェイスの RIP ステータスを表示します。
show {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***neighbor*** [*interface-type number*] [***vrf*** *vrf-name*]	RIP ネイバー テーブルを表示します。
show {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***route*** [{*ip-prefix/lengh* | *ipv6-prefix/length*} [***longer-prefixes*** | ***shorter--prefixes***]][***summary***] [***vrf*** *vrf-name*]	RIP ルート テーブルを表示します。
show running-configuration rip	現在実行中の RIP コンフィギュレーションを表示します。

RIP 統計情報の表示

RIP 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 {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***policy statistics redistribute*** {***bgp*** *as* | ***direct*** | ***eigrp*** *as* | **isis** *tag* | ***rip*** *tag* | {***ospf*** *tag* | ***ospfv3*** *tag*} ***static***} [***vrf*** *vrf-name*]	RIP ポリシー ステータスを表示します。
show {***ip*** | ***ipv6***} ***rip*** [**instance** *instance-tag*] ***statistics*** *interface-type number*] [***vrf*** *vrf-name*]	RIP の統計情報を表示します。

ポリシーの統計情報を消去するには、***clear*** {***ip*** | ***ipv6***} ***rip policy*** コマンドを使用します。

RIP の統計情報を消去するには、***clear*** {***ip*** | ***ipv6***} ***rip statistics*** コマンドを使用します。

RIP の設定例

VRF で Enterprise RIP インスタンスを作成し、その RIP インスタンスに ethernet interface 1/2 を追加する例を示します。さらに、enthernet interface 1/2 の認証を設定し、この RIP ドメインに EIGRP を再配布します。

```
vrf context NewVRF
!
 feature rip
 router rip Enterprise
  vrf NewVRF
   address-family ip unicast
    redistribute eigrp 201 route-map RIPmap
    max-paths 10
!
interface ethernet 1/2
 vrf NewVRF
 ip address 209.0.2.1/16
 ip router rip Enterprise
 ip rip authentication mode md5
 ip rip authentication keychain RIPKey
```

関連項目

ルート マップの詳細については、第 14 章「Route Policy Manager の設定」を参照してください。

関連情報

RIP のパラメータのデフォルト設定については、次の項を参照してください。

デフォルト設定

表 11-1 に、RIP パラメータのデフォルト設定を示します。

表 11-1　デフォルトの RIP パラメータ

パラメータ	デフォルト
ロード バランシングを行う最大パス数	16
RIP 機能	ディセーブル
スプリット ホライズン	イネーブル

その他の関連資料

RIP 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。

- 関連資料（p.11-19）
- 規格（p.11-19）

関連資料

関連項目	マニュアル名
RIP CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 12

スタティック ルーティングの設定

この章では、ルータ上でスタティック ルーティングを設定する方法について説明します。

ここでは、次の内容を説明します。

- スタティック ルーティングの概要（p.12-2）
- スタティック ルーティングのライセンス要件（p.12-4）
- スタティック ルーティングの前提条件（p.12-4）
- 注意事項および制約事項（p.12-4）
- スタティック ルーティングの設定（p.12-5）
- スタティック ルーティングの設定確認（p.12-8）
- スタティック ルーティングの設定例（p.12-8）
- デフォルト設定（p.12-8）
- その他の関連資料（p.12-8）

スタティック ルーティングの概要

ルータは、ユーザが手動で設定したルート テーブル エントリのルート情報を使用するか、またはダイナミック ルーティング アルゴリズムで計算されたルート情報を使用して、パケットを転送します。

スタティック ルートは、2 つのルータ間の明示パスを定義するものであり、自動的にはアップデートされません。ネットワークに変更があった場合は、ユーザが手動でスタティック ルートを再設定する必要があります。スタティック ルートは、ダイナミック ルートに比べて使用する帯域幅が少なくなります。ルーティング アップデートの計算や分析に CPU サイクルを使用しません。

必要に応じて、スタティック ルートでダイナミック ルートを補うことができます。スタティック ルートをダイナミック ルーティング アルゴリズムに再配布できますが、ダイナミック ルーティング アルゴリズムで計算されたルーティング情報をスタティック ルーティング テーブルに再配布することはできません。

スタティック ルートは、ネットワーク トラフィックが予測可能で、ネットワーク設計が単純な環境で使用します。スタティック ルートはネットワークの変化に対応できないので、大規模でたえず変化しているネットワークでは、スタティック ルートを使用すべきではありません。大部分のネットワークは、ルータ間の通信にダイナミック ルートを使用しますが、特殊な状況でスタティック ルートを 1 つか 2 つ設定する場合があります。スタティック ルートは、最終手段としてのゲートウェイ（ルーティング不能なすべてのパケットの送信先となるデフォルト ルータ）を指定する場合にも便利です。

ここでは、次の内容について説明します。

- 管理ディスタンス（p.12-2）
- 直接接続のスタティック ルート（p.12-3）
- 完全指定のスタティック ルート（p.12-3）
- フローティング スタティック ルート（p.12-3）
- スタティック ルートのリモート ネクスト ホップ（p.12-3）

管理ディスタンス

管理ディスタンスは、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、最適なパスを選択するために、ルータが使用するメトリックです。複数のプロトコルがユニキャスト ルーティング テーブルに同じルートを追加した場合に、管理ディスタンスを手がかりに、他のルーティング プロトコル（またはスタティック ルート）ではなく、特定のルーティング プロトコル（またはスタティック ルート）が選択されます。各ルーティング プロトコルは、管理ディスタンス値を使用して、信頼性の高い順にプライオリティが与えられます。

スタティック ルートのデフォルトの管理ディスタンスは 1 です。ルータは値の小さいルートが最短であるとみなすので、スタティック ルートがダイナミック ルートより優先されます。ダイナミック ルートでスタティック ルートを上書きする場合は、スタティック ルートに管理ディスタンスを指定します。たとえば、管理ディスタンスが 120 のダイナミック ルートが 2 つある場合に、ダイナミック ルートでスタティック ルートを上書きするには、スタティック ルートに 120 より大きい管理ディスタンスを指定します。

直接接続のスタティック ルート

直接接続のスタティック ルートで指定しなければならないのは、出力インターフェイス（あらゆるパケットを宛先ネットワークに送り出すインターフェイス）だけです。ルータは宛先が出力インターフェイスに直接接続されているものとみなし、パケットの宛先をネクストホップ アドレスとして使用します。ネクストホップは、ポイントツーポイント インターフェイスの場合に限り、インターフェイスにできます。ブロードキャスト インターフェイスの場合は、ネクストホップを IPv4/IPv6 アドレスにする必要があります。

完全指定のスタティック ルート

完全指定のスタティック ルートでは、出力インターフェイス（あらゆるパケットを宛先ネットワークに送り出すインターフェイス）またはネクスト ホップ アドレスのどちらかを指定する必要があります。完全指定のスタティック ルートを使用できるのは、出力インターフェイスがマルチアクセス インターフェイスで、ネクストホップ アドレスを特定する必要がある場合です。ネクストホップ アドレスは、指定された出力インターフェイスに直接接続する必要があります。

フローティング スタティック ルート

フローティング スタティック ルートは、ダイナミック ルートをバックアップするためにルータが使用するスタティック ルートです。フローティング スタティック ルートには、バックアップするダイナミック ルートより大きい管理ディスタンスを設定する必要があります。この場合、ルータはフローティング スタティック ルートよりダイナミック ルートを優先させます。フローティング スタティック ルートは、ダイナミック ルートが失われた場合の代用として使用できます。

（注）　デフォルトでは、ルータはダイナミック　ルートよりスタティック　ルートを優先させます。スタティック ルートの方がダイナミック ルートより、管理ディスタンスが小さいからです。

スタティック ルートのリモート ネクスト ホップ

リモート（非直接接続）ネクストホップを指定したスタティック ルートの場合、ルータに直接接続されていない隣接ルータのネクストホップ アドレスを指定できます。データ転送時に、スタティック ルートにリモート ネクストホップがあると、そのネクスト ホップがユニキャスト ルーティング テーブルで繰り返し使用され、リモート ネクストホップに到達可能な、対応する直接接続のネクストホップ（複数可）が特定されます。

仮想化サポート

スタティック ルートは VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

スタティック ルーティングのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	スタティック ルーティングにライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

スタティック ルーティングの前提条件

スタティック ルーティングの前提条件は、次のとおりです。

- スタティック ルートのネクストホップ アドレスが到達不能な場合、そのスタティック ルートはユニキャスト ルーティング テーブルに追加されません。

注意事項および制約事項

スタティック ルーティングに関する注意事項および制約事項は、次のとおりです。

- スタティック　ルートのネクストホップ　アドレスとしてインターフェイスを指定できるのは、GRE トンネルなどのポイントツーポイント インターフェイスの場合に限られます。

スタティック ルーティングの設定

ここでは、次の内容について説明します。

- スタティック ルートの設定（p.12-5）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

スタティック ルートの設定

ルータ上でスタティック ルートを設定できます。

準備作業

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）。

手順概要

1. ***config t***
2. **ip route** {*ip-prefix* | *ip-addr ip-mask*} {*next-hop* | *nh-prefix* | *interface*} [**tag** *tag-value* [*pref*]
   または
3. **ipv6 route** *ip6-prefix*{ {*next-hop* | *nh-prefix*} | *interface* | {*link-local interface*}} [*pref*]
4. **show** {**ip** | **ipv6**} **static-route**
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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	***ip route*** {*ip-prefix* | *ip-addr ip-mask*} {*next-hop* | *nh-prefix* | *interface*} [**tag** *tag-value* [*pref*] **例：** switch(config)# ip route 192.0.2.0/8 ethernet 1/2	スタティック ルートおよびこのスタティック ルート用のインターフェイスを設定します。任意でネクストホップ アドレスを設定できます。*preference* 値で管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 1 です。
	ipv6 route {*ip6-prefix*{ {*next-hop* | *nh-prefix*} | *interface* |{*link-local interface*}} [*pref*] **例：** switch(config)# ipv6 route 2001:0DB8::/48 ethernet 1/2	スタティック ルートおよびこのスタティック ルート用のインターフェイスを設定します。任意でネクストホップ アドレスを設定できます。*preference* 値で管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 1 です。

	コマンド	目的
ステップ 3	***show*** {**ip** | **ipv6**} **static-route** **例：** `switch(config)# show ip static-route`	（任意）スタティック ルート情報を表示します。
ステップ 4	**例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

スタティック ルートの設定例を示します。

```
switch# config t
switch(config)# ip route 192.0.2.0/8 192.0.2.10
switch(config)# copy running-config startup-config
```

スタティック ルートを削除するには、***no*** {**ip** | **ipv6**} ***static-route*** コマンドを使用します。

仮想化の設定

VRF でスタティック ルートを設定できます。

準備作業

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）。

手順概要

1. ***config t***
2. **vrf context** *vrf-name*
3. **ip route** {*ip-prefix* | *ip-addr ip-mask*} {*next-hop* | *nh-prefix* | *interface*} [**tag** *tag-value* [*pref*]

   または
4. **ipv6 route** *ip6-prefix*{ {*next-hop* | *nh-prefix*} | *interface* | {*link-local interface*}} [*pref*]
5. **show** {**ip** | **ipv6**} **static-route vrf** *vrf-name*
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 StaticVrf`	VRF を作成し、VRF コンフィギュレーション モードを開始します。

	コマンド	目的
ステップ 3	***ip route*** {*ip-prefix* | *ip-addr ip-mask*} {*next-hop* | *nh-prefix* | *interface*} [**tag** *tag-value* [*pref*] **例：** `switch(config-vrf)# ip route 192.0.2.0/8 ethernet 1/2`	スタティック ルートおよびこのスタティック ルート用のインターフェイスを設定します。任意でネクストホップ アドレスを設定できます。*preference* 値で管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 1 です。
	ipv6 route {*ip6-prefix*{ {*next-hop* | *nh-prefix*} | *interface* |{*link-local interface*}} [*pref*] **例：** `switch(config-vrf)# ipv6 route 2001:0DB8::/48 ethernet 1/2`	スタティック ルートおよびこのスタティック ルート用のインターフェイスを設定します。任意でネクストホップ アドレスを設定できます。*preference* 値で管理ディスタンスを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 1 です。
ステップ 4	***show*** {**ip** | **ipv6**} ***static-route vrf vrf-name*** **例：** `switch(config-vrf)# show ip static-route`	（任意）スタティック ルート情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-vrf)# copy running-config startup-config`	（任意）この設定変更を保存します。

スタティック ルートの設定例を示します。

```
switch# config t
switch(config)# vrf context StaticVrf
switch(config-vrf)# ip route 192.0.2.0/8 192.0.2.10
switch(config-vrf)# copy running-config startup-config
```

スタティック ルーティングの設定確認

設定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 ip static-route	設定されているスタティック ルートを表示します。
show ipv6 static-route vrf *vrf-name*	VRF 単位のスタティック ルート情報を表示します。
show ipv6 static-route	設定されているスタティック ルートを表示します。

スタティック ルーティングの設定例

スタティック ルーティングの設定例を示します。

```
config t
 ip route 192.0.2.0/8 192.0.2.10
 copy running-config startup-config
```

デフォルト設定

表 12-1 に、スタティック ルーティング パラメータのデフォルト設定を示します。

表 12-1　デフォルトのスタティック ルーティング パラメータ

パラメータ	デフォルト
管理ディスタンス	1
RIP 機能	ディセーブル

その他の関連資料

スタティック ルーティング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。

- 関連資料（p.12-8）

関連資料

関連項目	マニュアル名
スタティック ルーティング CLI	『*Cisco NX-OS Unicast Routing Command Line Reference*』
VDC	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

CHAPTER 13

レイヤ 3 仮想化の設定

この章では、レイヤ 3 仮想化の設定手順について説明します。

ここでは、次の内容を説明します。

- レイヤ 3 仮想化（p.13-2）
- VRF のライセンス要件（p.13-6）
- VRF の前提条件（p.13-6）
- 注意事項および制約事項（p.13-6）
- VRF の設定（p.13-7）
- VRF の設定確認（p.13-13）
- VRF の設定例（p.13-13）
- 関連資料（p.13-13）
- デフォルト設定（p.13-13）
- デフォルト設定（p.13-13）
- その他の関連資料（p.13-14）

レイヤ 3 仮想化

ここでは、次の内容について説明します。

- レイヤ 3 仮想化の概要（p.13-2）
- VRF およびルーティング（p.13-3）
- VRF 認識サービス（p.13-3）

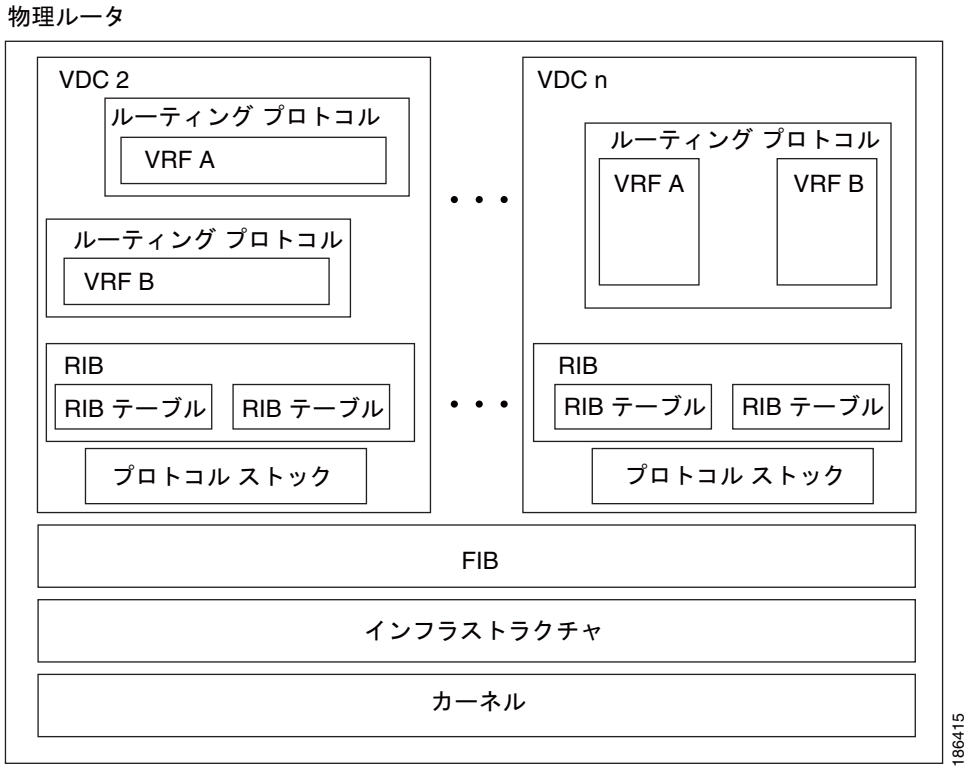
レイヤ 3 仮想化の概要

Cisco NX-OS は、物理システム リソースを複数の virtual device context（仮想デバイス コンテキスト ; VDC）に分割できる、仮想化の階層構造をサポートします。各 VDC は、レイヤ 2 サービスとレイヤ 3 サービスの両方が使用できる、独立型デバイスとして動作します。デフォルト VDC を含め、最大 15 の VDC を設定できます。VDC の詳細については、『*Cisco Cisco NX-OS Virtualization Configuration Guide*』を参照してください。

Cisco NX-OS は、各 VDC の仮想化をさらに進めて、VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。1 つの VDC に複数の VRF を設定できます。各 VRF には、IPv4 および IPv6 に対応するユニキャストおよびマルチキャスト ルート テーブルを備えた、独立したアドレス スペースが 1 つずつあり、他の VRF と無関係にルーティングを決定できます。

図 13-1 に、2 つの異なる VDC にある複数の独立した VRF を示します。

図 13-1　VDC 内の複数の VRF



VRF 名は VDC ローカルなので、VRF が異なる VDC に存在する場合は、同じ名前で 2 つの VRF を設定できます。図 13-1 では、VDC 2 の VRF A は、VDC n の VRF B および VRF A と無関係です。

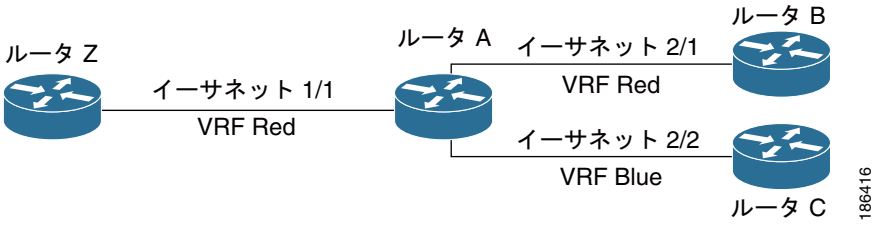
ルータごとに、デフォルト VRF および管理 VRF があります。すべてのレイヤ 3 インターフェイスおよびルーティング プロトコルは、ユーザが別の VRF に割り当てないかぎり、デフォルト VRF に存在します。mgmt0 インターフェイスは管理 VRF にあり、あらゆる VDC からアクセスできます。各 VDC は、mgmt0 インターフェイスに固有の IP アドレスを使用します（『*Cisco NX-OS Fundamentals Configuration Guide*』Release 4.0 を参照）。

VRF およびルーティング

すべてのユニキャストおよびマルチキャスト ルーティング プロトコルは VRF をサポートします。VRF でルーティング プロトコルを設定する場合は、同じルーティング プロトコル インスタンスの別の VRF のルーティング パラメータに依存しないルーティング パラメータをその VRF に設定します。

VRF にインターフェイスおよびルーティング プロトコルを割り当てることによって、仮想レイヤ 3 ネットワークを作成できます。インターフェイスが存在する VRF は 1 つだけです。図 13-2 に、1 つの物理ネットワークが 2 つの VRF からなる 2 つの仮想ネットワークに分割されている例を示します。ルータ Z、A、および B は、VRF Red にあり、1 つのアドレス ドメインを形成しています。これらのルータは、ルータ C が含まれないルート アップデートを共有します。ルータ C は別の VRF で設定されているからです。

図 13-2　ネットワーク内の VRF



Cisco NX-OS はデフォルトで、着信インターフェイスの VRF を使用して、ルート検索に使用するルーティング テーブルを選択します。ルート ポリシーを設定すると、この動作を変更し、Cisco NX-OS が着信パケットに使用する VRF を設定できます。詳細については、第 15 章「ポリシーベース ルーティングの設定」を参照してください。

VRF 認識サービス

VRF 認識サービスは、特定の VRF を選択することによって、リモート サーバに接続したり、選択した VRF に基づいて情報をフィルタリングしたりできます。

VRF 認識サービスに含まれるものは、次のとおりです。

- AAA（認証、認可、アカウンティング）— 詳細については、『*Cisco Cisco NX-OS Security Configuration Guide*』を参照してください。
- コール ホーム — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- DNS（ドメイン ネーム システム）— 詳細については、第 4 章「DNS の設定」を参照してください。
- GLBP — 詳細については、第 16 章「GLBP の設定」を参照してください。
- HSRP — 詳細については、第 17 章「HSRP の設定」を参照してください。
- HTTP（ハイパーテキスト転送プロトコル）— 詳細については、『*Cisco Cisco NX-OS Getting Started Configuration Guide*』を参照してください。

- ライセンス — 詳細については、『*Cisco Cisco NX-OS Licensing Configuration Guide*』を参照してください。
- Netflow — 詳細については、『*Cisco Cisco NX-OS Network Management Configuration Guide*』を参照してください。
- NTP（ネットワーク タイム プロトコル） — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- RADIUS — 詳細については、『*Cisco Cisco NX-OS Security Configuration Guide*』を参照してください。
- ping および traceroute — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- SSH（セキュア シェル） — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- SNMP（簡易ネットワーク管理プロトコル） — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- syslog — 詳細については、『*Cisco Cisco NX-OS System Management Configuration Guide*』を参照してください。
- TACACS+ — 詳細については、『*Cisco Cisco NX-OS Security Configuration Guide*』を参照してください。
- TFTP — 詳細については、『*Cisco Cisco NX-OS Getting Started Configuration Guide*』を参照してください。
- VRRP（仮想ルータ冗長プロトコル） — 詳細については、第 18 章「VRRP の設定」を参照してください。
- XML — 詳細については、『*Cisco Cisco NX-OS XML Configuration Guide*』を参照してください。

各サービスで VRF サポートを設定する詳細については、各サービスの適切なコンフィギュレーション ガイドを参照してください。

ここでは、次の内容について説明します。

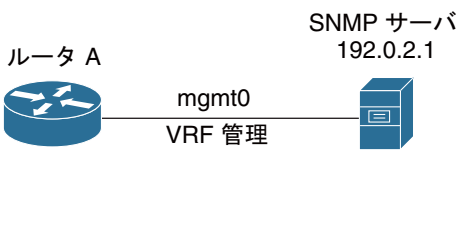
- 到達可能性（p.13-4）
- フィルタリング（p.13-5）
- 到達可能性とフィルタリングの組み合わせ（p.13-5）

到達可能性

到達可能性は、どの VRF にサービスを提供するサーバに到達するために必要なルーティング情報があるかを示します。たとえば、管理 VRF で到達可能な SNMP サーバを設定できます。ルータ上でサーバ アドレスを設定する場合は、サーバに到達するために Cisco NX-OS が使用しなければならない VRF も設定します。

図 13-3 に、管理 VRF を介して到達できる SNMP サーバを示します。SNMP サーバ ホスト 192.0.2.1 には管理 VRF を使用するように、ルータ A を設定します。

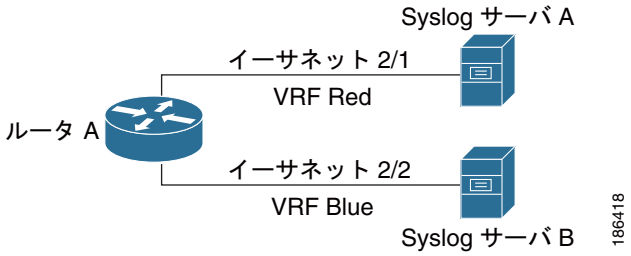
図 13-3　サービス VRF の到達可能性



フィルタリング

フィルタリングによって、VRF に基づいて VRF 認識サービスに渡す情報のタイプを制限できます。たとえば、特定の VRF をサポートするように、Syslog サーバを設定できます。図 13-4 に、それぞれが VRF を 1 つずつサポートする、2 つの Syslog サーバを示します。Syslog サーバ A は VRF Red で設定されているので、Cisco NX-OS は VRF Red で生成されたシステム メッセージだけを Syslog サーバ A に送信します。

図 13-4　サービス VRF のフィルタリング

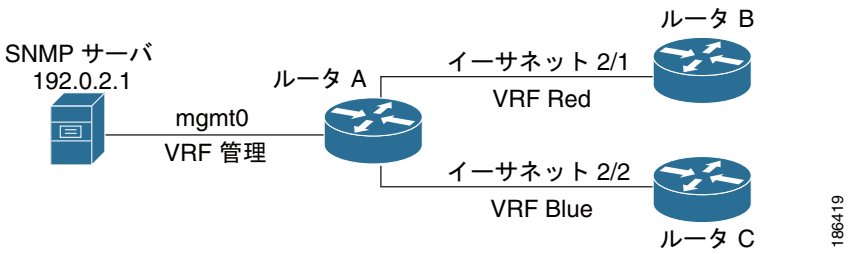


到達可能性とフィルタリングの組み合わせ

VRF 認識サービスの到達可能性とフィルタリングを組み合わせることができます。そのサービスに接続するために Cisco NX-OS が使用する VRF とともに、サービスがサポートする VRF も設定できます。デフォルト VRF でサービスを設定する場合は、任意で、すべての VRF をサポートするようにサービスを設定できます。

図 13-5 に、管理 VRF 上で到達できる SNMP サーバを示します。たとえば、SNMP サーバが VRF Red からの SNMP 通知だけをサポートするように設定できます。

図 13-5　サービス VRF の到達可能性とフィルタリング



VRF のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	VRF にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

VRF の前提条件

VRF の前提条件は、次のとおりです。

- デフォルト VDC 以外の VDC を使用するには、Advanced Services ライセンスをインストールする必要があります。

注意事項および制約事項

VRF に関する注意事項および制約事項は、次のとおりです。

- インターフェイスを既存 VRF のメンバにすると、Cisco NX-OS はあらゆるレイヤ 3 設定を削除します。VRF にインターフェイスを追加したあとで、すべてのレイヤ 3 パラメータを設定する必要があります。
- 管理 VRF に mgmt0 インターフェイスを追加し、そのあとで mgmt0 の IP アドレスおよびその他のパラメータを設定します。
- VRF が存在しないうちに VRF のインターフェイスを設定した場合は、VRF を作成するまで、そのインターフェイスは運用上のダウンになります。
- Cisco NX-OS はデフォルトで、デフォルト VRF および管理 VRF を作成します。mgmt0 は管理 VRF のメンバにする必要があります。

VRF の設定

ここでは、次の内容について説明します。

- VRF の作成（p.13-7）
- インターフェイスへの VRF メンバシップの割り当て（p.13-8）
- ルーティング プロトコルに関する VRF パラメータの設定（p.13-9）
- VRF 認識サービスの設定（p.13-11）
- VRF スコープの設定（p.13-12）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

VRF の作成

VDC に VRF を作成できます。

準備作業

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）。

手順概要

1. **config t**
2. **vrf context** *vrf-name*
3. オプション パラメータの設定
4. ***show vrf*** [*vrf-name*]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *name*
	
	例：
	`switch(config)# vrf definition Enterprise`
	`switch(config-vrf)#`
		新しい VRF を作成し、VRF 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	**ip route** *ip-prefix interface-type number* **例：** `switch(config-vrf)# ip route 192.0.2.0/8 ethernet 1/2`	（任意）スタティック ルートを設定します。
ステップ 4	**show vrf** [*vrf-name*] **例：** `switch(config-vrf)# show vrf Enterprise`	（任意）VRF 情報を表示します。

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

VRF および関連する設定を削除するには、***no vrf context*** コマンドを使用します。

コマンド	目的
no vrf context *name* **例：** `switch(config)# no vrf context Enterprise`	VRF および関連するすべての設定を削除します。

グローバル コンフィギュレーション モードで使用できるコマンドはすべて、VRF コンフィギュレーション モードでも使用できます。

VRF を作成し、VRF にスタティック ルートを追加する例を示します。

```
switch# config t
switch(config)# vrf context Enterprise
switch(config-vrf)# ip route 192.0.2.0/8 ethernet 1/2
switch(config-vrf)# exit
switch(config)# copy running-config startup-config
```

インターフェイスへの VRF メンバシップの割り当て

インターフェイスを VRF のメンバにできます。

準備作業

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）。

VRF 用のインターフェイスを設定したあとで、インターフェイスに IP アドレスを割り当てます。

手順概要

1. **config t**
2. ***interface*** *interface-type slot/port*
3. ***vrf member*** *vrf-name*
4. ***ip-address*** *ip-prefix/length*
5. ***show vrf*** *vrf-name* ***interface*** *interface-type number*
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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** *interface-type slot/port* **例：** switch(config)# interface ethernet 1/2 switch(config-if)#	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**vrf member** *vrf-name* **例：** switch(config-if)# vrf member RemoteOfficeVRF	VRF にこのインターフェイスを追加します。
ステップ 4	**ip address** *ip-prefix/length* **例：** switch(config-if)# ip address 209.0.2.1/16	このインターフェイスの IP アドレスを設定します。この作業は、VRF にこのインターフェイスを割り当てたあとで行う必要があります。
ステップ 5	**show vrf** *vrf-name* **interface** *interface-type number* **例：** switch(config-vrf)# show vrf Enterprise interface ethernet 1/2	（任意）VRF 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** switch(config)# copy running-config startup-config	（任意）この設定変更を保存します。

VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# vrf member RemoteOfficeVRF
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# copy running-config startup-config
```

ルーティング プロトコルに関する VRF パラメータの設定

1 つまたは複数の VRF にルーティング プロトコルを関連付けることができます。ルーティング プロトコルに関する VRF の設定については、該当する章を参照してください。ここでは、詳細な設定手順の例として、OSPFv2 プロトコルを使用します。

準備作業

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）。

手順概要

1. **config t**
2. **router** *protocol tag*
3. ***vrf*** *vrf-name*

4. VRF のプロトコルに対応するオプション パラメータの設定
5. *copy running-config startup-config*

手順詳細

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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 ospf** *instance-tag*
	例：
	`switch(config-vrf)# router ospf 201`
	`switch(config-router)#`
		instance tag を設定して、新しい OSPFv2 インスタンスを作成します。

ステップ 3	**vrf** *vrf-name*
	例：
	`switch(config-router)# vrf RemoteOfficeVRF`
	`switch(config-router-vrf)#`
		VRF 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	**maximum-paths** *paths* **例：** `switch(config-router-vrf)# maximum-paths 4`	（任意）この VRF のルート テーブルの宛先に到達する、等しい OSPFv2 パスの最大数を設定します。ロード バランシング用です。

ステップ 5	**interface** *interface-type slot/port*
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 6	**vrf member** *vrf-name* **例：** `switch(config-if)# vrf member RemoteOfficeVRF`	VRF にこのインターフェイスを追加します。
ステップ 7	**ip address** *ip-prefix/length* **例：** `switch(config-if)# ip address 209.0.2.1/16`	このインターフェイスの IP アドレスを設定します。この作業は、VRF にこのインターフェイスを割り当てたあとで行う必要があります。
ステップ 8	**ip router ospf** *instance-tag* **area** *area-id* **例：** `switch(config-if)# ip router ospf 201 area 0`	このインターフェイスを OSPFv2 インスタンスおよび設定エリアに割り当てます。
ステップ 9	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRF を作成し、VRF にインターフェイスを追加する例を示します。

```
switch# config t
switch(config)# vrf context RemoteOfficeVRF
switch(config-vrf)# exit
switch(config)# router ospf 201
switch(config-router)# vrf RemoteOfficeVRF
switch(config-router-vrf)# maximum-paths 4
switch(config-router-vrf)# interface ethernet 1/2
switch(config-if)# vrf member RemoteOfficeVRF
switch(config-if)# ip address 209.0.2.1/16
switch(config-if)# ip router ospf 201 area 0
switch(config-if)# exit
switch(config)# copy running-config startup-config
```

VRF 認識サービスの設定

VRF 認識サービスの到達可能性およびフィルタリングを設定できます。VRF 用サービスの設定手順を扱っている、該当する章またはコンフィギュレーション ガイドへのリンクについては、「VRF 認識サービス」（p.13-3）を参照してください。ここでは、サービスの詳細な設定手順の例として、SNMP および IP ドメイン リストを使用します。

準備作業

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）。

手順概要

1. **config t**
2. *service parameters* [***filter_vrf*** *vrf-name*] [***use-vrf*** *vrf-name*]
3. ***vrf context*** [*vrf-name*]
4. *service parameters* [**all-vrfs**][***use-vrf*** *vrf-name*]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**snmp-server host** *ip-address* [***filter-vrf*** *vrf-name*] [***use-vrf*** *vrf-name*]
	
	例：
	`switch(config)# snmp-server host 192.0.2.1 use-vrf Red`
	`switch(config-vrf)#`
		グローバル SNMP サーバを設定し、サービスに接続するために Cisco NX-OS が使用する VRF を設定します。選択した VRF からこのサーバへの情報をフィルタリングするには、***filter-vrf*** キーワードを使用します。

ステップ 3	**vrf context** *vrf-name*
	
	例：
	`switch(config)# vrf context Blue`
	`switch(config-vrf)#`
		新しい VRF を作成します。

	コマンド	目的

ステップ 4	**ip domail-list** *domain-name* [**all-vrfs**][***use-vrf*** *vrf-name*]
	
	例：
	`switch(config-vrf)# ip domain-list List all-vrfs use-vrf Blue`
	`switch(config-vrf)#`
		VRF でドメイン リストを設定し、さらに任意で、指定されたドメイン名に接続するために Cisco NX-OS が使用する VRF を設定します。

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

VRF Red で到達可能な SNMP ホスト 192.0.2.1 に、すべての VRF の SNMP 情報を送信する例を示します。

```
switch# config t
switch(config)# snmp-server host 192.0.2.1 for-all-vrfs use-vrf Red
switch(config)# copy running-config startup-config
```

VRF Red で到達可能な SNMP ホスト 192.0.2.12 に対して、VRF Blue の SNMP 情報をフィルタリングする例を示します。

```
switch# config t
switch(config)# vrf definition Blue
switch(config-vrf)# snmp-server host 192.0.2.12 use-vrf Red
switch(config)# copy running-config startup-config
```

VRF スコープの設定

すべての EXEC コマンド（**show** コマンドなど）に対応する VRF スコープを設定できます。VRF スコープを設定すると、EXEC コマンド出力のスコープが設定された VRF に自動的に限定されます。このスコープは、一部の EXEC コマンドで使用できる VRF キーワードによって上書きできます。

VRF スコープを設定するには、EXEC モードで次のコマンドを使用します。

コマンド	目的

routing-context vrf *vrf-name*
 	
例：
`switch# routing-context vrf red`
`switch%red#`
	すべての EXEC コマンドに対応するルーティング コンテキストを設定します。デフォルトのルーティング コンテキストはデフォルト VRF です。

デフォルトの VRF スコープに戻すには、EXEC モードで次のコマンドを使用します。

コマンド	目的

routing-context vrf default
 	
例：
`switch%red# routing-context vrf default`
`switch#`
	デフォルトのルーティング コンテキストを設定します。

VRF の設定確認

VRF の設定情報を表示するには、次のコマンドのいずれかを使用します。

コマンド	目的
show vrf [*vrf-name*]	すべてまたは 1 つの VRF の情報を表示します。
show vrf [*vrf-name*] **detail**	すべてまたは 1 つの VRF の詳細情報を表示します。
show vrf [*vrf-name*] [**interface** *interface-type slot/port*]	インターフェイスの VRF ステータスを表示します。

VRF の設定例

VRF Red を設定し、その VRF に SNMP サーバを追加し、VRF Red に OSPF インスタンスを追加する例を示します。

```
config t
vrf context Red
 snmp-server host 192.0.2.12 use-vrf Red
router ospf 201
interface ethernet 1/2
 vrf member Red
 ip address 209.0.2.1/16
 ip router ospf 201 area 0
```

関連資料

VRF の詳細については、次の関連資料を参照してください。

- 『*Cisco NX-OS Getting Started Configuration Guide*』
- 『*Cisco NX-OS Virtual Device Context Configuration Guide*』

デフォルト設定

表 13-1 に、VRF パラメータのデフォルト設定を示します。

表 13-1　デフォルトの VRF パラメータ

パラメータ	デフォルト
設定されている VRF	デフォルト、管理
ルーティング コンテキスト	デフォルト VRF

その他の関連資料

仮想化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。

- 関連資料（p.13-14）
- 規格（p.13-14）

関連資料

関連項目	マニュアル名
VRF CLI	『*Cisco NX-OS Unicast Routing Command Line Reference*』
VDC	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 14

Route Policy Manager の設定

この章では、Route Policy Manager の設定手順について説明します。

ここでは、次の内容を説明します。

- Route Policy Manager の概要（p.14-2）
- Route Policy Manager のライセンス要件（p.14-5）
- Route Policy Manager の前提条件（p.14-5）
- 注意事項および制約事項（p.14-5）
- Route Policy Manager の設定（p.14-6）
- Route Policy Manager の設定確認（p.14-14）
- Route Policy Manager の設定例（p.14-14）
- 関連項目（p.14-14）
- デフォルト設定（p.14-14）

Route Policy Manager の概要

Route Policy Manager は、ルート マップおよび IP プレフィクス リストをサポートします。これらの機能は、ルートの再配布およびポリシーベース ルーティングに使用します。プレフィクス リストには、1 つまたは複数の IPv4 または IPv6 ネットワーク プレフィクスおよび関連付けられたプレフィクス長の値を指定します。プレフィクス リストは、BGP（ボーダー ゲートウェイ プロトコル）テンプレート、ルート フィルタリング、またはルーティング ドメイン間で交換されるルートの再配布などの機能で、単独で使用できます。

ルート マップは、ルートおよび IP パケットの両方に適用できます。ルート フィルタリングおよび再配布では、ルート マップを介してルートを渡すのに対して、ポリシーベース ルーティングでは、ルート マップを介して IP パケットを渡します。

ここでは、次の内容について説明します。

- プレフィクス リスト（p.14-2）
- ルート マップ（p.14-2）
- ルートの再配布およびルート マップ（p.14-4）
- ポリシーベース ルーティング（p.14-4）

プレフィクス リスト

プレフィクス リストを使用すると、アドレスまたはアドレス範囲を許可または拒否できます。プレフィクス リストによるフィルタリングには、ルートまたはパケットのプレフィクスとプレフィクス リストに指定されているプレフィクスの照合が必要です。所定のプレフィクスがプレフィクス リストのどのエントリとも一致しなかった場合は、暗黙の拒否とみなされます。

プレフィクス リストに複数のエントリを設定し、エントリと一致したプレフィクスを許可または拒否できます。各エントリには、ユーザ設定可能なシーケンス番号が関連付けられます。シーケンス番号を設定しなかった場合は、Cisco NX-OS がシーケンス番号を自動的に割り当てます。Cisco NX-OS は、最小シーケンス番号からプレフィクス リストを評価していきます。Cisco NX-OS は、所定のプレフィクスとの最初の一致を処理します。一致すると、Cisco NX-OS は許可または拒否文を処理し、残りの許可リストは評価しません。

（注） 空の許可リストの場合は、すべてのルートが許可されます。

ルート マップ

ルート マップは、ルートの再配布またはポリシーベース ルーティングに使用できます。ルート マップ エントリは、一致基準および設定基準のリストからなります。一致基準では、着信ルートまたはパケットの一致条件を指定します。設定基準では、一致基準を満たした場合のアクションを指定します。

同じルート マップに複数のエントリを設定できます。これらのエントリには、同じルート マップ名を指定し、シーケンス番号で区別します。

一意のルート マップ名の下に 1 つまたは複数の ルート マップ エントリをシーケンス番号に従って並べ、ルート マップを作成します。ルート マップ エントリのパラメータは、次のとおりです。

- シーケンス番号
- アクセス権 — 許可または拒否
- 一致基準
- 設定変更

ルート マップではデフォルトで、最小のシーケンス番号から順にルートまたは IP パケットが処理されます。**continue** 文を使用すると、次に処理するルート マップ エントリを決定できるので、別の順序で処理するようにルート マップを設定できます。

一致基準

さまざまな基準を使用して、ルート マップのルートまたは IP パケットを照合できます。BGP コミュニティ リストのように、特定のルーティング プロトコルだけに適用できる基準もありますが、IP 送信元または宛先アドレスなど、その他の基準はあらゆるルートまたは IP パケットに使用できます。

ルート マップに従ってルートまたはパケットを処理する場合、Cisco NX-OS は設定されている個々の match 文とルートまたはパケットを比較します。ルートまたはパケットが設定されている基準と一致した場合、Cisco NX-OS はルート マップの一致エントリに設定されている許可または拒否、および設定されている設定基準に基づいて、ルートまたはパケットを処理します。

一致のカテゴリおよびパラメータは、次のとおりです。

- IP アクセス リスト ― （ポリシーベース ルーティングの場合のみ）送信元または宛先 IP アドレス、プロトコル、または QoS（Quality of Service）パラメータに基づく一致
- BGP パラメータ ― AS（自律システム）パスまたはコミュニティ アトリビュートに基づく一致
- プレフィクス リスト ― アドレスまたはアドレス範囲に基づく一致
- マルチキャスト パラメータ ― ランデブー ポイント、グループ、または送信元に基づく一致
- その他のパラメータ ― IP ネクストホップ アドレスまたはパケット長に基づく一致

設定変更

ルートまたはパケットがルート マップ エントリと一致すると、設定した 1 つまたは複数の set 文に基づいて、そのルートまたはパケットを変更できます。

設定変更は次のとおりです。

- BGP パラメータ ― AS パス、タグ、コミュニティ、ダンプニング、ローカル プリファレンス、オリジン、または重み値アトリビュートの変更
- メトリック ― ルート メトリック、ルート タグ、またはルート タイプの変更
- ポリシーベース ルーティングのみ ― インターフェイスまたはデフォルト ネクストホップ アドレスの変更
- その他のパラメータ ― フォワーディング アドレスまたは IP ネクストホップ アドレスの変更

アクセス リスト

IP アクセス リストでは、次のような IP パケット フィールドとパケットを照合できます。

- 送信元または宛先 IPv4 または IPv6 アドレス
- プロトコル
- プレシデンス
- ToS（タイプ オブ サービス）

ルート マップで ACL（アクセス コントロール リスト）を使用できるのは、ポリシーベース ルーティングの場合に限られます。ACL の詳細については、『*Cisco NX-OS Security Configuration Guide*』Release 4.0 を参照してください。

BGP の AS パス リスト

AS パス リストを設定すると、着信または発信 BGP ルート アップデートをフィルタリングできます。ルート アップデートに AS パス リストのエントリと一致する AS パス アトリビュートが含まれている場合、ルータは設定されている許可または拒否条件に基づいてルートを処理します。ルート マップの中で AS パス リストを設定できます。

同じ AS パス リスト名を使用することによって、AS パス リストで複数の AS パス エントリを設定できます。ルータは最初に一致したエントリを処理します。

BGP のコミュニティ リスト

ルート マップのコミュニティ リストを使用すると、BGP コミュニティに基づいて BGP ルート アップデートをフィルタリングできます。コミュニティ リストには、1 つまたは複数のコミュニティ アトリビュートを指定します。同じコミュニティ リスト エントリに複数のコミュニティ アトリビュートを設定した場合、BGP ルートが一致とみなされるには、指定されたすべてのコミュニティ アトリビュートと一致しなければなりません。

同じコミュニティ リスト名を使用することによって、コミュニティ リストのそれぞれ個別のエントリとして、複数のコミュニティ アトリビュートを設定することもできます。この場合、ルータは最初に BGP ルートと一致したコミュニティ アトリビュートを、そのエントリの許可または拒否設定に基づいて処理します。

コミュニティ リストのコミュニティ アトリビュートは、次の形式のいずれか 1 つで設定できます。

- 名前付きコミュニティ アトリビュート（**internet**、**no-export** など）
- AS 番号およびユーザ定義のネットワーク番号を表す 4 バイト値
- 正規表現

正規表現の詳細については、『*Cisco NX-OS Unicast Routing Command Reference*』Release 4.0 を参照してください。

ルートの再配布およびルート マップ

ルート マップを使用すると、ルーティング ドメイン間でルートの再配布を制御できます。ルート マップではルートのアトリビュートを照合し、一致基準に合格したルートだけを再配布します。設定変更を使用することによって、再配布時に、ルート マップでルート アトリビュートを変更することもできます。

ルータは再配布ルートを各ルート マップ エントリと照合します。複数の match 文がある場合、ルートはすべての一致基準に合格しなければなりません。ルートがルート マップ エントリで定義されている一致基準に合格した場合は、エントリで定義されているアクションが実行されます。ルートが基準と一致しなかった場合、ルータは後続のルート マップ エントリとルートを比較します。一致するか、または一致しないまま、ルート マップのすべてのエントリによってルートが処理されるまで、ルート処理が続けられます。ルータがルート マップの全エントリとルートを比較しても、一致しなかった場合、ルータはそのルートを受け付けないか（着信ルート マップ）またはルートを転送します（発信ルート マップ）。

ポリシーベース ルーティング

ポリシーベース ルーティングを使用すると、パケットの送信元またはパケット ヘッダーのその他のフィールドに基づいて、特定のネクストホップ アドレスにパケットを転送できます。第 15 章「ポリシーベース ルーティングの設定」を参照してください。

Route Policy Manager のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	Route Policy Manager にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

Route Policy Manager の前提条件

Route Policy Manager の前提条件は、次のとおりです。

- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

注意事項および制約事項

Route Policy Manager に関する注意事項および制約事項は、次のとおりです。

- 空のルート マップの場合は、すべてのルートが拒否されます。
- 空のプレフィクス リストの場合は、すべてのルートが許可されます。
- ルート マップ エントリに match 文がなかった場合、ルート マップ エントリのアクセス権（許可または拒否）によって、すべてのルートまたはパケットの結果が決まります。
- ルート マップ エントリの match 文の中で参照されたポリシー（プレフィクス リストなど）から no-match または deny-match が戻った場合、Cisco NX-OS は match 文を失敗として、次のルート マップ エントリを処理します。
- ルート マップを変更しても、ルート マップ コンフィギュレーション サブモードを終了するまでは、Cisco NX-OS がすべての変更を保留します。その後、Cisco NX-OS はすべての変更が有効になるように、プロトコル クライアントに送信します。
- ルート　マップは定義する前に使用できるので、設定変更を終えるときには、すべてのルート マップが存在していることを確認してください。
- ルート マップの用途が再配布なのか、フィルタリングなのかを確認できます。これらの統計の表示方式は、個々のルーティング プロトコルによって決まります。

Route Policy Manager の設定

Route Policy Manager の設定では、次の内容を扱います。

- IP プレフィクス リストの設定（p.14-6）
- AS パス リストの設定（p.14-7）
- コミュニティ リストの設定（p.14-8）
- ルート マップの設定（p.14-9）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

IP プレフィクス リストの設定

IP プレフィクス リストでは、プレフィクスおよびプレフィクス長のリストに対して IP パケットまたはルートを照合します。IPv4 には IP プレフィクス リスト、IPv6 には IPv6 プレフィクス リストを作成します。

手順概要

1. **config t**
2. ***{ip | ipv6} prefix-list*** *name* **description** *string*
3. ***ip prefix-list*** *name* [***seq*** *number*] [{***permit*** | ***deny***} *prefix* {[***eq*** *prefix-length*] | [***ge*** *prefix-length*] [***le*** *prefix-length*]}]

   または

   ipv6 prefix-list *name* [***seq*** *number*] [{***permit*** | ***deny***} *prefix* **{[*eq* *prefix-length*] | [*ge* *prefix-length*] [*le* *prefix-length*]}]**
4. ***show*** {***ip*** | ***ipv6***} ***prefix-list*** *name*
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	`config t`
	
	例：
	`switch# config t`
	`switch(config)#`
		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	{***ip*** | ***ipv6***} ***prefix-list*** *name* **description** *string* **例：** `switch(config)# ip prefix-list AllowPrefix description allows engineering server`	（任意）プレフィクス リストについての情報ストリングを追加します。

	コマンド	目的
ステップ 3	***ip prefix-list*** *name* [***seq*** *number*] [{***permit*** | ***deny***} *prefix* {[***eq*** *prefix-length*] | [***ge*** *prefix-length*] [***le*** *prefix-length*]}] **例：** `switch(config)# ip prefix-list AllowPrefix 10 permit 192.0.2.0 eq 24`	IPv4 プレフィクス リストを作成するか、または既存のプレフィクス リストにプレフィクスを追加します。プレフィクス長の一致は次のようになります。 • eq ― *prefix length* と厳密に一致 • ge ― 設定された *prefix length* 以上のプレフィクス長と一致 • le ― 設定された *prefix length* 以下のプレフィクス長と一致
	ipv6 prefix-list *name* [***seq*** *number*] [{***permit*** | ***deny***} *prefix* {[***eq*** *prefix-length*] | [***ge*** *prefix-length*] [***le*** *prefix-length*]}] **例：** `switch(config)#ipv6 prefix-list AllowIPv6Prefix 10 permit 2001:0DB8:: le 32`	IPv6 プレフィクス リストを作成するか、または既存のプレフィクス リストにプレフィクスを追加します。プレフィクス長の設定は次のとおりです。 • eq ― *prefix length* と厳密に一致 • ge ― 設定された *prefix length* 以上のプレフィクス長と一致 • le ― 設定された *prefix length* 以下のプレフィクス長と一致
ステップ 4	***show*** {***ip*** | ***ipv6***} ***prefix-list*** *name* **例：** `switch(config)# show ip prefix-list AllowPrefix`	（任意）プレフィクス リスト情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch# copy running-config startup-config`	（任意）この設定変更を保存します。

2 つのエントリからなる IPv4 プレフィクス リストを作成し、BGP ネイバーにプレフィクス リストを適用する例を示します。

```
switch# config t
switch(config)# ip prefix-list allowprefix 10 permit 192.0.2.0 eq 24
switch(config)# ip prefix-list allowprefix 20 permit 209.165.201.0 eq 27
switch(config)# router bgp 33:20
switch(config-router)# neighbor 192.0.2.1/16 remote-as 99:20
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# prefix-list allowprefix in
```

AS パス リストの設定

発信および着信 BGP ルートの両方に、AS パス リスト フィルタを指定できます。各フィルタは、正規表現を使用するアクセス リストです。正規表現が ASCII ストリングとして表されたルートの AS パス アトリビュートと一致した場合は、許可または拒否条件が適用されます。

手順概要

1. **config t**
2. **ip as-path access-list** *name* {***deny*** | ***permit***} *expression*
3. ***show*** {***ip*** | ***ipv6***} ***as-path list*** *name*
4. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***ip as-path access-list*** *name* {***deny*** | ***permit***} *expression* **例：** `switch(config)# ip as-path access-list Allow40 permit 40`	正規表現を使用して BGP AS パス リストを作成します。
ステップ 3	***show*** {***ip*** | ***ipv6***} ***as-path-access-list*** *name* **例：** `switch(config)# show ip as-path-access-list Allow40`	（任意）AS パス アクセス リスト情報を表示します。
ステップ 4	***copy running-config startup-config*** **例：** `switch# copy running-config startup-config`	（任意）この設定変更を保存します。

2 つのエントリからなる AS パス リストを作成し、BGP ネイバーに AS パス リストを適用する例を示します。

```
switch# config t
switch(config)# ip as-path access-list AllowAS permit 40
switch(config)# ip as-path access-list AllowAS permit 40000
switch(config)# copy running-config startup-config
switch(config)# router bgp 33:20
switch(config-router)# neighbor 192.0.2.1/16 remote-as 99:20
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# filter-list AllowAS in
```

コミュニティ リストの設定

コミュニティ リストを使用すると、コミュニティ アトリビュートに基づいて BGP ルートをフィルタリングできます。コミュニティ番号は *aa:nn* 形式の 4 バイト値です。最初の 2 バイトは AS 番号を表し、後続の 2 バイトはユーザ定義のネットワーク番号です。

同じコミュニティ リスト文で複数の値を設定した場合、コミュニティ リスト フィルタを満足させるには、すべてのコミュニティ値が一致しなければなりません。複数の値をそれぞれ個別のコミュニティ リスト文で設定した場合は、最初に条件が一致したリストが処理されます。

手順概要

1. **config t**
2. ***ip community-list standard*** *list-name* [*community-list*] [**internet**] [**local-AS**] [**no-advertise**] [**no-export**]

   または

   ip community-list expanded *list-name* {***deny*** | ***permit***} *expresion*
3. ***show ip community-list*** *name*
4. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***ip community-list standard*** *list-name* {***deny*** | ***permit***} [*community-list*] [***internet***] [***local-AS***] [***no-advertise***] [***no-export***] **例：** `switch(config)# ip community-list standard BGPCommunity permit no-advertise 40000:20`	標準 BGP コミュニティ リストを作成します。*community-list* は、次のうちの 1 つまたは複数にできます。 • ***internet*** • ***local-AS*** • ***no-advertise*** • ***no-export*** • *aa:nn* 形式の 1 つまたは複数のコミュニティ
	ip community-list expanded *list-name* {***deny*** | ***permit***} *line* **例：** `switch(config)# ip community-list expanded BGPComplex deny 50000:[0-9][0-9]_`	正規表現を使用して拡張 BGP AS コミュニティ リストを作成します。
ステップ 3	***show ip community-list*** *name* **例：** `switch(config)# show ip community-list BGPCommunity`	（任意）プレフィクス リスト情報を表示します。
ステップ 4	***copy running-config startup-config*** **例：** `switch# copy running-config startup-config`	（任意）この設定変更を保存します。

2 つのエントリからなるコミュニティ リストの作成例を示します。

```
switch# config t
switch(config)# ip community-list standard BGPCommunity permit no-advertise 40000:20
switch(config)# ip community-list standard BGPCommunity permit local-AS no-export
switch(config)# copy running-config startup-config
```

ルート マップの設定

ルート マップは、ルートの再配布またはルート フィルタリングに使用できます。ルート マップには、複数の一致基準と複数の設定基準を含めることができます。

BGP にルート マップを設定すると、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のトリガーになります。

手順概要

1. **config t**
2. ***route-map*** *map-name* [***permit*** | ***deny***] [*seq*]
3. ルート マップ コンフィギュレーション モードでオプションの一致または設定パラメータを追加

4. *exit*
5. *copy running-config startup-config*

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***route-map*** *map-name* [***permit*** | ***deny***] [*seq*]
	
	例：
	`switch(config)# route-map Testmap permit 10`
	`switch(config-route-map)#`
		ルート マップを作成するか、または既存のルート マップに対応するルート マップ コンフィギュレーション モードを開始します。*seq* を使用して、ルート マップ エントリを順序付けます。

ステップ 3	**description** *string* **例：** `switch(config-route-map)# description A test route map`	（任意）ルート マップ シーケンスの説明を追加します。
ステップ 4	**continue** *seq* **例：** `switch(config-route-map)# continue 10`	（任意）ルート マップで次を処理するシーケンス文を決定します。使用するのは、フィルタリングおよび再配布の場合だけです。
ステップ 5	**exit** **例：** `switch(config-route-map)# exit`	（任意）ルート マップ コンフィギュレーション モードを終了します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

ルート マップ コンフィギュレーション モードで、オプションとして、ルート マップに次の match パラメータを設定できます。

コマンド	目的
match as-path *name* [*name...*] **例：** `switch(config-route-map)# match as-path Allow40`	1 つまたは複数の AS パス リストと照合。AS パス リストは、***ip as-path access-list*** コマンドで作成します。
match community *name* [*name...*][***exact-match***] **例：** `switch(config-route-map)# match community BGPCommunity`	1 つまたは複数のコミュニティ リストと照合。AS パス リストは、***ip community-list*** コマンドで作成します。
match ip address prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip address prefix-list AllowPrefix`	1 つまたは複数の IPv4 プレフィクス リストと照合。プレフィクス リストは **ip prefix-list** コマンドを使用して作成します。

コマンド	目的
match ipv6 address prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip address prefix-list AllowIPv6Prefix`	1 つまたは複数の IPv6 プレフィクス リストと照合。プレフィクス リストは **ipv6 prefix-list** コマンドを使用して作成します。
match ip multicast [***source*** *ipsource*] [[***group*** *ipgroup*] [*rp iprp*]] **例：** `switch(config-route-map)# match ip multicast rp 192.0.2.1`	マルチキャスト送信元、グループ、またはランデブーポイントに基づいて IPv4 マルチキャスト パケットを照合
match ipv6 multicast [***source*** *ipsource*] [[***group*** *ipgroup*] [*rp iprp*]] **例：** `switch(config-route-map)# match ip multicast source 2001:0DB8::1`	マルチキャスト送信元、グループ、またはランデブーポイントに基づいて IPv6 マルチキャスト パケットを照合
match ip next-hop prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip next-hop prefix-list AllowPrefix`	1 つまたは複数の IP プレフィクス リストに対して、ルートの IP ネクストホップ アドレスを照合。プレフィクス リストは **ip prefix-list** コマンドを使用して作成します。
match ipv6 next-hop prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip next-hop prefix-list AllowIPv6Prefix`	1 つまたは複数の IP プレフィクス リストに対して、ルートの IPv6 ネクストホップ アドレスを照合。プレフィクス リストは **ipv6 prefix-list** コマンドを使用して作成します。
match ip route-source prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip route-source prefix-list AllowPrefix`	1 つまたは複数の IP プレフィクス リストに対して、ルートの IPv4 ルート送信元アドレスを照合。プレフィクス リストは **ip prefix-list** コマンドを使用して作成します。
match ipv6 route-source prefix-list *name* [*name...*] **例：** `switch(config-route-map)# match ip route-source prefix-list AllowIPv6Prefix`	1 つまたは複数の IP プレフィクス リストに対して、ルートの IPv6 ルート送信元アドレスを照合。プレフィクス リストは **ipv6 prefix-list** コマンドを使用して作成します。
match route-type *route-type* **例：** `switch(config-route-map)# match route-type level 1 level 2`	ルート タイプと照合。*route-type* は、次のうちの 1 つまたは複数にできます。 • external • internal • level-1 • level-2 • local • nssa-external • type-1 • type-2
match tag *tagid* [*tagid...*] **例：** `switch(config-route-map)# match tag 2`	フィルタリングまたは再配布に関する 1 つまたは複数のタグとルートを照合。

ルート マップ コンフィギュレーション モードで、オプションとして、ルート マップに次の set パラメータを設定できます。

コマンド	目的
set as-path {**tag** | ***prepend*** {***last-as*** *number* | *as-1* [*as-2...*]}} **例：** `switch(config-route-map)# set as-path prepend 10 100 110`	BGP ルートの AS パス アトリビュートを変更します。最後の AS 番号として設定された *number* または特定の AS パス値としてのストリング（*as-1 as-2...as-n*）をプリペンドできます。
set comm-list *name* ***delete*** **例：** `switch(config-route-map)# set comm-list BGPCommunity delete`	着信または発信 BGP ルート アップデートのコミュニティ アトリビュートから、コミュニティを削除します。コミュニティ リストは ***ip community-list*** コマンドを使用して作成します。
set community {***none*** |{***additive*** | ***local-AS*** | ***no-advertise*** | ***no-export*** | *community-1* [community-2...]} **例：** `switch(config-route-map)# set community local-AS`	BGP ルート アップデートのコミュニティ アトリビュートを設定します。 **（注）** ルート マップ アトリビュートの同じシーケンスで、***set community*** コマンドと ***set comm-list delete*** コマンドを両方使用すると、設定処理より先に削除処理が実行されます。
set dampening *halflife reuse suppress duration* **例：** `switch(config-route-map)# set dampening 30 1500 10000 120`	BGP ルート ダンプニング パラメータを設定します。 • *halflife* ― 範囲は 1 ～ 45 分です。デフォルトは 15 です。 • *reuse* ― 範囲は 1 ～ 20000 秒です。デフォルトは 750 です。 • *suppress* ― 範囲は 1 ～ 20000 です。デフォルトは 2000 です。 • *duration* ― 範囲は 1 ～ 255 分です。デフォルトは 60 です。
set forwarding-address **例：** `switch(config-route-map)# set forwarding-address`	OSPF のフォワーディング アドレスを設定します。
set level {***backbone*** | ***level-1*** | ***level-1-2*** | ***level-2***} **例：** `switch(config-route-map)#`	IS-IS 用にルートをインポートするエリアを設定します。IS-IS のオプションは level-1、level-1-2、または level-2 です。デフォルトは level-1 です。
set local-preference *value* **例：** `switch(config-route-map)# set local-preference 4000`	BGP ローカル プリファレンス値を設定します。有効値の範囲は 0 ～ 4294967295 です。
set metric *metric0* [*metric1 metric2 metric3 metric4*] **例：** `switch(config-route-map)#`	ルート メトリック値を設定します。メトリックは次のとおりです。 • *metric0* ― 帯域幅（kbps） • *metric1* ― 遅延（10 マイクロ秒単位） • *metric2* ― 信頼性。有効値の範囲は 0 ～ 255（100% の信頼性）です。 • *metric3* ― ロード。有効値の範囲は 1 ～ 200（100% のロード）です。 • *metric4* ― パスの MTU

コマンド	目的
set metric-type {***external*** | ***internal*** | ***type-1*** | ***type-2***} **例：** `switch(config-route-map)# set metric-type internal.`	宛先ルーティング プロトコルのメトリック タイプを設定します。オプションは次のとおりです。 external ― IS-IS 外部メトリック internal ― BGP の MED として IGP メトリックを使用 type-1 ― OSPF 外部 タイプ 1 メトリック type-2 ― OSPF 外部 タイプ 2 メトリック
set origin {***egp*** *as-number* | ***igp*** | ***incomplete***} **例：** `switch(config-route-map)#`	BGP オリジン アトリビュートを設定します。EGP *as-number* の範囲は 0 ～ 65535 です。
set tag *name* **例：** `switch(config-route-map)# set tag 33`	宛先ルーティング プロトコルのタグ値を設定します。*name* パラメータは符号なし整数です。
set weight *count* **例：** `switch(config-route-map)#`	BGP ルートの重み値を設定します。有効値の範囲は 0 ～ 65535 です。

set metric-type internal コマンドが作用するのは、発信ポリシーに限られます。また、eBGP ネイバーだけに作用します。同じ BGP ピア発信ポリシーに ***metric*** コマンドと ***metric-type internal*** コマンドを両方設定した場合、Cisco NX-OS は ***metric-type internal*** コマンドを無視します。

Route Policy Manager の設定確認

Route Policy Manager の設定を確認するには、**show route-map** コマンドを使用します。

Route Policy Manager の設定例

アドレス ファミリを使用して BGP を設定し、ネイバー 209.0.2.1 からのユニキャストおよびマルチキャスト ルートがアクセス リスト 1 と一致した場合に、受け付けられるようにする例を示します。

```
router bgp 40000
  address-family ipv4 unicast
    network 192.0.2.0/24
    network 209.165.201.0/27 route-map filterBGP
route-map filterBGP
 match ip next-hop prefix-list AllowPrefix

ip prefix-list AllowPrefix 10 permit 192.0.2.0 eq 24
ip prefix-list AllowPrefix 20 permit 209.165.201.0 eq 27
```

関連項目

Route Policy Manager の詳細については、次の項目を参照してください。

- 第 9 章「ベーシック BGP の設定」
- 第 15 章「ポリシーベース ルーティングの設定」

デフォルト設定

表 14-1 に、Route Policy Manager のデフォルト設定を示します。

表 14-1　Route Policy Manager のデフォルト パラメータ

パラメータ	デフォルト
Route Policy Manager	イネーブル

その他の関連資料

IP 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。

- 関連資料（p.14-15）
- 規格（p.14-15）

関連資料

関連項目	マニュアル名
Route Policy Manager CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 15

ポリシーベース ルーティングの設定

この章では、ポリシーベース ルーティングの設定手順について説明します。

ここでは、次の内容を説明します。

- ポリシーベース ルーティングの概要（p.15-2）
- ポリシーベース ルーティングのライセンス要件（p.15-4）
- ポリシーベース ルーティングの前提条件（p.15-4）
- 注意事項および制約事項（p.15-4）
- ポリシーベース ルーティングの設定（p.15-5）
- ポリシーベース ルーティングの設定確認（p.15-9）
- ポリシーベース ルーティングの設定例（p.15-9）
- 関連項目（p.15-9）
- デフォルト設定（p.15-9）

ポリシーベース ルーティングの概要

ポリシーベース ルーティングを使用すると、トラフィック フローに定義済みのポリシーを設定し、ルーティング プロトコルから派生したルートへの依存を弱めることができます。ポリシーベース ルーティングがイネーブルのインターフェイスで受信するすべてのパケットは、拡張パケット フィルタまたは*ルート マップ*を経由して渡されます。ルート マップでは、パケットの転送先を決定するポリシーを記述します。

ルート マップは match 文および set 文からなり、許可または拒否を指定できます。文の解釈は次のとおりです。

- パケットがあらゆるルート マップと一致する match 文の場合、すべての set 文が適用されます。そのアクションの 1 つに、ネクストホップの選択が含まれます。
- 文に拒否が指定されている場合、一致基準を満たすパケットは標準のフォワーディング チャネルを通じて送り返され、宛先ベース ルーティングが実行されます。
- 文に許可が指定されていて、なおかつパケットがルート マップ文と一致しない場合、パケットは標準のフォワーディング チャネルを通じて送り返され、宛先ベース ルーティングが実行されます。

「ルート マップ」（p.14-2）を参照してください。

ポリシーベース ルーティングには、次の機能が含まれます。

- 送信元ベース ルーティング ― 異なるユーザ セットを起点とするトラフィックをポリシー ルータ上のそれぞれ異なる接続を使用してルーティングします。
- QoS（Quality of Service）― ネットワークの周辺で IP パケット ヘッダーに優先または ToS（タイプ オブ サービス）値を設定することによって、またはキューイング メカニズムを利用して、ネットワークのコアまたはバックボーンでトラフィックにプライオリティを設定することによって、トラフィックを差別化します（『*Cisco NX-OS Quality of Service Configuration Guide*』Release 4.0 を参照）。
- ロード シェアリング ― トラフィックの特性に基づいて、複数のパスにトラフィックを分散します。

ここでは、次の内容について説明します。

- ポリシー ルート マップ（p.15-2）
- ポリシーベース ルーティングの set 基準（p.15-3）

ポリシー ルート マップ

ルート マップの各エントリで、match および set 文のコンビネーションを指定します。match 文では、該当するパケットが特定のポリシーを満たす基準（すなわち、満たすべき条件）を定義します。set 文節で、match 基準を満たしたパケットをどのようにルーティングするかを説明します。

ルート マップ文を許可または拒否として指定できます。文に拒否が指定されている場合、一致基準を満たすパケットは標準のフォワーディング チャネルを通じて送り返されます（宛先ベース ルーティングが実行されます）。文に許可が指定されていて、なおかつパケットが一致基準を満たしている場合は、すべての set 文節が適用されます。文に許可が指定されていて、なおかつパケットが一致基準を満たしていない場合は、それらのパケットも標準のルーティング チャネルを通じて転送されます。

（注） ポリシー　ルーティングは、パケットの送信元となるインターフェイスではなく、パケットを受信するインターフェイス上で指定します。

ポリシーベース ルーティングの set 基準

ルート マップの set 基準は、ルート マップに指定された順番で評価されます。ポリシーベース ルーティング用のルート マップに固有の set 基準は、次のとおりです。

1. パケットを通過させてルーティングできるインターフェイスのリスト ― 複数のインターフェイスを指定した場合は、最初にアップとして検出されたインターフェイスがパケット転送に使用されます。
2. 指定 IP アドレスのリスト ― IP アドレスでは、パケットの転送先である宛先へのパス上の隣接ネクストホップ ルータを指定できます。その時点でアップの接続インターフェイスに関連付けられた最初の IP アドレスがパケットのルーティングに使用されます。
3. デフォルト インターフェイス リスト ― ポリシー ルーティング対象とされるパケットの宛先アドレスに使用できる明示的ルートがない場合は、ルート マップによって、指定デフォルト インターフェイス リストで最初にアップだったインターフェイスにパケットがルーティングされます。
4. デフォルト ネクストホップ IP アドレスのリスト ― ルーティング テーブルに、パケットの宛先アドレスに対応する明示的ルートがない場合は、この set 文で指定されたインターフェイスまたはネクストホップ アドレスだけにルーティングされます。

パケットが定義された一致基準のいずれにも一致しない場合、そのパケットは標準の宛先ベース ルーティング プロセスを使用してルーティングされます。

ポリシーベース ルーティングのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	ポリシーベース ルーティングには Enterprise Services ライセンスが必要です。NX-OS ライセンス方式の詳細、ライセンスを取得して適用する方法については、『*Cisco NX-OS Licensing Guide*』を参照してください。

ポリシーベース ルーティングの前提条件

ポリシーベース ルーティングの前提条件は、次のとおりです。

- 有効なライセンスをインストールします。
- ポリシーベース ルーティング機能をイネーブルにする必要があります（「ポリシーベース ルーティング機能のイネーブル化」[p.15-5] を参照）。
- インターフェイスに IP アドレスを割り当て、インターフェイスをアップにしてから、ポリシーベース ルーティング用のルート マップをインターフェイス上で適用します。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

注意事項および制約事項

ポリシーベース ルーティングに関する注意事項および制約事項は、次のとおりです。

- ポリシーベース ルーティングのルート マップに指定できるルート マップ エントリ（match および set を指定）は 1 つだけです。
- ポリシーベース ルーティングのルート マップでは、ルート マップ エントリに複数の match および set コマンドを指定することはできません。
- match コマンドで、ポリシーベース ルーティング用ルート マップの複数の ACL を参照することはできません。
- インターフェイスが同じ VRF に所属している場合は、ポリシーベース ルーティング対応のさまざまなインターフェイス間で、同じルート マップを共有できます。

ポリシーベース ルーティングの設定

ここでは、次の内容について説明します。

- ポリシーベース ルーティング機能のイネーブル化（p.15-5）
- ルート ポリシーの設定（p.15-6）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

ポリシーベース ルーティング機能のイネーブル化

ルート ポリシーを設定する前に、ポリシーベース ルーティング機能をイネーブルにしておく必要があります。

準備作業

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）。

手順概要

1. **config t**
2. ***feature pbr***
3. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature pbr** **例：** `switch(config)# feature pbr`	ポリシーベース ルーティング機能をイネーブルにします。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

ポリシーベース ルーティング機能をディセーブルにして、関連するすべての設定を削除する場合は、***no feature pbr*** コマンドを使用します。

コマンド	目的
no feature pbr **例：** `switch(config)# no feature pbr`	ポリシーベース ルーティング機能をディセーブルにして、関連するすべての設定を削除します。

ルート ポリシーの設定

ポリシーベース ルーティングでルート マップを使用すると、着信インターフェイスにルーティング ポリシーを割り当てることができます。「ルート マップの設定」（p.14-9）を参照してください。

手順概要

1. **config t**
2. ***interface*** *type slot/port*
3. ***ip policy route-map*** *map-name*

   または
4. ***ipv6 policy route-map*** *map-nam*
5. ***exit***
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 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	***ip policy route-map*** *map-name* **例：** `switch(config-if)# ip policy route-map Testmap`	IPv4 ポリシーベース ルーティング用のルート マップをインターフェイスに割り当てます。
	ipv6 policy route-map *map-name* **例：** `switch(config-if)# ipv6 policy route-map TestIPv6map`	IPv6 ポリシーベース ルーティング用のルート マップをインターフェイスに割り当てます。
ステップ 4	**exit** **例：** `switch(config-route-map)# exit`	（任意）ルート マップ コンフィギュレーション モードを終了します。
ステップ 5	**exit** **例：** `switch(config)# exit`	（任意）ルート マップ コンフィギュレーション モードを終了します。
ステップ 6	***copy running-config startup-config*** **例：** `switch# copy running-config startup-config`	（任意）この設定変更を保存します。

インターフェイスにルート マップを追加する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ip policy route-map Testmap
switch(config)# exit
switch(config)# copy running-config startup-config
```

ルート マップ コンフィギュレーション モードで、オプションとして、ルート マップに次の match パラメータを設定できます。

コマンド	目的
match ip address access-list-name *name* [*name...*] **例：** `switch(config-route-map)# match ip address access-list-name ACL1`	1 つまたは複数の IP ACL（アクセス コントロール リスト）に対して IPv4 アドレスを照合します。このコマンドはポリシーベース ルーティング用であり、ルート フィルタリングまたは再配布では無視されます。
match ipv6 address access-list-name *name* [*name...*] **例：** `switch(config-route-map)# match ipv6 address access-list-name ACLv6`	1 つまたは複数の IPv6 ACL に対して IPv6 アドレスを照合します。このコマンドはポリシーベース ルーティング用であり、ルート フィルタリングまたは再配布では無視されます。
match length *min max* **例：** `switch(config-route-map)# match length 64 1500`	パケット長と照合します。このコマンドはポリシーベース ルーティング用です。

ルート マップ コンフィギュレーション モードで、オプションとして、ルート マップに次の set パラメータを設定できます。

コマンド	目的
set interface *ifname-1* [*ifname-2...*] **例：** `switch(config-route-map)# set interface ethernet 1/2`	ポリシーベース ルーティング対応として出力パケット インターフェイスを設定します。*ifname-1* が動作可能ではない場合、*ifname-2* など、設定されているオプション インターフェイスの 1 つが使用されます。
set default interface *ifname-1* [*ifname-2...*] **例：** `switch(config-route-map)# set default interface ethernet 2/2`	宛先への明示的ルートがない場合に使用する、ポリシーベース ルーティング用の出力パケット インターフェイスを設定します。*ifname-1* が動作可能ではない場合、*ifname-2* など、設定されているオプション インターフェイスの 1 つが使用されます。
set ip next-hop *address1* [*address2...*] **例：** `switch(config-route-map)# set ip next-hop 209.0.2.1`	ポリシーベース ルーティング用の IPv4 ネクストホップ アドレスを設定します。このコマンドでは、複数のアドレスが設定されている場合に、最初の有効なネクストホップ アドレスが使用されます。
set ip default next-hop *address1* [*address2...*] **例：** `switch(config-route-map)# set ip default next-hop 209.0.2.2`	宛先への明示的ルートがない場合に使用する、ポリシーベース ルーティング用の IPv4 ネクストホップ アドレスを設定します。このコマンドでは、複数のアドレスが設定されている場合に、最初の有効なネクストホップ アドレスが使用されます。
set ipv6 next-hop *address1* [*address2...*] **例：** `switch(config-route-map)# set ipv6 next-hop 2001:0DB8::1`	ポリシーベース ルーティング用の IPv6 ネクストホップ アドレスを設定します。このコマンドでは、複数のアドレスが設定されている場合に、最初の有効なネクストホップ アドレスが使用されます。

コマンド	目的
set ipv6 default next-hop *address1* [*address2...*] **例：** `switch(config-route-map)# set ipv6 default next-hop 2001:0DB8::2`	宛先への明示的ルートがない場合に使用する、ポリシーベース ルーティング用の IPv6 ネクストホップ アドレスを設定します。このコマンドでは、複数のアドレスが設定されている場合に、最初の有効なネクストホップ アドレスが使用されます。
set vrf *vrf-name* **例：** `switch(config-route-map)# set vrf MainVRF`	ネクストホップ解決用の Virtual Routing and Forwarding（VRF）を設定します。

Cisco NX-OS はネクストホップおよびインターフェイスを検出すると、ただちにパケットをルーティングします。

ポリシーベース ルーティングの設定確認

ポリシーベース ルーティングを確認するには、***show ip policy*** コマンドを使用します。

ポリシー統計をイネーブルにするには、***route-map*** *map-name* ***pbr-statistics*** を使用します。ポリシー統計を表示するには、***show route-map*** *map-name* ***pbr-statistics*** を使用します。ポリシー統計を消去するには、***clear route-map*** *map-name* ***pbr-statistics*** を使用します。

ポリシーベース ルーティングの設定例

インターフェイス上で単純なルート ポリシーを設定する例を示します。

```
feature pbr
access-list 1 permit ip 209.0.2.1
!
 interface ethernet 1/2
  ip policy route-map equal-access
 route-map equal-access permit 10
  match ip address 1
   set ip default next-hop 209.0.2.10
```

関連項目

ポリシーベース ルーティングの詳細については、次の項目を参照してください。

- 第 14 章「Route Policy Manager の設定」

デフォルト設定

表 15-1 に、ポリシーベース ルーティング パラメータのデフォルト設定を示します。

表 15-1　デフォルトのポリシーベース ルーティング パラメータ

パラメータ	デフォルト
ポリシーベース ルーティング	ディセーブル

その他の関連資料

IP 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。

- 関連資料（p.15-10）
- 規格（p.15-10）

関連資料

関連項目	マニュアル名
ポリシーベース ルーティング CLI コマンド	『*Cisco NX-OS Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

PART 3

FHRP

CHAPTER 16

GLBP の設定

この章では、Gateway Load Balancing Protocol（GLBP）の設定方法について説明します。

ここでは、次の内容を説明します。

- GLBP の概要（p.16-2）
- GLBP のライセンス要件（p.16-7）
- GLBP の前提条件（p.16-7）
- 注意事項および制約事項（p.16-7）
- GLBP の設定（p.16-8）
- GLBP の設定確認（p.16-16）
- GLBP の設定例（p.16-16）
- デフォルト設定（p.16-17）
- その他の関連資料（p.16-18）

GLBP の概要

GLBP は、冗長ゲートウェイ間でプロトコルおよび MAC（メディア アクセス制御）アドレスを共有することによって、IP にパスの冗長性をもたらします。GLBP はさらに、レイヤ 3 ルータ グループ間で、LAN 上のデフォルト ゲートウェイの負荷が分担されるようにします。GLBP ルータは他のルータで障害が発生した場合、グループの別のルータのフォワーディング機能を自動的に引き受けます。

ここでは、次の内容について説明します。

- GLBP の概要（p.16-2）
- GLBP アクティブ バーチャル ゲートウェイ（p.16-2）
- GLBP バーチャル MAC アドレス割り当て（p.16-3）
- GLBP によるバーチャル ゲートウェイの冗長性（p.16-3）
- GLBP によるバーチャル フォワーダの冗長性（p.16-3）
- GLBP 認証（p.16-4）
- GLBP ロード バランシングおよびトラッキング（p.16-5）
- ハイ アベイラビリティ（p.16-6）
- 仮想化サポート（p.16-6）

GLBP の概要

GLBP は、IEEE 802.3 LAN 上でデフォルト ゲートウェイを 1 つだけ指定して設定された IP ホストの自動*ゲートウェイ* バックアップを行います。LAN 上の複数のルータが結びついて、1 つのバーチャル ファーストホップ IP ゲートウェイを提供し、なおかつ IP パケット転送の負荷を分担します。LAN 上の他のルータは、冗長 GLBP ゲートウェイとして動作可能であり、既存のフォワーディング ゲートウェイのいずれかで障害が発生した場合にアクティブになります。

GLBP は、HSRP（ホットスタンバイ冗長プロトコル）および VRRP（仮想ルータ冗長プロトコル）と同様の機能を実行します。HSRP および VRRP は、バーチャル IP アドレスを指定して設定されたバーチャル グループに、複数のルータを参加させます。これらのプロトコルでは、グループのバーチャル IP アドレスにパケットを転送するアクティブ ルータとして、メンバを 1 つ選択します。グループ内の他のルータは、アクティブ ルータで障害が発生するまで冗長構成になります。

GLBP は、他のプロトコルにはないロード バランシング機能を実行します。GLBP は、1 つのバーチャル IP アドレスと複数のバーチャル MAC アドレスを使用し、複数のルータ（ゲートウェイ）間でロード バランスを図ります。GLBP の場合は、1 つのルータに負荷全体を引き受けさせ、他のルータを遊ばせておくのではなく、GLBP グループのすべてのルータ間で転送負荷を分担します。各ホストに同じバーチャル IP アドレスを設定し、バーチャル グループ内のすべてのルータがパケット転送に関与するようにします。GLBP メンバは定期的な hello パケットによって、相互に通信します。

GLBP アクティブ バーチャル ゲートウェイ

GLBP はゲートウェイにプライオリティを設定して、アクティブ バーチャル ゲートウェイ（*AVG*）を選択します。複数のゲートウェイに同じプライオリティを与えた場合は、実 IP アドレスが最も大きいゲートウェイが AVG になります。AVG は GLBP グループの各メンバにバーチャル MAC アドレスを割り当てます。各メンバはそれぞれ割り当てられたバーチャル MAC アドレスに対応するアクティブ バーチャル フォワーダ（*AVF*）となり、割り当てられたバーチャル MAC アドレスにパケットを転送します。

AVG は、バーチャル IP アドレスに対する ARP（アドレス解決プロトコル）要求にも応答します。ロード シェアリングは、AVG が ARP 要求に異なるバーチャル MAC アドレスで応答したときに行われます。

GLBP バーチャル MAC アドレス割り当て

AVG はグループの各メンバにバーチャル MAC アドレスを割り当てます。グループ メンバは hello メッセージを通じて AVG を検出したあとで、バーチャル MAC アドレスを要求します。AVG は選択されたロード バランシング アルゴリズムに基づいて、ネクスト MAC アドレスを割り当てます（「GLBP ロード バランシングおよびトラッキング」[p.16-5] を参照）。AVG によってバーチャル MAC アドレスが割り当てられたゲートウェイは、プライマリ バーチャル フォワーダになります。hello メッセージから バーチャル MAC アドレスを学習する、GLBP グループの他のメンバは、セカンダリ バーチャル フォワーダです。

GLBP によるバーチャル ゲートウェイの冗長性

GLBP は、バーチャル ゲートウェイの冗長性を実現します。グループ メンバは、アクティブ、スタンバイ、またはリッスン ステートになります。GLBP はプライオリティ アルゴリズムを使用し、1 つのゲートウェイを AVG として選択し、もう 1 つのゲートウェイをスタンバイ バーチャル ゲートウェイとして選択します。残りのゲートウェイはリッスン ステートになります。各ゲートウェイ上で GLBP プライオリティを設定できます。GLBP プライオリティが複数のゲートウェイで同じ場合、GLBP は IP アドレスが最大のゲートウェイを AVG として使用します。

AVG で障害が発生すると、スタンバイ バーチャル ゲートウェイがバーチャル IP アドレスに対応する役割を引き受けます。GLBP はリッスン ステートのゲートウェイから新しいスタンバイ バーチャル ゲートウェイを選択します。

GLBP によるバーチャル フォワーダの冗長性

GLBP は、バーチャル フォワーダの冗長性を実現します。バーチャル フォワーダの冗長性は、アクティブ バーチャル フォワーダ（AVF）の点で、バーチャル ゲートウェイの冗長性と類似しています。AVF で障害が発生すると、リッスン ステートのセカンダリ バーチャル フォワーダがバーチャル MAC アドレスに対応する役割を引き受けます。このセカンダリ バーチャル フォワーダは、別のバーチャル MAC アドレスのプライマリ バーチャル フォワーダでもあります。GLBP は次の 2 種類のタイマーを使用して、障害 AVF の古いバーチャル MAC アドレスからホストを移行させます。

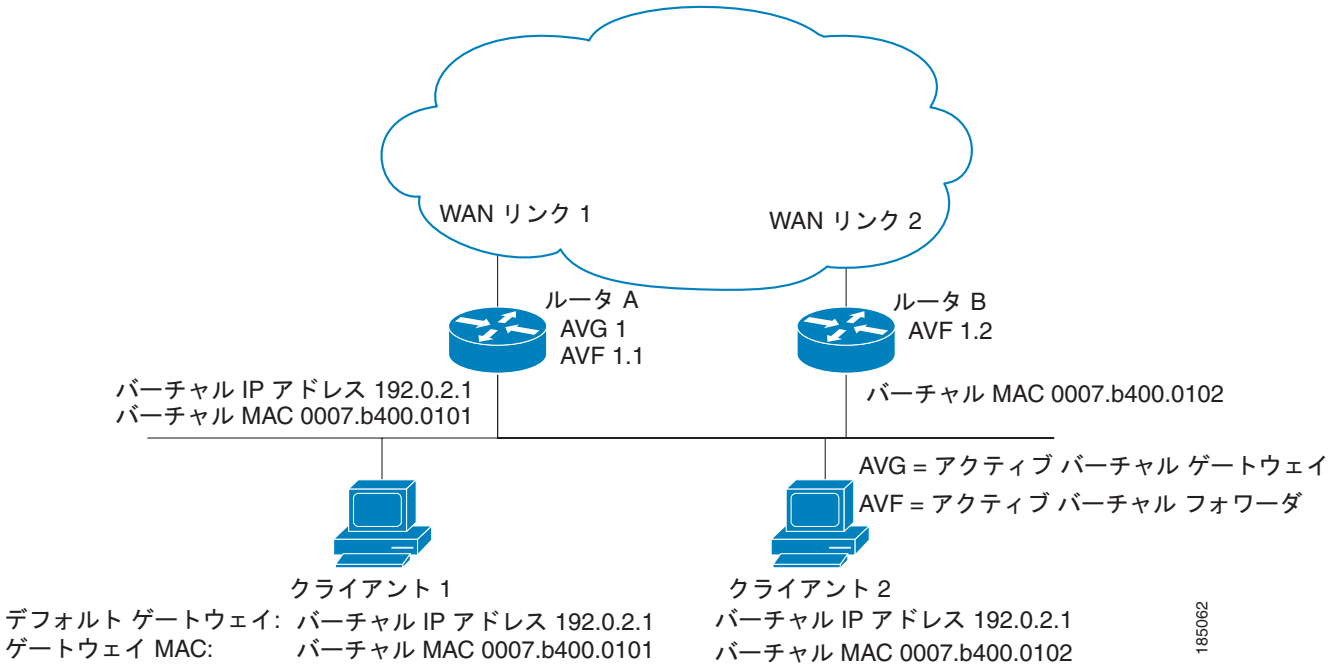
- リダイレクト タイマー ― AVG が古いバーチャル MAC アドレスにホストをリダイレクトし続ける時間の長さを指定します。リダイレクト タイムが経過すると、AVG は ARP 応答での古いバーチャル MAC アドレスの使用を中止しますが、セカンダリ バーチャル フォワーダは引き続き、古いバーチャル MAC アドレスに送信されたパケットを転送します。
- セカンダリ ホールド タイマー ― バーチャル MAC アドレスが有効な時間の長さを指定します。セカンダリ ホールド タイムが経過すると、GLBP が GLBP グループのすべてのゲートウェイからバーチャル MAC アドレスを削除し、残りの AVF 間でトラフィックのロード バランスが図られます。時間切れになったバーチャル MAC アドレスは、AVG による再割り当ての対象になります。

GLBP は hello メッセージを使用して、タイマーの現在のステートを伝えます。

図 16-1 では、ルータ A は GLBP グループの AVG であり、バーチャル IP アドレス 192.0.2.1 を担当します。ルータ A は、バーチャル MAC アドレス 0007.b400.0101 に対応する AVF でもあります。ルータ B は、同じ GLBP グループのメンバであり、バーチャル MAC アドレス 0007.b400.0102 の AVF として指定されています。クライアント 1 にはデフォルト ゲートウェイ IP アドレス 192.0.2.1、バーチャル IP アドレス、およびゲートウェイ MAC アドレス 0007.b400.0101（ルータ A を指す）が

設定されています。クライアント 2 は、同じデフォルト ゲートウェイ IP アドレスを共有しますが、ルータ B がルータ A とトラフィック負荷を分担するので、与えられているゲートウェイ MAC アドレスは 0007.b400.0102 です。

図 16-1　GLBT トポロジ



ルータ A が使用不能になっても、ルータ B がルータ A のバーチャル MAC アドレス宛てのパケットの転送を引き受け、自分のバーチャル MAC アドレス宛てのパケットに応答するので、クライアント 1 が WAN にアクセスできなくなることはありません。ルータ B は、GLBP グループ全体の AVG の役割も引き受けます。GLBP グループ内のルータで障害が発生しても、GLBP メンバの通信は継続されます。

GLBP 認証

GLBP の認証タイプは、次の 3 種類です。

- MD5 認証
- プレーンテキスト認証
- 認証なし

MD5 認証を使用すると、プレーンテキスト認証より強力なセキュリティが得られます。MD5 認証の場合、各 GLBP グループ メンバが秘密鍵を使用して、発信パケットに組み込まれる鍵付き MD5 ハッシュを生成します。受信側では、着信パケットの鍵付きハッシュが生成されます。着信パケット内のハッシュが生成されたハッシュと一致しなかった場合、そのパケットは無視されます。MD5 ハッシュの鍵は、鍵ストリングを使用してコンフィギュレーションに直接指定することも、キーチェーンによって間接的に供給することもできます。

プレーンテキストの単純なパスワードを使用して GLBP を認証する、または GLBP に関して認証を行わないという選択も可能です。

GLBP は次の場合に、パケットを拒否します。

- 認証方式がルータと着信パケット間で異なっている。
- MD5 ダイジェストがルータと着信パケット間で異なっている。
- テキスト認証ストリングがルータと着信パケット間で異なっている。

GLBP ロード バランシングおよびトラッキング

GLBP で設定できるロード バランシング方式は、次のとおりです。

- ラウンドロビン — GLBP は ARP 応答で送信されたバーチャル MAC アドレスを循環させ、すべての AVF 間でトラフィックのロード バランシングを図ります。
- 重み付き — AVG はアドバタイズされた AVF の重み値を使用して、AVF に与える負荷を決定します。重み値が大きいほど、AVG が AVF に与えるトラフィックが多くなります。
- ホスト依存 — GLBP はホストの MAC アドレスを使用して、使用するホストに指示するバーチャル MAC アドレスを決定します。このアルゴリズムでは、バーチャル フォワーダの数が変わらないかぎり、ホストに同じバーチャル MAC アドレスが与えられることが保証されます。

IPv4 ネットワークのデフォルトは、ラウンドロビンです。インターフェイスで、GLBP に関するすべてのロード バランシングをディセーブルにできます。ロード バランシングを設定しなかった場合、AVG がホストへのすべてのトラフィックを引き受け、他の GLBP グループ メンバーはスタンバイまたはリッスン モードになります。

インターフェイスまたはルートを追跡し、追跡対象のリンクがダウンした場合に、セカンダリ バーチャル フォワーダが引き継ぐように GLBP を設定できます。GLBP トラッキングでは、重み付きロード バランシングを使用して、GLBP グループ メンバが AVF として動作するかどうかを判別します。AVF としてのそのグループ メンバを使用できるか、または使用できないかを決定するには、初期重み値およびオプションのしきい値を設定する必要があります。追跡するインターフェイスも設定できます。また、インターフェイスがダウンしたときに、インターフェイスの重みがどれだけ減るか、その値も設定できます。GLBP グループの重みが下限しきい値を下回ると、メンバは AVF ではなくなり、セカンダリ バーチャル フォワーダが引き継ぎます。重みが上限しきい値を上回ると、メンバは AVF としての役割を再び得ます。

図 16-2 に、GLBP トランキングおよび重み付けの例を示します。

図 16-2　GLBP オブジェクト トラッキングおよび重み付け

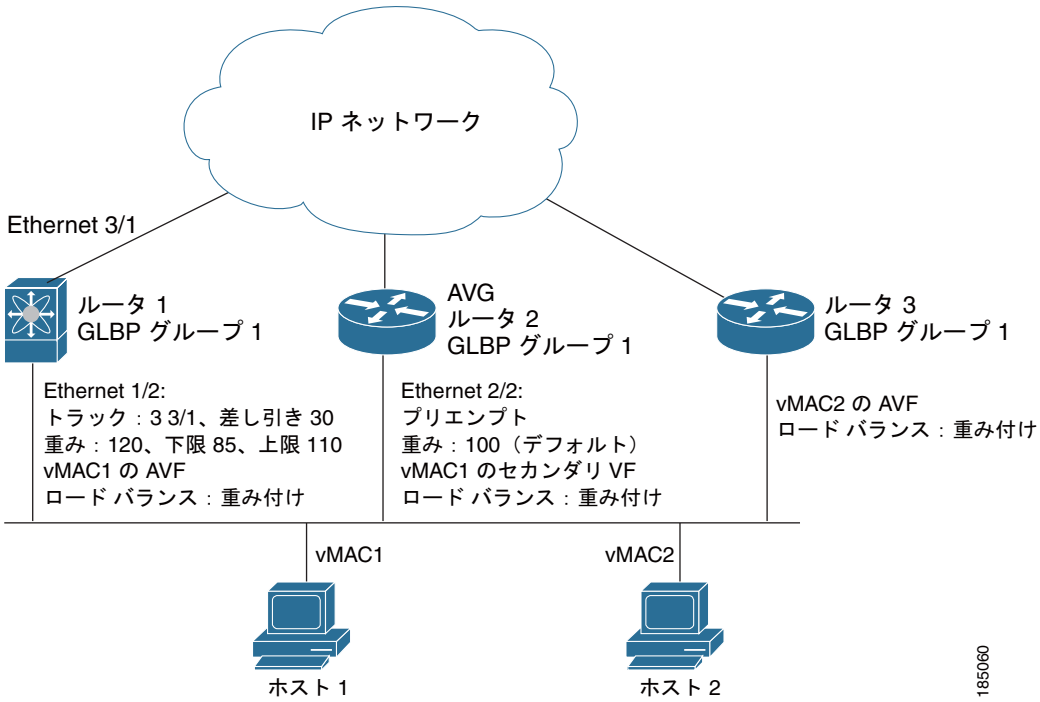


図 16-2 では、ルータ 1 上のインターフェイス Ethernet 1/2 がホスト 1 のゲートウェイ（バーチャル MAC アドレス vMAC に対応する AVF）です。一方、ルータ 2 上の Ethernet 2/2 は、ホスト 1 のセカンダリ バーチャル フォワーダとして動作します。Ethernet 1/2 は、ルータ 1 のネットワーク接続である Ethernet 3/1 を追跡します。Ethernet 3/1 がダウンすると、Ethernet 1/2 の重み値が 90 に下がります。ルータ 2 上の Ethernet 2/2 が Ethernet 1/2 に代わり、AVF として引き継ぎます。Ethernet 2/2 はデフォルトの重み値が 100 であり、AVF に関する優先権が設定されているからです。

重み付けおよびトラッキングの詳細については、「GLBP 重み付けおよびトラッキングの設定」（p.16-11）を参照してください。

ハイ アベイラビリティ

GLBP は、ステートフル リスタートおよびステートフル スイッチオーバーをサポートします。ステートフル リスタートは、GLBP が障害を処理してリスタートするときに行われます。ステートフル スイッチオーバーは、アクティブ スーパーバイザがスタンバイ スーパーバイザに切り替わるときに行われます。Cisco NX-OS は、スイッチオーバー後に実行コンフィギュレーションを適用します。

仮想化サポート

GLBP は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。

インターフェイスの VRF メンバシップを変更すると、Cisco NX-OS によって GLBP を含め、すべてのレイヤ 3 設定が削除されます。

詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

GLBP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	GLBP にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

GLBP の前提条件

GLBP の前提条件は、次のとおりです。

- GLBP 機能をグローバルでイネーブルにします（「GLBP 機能のイネーブル化」[p.16-8] を参照）。
- GLBP を設定できるのは、レイヤ 3 インターフェイス上に限られます（『*Cisco NX-OS Interface Configuration Guide*』を参照）。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

注意事項および制約事項

GLBP に関する注意事項および制約事項は、次のとおりです。

- バーチャル IP アドレスを設定することによって GLBP グループをイネーブルにするには、その前にすべての GLBP メンバ ゲートウェイ上で、GLBP に関するすべてのカスタマイズ オプションを設定する必要があります。
- GLBP がサポートする最小 hello タイムは 250 ms、最小ホールド タイムは 1020 ms です。
- GLBP を設定するインターフェイスに IP アドレスを設定し、そのインターフェイスをイネーブルにしてからでなければ、GLBP はアクティブになりません。
- GLBP バーチャル IP アドレスは、インターフェイス IP アドレスと同じサブネットになければなりません。
- 同一インターフェイス上では、複数のファーストホップ冗長プロトコルを設定しないことを推奨します。
- VDC、インターフェイス VRF メンバシップ、ポート チャネル メンバシップを変更した場合、またはポート モードをレイヤ 2 に変更した場合は、Cisco NX-OS によってインターフェイス上のすべてのレイヤ 3 設定が削除されます。

GLBP の設定

ここでは、次の内容について説明します。

- GLBP 機能のイネーブル化（p.16-8）
- GLBP 認証の設定（p.16-8）
- GLBP ロード バランシングの設定（p.16-10）
- GLBP 重み付けおよびトラッキングの設定（p.16-11）
- GLBP のカスタマイズ（p.16-13）
- GLBP グループのイネーブル化（p.16-14）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

GLBP 機能のイネーブル化

GLBP グループを設定してイネーブルにする前に、GLBP 機能をイネーブルにする必要があります。

準備作業

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）。

手順詳細

GLBP 機能をイネーブルにするには、グローバル 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。

コマンド	目的
feature glbp **例：** `switch(config)# feature glbp`	GLBP をイネーブルにします。

VDC で GLBP 機能をディセーブルにするには、グローバル 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。

コマンド	目的
no feature glbp **例：** `switch(config)# no feature glbp`	VDC で GLBP 機能をディセーブルにします。

GLBP 認証の設定

クリアテキストまたは MD5 ダイジェストを使用してプロトコルを認証するように、GLBP を設定できます。MD5 認証ではキーチェーンを使用します（『*Cisco NX-OS Security Configuration Guide*』を参照）。

準備作業

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）。

GLBP 機能をイネーブルにします（「GLBP 機能のイネーブル化」[p.16-8] を参照）。

（注） GLBP グループのすべてのメンバに同じ認証および鍵を設定する必要があります。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **ip** *ip-address/length*
4. **glbp** *group-number*
5. **authentication text** *string*

   または

   authentication md5 {**key-chain** *key-chain* | ***key-string*** {*text* | ***encrypted*** *text*}
6. **ip** [*ip-address* [***secondary***]]
7. **show glbp** [***group*** *group-number*]
8. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip** *ip-address*/length **例：** `switch(config-if)# ip 192.0.2.1/8`	インターフェイスの IPv4 アドレスを設定します。

ステップ 4	**glbp** *group-number*
	
	例：
	`switch(config-if)# glbp 1`
	`switch(config-if-glbp)#`
		GLBP グループを作成し、GLBP コンフィギュレーション モードを開始します。有効値の範囲は 0 ～ 1024 です。

	コマンド	目的
ステップ 5	**authentication text** *string* **例：** `switch(config-if-glbp)# authentication text mypassword`	このインターフェイス上で、GLBP のクリアテキスト認証を設定します。
	authentication md5 {***key-chain*** *key-chain* | ***key-string*** {*text* | ***encrypted*** *text*} **例：** `switch(config-if-glbp)# authentication md5 key-chain glbp-keys`	このインターフェイス上で、GLBP の MD5 認証を設定します。
ステップ 6	**ip** [*ip-address* [**secondary**]] **例：** `switch(config-if-glbp)# ip 192.0.2.10`	インターフェイス上で GLBP を設定し、バーチャル ゲートウェイのプライマリ IP アドレスを指定します。 プライマリ IP アドレスの指定後は、***secondary*** キーワードを指定して **glbp** *group* **ip** コマンドを再び使用し、このグループでサポートする他の IP アドレスを指定できます。**ip** キーワードだけを指定した場合、GLBP はネイバーからバーチャル IP アドレスを学習します。
ステップ 7	***show glbp*** [***group*** *group-number*] **例：** `switch(config-if-glbp)# show glbp 1`	（任意）GLBP 情報を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-if-glbp)# copy running-config startup-config`	（任意）この設定変更を保存します。

キーチェーンの作成後に、Ethernet 1/2 で GLBP の MD5 認証を設定する例を示します。

```
switch# config t
switch(config)# key chain glbp-keys
switch(config-keychain)# key 0
switch(config-keychain-key)# key-string 7 zqdest
switch(config-keychain-key) accept-lifetime 00:00:00 Jun 01 2008 23:59:59 Sep 12 2008
switch(config-keychain-key) send-lifetime 00:00:00 Jun 01 2008 23:59:59 Aug 12 2008
switch(config-keychain-key) key 1
switch(config-keychain-key) key-string 7 uaeqdyito
switch(config-keychain-key) accept-lifetime 00:00:00 Aug 12 2008 23:59:59 Dec 12 2008
switch(config-keychain-key) send-lifetime 00:00:00 Sep 12 2008 23:59:59 Nov 12 2008
switch(config)# interface ethernet 1/2
switch(config-if)# glbp 1
switch(config-if-glbp)# authenticate md5 key-chain glbp-keys
switch(config-if-glbp)# copy running-config startup-config
```

GLBP ロード バランシングの設定

ラウンドロビン、重み付き、またはホスト依存方式に基づいて、ロード バランシングを使用するように GLBP を設定できます（「GLBP ロード バランシングおよびトラッキング」[p.16-5] を参照）。

GLBP ロード バランシングを設定するには、GLBP 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。

コマンド	目的
load-balancing [host-dependent | round-robin | weighted] **例：** `switch(config-if-glbp)# load-balancing weighted`	GLBP ロード バランシングの方式を設定します。デフォルトはラウンドロビンです。

GLBP 重み付けおよびトラッキングの設定

GLBP 重み値および GLBP 重み付きロード バランシング方式と連動するオブジェクト トラッキングを設定できます。

インターフェイスが最初にバーチャル MAC アドレスを指定して割り当てられている場合、またはインターフェイスの重み値が AVF より大きい場合に、そのインターフェイスによる AVF のプリエンプトを任意で設定できます。

準備作業

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）。

GLBP 機能をイネーブルにします（「GLBP 機能のイネーブル化」[p.16-8] を参照）。

手順概要

1. **config t**
2. **track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
3. **track** *object-id* **ip route** *ip-prefix/length* **reachability**
4. **interface** *interface-type slot/port*
5. **ip** *ip-address/length*
6. **glbp** *group-number*
7. **weighting** *maximum* [***lower*** *lower*] [***upper*** *upper*]
8. **weighting track** *object-number* [***decrement*** *value*]
9. **forwarder preempt** [***delay minimum*** *seconds*]
10. **ip** [*ip-address* [***secondary***]]
11. **show glbp** *interface-type number*
12. ***copy running-config startup-config***

手順詳細

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
	
	例：
	`switch(config)# track 1 interface ethernet 2/2 line-protocol`
	`switch(config-track#`
		この GLBP インターフェイスが追跡するインターフェイスを設定します。インターフェイスのステート変化は次のように、この GLBP のプライオリティを左右します。
		• GLBP コンフィギュレーション モードで、**track** コマンドで使用するインターフェイスおよび対応するオブジェクト番号を設定します。
		• **line-protocol** キーワードを指定すると、インターフェイスがアップかどうかが追跡されます。**ip** キーワードを指定すると、インターフェイス上で IP ルーティングがイネーブルであり、IP アドレスが設定されているかどうかもチェックされます。

	track *object-id* **ip route** *ip-prefix/length* **reachability**
	
	例：
	`switch(config)# track 2 ip route 192.0.2.0/8 reachability`
	`switch(config-track#`
		ルートの追跡対象オブジェクトを作成し、トラッキング コンフィギュレーション モードを開始します。object-id の範囲は 1 ～ 500 です。

ステップ 3	**interface** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**ip** *ip-address*/length **例：** `switch(config-if)# ip 192.0.2.1/8`	インターフェイスの IPv4 アドレスを設定します。

ステップ 5	**glbp** *group-number*
	
	例：
	`switch(config-if)# glbp 1`
	`switch(config-if-glbp)#`
		GLBP グループを作成し、GLBP 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	**weighting** *maximum* [**lower** *lower*] [**upper** *upper*] **例：** `switch(config-if-glbp)# weighting 110 lower 95 upper 105`	GLBP ゲートウェイの初期重み値、上限しきい値、および下限しきい値を指定します。最大範囲は 1 ～ 254 です。デフォルトの重み値は 100 です。下限の範囲は 1 ～ 253 です。上限の範囲は 1 ～ 254 です。
ステップ 7	**weighting track** *object-number* [**decrement** *value*] **例：** `switch(config-if-glbp)# weighting track 2 decrement 20`	GLBP ゲートウェイの重み付けを左右する、追跡対象のオブジェクトを指定します。*value* 引数には、追跡対象のオブジェクトで障害が発生した場合に、GLBP ゲートウェイの重み値から差し引く値を指定します。有効値の範囲は 1 ～ 255 です。

	コマンドまたはアクション	目的
ステップ 8	**forwarder preempt** [**delay minimum** *seconds*] **例：** `switch(config-if-glbp)# forwarder preempt delay minimum 60`	（任意）GLBP グループの現在の AVF が重みの下限しきい値を下回った場合に、GLBP グループの AVF を引き継ぐようにルータを設定します。有効値の範囲は 0 ～ 3600 秒です。 このコマンドはデフォルトでイネーブルであり、遅延は 30 秒です。
ステップ 9	**ip** [*ip-address* [**secondary**]] **例：** `switch(config-if-glbp)# ip 192.0.2.10`	インターフェイス上で GLBP を設定し、バーチャル ゲートウェイのプライマリ IP アドレスを指定します。 プライマリ IP アドレスの指定後は、***secondary*** キーワードを指定して **glbp** *group* **ip** コマンドを再び使用し、このグループでサポートする他の IP アドレスを指定できます。**ip** キーワードだけを指定した場合、GLBP はネイバーからバーチャル IP アドレスを学習します。
ステップ 10	**show glbp** *interface-type number* **例：** `switch(config-if-glbp)# show glbp ethernet 1/2`	（任意）インターフェイスの GLBP 情報を表示します。
ステップ 11	***copy running-config startup-config*** **例：** `switch(config-if-glbp)# copy running-config startup-config`	（任意）この設定変更を保存します。

Ethernet 1/2 上で GLBP の重み付けおよびトラッキングを設定する例を示します。

```
switch# config t
switch(config)# track 2 interface ethernet 2/2 ip routing
switch(config)# interface ethernet 1/2
switch(config-if)# glbp 1
switch(config-if-glbp)# weighting 110 lower 95 upper 105
switch(config-if-glbp)# weighting track 2 decrement 20
switch(config-if-glbp)# copy running-config startup-config
```

GLBP のカスタマイズ

GLBP 動作のカスタマイズは任意です。バーチャル IP アドレスを設定することによって、GLBP グループをイネーブルにすると、そのグループがただちに動作可能になることに注意してください。GLBP をカスタマイズする前に GLBP グループをイネーブルにした場合、機能のカスタマイズが完了しないうちに、ルータがグループの制御を引き継いで AVG になる可能性があります。GLBP のカスタマイズを予定している場合は、GLBP をイネーブルにする前に行ってください。

GLBP をカスタマイズするには、GLBP 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。

コマンドまたはアクション	目的
timers [**msec**] *hellotime* [**msec**] *holdtime* **例：** `switch(config-if-glbp)# timers 5 18`	この GLBP メンバに次の hello タイムおよびホールド タイムを設定します。 • *hellotime* ― GLBP グループの AVG が hello パケットを送信してから、次の hello パケットを送信するまでのインターバル。範囲は 1 ～ 60 秒または 250 ～ 60000 ミリ秒です。デフォルト値は 3 秒です。 • *holdtime* ― hello パケットのバーチャル ゲートウェイおよびバーチャル フォワーダ情報が無効とみなされるまでのインターバル。範囲は 2 ～ 180 秒または 1020 ～ 180000 ミリ秒です。デフォルト値は 10 秒です。 オプションの ***msec*** キーワードでは、引数をデフォルトの秒単位ではなく、ミリ秒単位で表すことを指定します。
timers redirect *redirect timeout* **例：** `switch(config-if-glbp)# timers redirect 600 7200`	次のタイマーを設定します。 • *redirect* ― AVG が AVF にクライアントのリダイレクトを続ける時間の長さ（秒数）。有効値の範囲は 0 ～ 3600 秒です。デフォルト値は 600 秒です。 • *timeout* ― セカンダリ バーチャル フォワーダが無効になるまでの時間の長さ（秒数）。有効値の範囲は 610 ～ 64800 秒です。デフォルト値は 14,440 秒です。
priority *level* **例：** `switch(config-if-glbp)# priority 254`	GLBP グループでの AVG 選択に使用するプライオリティ レベルを設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 100 です。
preempt [**delay minimum** *seconds*] **例：** `switch(config-if-glbp)# preempt delay minimum 60`	現在の AVG よりプライオリティが高い場合に、GLBP グループの AVG として引き継ぐようにルータを設定します。このコマンドはデフォルトでディセーブルです。 AVG の交替が行われるまでの最小遅延インターバルを秒数で指定するには、オプションの **delay minimum** キーワードおよび *seconds* 引数を指定します。有効値の範囲は 0 ～ 3600 秒です。最小遅延のデフォルト値は 30 秒です。

GLBP グループのイネーブル化

GLBP グループをイネーブルにするインターフェイス上で、バーチャル IP アドレスを設定できます。同じグループ番号を指定して、GLBP グループの各ゲートウェイを設定する必要があります。GLBP メンバは別の GLBP メンバから必要な他のあらゆるパラメータを学習できます。

準備作業

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）。

GLBP 機能をイネーブルにします（「GLBP 機能のイネーブル化」[p.16-8] を参照）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **ip** *ip-address/length*

4. **glbp** *group-number*
5. **ip** [*ip-address* [***secondary***]]
6. **show glbp** [***brief***]
7. ***copy running-config startup-config***

手順詳細

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip** *ip-address*/length **例：** `switch(config-if)# ip 192.0.2.1/8`	インターフェイスの IPv4 アドレスを設定します。

ステップ 4	**glbp** *group-number*
	
	例：
	`switch(config-if)# glbp 1`
	`switch(config-if-glbp)#`
		GLBP グループを作成し、GLBP 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	**ip** [*ip-address* [**secondary**]] **例：** `switch(config-if-glbp)# ip 192.0.2.10`	インターフェイス上で GLBP をイネーブルにして、バーチャル IP アドレスを指定します。バーチャル IP は、インターフェイス IP アドレスと同じサブネットになければなりません。 バーチャル IP アドレスの指定後は、***secondary*** キーワードを指定して **glbp** *group* **ip** コマンドを再び使用し、このグループでサポートする他の IP アドレスを指定できます。**ip** キーワードだけを指定した場合、GLBP はネイバーからバーチャル IP アドレスを学習します。
ステップ 6	**show glbp [group** *group-number***]** [***brief***] **例：** `switch(config-if-glbp)# show glbp brief`	（任意）GLBP 情報の要約を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-if-glbp)# copy running-config startup-config`	（任意）この設定変更を保存します。

Ethernet 1/2 上で GLBP をイネーブルにする例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# glbp 1
switch(config-if-glbp)# ip 192.0.2.10
```

GLBP の設定確認

GLBP の設定情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show glbp [**group** *group-number*]	すべてまたは特定のグループの GLBP ステータスを表示します。
show glbp [**group** *group-number*] **capability**	すべてまたは特定のグループの GLBP 機能を表示します。
show glbp [***interface*** *interface-type slot/port*]	インターフェイスの GLBP ステータスを表示します。
show glbp [**group** *group-number*] [***interface*** *interface-type slot/port*] [***active***] [**disabled**] [***init***] [***listen***] [***standby***]	選択された状態のバーチャル フォワーダに対応するグループまたはインターフェイスについて、GLBP ステータスを表示します。
show glbp [**group** *group-number*] [***interface*** *interface-type slot/port*] [***active***] [**disabled**] [***init***] [***listen***] [***standby***] ***brief***	選択された状態のバーチャル フォワーダに対応するグループまたはインターフェイスについて、GLBP ステータスの要約を表示します。

GLBP の設定例

次の例では、インターフェイス上で GLBP をイネーブルにして、MD5 認証、インターフェイス トラッキングおよび重み付きロード バランスを指定します。

```
key chain glbp-keys
 key 0
   key-string 7 zqdest
  accept-lifetime 00:00:00 Jun 01 2008 23:59:59 Sep 12 2008
   send-lifetime 00:00:00 Jun 01 2008 23:59:59 Aug 12 2008
  key 1
   key-string 7 uaeqdyito
   accept-lifetime 00:00:00 Aug 12 2008 23:59:59 Dec 12 2008
   send-lifetime 00:00:00 Sep 12 2008 23:59:59 Nov 12 2008
feature glbp
track 2 interface ethernet 2/2 ip
interface ethernet 1/2
 ip address 192.0.2.2/8
 glbp 1
  authentication md5 key-chain glbp-keys
  weighting 110 lower 95 upper 105
  weighting track 2 decrement 20
  ip 192.0.2.10
 no shutdown
```

デフォルト設定

表 16-1 に、GLBP パラメータのデフォルト設定を示します。

表 16-1　デフォルトの GLBP パラメータ

パラメータ	デフォルト
認証	認証なし
フォワーダ プリエンプト遅延	30 秒
フォワーダ タイムアウト	14400 秒
hello タイマー	3 秒
ホールド タイマー	10 秒
GLBP 機能	ディセーブル
ロード バランス	ラウンドロビン
プリエンプト	ディセーブル
プライオリティ	100
リダイレクト タイマー	600 秒
重み付け	100

その他の関連資料

GLBP 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。

- 関連資料（p.16-18）
- 規格（p.16-18）

関連資料

関連項目	マニュアル名
HSRP の設定	第 17 章「HSRP の設定」
VRRP の設定	第 18 章「VRRP の設定」
GLBP CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』
ハイ アベイラビリティの設定	『*Cisco NX-OS HA Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 17

HSRP の設定

この章では、*HSRP*（ホットスタンバイ ルータ プロトコル）の設定方法について説明します。

ここでは、次の内容を説明します。

- HSRP 情報（p.17-2）
- HSRP のライセンス要件（p.17-7）
- HSRPP の前提条件（p.17-7）
- 注意事項および制約事項（p.17-7）
- HSRP の設定（p.17-8）
- HSRP の設定確認（p.17-17）
- HSRP の設定例（p.17-17）
- デフォルト設定（p.17-18）
- その他の関連資料（p.17-18）

HSRP 情報

HSRP は、ファーストホップ IP ルータの透過的フェールオーバーが可能な、*FHRP*（ファーストホップ冗長プロトコル）です。HSRP は、デフォルト ゲートウェイの IP アドレスを指定して設定された、イーサネット ネットワーク上の IP ホストにファーストホップ ルーティングの冗長性を提供します。ルータ グループでは HSRP を使用して、アクティブ ゲートウェイおよびスタンバイ ゲートウェイを選択します。ゲートウェイ グループでは、アクティブ ゲートウェイはパケットをルーティングするゲートウェイです。スタンバイ ゲートウェイは、アクティブ ゲートウェイで障害が発生した場合、または事前に設定された条件が満たされた場合に、引き継ぐゲートウェイです。

実装されているホストの多くは、ダイナミック ルータ ディスカバリ メカニズムをサポートしませんが、デフォルト ルータを指定して設定できます。あらゆるホストでダイナミック ルータ ディスカバリ メカニズムを実行することは、管理上の負担、処理上の負担、セキュリティの問題など、さまざまな理由で現実的ではありません。HSRP はこのようなホストにフェールオーバー サービスを提供します。

ここでは、次の内容について説明します。

- HSRP の概要（p.17-2）
- HSRP のバージョン（p.17-3）
- HSRP 認証（p.17-4）
- HSRP アドレッシング（p.17-4）
- HSRP メッセージ（p.17-4）
- HSRP ロード シェアリング（p.17-5）
- オブジェクト トラッキングおよび HSRP（p.17-5）

HSRP の概要

HSRP を使用する場合は、ホストのデフォルト ゲートウェイとして（実際のゲートウェイの IP アドレスではなく）、HSRP *バーチャル IP アドレス*を設定します。バーチャル IP アドレスは、HSRP が動作するゲートウェイのグループで共有される IP アドレスです。

ネットワーク セグメントに HSRP を設定する場合は、HSRP グループ用の*バーチャル MAC アドレス*およびバーチャル IP アドレスを指定します。グループの各 HSRP 対応インターフェイス上で、同じバーチャル アドレスを指定します。各インターフェイス上で、実アドレスとして機能する固有の IP アドレスおよび MAC アドレスも設定します。HSRP はこれらのインターフェイスの 1 つを*アクティブ ルータ*として選択します。アクティブ ルータは、グループのバーチャル MAC アドレス宛てのパケットを受信してルーティングします。

指定されたアクティブ ルータで障害が発生すると、HSRP によって検出されます。この時点で、選択されている*スタンバイ ルータ*が HSRP グループのバーチャル MAC および IP アドレスの制御を引き継ぎます。HSRP はこの時点で、新しいスタンバイ ルータの選択も行います。

HSRP ではプライオリティ メカニズムを使用して、デフォルトのアクティブ ルータにする HSRP 設定インターフェイスを決定します。アクティブ ルータとしてインターフェイスを設定するには、グループ内の他のすべての HSRP 設定インターフェイスより高いプライオリティを与えます。デフォルトのプライオリティは 100 なので、それよりプライオリティが高いインターフェイスを 1 つ設定すると、そのインターフェイスがデフォルトのアクティブ ルータになります。

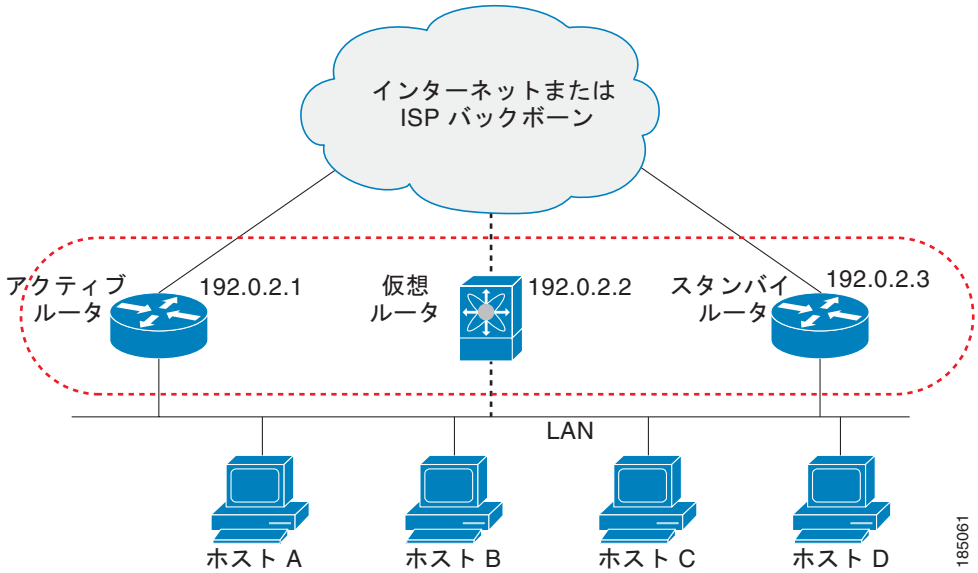
HSRP が動作するインターフェイスは、マルチキャスト UDP（ユーザ データグラム プロトコル）ベースの hello メッセージを送受信して、障害を検出し、アクティブおよびスタンバイ ルータを指定します。アクティブ ルータが設定された時間内に hello メッセージを送信できなかった場合は、

最高のプライオリティのスタンバイ ルータがアクティブ ルータになります。アクティブ ルータとスタンバイ ルータ間のパケット フォワーディング機能の移動は、ネットワーク上のすべてのホストに対して完全に透過的です。

1 つのインターフェイス上で複数の HSRP グループを設定できます。

図 17-1 に、HSRP 対応として設定されたネットワークを示します。バーチャル MAC アドレスおよびバーチャル IP アドレスを共有することによって、2 つ以上のインターフェイスを 単一の*仮想ルータ*として動作させることができます。バーチャル ルータは物理的には存在しませんが、相互にバックアップするように設定されたインターフェイスにとって、共通のデフォルト ゲートウェイになります。アクティブ ルータの IP アドレスを使用して、LAN 上でホストを設定する必要はありません。代わりに、デフォルト ゲートウェイとしてバーチャル ルータの IP アドレス（バーチャル IP アドレス）を使用して、ホストを設定します。アクティブ ルータが設定時間内に hello メッセージを送信できなかった場合は、スタンバイ ルータが引き継いでバーチャル アドレスに応答し、アクティブ ルータになってアクティブ ルータの役割を引き受けます。ホストの観点からは、バーチャル ルータは同じままです。

図 17-1　2 つの対応ルータからなる HSRP トポロジ



HSRP のバージョン

Cisco NX-OS は、デフォルトで HSRP バージョン 1 をサポートします。HSRP バージョン 2 を使用するようにインターフェイスを設定できます。

HSRP バージョン 2 では、HSRP バージョン 1 から次のように拡張されています。

- ミリ秒タイマーのサポートが追加されました。
- グループ番号の範囲が拡大されました。HSRP バージョン 1 がサポートするグループ番号は 0 ～ 255 です。HSRP バージョン 2 は、0 ～ 4095 のグループ番号をサポートします。
- HSRP バージョン 2 では、HSRP バージョン 1 で使用する IP マルチキャスト アドレス 224.0.0.2 の代わりに、新しい IP マルチキャスト アドレス 224.0.0.102 を使用して hello パケットを送信します。

- 0000.0C9F.F000 ～ 0000.0C9F.FFFF の MAC アドレス範囲を使用します。HSRP バージョン 1 で使用する MAC アドレス範囲は、0000.0C07.AC00 ～ 0000.0C07.ACFF です。
- MD 5 ダイジェスト認証のサポートが追加されました。

HSRP のバージョンを変更すると、Cisco NX-OS がグループを再初期化します。新しいバーチャル MAC アドレスがグループに与えられるからです。

HSRP バージョン 2 では HSRP バージョン 1 とは異なるパケット フォーマットを使用します。パケット フォーマットは TLV です。HSRP バージョン 1 ルータは、HSRP バージョン 2 パケットを受信しても無視します。

HSRP 認証

HSRP MD5（メッセージ ダイジェスト 5）アルゴリズム方式の認証は、HSRP スプーフィング ソフトウェアから保護し、業界標準である MD5 アルゴリズムを使用して、信頼性およびセキュリティを向上させます。

HSRP アドレッシング

HSRP ルータは HSRP hello パケットを交換することによって、相互に通信します。これらのパケットは、UDP ポート 1985 上の宛先 IP マルチキャスト アドレス 224.0.0.2（すべてのルートと通信するための予約済みマルチキャスト アドレス）に送信されます。アクティブ ルータは設定 IP アドレスおよび HSRP バーチャル MAC アドレスから hello パケットを得るのに対して、スタンバイ ルータは設定 IP アドレスおよびインターフェイス MAC アドレスから hello パケットを取得します。インターフェイス MAC アドレスは、BIA（バーンドイン MAC アドレス）のこともあれば、そうではないこともあります。

ホストはデフォルト ゲートウェイが HSRP バーチャル IP アドレスとして設定されているので、HSRP バーチャル IP アドレスに関連付けられた MAC アドレスと通信する必要があります。この MAC アドレスは、バーチャル MAC アドレス 0000.0C07.ACxy です。この場合、xy はそれぞれのインターフェイスに基づく、16 進数の HSRP グループ番号です。たとえば、HSRP グループ 1 は 0000.0C07.AC01 という HSRP バーチャル MAC アドレスを使用します。隣接 LAN セグメント上のホストは、標準の ARP（アドレス解決プロトコル）プロセスを使用して、関連付けられた MAC アドレスを解決します。

HSRP バージョン 2 では新しい IP マルチキャスト アドレス 224.0.0.102 を使用して hello パケットを送信します。バージョン 1 では、このマルチキャスト アドレスが 224.0.0.2 です。HSRP バージョン 2 では、拡張グループ番号範囲 0 ～ 4095 を使用できます。また、新しい MAC アドレス範囲 0000.0C9F.F000 ～ 0000.0C9F.FFFF を使用します。

HSRP メッセージ

HSRP が設定されたルータは、次の 3 種類のマルチキャスト メッセージを交換できます。

- hello ― hello メッセージは、ルータの HSRP プライオリティおよびステート情報を他の HSRP ルータに伝えます。
- coup ― スタンバイ ルータがアクティブ ルータの機能を引き受けるときに、coup メッセージを送信します。
- resign ― このメッセージは、アクティブ ルータであるルータがシャットダウン直前、またはプライオリティの高いルータから hello または coup メッセージが送信されたときに、ルータから送信されます。

HSRP ロード シェアリング

HSRP を使用すると、複数のグループで 1 つのインターフェイスを設定できます。オーバーラップする 2 つの HSRP グループを設定すると、期待されるデフォルト ゲートウェイの冗長性を HSRP から提供しながら、接続ホストからのトラフィックのロード シェアリングが可能です。図 17-2 に、ロード シェアリングが行われる HSRP 構成の例を示します。

図 17-2　HSRP ロード シェアリング

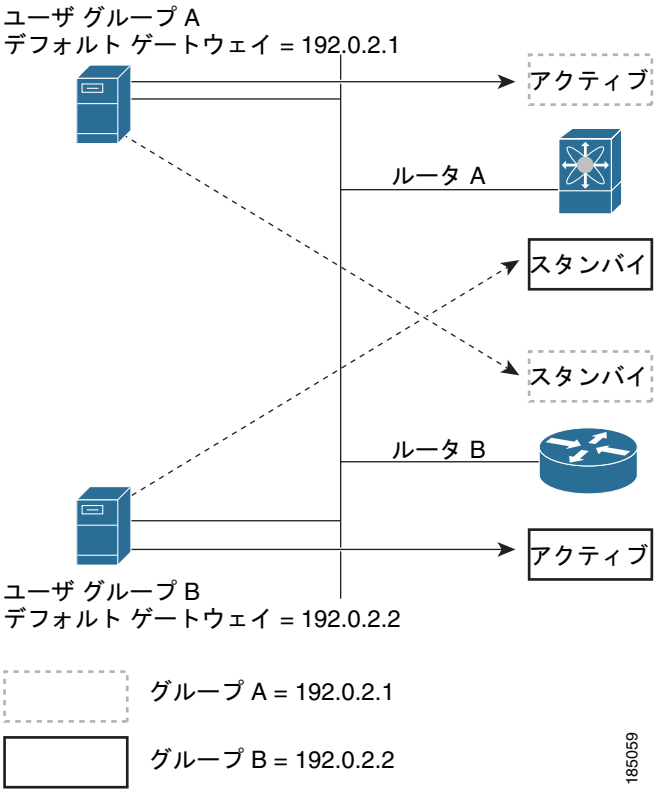


図 17-2 に、ルータ A、ルータ B、および 2 つの HSRP グループを示します。ルータ A はグループ A のアクティブ ルータであり、グループ B のスタンバイ ルータです。同様に、ルータ B は グループ B のアクティブ ルータであり、グループ A のスタンバイ ルータです。両方のルータがアクティブであるかぎり、HSRP は両方のルータにわたって、ホストからのトラフィックのロード バランスを図ります。どちらかのルータで障害が発生すると、残りのルータが引き続き、両方のホストのトラフィックを処理します。

オブジェクト トラッキングおよび HSRP

オブジェクト トラッキングを使用すると、別のインターフェイスの動作状態に基づいて、HSRP インターフェイスのプライオリティを変更できます。オブジェクト トラッキングによって、メイン ネットワークへのインターフェイスで障害が発生した場合に、スタンバイ ルータにルーティングできます。

トラッキング可能なオブジェクトは、インターフェイスのライン プロトコル ステートまたは IP ルートの到達可能性の 2 種類です。指定したオブジェクトがダウンすると、設定された値だけ、Cisco DC-OS が HSRP プライオリティを引き下げます。詳細については、「HSRP オブジェクト トラッキングの設定」（p.17-13）を参照してください。

ハイ アベイラビリティ

HSRP はステートフル リスタートをサポートします。ステートフル リスタートは、スーパーバイザのスイッチオーバー時に発生します。スイッチオーバー後に Cisco NX-OS が実行コンフィギュレーションを適用します。

仮想化サポート

HSRP は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。

インターフェイスの VRF メンバシップを変更すると、Cisco NX-OS によって HSRP を含め、すべてのレイヤ 3 設定が削除されます。

詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

HSRP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	HSRP にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

HSRPP の前提条件

HSRP の前提条件は、次のとおりです。

- HSRP グループを設定してイネーブルにするには、その前に HSRP をグローバルでイネーブルにする必要があります。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

注意事項および制約事項

HSRP に関する注意事項および制約事項は、次のとおりです。

- 最小 hello タイマー値は 250 ミリ秒です。
- 最小ホールド タイマー値は 750 ミリ秒です。
- HSRP を設定するインターフェイスに IP アドレスを設定し、そのインターフェイスをイネーブルにしてからでなければ、HSRP はアクティブになりません。
- バーチャル IP アドレスは、インターフェイス IP アドレスと同じサブネットになければなりません。
- 同一インターフェイス上では、複数のファーストホップ冗長プロトコルを設定しないことを推奨します。
- HSRP バージョン 2 は HSRP バージョン 1 と相互運用できません。どちらのバージョンも相互に排他的なので、インターフェイスがバージョン 1 およびバージョン 2 の両方を運用することはできません。しかし、同一ルータの異なる物理インターフェイス上であれば、異なるバージョンを実行できます。
- バージョン 1 で認められるグループ番号範囲（0 ～ 255）を超えるグループを設定している場合は、バージョン 2 からバージョン 1 への変更はできません。
- インターフェイス VRF メンバシップ、ポート チャネル メンバシップを変更した場合、またはポート モードをレイヤ 2 に変更した場合は、Cisco NX-OS によってインターフェイス上のすべてのレイヤ 3 設定が削除されます。

HSRP の設定

ここでは、次の内容について説明します。

- HSRP 機能のイネーブル化（p.17-8）
- HSRP グループの設定（p.17-8）
- HSRP バージョン設定（p.17-10）
- HSRP バーチャル MAC アドレスの設定（p.17-10）
- HSRP の認証（p.17-11）
- HSRP オブジェクト トラッキングの設定（p.17-13）
- HSRP のカスタマイズ（p.17-15）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

HSRP 機能のイネーブル化

HSRP グループを設定してイネーブルにするには、その前に HSRP 機能をグローバルでイネーブルにする必要があります。

VDC で HSRP 機能をイネーブルにするには、グローバル 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。

コマンド	目的
feature hsrp **例：** `switch(config)# feature hsrp`	HSRP をイネーブルにします。

VDC で HSRP 機能をディセーブルにするには、グローバル 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。

コマンド	目的
no feature hsrp **例：** `switch(config)# no feature hsrp`	VDC ですべてのグループの HSRP をディセーブルにします。

HSRP グループの設定

インターフェイス上で HSRP グループを設定し、その HSRP グループにバーチャル IP アドレスおよびバーチャル MAC アドレスを設定できます。

準備作業

HSRP 機能がイネーブルになっていることを確認します（「HSRP 機能のイネーブル化」[p.17-8] を参照）。

グループのいずれかのメンバ インターフェイス上でバーチャル IP アドレスを設定すると、Cisco NX-OS によって HSRP がイネーブルになります。HSRP をイネーブルにする前に、認証、タイマー、プライオリティなどの HSRP アトリビュートを設定する必要があります。

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）。

手順概要

1. **config t**
2. **interface** *type number*
3. ***ip*** *ip-address/length*
4. ***hsrp*** *group-number*
5. ***ip*** [*ip-address* [***secondary***]]
6. **exit**
7. ***no shutdown***
8. **show hsrp** [**group** *group-number*]
9. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 number*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**ip** *ip-address/length* **例：** `switch(config-if)# ip 192.0.2.2/8`	インターフェイスの IPv4 アドレスを設定します。

ステップ 4	**hsrp** *group-number*
	
	例：
	`switch(config-if)# hsrp 2`
	`switch(config-if-hsrp)#`
		HSRP グループを作成し、HSRP コンフィギュレーション モードを開始します。有効値の範囲は 0 ～ 4095 であり、デフォルト値は 0 です。

ステップ 5	**ip** [*ip-address* [***secondary***]] **例：** `switch(config-if-hsrp)# ip 192.0.2.1`	HSRP グループのバーチャル IP アドレスを設定し、グループをイネーブルにします。このアドレスは、インターフェイスの IPv4 アドレスと同じサブネットになければなりません。
ステップ 6	**exit** **例：** `switch(config-if-hsrp)# exit`	HSRP 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	**no shutdown** **例：** `switch(config-if)# no shutdown`	インターフェイスをイネーブルにします。

	コマンド	目的
ステップ 8	***show hsrp*** [***group*** *group-number*] **例：** `switch(config-if)# show hsrp group 2`	（任意）HSRP 情報を表示します。
ステップ 9	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

（注） 設定完了後にインターフェイスをイネーブルにするには、no shutdown コマンドを使用する必要があります。

Ethernet 1/2 上で HSRP グループを設定する例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# ip 192.0.2.2/8
switch(config-if)# hsrp 2
switch(config-if-hsrp)# ip 192.0.2.1
switch(config-if-hsrp)# exit
switch(config-if)# no shutdown
switch(config-if)# copy running-config startup-config
```

HSRP バージョン設定

HSRP のバージョンを設定できます。既存グループのバージョンを変更すると、バーチャル MAC アドレスが変更されるので、Cisco NX-OS がそれらのグループの HSRP を再初期化します。HSRP のバージョンは、インターフェイス上のすべてのグループに適用されます。

HSRP のバージョンを設定するには、インターフェイス 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。

コマンド	目的
hsrp **version** {**1** | **2**} **例：** `switch(config-if)# hsrp version 2`	HSRP バージョンを設定します。デフォルトはバージョン 1 です。

HSRP バーチャル MAC アドレスの設定

設定されたグループ番号に基づいて HSRP が生成したデフォルト バーチャル MAC アドレスを変更できます。

HSRP グループのバーチャル MAC アドレスを手動で設定するには、HSRP 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。

コマンド	目的
mac-address *string* **例：** `switch(config-if-hsrp)# mac-address 5000.1000.1060`	HSRP グループのバーチャル MAC アドレスを設定します。ストリングには標準の MAC アドレス フォーマット（xxxx.xxxx.xxxx）を使用します。

バーチャル MAC アドレスに BIA（バーンドイン MAC アドレス）を使用するように HSRP を設定するには、インターフェイス 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。

コマンド	目的
hsrp use-bia [***scope interface***] **例：** `switch(config-if)# hsrp use-bia`	HSRP バーチャル MAC アドレスにインターフェイスの BIA を使用するように、HSRP を設定します。任意で ***scope interface*** キーワードを使用すると、このインターフェイス上のすべてのグループに BIA を使用するように HSRP を設定できます。

HSRP の認証

クリアテキストまたは MD5 ダイジェスト認証を使用してプロトコルを認証するように、HSRP を設定できます。MD5 認証ではキーチェーンを使用します（『*Cisco NX-OS Security Configuration Guide*』を参照）。

（注） HSRP バージョン 1 がサポートするのは、テキスト認証だけです。

準備作業

HSRP 機能がイネーブルになっていることを確認します（「HSRP 機能のイネーブル化」[p.17-8] を参照）。

HSRP グループのすべてのメンバに同じ認証および鍵を設定する必要があります。

MD5 認証を使用する場合は、キーチェーンが作成してあることを確認します。

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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **hsrp** *group-number*
4. **authentication text** *string*

   または

   authentication md5 {**key-chain** *key-chain* | ***key-string*** {***0*** | ***7***} *text* [**timeout** *seconds*]}
5. **show hsrp [group** *group-number*]
6. ***copy running-config startup-config***

手順詳細

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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** *interface-type slot/port* **例：** switch(config)# interface ethernet 1/2 switch(config-if)#	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**hsrp** *group-number* **例：** switch(config-if)# hsrp 2 switch(config-if-hsrp)#	HSRP グループを作成し、HSRP 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	**authentication text** *string* **例：** switch(config-if-hsrp)# authentication text mypassword	このインターフェイス上で、HSRP のクリアテキスト認証を設定します。
	authentication md5 {***key-chain*** *key-chain* | ***key-string*** {***0*** | ***7***} *text* [**timeout** *seconds*]} **例：** switch(config-if-hsrp)# authentication md5 key-chain hsrp-keys	このインターフェイス上で、HSRP の MD5 認証を設定します。キーチェーンまたはキー ストリングを使用できます。キー ストリングを使用する場合は、HSRP が新しいキーだけを受け付けるように、任意でタイムアウトを設定できます。有効値の範囲は 0 ～ 32767 秒です。
ステップ 5	***show hsrp*** [***group*** *group-number*] **例：** switch(config-if-hsrp)# show hsrp group 2	（任意）HSRP 情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** switch(config-if-hsrp)# copy running-config startup-config	（任意）この設定変更を保存します。

キーチェーンの作成後に、Ethernet 1/2 で HSRP の MD5 認証を設定する例を示します。

```
switch# config t
switch(config)# key chain hsrp-keys
switch(config-keychain)# key 0
switch(config-keychain-key)# key-string 7 zqdest
switch(config-keychain-key) accept-lifetime 00:00:00 Jun 01 2008 23:59:59 Sep 12 2008
switch(config-keychain-key) send-lifetime 00:00:00 Jun 01 2008 23:59:59 Aug 12 2008
switch(config-keychain-key) key 1
switch(config-keychain-key) key-string 7 uaeqdyito
switch(config-keychain-key) accept-lifetime 00:00:00 Aug 12 2008 23:59:59 Dec 12 2008
switch(config-keychain-key) send-lifetime 00:00:00 Sep 12 2008 23:59:59 Nov 12 2008
switch(config-keychain-key)# interface ethernet 1/2
switch(config-if)# hsrp 2
switch(config-if-hsrp)# authenticate md5 key-chain hsrp-keys
switch(config-if-hsrp)# copy running-config startup-config
```

HSRP オブジェクト トラッキングの設定

他のインターフェイスまたはルータの可用性に基づいて、プライオリティが調整されるように HSRP グループを設定できます。デバイスがオブジェクト トラッキング対応として設定されていて、なおかつトラッキング対象のオブジェクトがダウンした場合、デバイスのプライオリティはダイナミックに変更されます。トラッキング プロセスはトラッキング対象オブジェクトに定期的にポーリングを実行し、値の変化を記録します。値が変化すると、HSRP がプライオリティを再計算します。HSRP インターフェイスにプリエンプトを設定している場合は、プライオリティの高い HSRP インターフェイスがアクティブ ルータになります。オブジェクト トラッキングの詳細については、「HSRP オブジェクト トラッキングの設定」（p.17-13）を参照してください。

準備作業

HSRP 機能がイネーブルになっていることを確認します（「HSRP 機能のイネーブル化」[p.17-8] 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）。

手順概要

1. **config t**
2. **track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
3. **track** *object-id* **ip route** *ip-prefix/length* **reachability**
4. **interface** *interface-type slot/port*
5. **hsrp** *group-number*
6. **priority** [*value*]
7. **track** *object-number* [***decrement*** *value*]
8. **preempt** [***delay minimum*** *seconds*] [***reload*** *seconds*] [***sync*** *seconds*]
9. **show hsrp interface** *interface-type number*
10. ***copy running-config startup-config***

手順詳細

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
	
	例：
	`switch(config)# track 1 interface ethernet 2/2 line-protocol`
	`switch(config-track#`
		この HSRP インターフェイスが追跡するインターフェイスを設定します。インターフェイスのステート変化は次のように、この HSRP のプライオリティを左右します。
		• HSRP コンフィギュレーション モードで、**track** コマンドで使用するインターフェイスおよび対応するオブジェクト番号を設定します。
		• **line-protocol** キーワードを指定すると、インターフェイスがアップかどうかが追跡されます。**ip** キーワードを指定すると、インターフェイス上で IP ルーティングがイネーブルであり、IP アドレスが設定されているかどうかもチェックされます。

	track *object-id* **ip route** *ip-prefix/length* **reachability**
	
	例：
	`switch(config)# track 2 ip route 192.0.2.0/8 reachability`
	`switch(config-track#`
		ルートの追跡対象オブジェクトを作成し、トラッキング コンフィギュレーション モードを開始します。object-id の範囲は 1 ～ 500 です。

ステップ 3	**interface** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**hsrp** *group-number*
	
	例：
	`switch(config-if)# hsrp 2`
	`switch(config-if-hsrp)#`
		HSRP グループを作成し、HSRP 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	**priority** [*value*] **例：** `switch(config-if-hsrp)# priority 254`	HSRP グループでのアクティブ ルータ選択に使用するプライオリティ レベルを設定します。有効値の範囲は 0 ～ 255 であり、デフォルトは 100 です。
ステップ 6	**track** *object-number* [**decrement** *value*] **例：** `switch(config-if-hsrp)# track 1 decrement 20`	HSRP インターフェイスの重み付けを左右する、トラッキング対象のオブジェクトを指定します。 *value* 引数には、トラッキング対象のオブジェクトで障害が発生した場合に、HSRP インターフェイスのプライオリティから差し引く値を指定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 10 です。
ステップ 7	**preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] **例：** `switch(config-if-hsrp)# preempt delay minimum 60`	現在のアクティブ ルータよりプライオリティが高い場合に、HSRP グループのアクティブ ルータとして引き継ぐようにルータを設定します。このコマンドはデフォルトでディセーブルです。有効値の範囲は 0 ～ 3600 秒です。

	コマンドまたはアクション	目的
ステップ 8	**show hsrp interface** *interface-type number* **例：** `switch(config-if-hsrp)# show hsrp interface ethernet 1/2`	（任意）インターフェイスの HSRP 情報を表示します。
ステップ 9	***copy running-config startup-config*** **例：** `switch(config-if-hsrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

Ethernet 1/2 上で HSRP オブジェクト トラッキングを設定する例を示します。

```
switch# config t
switch(config)# track 1 interface ethernet 2/2 line-protocol
switch(config)# interface ethernet 1/2
switch(config-if)# hsrp 2
switch(config-if-hsrp)# track 1 decrement 20
switch(config-if-hsrp)# copy running-config startup-config
```

HSRP のカスタマイズ

任意で、HSRP の動作をカスタマイズできます。バーチャル IP アドレスを設定することによって、HSRP グループをイネーブルにすると、そのグループがただちに動作可能になることに注意してください。HSRP をカスタマイズする前に HSRP グループをイネーブルにすると、機能のカスタマイズが完了しないうちに、ルータがグループの制御を引き継ぎ、AVG になる可能性があります。HSRP のカスタマイズを予定している場合は、HSRP グループをイネーブルにする前に行ってください。

HSRP をカスタマイズするには、HSRP 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。

コマンドまたはアクション	目的
name *string* **例：** `switch(config-if-hsrp)# name HSRP-1`	HSRP グループの IP 冗長名を指定します。*string* は 1 ～ 255 文字です。デフォルト ストリングのフォーマットは、 hsrp-<interface-short-name>-<group-id> です。たとえば、hsrp-Eth2/1-1 です。
priority [*level*] **例：** `switch(config-if-hsrp)# priority 254`	HSRP グループでのアクティブ ルータ選択に使用するプライオリティ レベルを設定します。有効値の範囲は 0 ～ 255 であり、デフォルトは 100 です。
preempt [**delay** [**minimum** *seconds*] [***reload*** *seconds*] [***sync*** *seconds*]] **例：** `switch(config-if-hsrp)# preempt delay minimum 60`	現在のアクティブ ルータよりプライオリティが高い場合に、HSRP グループのアクティブ ルータとして引き継ぐようにルータを設定します。このコマンドはデフォルトでディセーブルです。有効値の範囲は 0 ～ 3600 秒です。

コマンドまたはアクション	目的
timers [**msec**] *hellotime* [**msec**] *holdtime* **例：** `switch(config-if-hsrp)# timers 5 18`	次のように、この HSRP メンバーの hello タイムおよびホールド タイムを設定します。 • *hellotime* ― hello パケットを送信してから、次の hello パケットを送信するまでのインターバル。有効値の範囲は 1 ～ 254 秒です。 • *holdtime* ― hello パケットの情報が無効とみなされるまでのインターバル。範囲は 3 ～ 255 です。 オプションの ***msec*** キーワードでは、引数をデフォルトの秒単位ではなく、ミリ秒単位で表すことを指定します。

HSRP をカスタマイズするには、インターフェイス 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。

コマンドまたはアクション	目的
hsrp delay minimum *seconds* **例：** `switch(config-if)# hsrp delay minimum 30`	グループがイネーブルになってから、グループに参加するまでに HSRP が待機する最小時間を指定します。有効値の範囲は 0 ～ 10000 秒です。デフォルト値は 0 です。
hsrp delay reload *seconds* **例：** `switch(config-if)# hsrp delay reload 30`	リロード後、グループに参加するまでに HSRP が待機する最小時間を指定します。有効値の範囲は 0 ～ 10000 秒です。デフォルト値は 0 です。

HSRP の設定確認

HSRP の設定情報を確認するには、次のコマンドのいずれか 1 つを使用します。

コマンド	目的
show hsrp [**group** *group-number*]	すべてまたは特定のグループの HSRP ステータスを表示します。
show hsrp delay [***interface*** *interface-type slot/port*]	すべてまたは特定のインターフェイスの HSRP 遅延値を表示します。
show hsrp [***interface*** *interface-type slot/port*]	インターフェイスの HSRP ステータスを表示します。
show hsrp [**group** *group-number*] [***interface*** *interface-type slot/port*] [***active] [all***] [***init***] [**learn**] [***listen***] [**speak**] [***standby***]	ステートが active、init、listen、または standby のバーチャル フォワーダについて、グループまたはインターフェイスの HSRP ステータスを表示します。disabled を含めてすべてのステートを表示する場合は、**all** キーワードを使用します。
show hsrp [**group** *group-number*] [***interface*** *interface-type slot/port*] ***active***] [**all**] [***init***] [**learn**] [***listen***] [**speak**] [***standby***] ***brief***	ステートが active、init、listen、または standby のバーチャル フォワーダについて、グループまたはインターフェイスの HSRP ステータスの要約を表示します。disabled を含めてすべてのステートを表示する場合は、**all** キーワードを使用します。

HSRP の設定例

MD5 認証およびインターフェイス トラッキングを指定して、インターフェイス上で HSRP をイネーブルにする例を示します。

```
key chain hsrp-keys
 key 0
   key-string 7 zqdest
  accept-lifetime 00:00:00 Jun 01 2008 23:59:59 Sep 12 2008
   send-lifetime 00:00:00 Jun 01 2008 23:59:59 Aug 12 2008
  key 1
   key-string 7 uaeqdyito
   accept-lifetime 00:00:00 Aug 12 2008 23:59:59 Dec 12 2008
   send-lifetime 00:00:00 Sep 12 2008 23:59:59 Nov 12 2008
feature hsrp
track 2 interface ethernet 2/2 ip
interface ethernet 1/2
 ip address 192.0.2.2/8
 hsrp 1
  authenticate md5 key-chain hsrp-keys
  priority 90
  track 2 decrement 20
  ip-address 192.0.2.10
 no shutdown
```

デフォルト設定

表 17-1 に、HSRP パラメータのデフォルト設定を示します。

表 17-1　デフォルトの HSRP パラメータ

パラメータ	デフォルト
HSRP	ディセーブル
認証	バージョン 1 の場合はテキストとしてイネーブル、パスワードは cisco
HSRP のバージョン	バージョン 1
プリエンプト	ディセーブル
プライオリティ	100
バーチャル MAC アドレス	HSRP グループ番号から生成

その他の関連資料

IP 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。

- 関連資料（p.17-18）
- 規格（p.17-18）

関連資料

関連項目	マニュアル名
ゲートウェイ ロード バランシング プロトコルの設定	第 16 章「GLBP の設定」
VRRP の設定	第 18 章「VRRP の設定」
HSRP CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』
ハイ アベイラビリティの設定	『*Cisco NX-OS HA Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 18

VRRP の設定

この章では、デバイス上で VRRP（仮想ルータ冗長プロトコル）を設定する方法について説明します。

ここでは、次の内容を説明します。

- VRRP の概要（p.18-2）
- VRRP のライセンス要件（p.18-7）
- 注意事項および制約事項（p.18-7）
- VRRP の設定（p.18-8）
- VRRP の設定確認（p.18-18）
- VRRP 統計情報の表示（p.18-18）
- VRRP の設定例（p.18-19）
- デフォルト設定（p.18-20）
- その他の関連資料（p.18-20）

VRRP の概要

VRRP を使用すると、バーチャル IP アドレスを共有するルータ グループを設定することによって、ファーストホップ IP ルータで透過的フェールオーバーが可能になります。VRRP では、バーチャル IP アドレスへのすべてのパケットを引き受ける、そのグループのマスター ルータが選択されます。残りのルータはスタンバイになり、マスター ルータで障害が発生した場合に引き継ぎます。

ここでは、次の内容について説明します。

- VRRP の動作（p.18-2）
- VRRP の利点（p.18-3）
- マルチ VRRP グループ（p.18-4）
- VRRP ルータのプライオリティおよびプリエンプト（p.18-4）
- VRRP のアドバタイズメント（p.18-5）
- VRRP 認証（p.18-5）
- VRRP トラッキング（p.18-5）
- ハイ アベイラビリティ（p.18-6）
- 仮想化サポート（p.18-6）

VRRP の動作

LAN クライアントは、ダイナミック プロセスまたはスタティック設定を使用することによって、特定のリモート宛先へのファーストホップにするルータを決定できます。ダイナミック ルータ ディスカバリの例を示します。

- プロキシ ARP ― クライアントは ARP（アドレス解決プロトコル）を使用して到達すべき宛先を取得します。ルータは独自の MAC アドレスで ARP 要求に応答します。
- ルーティング プロトコル ― クライアントはダイナミック ルーティング プロトコルのアップデート（RIP など）を受信し、独自のルーティング テーブルを形成します。
- IRDP クライアント ― クライアントは ICMP(インターネット制御メッセージ プロトコル)ルータ ディスカバリ クライアントを実行します。

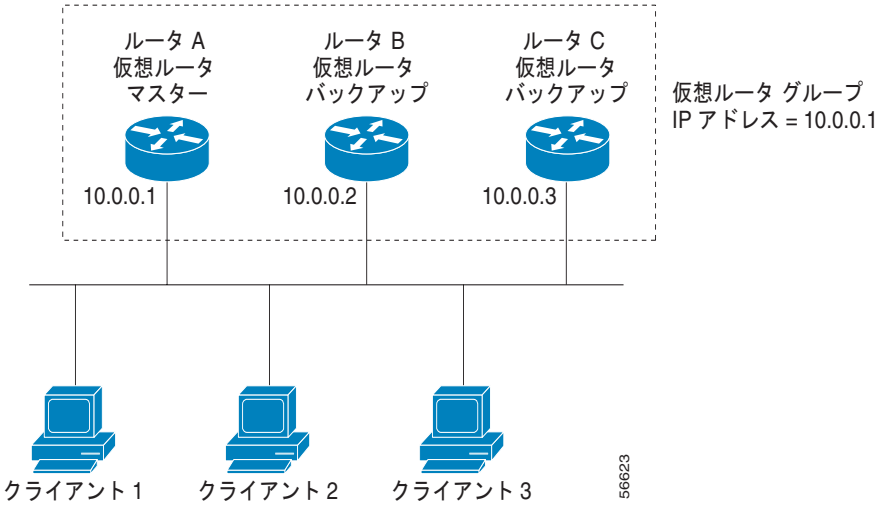
ダイナミック ディスカバリ プロトコルのデメリットは、LAN クライアントにある程度、設定および処理のオーバーヘッドが発生することです。また、ルータで障害が発生した場合に、別のルータへの切り替え処理が遅くなる可能性があります。

ダイナミック ディスカバリ プロトコルの代わりに、クライアント上でデフォルト ルータをスタティ ｘックに設定することもできます。この方法を使用すると、クライアントの設定および処理が簡素化されますが、シングルポイント障害が生じます。デフォルト ゲートウェイで障害が発生した場合、LAN クライアントの通信はローカル IP ネットワーク セグメントに限定され、ネットワークの他の部分から切り離されます。

VRRP では、ルータ グループ（VRRP グループ）が単一のバーチャル IP アドレスを共有できるようにすることによって、スタティック設定に伴う問題を解決できます。さらに、デフォルト ゲートウェイとしてバーチャル IP アドレスを指定して、LAN クライアントを設定できます。

図 18-1 に、基本的な LAN トポロジを示します。この例では、ルータ A、B、および C が VRRP グループを形成します。グループの IP アドレスは、ルータ A のインターフェイス インターフェイスに設定されているアドレス（10.0.0.1）と同じです。

図 18-1　基本的な VRRP トポロジ



バーチャル IP アドレスにルータ A の物理イーサネット インターフェイスの IP アドレスを使用するので、ルータ A がマスター（別名、*IP アドレス オーナー*）です。ルータ A はマスターとして、VRRP グループのバーチャル IP アドレスを所有し、送信されたパケットをこの IP アドレスに転送します。クライアント 1 ～ 3 には、デフォルト ゲートウェイの IP アドレス 10.0.0.1 が設定されています。

ルータ B および C の役割はバックアップです。マスターで障害が発生すると、プライオリティの高いバックアップ ルータがマスターになり、バーチャル IP アドレスを引き継いで、LAN ホストへのサービスが途切れないようにします。ルータ A が回復すると、再びマスターになります。詳細については、「VRRP ルータ プライオリティおよびプリエンプト」を参照してください。

VRRP の利点

VRRP の利点は、次のとおりです。

- 冗長性 ― 複数のルータをデフォルト ゲートウェイ ルータとして設定できるので、ネットワークにシングルポイント障害が発生する確率が下がります。
- ロード シェアリング ― 複数のルータで LAN クライアントとの間のトラフィックを分担できます。トラフィックの負荷が使用可能なルータ間でより公平に分担されます。
- マルチ VRRP グループ ― プラットフォームがマルチ MAC アドレスをサポートする場合、ルータの物理インターフェイス上で、最大 255 の VRRP グループをサポートします。マルチ VRRP グループによって、LAN トポロジで冗長性およびロード シェアリングを実現できます。
- マルチ IP アドレス ― セカンダリ IP アドレスを含めて、複数の IP アドレスを管理できます。イーサネット インターフェイス上で複数のサブネットを設定しているバイアは、各サブネットで VRRP を設定できます。
- プリエンプト ― 障害マスターを引き継いでいたバックアップ ルータより、さらにプライオリティが高いバックアップ ルータが使用可能になったときに、プライオリティが高い方を優先させることができます。
- アドバタイズメント プロトコル ― VRRP アドバタイズメントに、専用の IANA（インターネット割り当て番号局）規格マルチキャスト アドレス（224.0.0.18）を使用します。このアドレッシング方式によって、マルチキャストを提供するルータ数が最小限になり、テスト機器でセグメント上の VRRP パケットを正確に識別できるようになります。IANA は VRRP に IP プロトコル番号 112 を割り当てています。
- VRRP トラッキング ― インターフェイスのステートに基づいて VRRP プライオリティを変更することによって、最適な VRRP ルータがグループのマスターになることが保証されます。

マルチ VRRP グループ

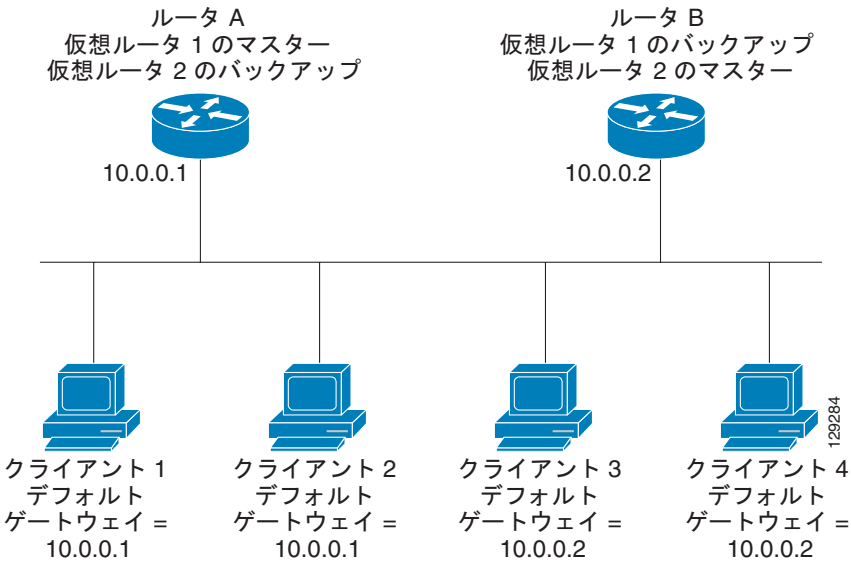
物理インターフェイス上で、最大 255 の VRRP グループを設定できます。ルータ インターフェイスがサポートできる VRRP グループの実際の数は、次の要因によって決まります。

- ルータの処理能力
- ルータのメモリの能力

ルータ インターフェイス上で複数の VRRP グループが設定されたトポロジでは、インターフェイスはある VRRP グループのマスターとして、また、他の 1 つまたは複数の VRRP グループのバックアップとして動作可能です。

図 18-2 に、ルータ A および B がクライアント 1 ～ 4 との間でトラフィックを共有するように VRRP が設定されている LAN トポロジを示します。

図 18-2　ロード シェアリングおよび冗長構成の VRRP トポロジ



このトポロジには、オーバーラップする 2 つの VRRP グループに対応する 2 つの IP アドレスが含まれています。VRRP グループ 1 では、ルータ A が IP アドレス 10.0.0.1 のオーナーであり、マスターです。ルータ B はルータ A のバックアップです。クライアント 1 および 2 には、デフォルト ゲートウェイ IP アドレス 10.0.0.1 が設定されています。

VRRP グループ 2 では、ルータ B が IP アドレス 10.0.0.2 のオーナーであり、マスターです。ルータ A はルータ B のバックアップです。クライアント 3 および 4 には、デフォルト ゲートウェイ IP アドレス 10.0.0.2 が設定されています。

VRRP ルータのプライオリティおよびプリエンプト

VRRP 冗長構成の重要なポイントは、VRRP ルータのプライオリティです。プライオリティによって、各 VRRP ルータが果たす役割が決まり、マスター ルータで障害が発生した場合のアクションが決まるからです。

VRRP ルータがバーチャル IP アドレスおよび物理インターフェイスの IP アドレスを所有する場合、そのルータはマスターとして機能します。マスターのプライオリティは 255 です。

プライオリティによって、VRRP ルータがバックアップ ルータとして動作するかどうかが決まり、さらに、マスターで障害が発生した場合にマスターになる順序も決まります。

たとえば、ルータ A が LAN トポロジにおけるマスターであり、そのルータ A で障害が発生した場合、VRRP はバックアップ B が引き継ぐのか、バックアップ C が引き継ぐのかを判断する必要があります。ルータ B にプライオリティ 101 が設定されていて、ルータ C がデフォルトのプライオリティ 100 の場合、VRRP はルータ B をマスターになるべきルータとして選択します。ルータ B の方がプライオリティが高いからです。ルータ B および C にデフォルトのプライオリティ 100 が設定されている場合は、VRRP は IP アドレスが大きい方のバックアップをマスターになるべきルータとして選択します。

VRRP ではプリエンプトを使用して、VRRP バックアップ ルータがマスターになってからのアクションを決定します。プリエンプトはデフォルトでイネーブルなので、VRRP は新しいマスターよりプライオリティの高いバックアップがオンラインになると、バックアップに切り替えます。たとえば、ルータ A がマスターであり、そのルータ A で障害が発生した場合、VRRP は（プライオリティの次の順位である）ルータ B を選択します。ルータ C がルータ B より高いプライオリティでオンラインになると、ルータ B で障害が発生していなくても、VRRP はルータ A を新しいマスターとして選択します。

プリエンプトをディセーブルにした場合、VRRP が切り替えるのは、元のマスターが回復した場合、または新しいマスターで障害が発生した場合に限られます。

VRRP のアドバタイズメント

VRRP マスターは同じグループ内の他の VRRP ルータに、VRRP アドバタイズメントを送信します。アドバタイズメントは、マスターのプライオリティおよびステートを伝えます。Cisco NX-OS は VRRP アドバタイズメントを IP パケットにカプセル化して、VRRP グループに割り当てられた IP マルチキャスト アドレスに送信します。Cisco NX-OS がアドバタイズメントを送信する間隔は、デフォルトで 1 秒ですが、ユーザ側で別のアドバタイズ インターバルを設定できます。

VRRP 認証

VRRP は、次の認証方式をサポートします。

- 認証なし
- プレーンテキスト認証

VRRP は次の場合に、パケットを拒否します。

- 認証方式がルータと着信パケット間で異なっている。
- テキスト認証ストリングがルータと着信パケット間で異なっている。

VRRP トラッキング

VRRP では、ネイティブのトラッキング方式を使用して、インターフェイス ステートを追跡します。インターフェイスのステートを追跡し、そのステートを使用して VRRP グループの VRRP ルータのプライオリティを判別できます。トラッキング対象インターフェイスがダウンすると、VRRP はユーザがトラッキング対象インターフェイスのステートとして新しいプライオリティをどのように設定するかに基づいて、プライオリティをアップデートします。トラッキング対象インターフェイスがオンラインになると、VRRP は仮想ルータ グループの元のプライオリティを復元します。

たとえば、ネットワークへのアップリンクがダウンした場合、別のグループ メンバーが VRRP グループのマスターとして引き継げるように、VRRP グループ メンバーのプライオリティを引き下げなければならないことがあります。

ハイ アベイラビリティ

VRRP はステートフル リスタートをサポートします。ステートフル リスタートは、スーパーバイザのスイッチオーバー時に発生します。スイッチオーバー後に Cisco NX-OS が実行コンフィギュレーションを適用します。

仮想化サポート

VRRP は VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。

インターフェイスの VRF メンバシップを変更すると、Cisco NX-OS によって VRRP を含め、すべてのレイヤ 3 設定が削除されます。

詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

VRRP のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	VRRP にライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

注意事項および制約事項

VRRP に関する注意事項および制約事項は、次のとおりです。

- 管理インターフェイス上で VRRP を設定することはできません。
- VRRP がイネーブルの場合は、ネットワーク上のデバイス全体で VRRP 設定を複製する必要があります。
- 同一インターフェイス上では、複数のファーストホップ冗長プロトコルを設定しないことを推奨します。
- VRRP を設定するインターフェイスに IP アドレスを設定し、そのインターフェイスをイネーブルにしてからでなければ、VRRP はアクティブになりません。
- Cisco NX-OS によって、インターフェイス上のすべてのレイヤ 3 設定が削除されるのは、インターフェイス VRF メンバシップ、ポート チャネル メンバシップを変更した場合、またはポート モードをレイヤ 2 に変更した場合です。

VRRP の設定

ここでは、次の内容について説明します。

- VRRP 機能のイネーブル化（p.18-8）
- VRRP グループの設定（p.18-8）
- VRRP プライオリティの設定（p.18-10）
- VRRP 認証の設定（p.18-11）
- アドバタイズメント パケットのタイム インターバル設定（p.18-13）
- プリエンプトのディセーブル化（p.18-14）
- VRRP インターフェイス ステート トラッキングの設定（p.18-16）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

VRRP 機能のイネーブル化

VRRP グループを設定してイネーブルにするには、その前に VRRP 機能をグローバルでイネーブルにする必要があります。

VRRP 機能をイネーブルにするには、グローバル 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。

コマンド	目的
feature vrrp **例：** `switch(config)# feature vrrp`	VRRP をイネーブルにします。

VDC で VRRP 機能をディセーブルにするには、グローバル 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。

コマンド	目的
no feature vrrp **例：** `switch(config)# no feature vrrp`	VDC で VRRP 機能をディセーブルにします。

VRRP グループの設定

VRRP グループを作成し、バーチャル IP アドレスを割り当て、グループをイネーブルにすることができます。

VRRP グループに設定できるバーチャル IPv4 アドレスは 1 つです。マスター VRRP ルータはデフォルトで、バーチャル IP アドレスを直接の宛先とするパケットを廃棄します。これは、VRRP マスターがパケットを転送するネクストホップ ルータとしてのみ想定されているからです。アプリケーションによって、Cisco NX-OS が仮想ルータ IP 宛てのパケットを受け付けるようにする必要があります。バーチャル IP アドレスに secondary オプションを使用して、ローカル ルータが VRRP マスターの場合に、これらのパケットを受け付けるようにします。

VRRP グループを設定した場合は、そのグループをアクティブにするために、グループを明示的にイネーブルにする必要があります。

準備作業

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）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] を参照）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **address** *ip-address* [**secondary**]
5. **no shutdown**
6. **show vrrp**
7. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)#`
	`switch(config-if)# interface ethernet 2/1`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。有効値の範囲は 1 ～ 255 です。

ステップ 4	**address** *ip-address* [**secondary**] **例：** `switch(config-if-vrrp)# address 192.0.2.8`	指定の VRRP グループにバーチャル IPv4 アドレスを設定します。このアドレスは、インターフェイスの IPv4 アドレスと同じサブネットになければなりません。 **secondary** オプションは、VRRP ルータが仮想ルータの IP アドレスに送信されたパケットを受け付けて、アプリケーションに配信することをアプリケーションが要求する場合に限られます。

ステップ 5	**no shutdown**
	
	例：
	`switch(config-if-vrrp)# no shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをイネーブルにします。デフォルトではディセーブルです。

	コマンド	目的
ステップ 6	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報を表示します。
ステップ 7	**copy running-config startup-config** **例：** `switch(config-if-vrrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRRP プライオリティの設定

仮想ルータの有効なプライオリティ範囲は 1 ～ 254 です（1 が最下位、254 が最上位のプライオリティ）。バックアップのデフォルトのプライオリティ値は 100 です。インターフェイス アドレスがプライマリ バーチャル IP アドレスと同じデバイス（マスター）の場合、デフォルト値は 255 です。

（注） VRRP グループをイネーブルにしている場合は、ディセーブルにしてから、設定変更を行う必要があります。設定変更の完了後、VRRP グループをイネーブルにします。

準備作業

VRRP 機能がイネーブルになっていることを確認します（「VRRP の設定」[p.18-8] を参照）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **shutdown**
5. **priority** *value*
6. **no shutdown**
7. **show vrrp**
8. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。

ステップ 4	**shutdown**
	
	例：
	`switch(config-if-vrrp)# shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをディセーブルにします。デフォルトではディセーブルです。

ステップ 5	**priority** *value* **例：** `switch(config-if-vrrp)# priority 2`	特定の VRRP のプライオリティを設定します。 有効値の範囲は 1 ～ 254 です。バックアップの場合、デフォルトは 100 です。インターフェイス IP アドレスがバーチャル IP アドレスと等しいマスターの場合は 255 です。

ステップ 6	**no shutdown**
	
	例：
	`switch(config-if-vrrp)# no shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをイネーブルにします。デフォルトではディセーブルです。

ステップ 7	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報の要約を表示します。
ステップ 8	**copy running-config startup-config** **例：** `switch(config-if-vrrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRRP 認証の設定

VRRP グループに単純なテキスト認証を設定できます。

（注） VRRP グループをイネーブルにしている場合は、ディセーブルにしてから、設定変更を行う必要があります。設定変更の完了後、VRRP グループをイネーブルにします。

準備作業

ネットワーク上のすべての VRRP デバイスで、認証設定が同じであることを確認します。

VRRP 機能がイネーブルになっていることを確認します（「VRRP の設定」[p.18-8] を参照）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **shutdown**
5. **authentication　text** *password*
6. **no shutdown**
7. **show vrrp**
8. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。

ステップ 4	**shutdown**
	
	例：
	`switch(config-if-vrrp)# shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをディセーブルにします。デフォルトではディセーブルです。

ステップ 5	**authentication　text** *password* **例：** `switch(config-if-vrrp)# authentication md5 prd555oln47espn0 spi 0x0`	単純なテキスト認証オプションを指定し、キーネーム パスワードを指定します。キーネームの範囲は 1 ～ 255 文字です。16 文字以上を推奨します。テキスト パスワードは、英数字で最大 8 文字です。

ステップ 6	**no shutdown**
	
	例：
	`switch(config-if-vrrp)# no shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをイネーブルにします。デフォルトではディセーブルです。

	コマンド	目的
ステップ 7	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報の要約を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-if-vrrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

アドバタイズメント パケットのタイム インターバル設定

アドバタイズメント パケットのタイム インターバルを設定できます。

（注） VRRP グループをイネーブルにしている場合は、ディセーブルにしてから、設定変更を行う必要があります。設定変更の完了後、VRRP グループをイネーブルにします。

準備作業

VRRP 機能がイネーブルになっていることを確認します（「VRRP の設定」[p.18-8] を参照）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **shutdown**
5. **advertisement-interval** *seconds*
6. **no shutdown**
7. **show vrrp**
8. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。

ステップ 4	**shutdown**
	
	例：
	`switch(config-if-vrrp)# shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをディセーブルにします。デフォルトではディセーブルです。

ステップ 5	**advertisement-interval** *seconds* **例：** `switch(config-if-vrrp)# advertisement-interval 15`	アドバタイズメント フレームの送信間隔を秒数で設定します。有効値の範囲は 1 ～ 255 であり、デフォルトは 1 秒です。

ステップ 6	**no shutdown**
	
	例：
	`switch(config-if-vrrp)# no shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをイネーブルにします。デフォルトではディセーブルです。

ステップ 7	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報の要約を表示します。
ステップ 8	**copy running-config startup-config** **例：** `switch(config-if-vrrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

プリエンプトのディセーブル化

VRRP グループ メンバのプリエンプトをディセーブルにできます。プリエンプトをディセーブルにすると、プライオリティの高いバックアップ ルータがプライオリティの低いマスター ルータに取って代わることはありません。プリエンプトはデフォルトでイネーブルです。

（注） VRRP グループをイネーブルにしている場合は、ディセーブルにしてから、設定変更を行う必要があります。設定変更の完了後、VRRP グループをイネーブルにします。

準備作業

VRRP 機能がイネーブルになっていることを確認します（「VRRP の設定」[p.18-8] を参照）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **shutdown**
5. **no preempt**
6. **no shutdown**
7. **show vrrp**
8. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。

ステップ 4	**no shutdown** **例：** `switch(config-if-vrrp)# no shutdown`	VRRP グループをイネーブルにします。デフォルトではディセーブルです。
ステップ 5	**no preempt** **例：** `switch(config-if-vrrp)# no preempt`	プリエンプト オプションをディセーブルにして、プライオリティが上位のバックアップが登場しても、マスターが変わらないようにします。
ステップ 6	**no shutdown** **例：** `switch(config-if-vrrp)# no shutdown`	VRRP グループをイネーブルにします。デフォルトではディセーブルです。
ステップ 7	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報の要約を表示します。

	コマンド	目的

ステップ 8	`copy running-config startup-config`
	
	例：
	`switch(config-if-vrrp)# copy running-config startup-config`
		（任意）この設定変更を保存します。

VRRP インターフェイス ステート トラッキングの設定

インターフェイス ステート トラッキングは、デバイスの別のインターフェイスのステートに基づいて、仮想ルータのプライオリティを変更します。トラッキング対象インターフェイスがダウンすると、Cisco NX-OS が仮想ルータにトラッキング プライオリティ値を割り当てます。トラッキング対象インターフェイスがオンラインになると、Cisco NX-OS が仮想ルータに設定されていたプライオリティを復元します（「VRRP プライオリティの設定」[p.18-10] を参照）。

（注） インターフェイス ステート トラッキングを動作させるには、インターフェイス上でプリエンプトをイネーブルにする必要があります。

（注） VRRP グループをイネーブルにしている場合は、ディセーブルにしてから、設定変更を行う必要があります。設定変更の完了後、VRRP グループをイネーブルにします。

準備作業

VRRP 機能がイネーブルになっていることを確認します（「VRRP の設定」[p.18-8] を参照）。

インターフェイス上で IP アドレスが設定されていることを確認します（「IPv4 アドレス指定の設定」[p.2-8] を参照）。

仮想ルータがイネーブルになっていることを確認します（「VRRP グループの設定」[p.18-8] 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）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **vrrp** *number*
4. **shutdown**
5. **track interface** *type number* **priority** *value*
6. **no shutdown**
7. **show vrrp**
8. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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** *interface-type slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		インターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**vrrp** *number*
	
	例：
	`switch(config-if)# vrrp 250`
	`switch(config-if-vrrp)#`
		仮想ルータ グループを作成します。

ステップ 4	**shutdown**
	
	例：
	`switch(config-if-vrrp)# shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをディセーブルにします。デフォルトではディセーブルです。

ステップ 5	**track interface** type *number* **priority** *value* **例：** `switch(config-if-vrrp)# track interface ethernet 2/10 priority 254`	VRRP グループのインターフェイス プライオリティ トラッキングをイネーブルにします。プライオリティの範囲は 1 ～ 254 です。

ステップ 6	**no shutdown**
	
	例：
	`switch(config-if-vrrp)# no shutdown`
	`switch(config-if-vrrp)#`
		VRRP グループをイネーブルにします。デフォルトではディセーブルです。

ステップ 7	**show vrrp** **例：** `switch(config-if-vrrp)# show vrrp`	（任意）VRRP 情報の要約を表示します。
ステップ 8	***copy running-config startup-config*** **例：** `switch(config-if-vrrp)# copy running-config startup-config`	（任意）この設定変更を保存します。

VRRP の設定確認

VRRP の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show vrrp	すべてのグループについて、VRRP ステータスを表示します。
show vrrp vr *group-number*	1 つの VRRP グループについて、VRRP ステータスを表示します。
show vrrp vr *number* **interface** *interface-type port* **configuration**	インターフェイスの仮想ルータ設定を表示します。
show vrrp vr *number* **interface** *interface-type port* **status**	インターフェイスの仮想ルータ ステータスを表示します。

VRRP 統計情報の表示

VRRP 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 vrrp vr *number* **interface** *interface-type port* **statistics**	仮想ルータ情報を表示します。
show vrrp statistics	VRRP の統計情報を表示します。

デバイスのすべてのインターフェイスについて、すべての VRRP 統計情報を消去するには、**clear vrrp statistics** コマンドを使用します。

特定のインターフェイスについて、IPv4 VRRP 統計情報を消去するには、**clear vrrp vr** コマンドを使用します。

特定の IPv4 仮想ルータについて、すべての統計情報を消去するには、**clear vrrp ipv4** コマンドを使用します。

VRRP の設定例

この例では、ルータ A およびルータ B はそれぞれ 3 つの VRRP グループに所属しています。コンフィギュレーションにおいて、各グループのプロパティは次のとおりです。

- グループ 1：
  - バーチャル IP アドレスは 10.1.0.10 です。
  - ルータ A はプライオリティ 120 で、このグループのマスターになります。
  - アドバタイズ インターバルは 3 秒です。
  - プリエンプトはイネーブルです。
- グループ 5：
  - ルータ B はプライオリティ 200 で、このグループのマスターになります。
  - アドバタイズ インターバルは 30 秒です。
  - プリエンプトはイネーブルです。
- グループ 100：
  - ルータ A は、IP アドレスが上位（10.1.0.2）なので、このグループのマスターになります。
  - アドバタイズ インターバルはデフォルトの 1 秒です。
  - プリエンプトはディセーブルです。

ルータ A

```
interface ethernet 1/0
   ip address 10.1.0.2/16
   no shutdown
    vrrp 1
     priority 120
     authentication text cisco
     advertisement-interval 3
     address 10.1.0.10
     no shutdown
    vrrp 5
     priority 100
     advertisement-interval 30
     address 10.1.0.50
     no shutdown
    vrrp 100
     no preempt
     address 10.1.0.100
     no shutdown
```

ルータ B

```
interface ethernet 1/0
 ip address 10.2.0.1/2
 no shutdown
    vrrp 1
     priority 100
     authentication text cisco
     advertisement-interval 3
     address 10.2.0.10
     no shutdown

    vrrp 5
     priority 200
     advertisement-interval 30
     address 10.2.0.50
     no shutdown
    vrrp 100
     no preempt
     address 10.2.0.100
     no shutdown
```

デフォルト設定

表 18-1 に、VRRP パラメータのデフォルト設定を示します。

表 18-1　デフォルトの VRRP パラメータ

パラメータ	デフォルト
アドバタイズ インターバル	1 秒
認証	認証なし
プリエンプト	イネーブル
プライオリティ	100
VRRP 機能	ディセーブル

その他の関連資料

VRRP 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。

- 関連資料（p.18-20）

関連資料

関連項目	マニュアル名
ゲートウェイ ロード バランシング プロトコルの設定	第 16 章「GLBP の設定」
HSRP の設定	第 17 章「HSRP の設定」
VRRP CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』
ハイ アベイラビリティの設定	『*Cisco NX-OS HA Configuration Guide*』

CHAPTER 19

オブジェクト トラッキングの設定

この章では、Cisco NX-OS デバイス上でオブジェクト トラッキングを設定する方法について説明します。

ここでは、次の内容を説明します。

- オブジェクト トラッキング情報（p.19-2）
- オブジェクト トラッキングのライセンス要件（p.19-4）
- オブジェクト トラッキングの前提条件（p.19-4）
- 注意事項および制約事項（p.19-4）
- オブジェクト トラッキングの設定（p.19-5）
- オブジェクト トラッキングの設定確認（p.19-8）
- オブジェクト トラッキングの設定例（p.19-8）
- 関連項目（p.19-8）
- デフォルト設定（p.19-9）
- その他の関連資料（p.19-9）

オブジェクト トラッキング情報

オブジェクト トラッキングを使用すると、インターフェイス ライン プロトコル ステート、IP ルーティング、ルータ到達可能性など、ネットワーク上の特定のオブジェクトをトラッキングし、トラッキング対象オブジェクトのステートが変化したときに対処できます。この機能によって、ネットワークの可用性を向上させ、オブジェクト ステートがダウンになった場合に、回復所要時間を短縮できます。

ここでは、次の内容について説明します。

- オブジェクト トラッキングの概要（p.19-2）
- ハイ アベイラビリティ（p.19-2）
- 仮想化サポート（p.19-3）

オブジェクト トラッキングの概要

オブジェクト トラッキング機能を使用すると、トラッキング対象オブジェクトを作成できます。複数のクライアントでこのオブジェクトを使用し、トラッキング対象オブジェクトが変化したときのクライアント動作を変更できます。複数のクライアントがそれぞれの関心をトラッキング プロセスに登録し、同じオブジェクトをトラッキングしますが、オブジェクトのステートが変化したときに取るアクションはクライアントごとに異なります。

クライアントには次の機能が含まれます。

- HSRP（ホットスタンバイ冗長プロトコル）
- Gateway Load Balancing Protocol（GLBP）
- Embedded Event Manager（EEM）

オブジェクト トラッキングは、トラッキング対象オブジェクトのステータスを監視し、変更があった場合は関係クライアントに伝えます。各トラッキング対象オブジェクトは、一意の番号で識別します。クライアントはこの番号を使用して、トラッキング対象オブジェクトのステートが変化したときに実行するアクションを設定できます。

Cisco NX-OS がトラッキングするオブジェクト タイプは、次のとおりです。

- インターフェイス ライン プロトコル ステート — ライン プロトコル ステートがアップまたはダウンかどうかをトラッキングします。
- インターフェイス IP ルーティング ステート — インターフェイスに IP アドレスが設定されていて、IP ルーティングがイネーブルでアクティブかどうかをトラッキングします。
- IP ルート到達可能性 — ルートが存在していて、ローカル デバイスから到達可能かどうかをトラッキングします。

たとえば、HSRP を設定すると、冗長ルータの 1 つをネットワークの他の部分に接続するインターフェイスのライン プロトコルをトラッキングできます。リンク プロトコルがダウンした場合、影響を受ける HSRP ルータのプライオリティを変更し、よりすぐれたネットワーク接続が得られるバックアップ ルータにスイッチオーバーされるようにできます。

ハイ アベイラビリティ

オブジェクト トラッキングは、ステートフル リスタートを通じてハイ アベイラビリティをサポートします。ステートフル リスタートが実行されるのは、オブジェクト トラッキング プロセスがクラッシュした場合です。オブジェクト トラッキングは、デュアル スーパーバイザ システムでのステートフル スイッチオーバーもサポートします。スイッチオーバー後に Cisco NX-OS が実行コンフィギュレーションを適用します。

オブジェクト トラッキングを使用して、ネットワーク全体の可用性が向上するように、クライアントの動作を変更することもできます。

仮想化サポート

オブジェクト トラッキングは VRF（仮想ルーティングおよびフォワーディング）インスタンスをサポートします。VRF は virtual device context（仮想デバイス コンテキスト ; VDC）内に存在します。特に VDC および VRF を設定しないかぎり、デフォルトで、Cisco NX-OS はユーザにデフォルト VDC およびデフォルト VRF を使用させます。Cisco NX-OS はデフォルトで、デフォルト VRF のオブジェクトのルート到達可能ステートをトラッキングします。別の VRF のオブジェクトをトラッキングする場合は、その VRF のメンバとしてオブジェクトを設定する必要があります（「非デフォルト VRF のオブジェクト トラッキング設定」[p.19-7] を参照）。

詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide*』および第 13 章「レイヤ 3 仮想化の設定」を参照してください。

オブジェクト トラッキングのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
NX-OS	オブジェクト トラッキングにライセンスは不要です。ライセンス パッケージに含まれていない機能は、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

オブジェクト トラッキングの前提条件

オブジェクト トラッキングの前提条件は、次のとおりです。

- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

注意事項および制約事項

オブジェクト トラッキングに関する注意事項および制約事項は、次のとおりです。

- VDC ごとに最大 500 のトラッキング対象オブジェクトをサポートします。
- サポートするのは、IPv4 アドレスに限られます。
- イーサネット、サブインターフェイス、ポート チャネル、ループバック インターフェイス、および VLAN インターフェイスをサポートします。
- 1 つの HSRP グループまたは GLBP グループでサポートするトラッキング対象オブジェクトは 1 つです。

オブジェクト トラッキングの設定

ここでは、次の内容について説明します。

- インターフェイスのオブジェクト トラッキング設定（p.19-5）
- ルート到達可能性のオブジェクト トラッキング設定（p.19-6）
- 非デフォルト VRF のオブジェクト トラッキング設定（p.19-7）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

インターフェイスのオブジェクト トラッキング設定

インターフェイスのライン プロトコルまたは IP ルーティングのステートをトラッキングするように Cisco NX-OS を設定できます。

準備作業

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）。

手順概要

1. **config t**
2. **track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
3. **show track** [*object-id*]
4. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**track** *object-id* **interface** *interface-type number* {**ip routing** | **line-protocol**}
	
	例：
	`switch(config)# track 1 interface ethernet 1/2 line-protocol`
	`switch(config-track#`
		インターフェイスの追跡対象オブジェクトを作成し、トラッキング コンフィギュレーション モードを開始します。*object-id* の範囲は 1 ～ 500 です。

ステップ 3	***show* track** [*object-id*] **例：** `switch(config-track)# show track 1`	（任意）オブジェクト トラッキング情報を表示します。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-track)# copy running-config startup-config`	（任意）この設定変更を保存します。

Ethernet 1/2 上でライン プロトコル ステートのオブジェクト トラッキングを設定する例を示します。

```
switch# config t
switch(config)# track 1 interface ethernet 1/2 line-protocol
switch(config-track)# copy running-config startup-config
```

ルート到達可能性のオブジェクト トラッキング設定

IP ルートの存在および到達可能性をトラッキングするように Cisco NX-OS を設定できます。

準備作業

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）。

手順概要

1. **config t**
2. **track** *object-id* **ip route** *ip-prefix/length* **reachability**
3. **show track** [*object-id*]
4. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**track** *object-id* **ip route** *ip-prefix/length* **reachability**
	
	例：
	`switch(config)# track 2 ip route 192.0.2.0/8 reachability`
	`switch(config-track)#`
		ルートの追跡対象オブジェクトを作成し、トラッキング コンフィギュレーション モードを開始します。*object-id* の範囲は 1 ～ 500 です。

ステップ 3	***show* track** [*object-id*] **例：** `switch(config-track)# show track 1`	（任意）オブジェクト トラッキング情報を表示します。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-track)# copy running-config startup-config`	（任意）この設定変更を保存します。

デフォルト VRF で、ルートのオブジェクト トラッキングを設定する例を示します。

```
switch# config t
switch(config)# track 2 ip route 192.0.2.0/8 reachability
switch(config-track)# copy running-config startup-config
```

非デフォルト VRF のオブジェクト トラッキング設定

特定の VRF でオブジェクトをトラッキングするように Cisco NX-OS を設定できます。

準備作業

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）。

手順概要

1. **config t**
2. **track** *object-id* **ip route** *ip-prefix/length* **reachability**
3. **vrf member** *vrf-name*
4. **show track** [*object-id*]
5. ***copy running-config startup-config***

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**track** *object-id* **ip route** *ip-prefix/length* **reachability**
	
	例：
	`switch(config)# track 3 ip route 209.165.201.0/8 reachability`
	`switch(config-track)#`
		ルートの追跡対象オブジェクトを作成し、トラッキング コンフィギュレーション モードを開始します。*object-id* の範囲は 1 ～ 500 です。

ステップ 3	***vrf member*** *vrf-name* **例：** `switch(config-track)# vrf member Red`	設定されたオブジェクトのトラッキングに使用する VRF を設定します。
ステップ 4	***show*** **track** [*object-id*] **例：** `switch(config-track)# show track 3`	（任意）オブジェクト トラッキング情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-track)# copy running-config startup-config`	（任意）この設定変更を保存します。

ルートのオブジェクト トラッキングを設定し、VRF Red を使用して、そのオブジェクトの到達可能性情報を調べる例を示します。

```
switch# config t
switch(config)# track 2 ip route 209.165.201.0/8 reachability
switch(config-track)# vrf member Red
switch(config-track)# copy running-config startup-config
```

トラッキング対象オブジェクト 2 を変更し、VRF Red の代わりに VRF Blue を使用して、そのオブジェクトの到達可能性情報を調べる例を示します。

```
switch# config t
switch(config)# track 2
switch(config-track)# vrf member Blue
switch(config-track)# copy running-config startup-config
```

オブジェクト トラッキングの設定確認

オブジェクト トラッキングの設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show track [*object-id*] [**brief**]	1 つまたは複数のオブジェクトについて、オブジェクト トラッキング情報を表示します。
show track [*object-id*] **interface** [**brief**]	インターフェイスベースのオブジェクト トラッキング情報を表示します。
show track [*object-id*] **ip route** [**brief**]	IP ルートベースのオブジェクト トラッキング情報を表示します。

オブジェクト トラッキングの設定例

ルート到達可能性のオブジェクト トラッキングを設定し、VRF Red を使用して、そのルートの到達可能性情報を調べる例を示します。

```
switch# config t
switch(config)# track 2 ip route 209.165.201.0/8 reachability
switch(config-track)# vrf member Red
switch(config-track)# copy running-config startup-config
```

関連項目

オブジェクト トラッキングの関連情報については、次の項目を参照してください。

- 第 13 章「レイヤ 3 仮想化の設定」
- 第 16 章「GLBP の設定」
- 第 17 章「HSRP の設定」

デフォルト設定

表 19-1 に、オブジェクト トラッキング パラメータのデフォルト設定を示します。

表 19-1　デフォルトのオブジェクト トラッキング パラメータ

パラメータ	デフォルト
トラック対象オブジェクト VRF	デフォルト VRF のメンバ

その他の関連資料

オブジェクト トラッキング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。

- 関連資料（p.19-9）
- 規格（p.19-9）

関連資料

関連項目	マニュアル名
オブジェクト トラッキング CLI コマンド	『*Cisco NX-OS Unicast Routing Command Reference*』
Embedded Event Manager の設定	『*Cisco NX-OS System Management Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

APPENDIX **A**

Cisco NX-OS Unicast Features Release 4.x がサポートする IETF RFC

この付録は、Cisco NX-OS Release 4.x がサポートする IETF RFC の一覧です。

BGP の FRC

RFC	タイトル
RFC 1997	『*BGP Communities Attribute*』
RFC 2385	『*Protection of BGP Sessions via the TCP MD5 Signature Option*』
RFC 2439	『*BGP Route Flap Damping*』
RFC 2519	『*A Framework for Inter-Domain Route Aggregation*』
RFC 2545	『*Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*』
RFC 2858	『*Multiprotocol Extensions for BGP-4*』
RFC 3065	『*Autonomous System Confederations for BGP*』
RFC 3392	『*Capabilities Advertisement with BGP-4*』
RFC 4271	『*A Border Gateway Protocol 4 (BGP-4)*』
RFC 4273	『*Definitions of Managed Objects for BGP-4*』
RFC 4456	『*BGP Route Reflection:An Alternative to Full Mesh Internal BGP (IBGP)*』
RFC 4486	『*Subcodes for BGP Cease Notification Message*』
RFC 4724	『*Graceful Restart Mechanism for BGP*』
RFC 4893	『*BGP Support for Four-octet AS Number Space*』
draft-ietf-idr-avoid-transition-05.txt	『*Avoid BGP Best Path Transitions from One External to Another*』
draft-ietf-idr-bgp4-mib-15.txt	『*BGP4-MIB*』
draft-kato-bgp-ipv6-link-local-00.txt	『*BGP4+ Peering Using IPv6 Link-local Address*』

First-Hop Redundancy Protocol の RFC

RFC	タイトル
RFC 2281	『*Hot Standby Redundancy Protocol*』
RFC 3768	『*Virtual Router Redundancy Protocol*』

IP サービスの RFC

RFC	タイトル
RFC 786	『*UDP*』
RFC 791	『*IP*』
RFC 792	『*ICMP*』
RFC 793	『*TCP*』
RFC 826	『*ARP*』
RFC 1027	『*Proxy ARP*』
RFC 1591	『*DNS Client*』
RFC 1812	『*IPv4 routers*』

IPv6 の RFC

RFC	タイトル
RFC 1981	『*Path MTU Discovery for IP version 6*』
RFC 2373	『*IP Version 6 Addressing Architecture*』
RFC 2374	『*An Aggregatable Global Unicast Address Format*』
RFC 2460	『*Internet Protocol, Version 6 (IPv6) Specification*』
RFC 2461	『*Neighbor Discovery for IP Version 6 (IPv6)*』
RFC 2462	『 *IPv6 Stateless Address Autoconfiguration*』
RFC 2463	『*Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*』
RFC 2464	『*Transmission of IPv6 Packets over Ethernet Networks*』
RFC 2467	『*Transmission of IPv6 Packets over FDDI Networks*』
RFC 2472	『*IP Version 6 over PPP*』
RFC 2492	『*IPv6 over ATM Networks*』
RFC 2590	『*Transmission of IPv6 Packets over Frame Relay Networks Specification*』
RFC 3152	『*Delegation of IP6.ARPA*』
RFC 3162	『*RADIUS and IPv6*』
RFC 3513	『*Internet Protocol Version 6 (IPv6) Addressing Architecture*』
RFC 3596	『*DNS Extensions to Support IP version 6*』
RFC 4193	『*Unique Local IPv6 Unicast Addresses*』

IS-IS の RFC

RFC	タイトル
RFC 1142	『*OSI 10589 Intermediate system to intermediate system intra-domain routing exchange protocol*』
RFC 1195	『*Use of OSI IS-IS for routng in TCP/IP and dual environment*』
RFC 2763	『*Dynamic Hostname Exchange Mechanism for IS-IS*』
RFC 2966	『*Domain-wide Prefix Distribution with Two-Level IS-IS*』
RFC 2972	『*IS-IS Mesh Groups*』
RFC 3273	『*Three-Way Handshake for IS-IS Point-to-Point Adjacencies*』
RFC 3277	『*IS-IS Transient Blackhole Avoidance*』
RFC 3567	『*IS-IS Cryptographic Authentication*』
RFC 3847	『*Restart Signaling for IS-IS*』
draft-ietf-isis-igp-p2p-over-lan-06.txt	『*Internet Draft Point-to-point operation over LAN in link-state routing protocols*』

OSPF の RFC

RFC	タイトル
RFC 2328	『*OSPF Version 2*』
RFC 2740	『*OSPF for IPv6*』
RFC 3623	『*Graceful OSPF Restart*』
RFC 3101	『*The OSPF Not-So-Stubby Area (NSSA) Option*』
RFC 2370	『*The OSPF Opaque LSA Option*』
RFC 3137	『*OSPF Stub Router Advertisement*』
draft-ietf-ospf-ospfv3-graceful-restart-04.txt	『*OSPFv3 Graceful Restart*』

RIP の RFC

RFC	タイトル
RFC 2453	『*RIP Version 2*』
RFC 2080	『*RIPng for IPv6*』
RFC 2082	『*RIP-2 MD5 Authentication*』

APPENDIX **B**

Cisco NX-OS レイヤ 3 ユニキャスト機能、リリース 4.x の設定の制限

DC3 がサポートする機能には、設定の最大制限があります。一部の機能では、設定でサポートされる制限が最大制限より小さいことが確認されています。表 B-1 は、シスコで確認済みの、DC3 Release 4.x を実行するスイッチの制限および最大制限の一覧です。

表 B-1　Cisco NX-OS リリース 4.x 設定の制限

機能	確認済みの制限	最大制限
BGP		300 のピア
		ピアごとに 150,000 のプレフィクス
		合計 2,000 万のパス
GLBP		1024 の GLBP グループ
HSRP		2048 の HSRP グループ
		インターフェイスごとに 20 の HSRP グループ
OSPF		200 のインターフェイス
		1000 台のルータ
		300 件の隣接関係
		200,000 の LSA

GLOSSARY

A

ABR エリア境界ルータを参照してください。

ARP Address Resolution Protocol; アドレス解決プロトコル。ARP は既知の IPv4 アドレスに対応する MAC アドレスを検出します。

AS 自律システムを参照してください。

ASBR 自律システム境界ルータを参照してください。

AVF アクティブ バーチャル フォワーダ。特定のバーチャル MAC アドレスにトラフィックを転送するために選定された、GLBP グループ内のゲートウェイ。

AVG アクティブ バーチャル ゲートウェイ。アクティブ バーチャル ゲートウェイとして選択され、プロトコルの動作を担当する、GLBP グループ内の 1 つのバーチャル ゲートウェイ。

B

BDR バックアップ代表ルータ。マルチアクセス OSPF ネットワークにおいて、代表ルータで障害が発生した場合に、バックアップとして動作するように選定されたルータ。すべてのネイバーは、代表ルータと同様、バックアップ代表ルータ（BDR）とも隣接関係を形成します。

BGP ボーダー ゲートウェイ プロトコル。BGP はドメイン間またはエクステリア ゲートウェイ プロトコルです。

BGP ピア ローカル BGP スピーカと隣接関係が確立されている、リモート BGP スピーカ。

BGP スピーカ BGP 対応ルータ。

D

DHCP 動的ホスト制御プロトコル。

Diffusing アップデート アルゴリズム DUAL を参照してください。

DNS クライアント ドメイン ネーム システム クライアント。DNS サーバと通信し、ホスト名を IP アドレスに変換します。

DR 代表ルータ。マルチアクセス OSPF ネットワークにおいて、すべての隣接ネイバーに代わって LSA を送信するように選定されたルータ。すべてのネイバーは、代表ルータおよびバックアップ代表ルータとだけ隣接関係を確立します。

DUAL Diffusing アップデート アルゴリズム。宛先への最適ルートを選択するための EIGRP アルゴリズム。

E

eBGP 外部 BGP（ボーダー ゲートウェイ プロトコル）。外部システム間で動作します。

EIGRP Enhanced IGRP。Diffusing アップデート アルゴリズムを使用して高速コンバージェンスを実現し、帯域幅の使用率を最小限に抑える、シスコのルーティング プロトコルです。

F

FIB 転送情報ベース。パケットごとにレイヤ 3 フォワーディングを決定するために使用される、各モジュール上のフォワーディング テーブル。

G

GLBP Gateway Load Balancing Protocol。エンド ホストにハイ アベイラビリティ機能を提供する、シスコ独自のプロトコル。

H

hello インターバル OSPF または EIGRP ルータが hello パケットを送信する、設定可能な間隔。

hello パケット OSPF または IS-IS がネイバー検出のために使用する、特殊なメッセージ。確立されたネイバー間のキープ アライブ メッセージとしても機能します。

HSRP

I

iBGP 内部 BGP（ボーダー ゲートウェイ プロトコル）。自律システム内で動作します。

ICMP

IETF RFC インターネット技術特別調査委員会コメント要求。

IGP インテリア ゲートウェイ プロトコル。同じ自律システム内のルータ間で使用されます。

IP トンネル

IPv4 インターネット プロトコル バージョン 4。

IPv6 インターネット プロトコル バージョン 6。

IS-IS Intermediate System to Intermediate System。ISO インテリア ゲートウェイ プロトコル。

L

LSA リンクステート アドバタイズメント。リンクの動作状態、リンク コスト、およびその他の OSPF ネイバー情報を共有するための OSPF メッセージ。

M

MD5 認証ダイジェスト 認証鍵および元のメッセージに基づいて計算される、暗号構築物。メッセージとともに宛先に送信されます。宛先は送信側の正統性を判別し、送信中にメッセージが改ざんされていない保証を得られます。

MTU Maximum Transmission Unit（最大伝送ユニット）。ネットワーク リンクで分割しないで送信できる、最大パケット サイズ。

N

NDP Neighbor Discovery Protocol（近隣探索プロトコル）。IPv6 アドレスに関連付けられた MAC アドレスを検索するために、IPv6 で使用されるプロトコル。

NSSA Not-So-Stubby-Area。OSPF エリアにおいて、AS External LSA を制限します。

O

OSPF Open Shortest Path First。IETF リンクステート プロトコル。OSPFv2 は IPv4 を、OSPFv3 は IPv6 をサポートします。

R

RIB ルーティング情報ベース。直接接続ルート、スタティック ルート、およびダイナミック ユニキャスト ルーティング プロトコルから学習したルートからなる、ルーティング テーブルを維持します。

Route Policy Manager ルート マップおよびポリシーベース ルーティングを制御するプロセス。

S

SPF アルゴリズム 最短パス優先アルゴリズム。ネットワーク経由で特定の宛先までの最短ルートを判別するために、OSPF で使用されるダイクストラ アルゴリズム。

SVI スイッチ仮想インターフェイス。

U

U6FIB ユニキャスト IPv6 転送情報ベース。

U6RIB ユニキャスト IPv6 ルーティング情報ベース。すべてのルーティング プロトコルから情報を集め、各モジュールの転送情報ベースをアップデートする、ユニキャスト ルーティング テーブル。

UFIB ユニキャスト IPv4 転送情報ベース。

URIB ユニキャスト IPv4 ルーティング情報ベース。すべてのルーティング プロトコルから情報を集め、各モジュールの転送情報ベースをアップデートする、ユニキャスト ルーティング テーブル。

V

VDC Virtual Device Context（仮想デバイス コンテキスト）。物理システムを安全で独立した論理システムに分割するために使用されます。

VRF Virtual Routing and Forwarding（仮想ルーティングおよびフォワーディング）。システム内部で別個の独立したレイヤ 3 エンティティを作成するための方法。

VRRP Virtual Router Redundancy Protocol（仮想ルータ冗長プロトコル）。

あ

アトリビュート BGP UPDATE メッセージで送信される、ルートのプロパティ。これらのアトリビュートには、アドバタイズされた宛先へのパスとともに、ベスト パス選択プロセスを変更する、設定可能なオプションが含まれます。

アドレス ファミリ ルーティング プロトコルがサポートする特定のネットワーク アドレッシング タイプ。IPv4 ユニキャスト、IPv4 マルチキャストなど。

い

インスタンス 独立した設定可能なエンティティ。通常はプロトコル。

え

エリア OSPF ドメイン内の独立したサブドメインを形成する、ルータおよびリンクからなる論理区分。LSA フラッディングはエリア内に封じ込められます。

エリア境界ルータ ある OSPF エリアを別の OSPF エリアに接続するルータ。

か

仮想化 物理エンティティを複数の独立した論理エンティティとして動作させる 1 つの方法。

管理ディスタンス ルーティング情報源の信頼性に関する格付け。通常、値が大きいほど、信頼性の格付けが下がります。

き

キーチェーン管理 認証鍵を制御する方法の 1 つ。『*Cisco NX-OS Security Configuration Guide*』を参照してください。

キープアライブ ルーティング ペア間の通信を確認して維持するために、ピア間で送信される特殊なメッセージ。

く

グレースフル リスタート ルーティング プロトコルのリブート時に、ルータがデータ転送パスにとどまるようにする機能。

け

ゲートウェイ LAN からネットワークのその他の部分にレイヤ 3 トラフィックを転送するスイッチまたはルータ。

こ

コンバージェンス 収束を参照してください。

さ

再配布 あるルーティング プロトコルが別のルーティング プロトコルからルート情報を受け入れ、ローカル自律システムでそれをアドバタイズします。

し

収束 ネットワーク内のすべてのルータが同じルーティング情報を得るポイント。

自律システム 単一のテクニカル アドミニストレーション エンティティによって制御されるネットワーク。

自律システム境界ルータ OSPF 自律システムを外部の自律システムに接続するルータ。

信頼性 各ネットワーク リンクに頼れるかどうか（通常は、ビット誤り率で表します）。

す

スタティック ルート 手動で設定されたルート。

スタブ エリア AS External（type 5）LSA を認めない OSPF エリア。

スタブ ルータ メイン ネットワークへの直接接続がなく、既知のリモート ルータを使用してメイン ネットワークにルーティングされるルータ。

スプリット ホライズン ルータが自身のルート アップデートを見ないように、ルートの学習元になったインターフェイスには、学習したルートをアドバタイズしません。

た

帯域幅 リンクの使用可能なトラフィック容量。

代表ルータ DR を参照してください。

ち

遅延 システムから宛先にインターネットワークを介してパケットを転送するために必要な時間。

つ

通信コスト リンクを介してルーティングする運用コストの算定基準。

て

ディスタンス ベクトル 距離（宛先までのホップ数など）および方向（ネクストホップ ルータなど）によってルートを定義し、さらに直接接続されたネイバー ルータにブロードキャストします。

デッド インターバル その範囲内で OSPF ルータが OSPF ネイバーから hello パケットを受信しなければならない時間。デッド インターバルは通常、hello インターバルの倍数です。hello パケットを受信しなかった場合、ネイバーの隣接関係は削除されます。

デフォルト ゲートウェイ あらゆるルーティング不能パケットの送信先となるルータ。ラスト リゾート ルータともいいます。

ね

ネクスト ホップ 宛先アドレスまでの間で、パケットの次の送信先になるルータ。

ネットワーク レイヤ到達可能性情報 BGP network layer reachability information（NRLI）。アドバタイズ側 BGP ピアから到達可能な、ネットワーク IP アドレスおよびネットワークに対応するネットワーク マスクのリストが含まれます。

は

ハイ アベイラビリティ コンポーネントで障害が発生したときに、システムまたはコンポーネントがネットワークの停止を制限または回避する能力。

パス長 送信元から宛先までのルーティングにおいて、パケットが経験するすべてのリンク コストおよびホップ カウントの合計。

バックアップ代表ルータ BDR を参照してください。

ふ

フィージブル サクセッサ 現在のフィージブル ディスタンスより短い宛先までの距離をアドバタイズした、EIGRP のネイバー。

フィージブル ディスタンス EIGRP で計算された、ネットワークの宛先までの最短距離。フィージブル ディスタンスは、ネイバーがアドバタイズした距離に、そのネイバーへのリンク コストを加えた合計です。

負荷 ルータなどのネットワーク リソースが使用中になっている程度。

ほ

ポイズン リバースを指定したスプリット ホライズン ルータが自身のルート アップデートを見ないように、インターフェイスから学習したルートを到達不能として設定し、ルートの学習元になったインターフェイスには、学習したルートをアドバタイズしません。

ホップ カウント ルート上で経由できるルータの数。RIP で使用されます。

ポリシーベース ルーティング パケットに選択されたルートをルート マップを使用して変更する方式。

ホールド タイム BGP において、UPDATE または KEEPALIVE メッセージの間隔として許容される最大時間限度。この時間を超えると、BGP ピア間の TCP 接続が終了します。

EIGRP では、EIGRP Hello メッセージの間隔として許容される最大時間。この時間を超えると、ネイバーが到達不能として宣言されます。

め

メッセージ ダイジェスト 共有パスワードを使用するメッセージに適用される、一方向ハッシュ。メッセージを認証し、メッセージが送信中に変更されていないことを保証するために、メッセージに付加されます。

メトリック パス帯域幅など、宛先への最適パスを決定するためにルーティング アルゴリズムが使用する、標準の測定単位。

り

リライアブル トランスポート プロトコル すべてのネイバーに EIGRP パケットを保証付きで順序正しく配信する役目を担います。

リンク コスト OSPF インターフェイス上で設定された、最短パス優先計算に含まれる任意の値。

リンクステート ネイバー ルータとのリンク、リンク コストに関する情報の共有。

リンクステート アドバタイズメント LSA を参照してください。

リンクステート データベース 受信したすべての LSA に関する OSPF データベース。OSPF ではこのデータベースを使用して、ネットワーク上の各宛先に最適なパスを計算します。

リンクステート リフレッシュ すべての OSPF ルータが同じ情報を持っていることを保証するために、OSPF が LSA をネットワークにフラッディングする時間。

隣接関係 コンフィギュレーションに互換性があり、リンクステート データベースが同期している 2 つの OSPF ルータ。

る

ルータ ID ルーティング プロトコルで使用される一意の識別情報。手動で設定しなかった場合は、ルーティング プロトコルがシステムに設定されている最大の IP アドレスを選択します。

ルーティング情報ベース RIB を参照してください。

ルート集約 ルート テーブル内の関連した一連の固有ルートを汎用性の高いルートに置き換えるプロセス。

ルート マップ 一致基準に基づいてルートまたはパケットをマッピングし、任意で設定基準に基づいてルートまたはパケットを変更するために使用される構築物。ルート再配布およびポリシー ベース ルーティングで使用されます。

ろ

ロード バランシング　所定の宛先に複数のパスを使用してネットワーク トラフィックを配信すること。

INDEX

A

address family コマンド　8-9, 9-8
aggregate-address コマンド　10-27
ARP
　Inverse ARP　2-4
　Reverse ARP（RARP）　2-5
　処理（図）　2-3
　説明　2-3
　プロキシ ARP　2-5
　ローカル プロキシ ARP　2-6

B

BGP
　パス選択　9-3
bgp グローバル アドレス ファミリ サブモード　9-8
　aggregate-address コマンド　10-27
bgp ネイバー アドレス ファミリ サブモード　9-9
　route-reflector-client コマンド　10-24, 10-25
bgp ネイバー サブモード　9-9
bgp ルータ サブモード　8-9, 9-8

D

DNS
　IP version 6 アドレス　3-12

G

GLBP
　重み付けおよびトラッキングの設定　16-11
　カスタマイズ　16-13
　機能のイネーブル化　16-8
　グループのイネーブル化　16-14
　説明　16-2
　認証の設定　16-8
　ロード バランシングの設定　16-10

I

ICMP
　説明　2-6
ICMPv6
　説明　3-13
　パケット ヘッダーのフォーマット（図）　3-13
Inverse ARP
　説明　2-4
IP version 4
　Address Resolution Protocol（ARP; アドレス解決プロトコル）　2-3
　Internet Control Message Protocol（ICMP; インターネット制御メッセージ プロトコル）　2-6
　サブネット マスク　2-2
　制約事項　2-7
　説明　2-2
　前提条件　2-7
　注意事項　2-7
　デフォルト設定　2-13
IP version 6
　DNS レコード タイプ　3-12
　Internet Control Message Protocol（ICMP; インターネット制御メッセージ プロトコル）　3-13
　アドレス フォーマット　3-2
　エニーキャスト アドレス　3-7
　近隣探索　3-14
　制約事項　3-19
　説明　3-2
　前提条件　3-19
　注意事項　3-19
　デフォルト設定　3-25
　ネイバー リダイレクト メッセージ　3-17
　ネイバー送信要求メッセージ　3-14
　パケット ヘッダー　3-10
　パス最大伝送ユニット（MTU）探索　3-13
　マルチキャスト アドレス　3-8
　ユニキャスト アドレス　3-3
　ルータ アドバタイズメント メッセージ　3-16

IPv4 互換 IPv6 アドレス、ユニキャストを参照
IS-IS アドレス ファミリ サブモード　8-9
IS-IS（Intermediate System-to-Intermediate System）
　説明　8-1

M

maximum-prefix コマンド　7-18, 7-22, 7-23
mgmt0 インターフェイス
　デフォルト設定　4-9, 5-43, 6-38, 7-27, 8-28, 9-17, 10-36, 11-18, 13-13, 14-14, 15-9, 16-17, 17-18, 18-20

N

neighbor address family コマンド　9-9
neighbor コマンド　9-9

O

OSPFv2（Open Shortest Path First Version 2）
　説明　5-1, 6-1, 7-1, 8-1, 9-1

R

Really Simple SyndicationRSS を参照
redistribute コマンド　7-17, 7-19, 7-20
Reverse ARP
　説明　2-5
router bgp コマンド　8-9, 9-8
router eigrp address family コンフィギュレーション サブモード
　maximum-prefix コマンド　7-18, 7-22, 7-23
　redistribute コマンド　7-17, 7-19, 7-20
route-reflector-client コマンド　10-24, 10-25
RSS
　マニュアル フィード　xxv

あ

アドレス フォーマット
　IP version 6　3-2

い

インターフェイス
　オブジェクト トラッキングの設定　19-5
　デフォルト設定　4-9, 5-43, 6-38, 7-27, 8-28, 9-17, 10-36, 11-18, 13-13, 14-14, 15-9, 16-17, 17-18, 18-20

え

エニーキャスト
　IP version 6 アドレス　3-7

お

オブジェクト トラッキング
　GLBP の設定　16-11
　インターフェイスでの設定　19-5
　説明　19-2
　ルート到達可能性の設定　19-6
オブジェクト追跡
　説明　1-18

か

管理インターフェイス
　デフォルト設定　4-9, 5-43, 6-38, 7-27, 8-28, 9-17, 10-36, 11-18, 13-13, 14-14, 15-9, 16-17, 17-18, 18-20
管理ディスタンス
　スタティック ルーティング　12-2
関連資料　xxiv

き

近隣探索
　説明　3-14

け

計算
　負荷　1-5

さ

サービス
リクエスト xxv
サイトローカル アドレス、ユニキャストを参照

し

集約可能グローバル アドレス、ユニキャストを参照
資料
その他の関連資料 xxiv

す

スタティック ルーティング
管理ディスタンス 12-2
制約事項 12-4
説明 12-2
前提条件 12-4
注意事項 12-4
デフォルト設定 12-8

せ

制限
説明（表） B-1
設定の制限
説明（表） B-1

て

ディスタンス ベクトル ルーティング アルゴリズム 1-10
デバイス
負荷 1-5

ね

ネイバー リダイレクト メッセージ
説明 3-17
ネイバー送信要求メッセージ
説明 3-14

は

パス MTU 探索
説明 3-13
パス選択
BGP 9-3

ひ

比較
リンクステート アルゴリズムおよびディスタンス ベクトル ルーティング アルゴリズム 1-10

ふ

ファイバ チャネル インターフェイス
デフォルト設定 4-9, 5-43, 6-38, 7-27, 8-28, 9-17, 10-36, 11-18, 13-13, 14-14, 15-9, 16-17, 17-18, 18-20
負荷 1-5
プロキシ ARP
説明 2-5

へ

ベスト パスの選択（BGP） 9-3
ベスト パス（BGP）
選択 9-3

ま

マニュアル
更新 xxv
マルチキャスト
IP version 6 アドレス 3-8

ゆ

ユニーク ローカル アドレス、ユニキャストを参照
ユニキャスト
IP version 6 アドレス 3-3

り

リンクローカル アドレス、ユニキャストを参照

リンクステート ルーティング アルゴリズム　1-10

る

ルータ アドバタイズメント メッセージ
　説明　3-16
ルーティング アルゴリズム
　ディスタンス ベクトル　1-10
　リンクステート　1-10

ろ

ローカル プロキシ ARP
　説明　2-6