



## OSPF の実装

Open Shortest Path First (OSPF) は、Internet Engineering Task Force (IETF) の OSPF ワーキンググループによって開発された内部ゲートウェイプロトコル (IGP) です。OSPF は特に IP ネットワーク向けに設計されており、IP サブネット化、および外部から取得したルーティング情報のタグgingをサポートしています。また、OSPF では、パケットの送受信時のパケット認証も可能になります。

OSPF Version 3 (OSPFv3) は OSPF Version 2 を拡張し、IPv6 ルーティングプレフィックスのサポートを提供します。

このモジュールでは、ソフトウェアで OSPF の両方のバージョンを実装するために必要な概念と作業について説明します。特に記載のないかぎり、用語「OSPF」は両方のバージョンのルーティングプロトコルを意味します。



- (注)
1. VPNv4、VPNv6 および VPN ルーティング/転送 (VRF) のアドレスファミリーは、今後のリリースでサポートされる予定です。
  2. GTSM TTL セキュリティはサポートされていません。

- [OSPF の実装の前提条件 \(2 ページ\)](#)
- [OSPF のイネーブル化 \(2 ページ\)](#)
- [OSPF の設定と動作の確認 \(5 ページ\)](#)
- [スタブエリア \(7 ページ\)](#)
- [OSPF のネイバーおよび隣接関係 \(10 ページ\)](#)
- [認証方法 \(14 ページ\)](#)
- [OSPF に同じ LSA が生成される頻度または受け入れられる頻度の制御 \(18 ページ\)](#)
- [OSPF の仮想リンクおよび中継エリア \(19 ページ\)](#)
- [OSPF ABR でのサブネットワーク LSA の集約 \(25 ページ\)](#)
- [OSPF のルート再配布 \(27 ページ\)](#)
- [OSPF Version 2 のノンストップフォワーディング \(30 ページ\)](#)
- [OSPF Shortest Path First スロットリング \(32 ページ\)](#)
- [OSPFv3 のグレースフルリスタート \(35 ページ\)](#)

- OSPF Version 2 のウォーム スタンバイとノンストップルーティング (38 ページ)
- OSPF Version 3 のウォーム スタンバイとノンストップルーティング (39 ページ)
- OSPFv2OSPF SPF プレフィックスのプライオリティ設定 (40 ページ)
- プロバイダー エッジからカスタマー エッジ (PE-CE) プロトコルとしての OSPF の設定 (45 ページ)
- 複数の OSPF インスタンスの作成 (OSPF プロセスおよび VRF) (47 ページ)
- OSPF のラベル配布プロトコル IGP 自動設定 (48 ページ)
- OSPF 認証のメッセージダイジェスト管理 (51 ページ)
- OSPF の GTSM TTL セキュリティ メカニズム (55 ページ)
- OSPF の参照 (58 ページ)

## OSPF の実装の前提条件

以下は、OSPF を実装するための前提条件です。

- OSPFv3 の設定作業では、IPv6 のアドレッシングと基本概念について精通していることを前提としています。IPv6 ルーティングとアドレッシングについては、『*Cisco IP Addresses and Services Configuration Guide IP Addresses and Services Configuration Guide for Cisco NCS 560 Series Routers*』の「*Implementing Network Stack IPv4 and IPv6*」を参照してください。
- インターフェイスで OSPFv3 をイネーブルにする前に、次の手順を実行する必要があります。
  - ご使用の IPv6 ネットワークに対する OSPF ネットワーク戦略と計画を完成させます。たとえば、複数のエリアが必要かどうかを決定します。
  - インターフェイスで IPv6 をイネーブルにします。
- 認証 (IPセキュリティ) の設定はオプションの作業です。認証を設定する場合、プレーンテキスト認証と Message Digest 5 (MD5) 認証のどちらを設定するかについて、また、認証をエリア全体に適用するか特定のインターフェイスに適用するかについて最初に決定する必要があります。

## OSPF のイネーブル化

このタスクでは、1つのルータ ID で OSPF プロセスをイネーブルにするルータで、最小の OSPF コンフィギュレーションを実行し、バックボーンまたはバックボーン以外のエリアを設定し、OSPF を実行する 1 つ以上のインターフェイスを割り当てる方法を説明します。

### 始める前に

IP アドレスを設定する前に OSPF を設定することはできますが、IP アドレスが設定されるまで、OSPF はルーティングされません。

## 手順

**ステップ1 configure**

**ステップ2** 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。

または

指定したルーティングプロセスに OSPFv3 ルーティングをイネーブルにし、**router ospfv3** コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

**ステップ3 router-id { router-id }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IP アドレスをルータ ID として使用することを推奨します。

**ステップ4 area area-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 0
```

エリア コンフィギュレーション モードを開始し、OSPF プロセスのエリアを設定します。

- バックボーンエリアには 0 のエリア ID があります。
- バックボーン以外のエリアにはゼロではないエリア ID があります。
- *area-id* 引数は、**area 1000** や **area 0.0.3.232** など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1 つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

**ステップ5 interface type interface-path-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、ステップ 4 で設定したエリアのインターフェイスを 1 つ以上関連付けます。

**ステップ 6** OSPF を使用する各インターフェイスでステップ 5 を繰り返します。

—

**ステップ 7** `log adjacency changes [ detail ] [ enable | disable ]`

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# log adjacency changes detail
```

(任意) ネイバー変更の通知を要求します。

- デフォルトでは、この機能はイネーブルです。
- ネイバー変更によって生成されたメッセージは通知と見なされます。このメッセージは `logging console` コマンドで重大度レベル 5 に分類されます。 `logging console` コマンドではどの重大度レベルのメッセージをコンソールに送信するかを制御します。デフォルトでは、すべての重大度レベルのメッセージが送信されます。

**ステップ 8** `commit`

### OSPF のイネーブル化：例

OSPF エリアは明示的に設定する必要があり、エリア コンフィギュレーション モードで設定されたインターフェイスは、そのエリアに明示的にバインドされています。この例では、インターフェイス 10.1.2.0/24 がエリア 0 に、インターフェイス 10.1.3.0/24 がエリア 1 にバインドされています。

```
interface TenGigE 0/0/0/0
 ip address 10.1.2.1 255.255.255.0
 negotiation auto
!
interface TenGigE 0/0/0/1
 ip address 10.1.3.1 255.255.255.0
 negotiation auto
!
router ospf 1
 router-id 10.2.3.4
 area 0
  interface TenGigE 0/0/0/0
  !
 area 1
  interface TenGigE 0/0/0/1
  !
!
```

# OSPF の設定と動作の確認

このタスクでは、OSPF の設定および操作を確認する方法について説明します。

## 手順

**ステップ 1** `show { ospf | ospfv3 } [ process-name ]`

例 :

```
RP/0/RP0/cpu 0: router# show ospf group1
```

(任意) OSPF ルーティング プロセスに関する一般情報を表示します。

**ステップ 2** `show { ospf | ospfv3 } [ process-name ] border-routers [ router-id ]`

例 :

```
RP/0/RP0/cpu 0: router# show ospf group1 border-routers
```

(任意) 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。

**ステップ 3** `show { ospf | ospfv3 } [ process-name ] database`

例 :

```
RP/0/RP0/cpu 0: router# show ospf group2 database
```

(任意) 特定のルータの OSPF データベースに関する情報の一覧を表示します。

- このコマンドは、さまざまな形式で、異なる OSPF LSA に関する情報を提供します。

**ステップ 4** `show { ospf | ospfv3 } [ process-name ] [ area-id ] flood-list interface type interface-path-id`

例 :

```
RP/0/RP0/cpu 0: router# show ospf 100 flood-list interface TenGigE 0/0/0/0
```

(任意) インターフェイス上でのフラッディングを待機している OSPF LSA のリストを表示します。

**ステップ 5** `show { ospf | ospfv3 } [ process-name ] [ vrf vrf-name ] [ area-id ] interface [ type interface-path-id ]`

例 :

```
RP/0/RP0/cpu 0: router# show ospf 100 interface TenGigE 0/0/0/0
```

例 :

```
RP/0/RP0/CPU0:router# show ospf interface
Interfaces for OSPF test
TenGigE0/0/0/7 is up, line protocol is up
Internet Address 7.7.7.3/16, Area 1
```

```

Label stack Primary label 0 Backup label 0 SRTE label 0
Process ID test, Router ID 7.7.7.3, Network Type BROADCAST, Cost: 1 (RSVP-TE)
Transmit Delay is 1 sec, State DR, Priority 1, MTU 1500, MaxPktSz 1500
Forward reference No, Unnumbered no, Bandwidth 10000000
Designated Router (ID) 7.7.7.3, Interface address 7.7.7.3
Backup Designated router (ID) 7.7.7.2, Interface address 7.7.7.2
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:01:801
Index 1/1, flood queue length 0
Next 0(0)/0(0)
Last flood scan length is 2, maximum is 2
Last flood scan time is 0 msec, maximum is 0 msec
LS Ack List: current length 0, high water mark 1
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 7.7.7.2 (Backup Designated Router)
Suppress hello for 0 neighbor(s)
Keychain-based authentication enabled
  Key id used is 1
  Cryptographic algorithm HMAC-SHA-256
Multi-area interface Count is 0

```

(任意) OSPF インターフェイス情報を表示します。

**ステップ 6** `show { ospf | ospfv3 } [ process-name ] [ area-id ] neighbor [ type interface-path-id ] [ neighbor-id ] [ detail ]`

例 :

```
RP/0/RP0/cpu 0: router# show ospf 100 neighbor
```

(任意) 個々のインターフェイスに基づいた OSPF ネイバー情報を表示します。

**ステップ 7** `clear { ospf | ospfv3 } [ process-name ] process`

例 :

```
RP/0/
/CPU0:router# clear ospf 100 process
```

(任意) OSPF ルータ プロセスを停止および再起動せずにリセットします。

**ステップ 8** `clear { ospf | ospfv3 } [ process-name ] redistribution`

例 :

```
RP/0/RP0/cpu 0: router#clear ospf 100 redistribution
```

OSPF ルート再配布をクリアします。

**ステップ 9** `clear { ospf | ospfv3 } [ process-name ] routes`

例 :

```
RP/0/RP0/cpu 0: router#clear ospf 100 routes
```

OSPF ルート テーブルをクリアします。

**ステップ 10** `clear { ospf | ospfv3 } [ process-name ] vrf [ vrf-name | all ] { process | redistribution | routes | statistics [ interface type interface-path-id | message-queue | neighbor ] }`

例 :

```
RP/0/RP0/cpu 0: router#clear ospf 100 vrf vrf_1 process
```

OSPF ルート テーブルをクリアします。

**ステップ 11** `clear { ospf | ospfv3 } [ process-name ] statistics [ neighbor [ type interface-path-id ] [ ip-address ] ]`

例 :

```
RP/0/RP0/cpu 0: router# clear ospf 100 statistics
```

(任意) ネイバー状態遷移の OSPF 統計情報をクリアします。

## スタブエリア

スタブエリアは、ルートアドバタイズメントや、エリアの外部にあるネットワークの詳細情報を受け入れないエリアです。スタブエリアには、通常、エリアと他の自律システムとのインターフェイスになるルータが1つだけがあります。スタブ ABR は、外部の宛先への単一のデフォルトルートをスタブエリアにアドバタイズします。スタブエリア内のルータはエリア外の宛先および自律システムに対してこのルートを使用します。この関係は、エリアにフラッディングされた外部 LSA を格納するためにも使用される LSA データベースのスペースを節約します。

## Not-So-Stubby Area

Not-So-Stubby Area (NSSA) はスタブエリアに似ています。NSSA はコアからエリアへとタイプ 5 の外部 LSA をフラッディングしませんが、限定的に自律システム外部ルートをエリア内にインポートできます。

NSSA は、再配布によって、タイプ 7 の自律システムの外部ルートを NSSA エリア内部にインポートできます。これらのタイプ 7 の LSA は、NSSA の ABR によってタイプ 5 の LSA に変換され、ルーティングドメイン全体にフラッディングされます。変換中は集約とフィルタリングがサポートされます。

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 OSPF を使用する中央サイトを接続する必要があるネットワーク管理者であれば、管理を簡素化するために NSSA 使用します。

スタブエリアにはリモートサイトのルートが再配布されないため、NSSA が実装される前は、企業サイトの境界ルータとリモートルータ間の接続に OSPF スタブエリアを利用できず、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した。RIP のようなシンプルなプロトコルを実行して再配布を処理する方法が一般的でした。NSSA が実装されたことで、企業ルータとリモートルータ間のエリアを NSSA として定義することにより、NSSA で OSPF を拡張してリモート接続をカバーできます。エリア 0 を NSSA にすることはできません。

## スタブエリアタイプおよび Not-So-Stubby Area タイプの設定

このタスクでは、OSPF のスタブ エリアおよび NSSA を設定する方法を説明します。

### 手順

#### ステップ 1 `configure`

ステップ 2 次のいずれかを実行します。

- `router ospf process-name`
- `router ospfv3 process-name`

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。

または

指定したルーティング プロセスに OSPFv3 ルーティングをイネーブルにし、`router ospfv3` コンフィギュレーション モードでルータを配置します。

(注) `process-name` 引数は、40 文字未満の英数字です。

#### ステップ 3 `router-id { router-id }`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IP アドレスをルータ ID として使用することを推奨します。

#### ステップ 4 `area area-id`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 1
```

エリア コンフィギュレーション モードを開始し、OSPF プロセスのバックボーン以外のエリアを設定します。

- `area-id` 引数は、`area 1000` や `area 0.0.3.232` など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。



ステップ5 次のいずれかを実行します。

- **stub** [ **no-summary** ]
- **nssa** [ **no-redistribution** ] [ **default-information-originate** ] [ **no-summary** ]

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# stub no summary
```

または

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# nssa no-redistribution
```

非バックボーンエリアをスタブエリアとして定義します。

- スタブエリアに送信される LSA の数をさらに減らすために **no-summary** キーワードを指定します。このキーワードにより、ABR がサマリーリンクステートアドバタイズメント (タイプ 3) をスタブエリアに送信しないようにします。

または

エリアを NSSA として定義します。

ステップ6 次のいずれかを実行します。

- **stub**
- **nssa**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# stub
```

または

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# nssa
```

(任意) スタブエリア、および NSSA エリアに設定されたオプションをオフにします。

- ステップ5 でオプションのキーワード (**no-summary**、**no-redistribution**、**default-information-originate**、および **no-summary**) を使用してスタブエリアおよび NSSA エリアを設定した場合、コマンドの **no** 形式を使用するのではなく、**stub** および **nssa** コマンドをこれらのキーワードなしで再度発行する必要があります。
- たとえば、コマンドの **no nssa default-information-originate** 形式は、NSSA エリアを通常のエリアに変更し、そのエリアの既存の隣接関係を意図せずダウンさせます。

ステップ7 **default-cost** *cost*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)#default-cost 15
```

(任意) スタブエリアまたは NSSA に送信されるデフォルトサマリールートのコストを指定します。

- このコマンドはNSSAにアタッチされているABRでのみ使用します。エリア内の他のルータには使用しないでください。
- デフォルトのコストは1です。

#### ステップ8 commit

ステップ9 スタブ エリアまたはNSSAにある他のすべてのルータでこのタスクを繰り返します。

#### スタブエリアの設定：例

エリア1がスタブエリアとして設定される例を次に示します。

```
router ospfv3 1
router-id 10.0.0.217
area 0
interface TenGigE 0/0/0/1
area 1
stub
interface TenGigE 0/0/0/0
```

## OSPFのネイバーおよび隣接関係

セグメントを共有するルータ（2つのインターフェイス間のレイヤ2リンク）は、そのセグメント上でネイバー同士となります。OSPFではHelloプロトコルをネイバー探索およびキープアラライブメカニズムとして使用します。Helloプロトコルでは定期的にhelloパケットを各インターフェイスで送受信します。helloパケットは、インターフェイス上のすべての既知のOSPFネイバーをリストします。ルータがネイバーのHelloパケット内に自身がリストされていることを認識すると、それらのルータはネイバー同士となります。2つのルータがネイバーになると、データベースの交換や同期化を行うことができるようになります。これにより、隣接が作成されます。ブロードキャストおよびNBMAネットワークのすべての隣接ルータに隣接があります。

## 非ブロードキャストネットワークのネイバーの設定

このタスクでは、非ブロードキャストネットワークにネイバーを設定する方法を説明します。このタスクはオプションです。

#### 始める前に

NBMAネットワークをブロードキャストまたは非ブロードキャストとして構成する場合は、各ルータから各ルータあるいはフルメッシュのネットワークにまで仮想回線があると想定されます。

## 手順

ステップ1 **configure**

ステップ2 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。

または

指定したルーティングプロセスに OSPFv3 ルーティングをイネーブルにし、**router ospfv3** コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

ステップ3 **router-id { router-id }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IP アドレスをルータ ID として使用することを推奨します。

ステップ4 **area area-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 0
```

エリア コンフィギュレーション モードを開始し、OSPF プロセスのエリアを設定します。

- この例ではバックボーン エリアを設定します。
- *area-id* 引数は、**area 1000** や **area 0.0.3.232** など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1 つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

ステップ5 **network { broadcast | non-broadcast }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# network non-broadcast
```

OSPF ネットワーク タイプをそのメディアのデフォルト以外のタイプに設定します。

- この例では、ネットワーク タイプを NBMA に設定します。

#### ステップ 6 **dead-interval** *seconds*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# dead-interval 40
```

(任意) ネイバーのダウンを宣言する前に、ネイバーからの hello パケットを待機する時間を設定します。

#### ステップ 7 **hello-interval** *seconds*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# hello-interval 10
```

(任意) OSPF がインターフェイスで送信する hello パケットの間隔を指定します。

#### ステップ 8 **interface** *type interface-path-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、ステップ 4 で設定したエリアのインターフェイスを 1 つ以上関連付けます。

- この例では、値がインターフェイス レベルで設定されていないため、インターフェイスはブロードキャスト ネットワーク タイプおよび hello および dead 間隔をそのエリアから継承します。

#### ステップ 9 次のいずれかを実行します。

- **neighbor** *ip-address* [ **priority** *number* ] [ **poll-interval** *seconds* ] [ **cost** *number* ]
- **neighbor** *ipv6-link-local-address* [ **priority** *number* ] [ **poll-interval** *seconds* ] [ **cost** *number* ] [ **database-filter** [ **all** ] ]

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# neighbor 10.20.20.1 priority 3 poll-interval 15
```

または

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# neighbor fe80::3203:a0ff:fe9d:f3fe
```

ブロードキャスト ネットワーク以外と相互接続する OSPF ネイバーの IPv4 アドレスを設定します。

または

OSPFv3 ネイバーのリンクローカル IPv6 アドレスを設定します。

- `ipv6-link-local-address` 引数は、RFC 2373 に記載されている形式である必要があります。このアドレスは 16 ビット値を使用する 16 進数をコロンで区切って指定します。
- **priority** キーワードでは、このネイバーが DR または BDR になる資格があることをルータに通知します。priority 値は隣接ルータの実際のプライオリティ設定と一致する必要があります。ネイバープライオリティのデフォルト値はゼロです。
- 特定のコストが設定されていないネイバーは、`cost` コマンドに基づいてインターフェイスのコストを想定します。
- **database-filter** キーワードでは OSPF ネイバーへの発信 LSA をフィルタリングします。`all` キーワードを指定すると、着信および発信 LSA はフィルタリングされます。

**ステップ 10** インターフェイスのすべてのネイバーでステップ 9 を繰り返します。

**ステップ 11** `exit`

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# exit  
エリア 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** `interface type interface-path-id`

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

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て、ステップ 4 で設定したエリアのインターフェイスを 1 つ以上関連付けます。

- この例では、値がインターフェイスレベルで設定されていないため、インターフェイスはブロードキャスト ネットワーク タイプおよび hello および dead 間隔をそのエリアから継承します。

**ステップ 13** 次のいずれかを実行します。

- `neighbor ip-address [ priority number ] [ poll-interval seconds ] [ cost number ] [ database-filter [ all ] ]`
- `neighbor ipv6-link-local-address [ priority number ] [ poll-interval seconds ] [ cost number ] [ database-filter [ all ] ]`

例：

```
RP/0/  
/CPU0:router(config-ospf-ar)# neighbor 10.34.16.6
```

または

```
RP/0/  
/CPU0:router(config-ospf-ar)# neighbor fe80::3203:a0ff:fe9d:f3f
```

ブロードキャスト ネットワーク以外と相互接続する OSPF ネイバーの IPv4 アドレスを設定します。

または

OSPFv3 ネイバーのリンクローカル IPv6 アドレスを設定します。

- **ipv6-link-local-address** 引数は、RFC 2373 に記載されている形式である必要があります。このアドレスは 16 ビット値を使用する 16 進数をコロンで区切って指定します。
- **priority** キーワードでは、このネイバーが DR または BDR になる資格があることをルータに通知します。priority 値は隣接ルータの実際のプライオリティ設定と一致する必要があります。ネイバー プライオリティのデフォルト値はゼロです。
- 特定のコストが設定されていないネイバーは、**cost** コマンドに基づいてインターフェイスのコストを想定します。
- **database-filter** キーワードでは OSPF ネイバーへの発信 LSA をフィルタリングします。**all** キーワードを指定すると、着信および発信 LSA はフィルタリングされます。フィルタリングによりルーティング トポロジが 2 つのネイバー間でまったく異なるように見え、「black-holing」化やルーティング ループを引き起こしたりすることがあるため、十分注意して使用してください。

**ステップ 14** インターフェイスのすべてのネイバーでステップ 13 を繰り返します。

—

**ステップ 15** **commit**

## 認証方法

プロセスまたはエリアの全体に対して、またはインターフェイスまたは仮想リンク上に認証を指定できます。インターフェイスまたは仮想リンクには 1 種類の認証だけを指定でき、両方とも設定することはできません。エリアまたはプロセス用に設定されたインターフェイスまたは仮想リンクのオーバーライド認証用に設定された認証。

エリアのすべてのインターフェイスで同じ認証タイプを使用する場合、エリア コンフィギュレーション サブモードで **authentication** コマンドを使用すると（また、エリア全体で MD5 認証を使用する場合は **message-digest** キーワードを指定すると）、より少ないコマンドを設定できます。この方法を使用すると、各インターフェイスに認証を指定するときに必要なコマンドよりも少ないコマンドで設定できます。

## OSPF Version 2 の異なる階層レベルでの認証の設定

このタスクでは、OSPF ルータ プロセスに MD5（セキュア）認証を設定する方法について説明します。プレーンテキスト認証を 1 エリアに設定し、次にクリアテキスト（null）認証を 1 インターフェイスに適用します。



- (注) インターフェイス レベルで設定された認証は、エリア レベルおよびルータ プロセス レベルで設定された認証を上書きします。インターフェイスに特別に設定された認証がない場合、そのインターフェイスは認証パラメータ値をより高い階層レベルから継承します。

### 始める前に

認証を設定する場合、プレーンテキスト認証または MD5 認証のどちらを設定するかをはじめに決定する必要があります。また、認証の適用対象がプロセス内のすべてのインターフェイスか、全エリアか、特定のインターフェイスかを決定する必要があります。ネットワークに特定のメソッドを使用する場合、それぞれの種類の認証に関する情報については、[OSPF の階層 CLI および CLI 継承 \(60 ページ\)](#) を参照してください。

### 手順

ステップ 1 **configure**

ステップ 2 **router ospf process-name**

例 :

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティング プロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

ステップ 3 **router-id { router-id }**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

ステップ 4 **authentication [ message-digest | null ]**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)#authentication message-digest
```

OSPF プロセスに対して MD5 認証が有効になります。

- エリアやインターフェイスなどのより低い階層レベルによって変更されないかぎり、この認証タイプはルータ プロセス全体に適用されます。

ステップ 5 **message-digest-key key-id md5 { key | clear key | encrypted key | LINE }**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)#message-digest-key 4 md5 yourkey
```

OSPF プロセスに対して MD5 認証キーを指定します。

- 隣接ルータが、同じキー ID を保持する必要があります。

#### ステップ 6 **area** *area-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 0
```

エリア コンフィギュレーション モードを開始して、OSPF プロセスのバックボーン エリアを設定します。

#### ステップ 7 **interface** *type interface-path-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、1つ以上のインターフェイスをバックボーン エリアに関連付けます。

- すべてのインターフェイスは、OSPF プロセスの指定された認証パラメータ値を継承します (ステップ 4、ステップ 5、ステップ 6)。

ステップ 8 同じ認証を使用して通信する必要があるインターフェイスごとにステップ 7 を繰り返します。

—

#### ステップ 9 **exit**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# exit
```

エリア OSPF 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 **area** *area-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 1
```

エリア コンフィギュレーション モードを開始し、OSPF プロセスの非バックボーンのエリア 1 を設定します。

- *area-id* 引数は、`area 1000` や `area 0.0.3.232` など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

#### ステップ 11 **authentication** [ **message-digest** | **null** ]

例：



```
RP/0/RP0/cpu 0: router(config-ospf-ar)# authentication
```

セキュリティのないタイプ 1 (プレーンテキスト) 認証をイネーブルにします。

- 例では、プレーンテキスト認証を (キーワードを指定しないことによって) 指定します。インターフェイス コンフィギュレーション モードで **authentication-key** コマンドを使用し、このプレーンテキストパスワードを指定します。

#### ステップ 12 **interface type interface-path-id**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、ステップ 7 で指定したバックボーン以外のエリア 1 に 1 つ以上のインターフェイスを関連付けます。

- 設定されているすべてのインターフェイスがエリア 1 に対して設定されている認証パラメータ値を継承します。

ステップ 13 同じ認証を使用して通信する必要があるインターフェイスごとにステップ 12 を繰り返します。

—

#### ステップ 14 **interface type interface-path-id**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

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、1 つ以上のインターフェイスを異なる認証タイプに関連付けます。

#### ステップ 15 **authentication [ message-digest | null ]**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# authentication null
```

TenGigE 0/0/0/0 に認証なしを指定し、エリア 1 に指定されたプレーンテキスト認証を上書きします。

- デフォルトでは、同じエリアで設定されるすべてのインターフェイスは、エリアと同じ認証パラメータ値を継承します。

#### ステップ 16 **commit**

---

# OSPFに同じLSAが生成される頻度または受け入れられる頻度の制御

このタスクでは、非常に短い間隔で多数のLSAがフラッディングされる必要がある場合に、ルーティングテーブルのOSPFルートのコンバージェンス時間を調整する方法を説明します。

## 手順

### ステップ1 **configure**

ステップ2 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router:router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスにOSPFルーティングをイネーブルにし、ルータ コンフィギュレーションモードでルータを配置します。

または

指定したルーティングプロセスにOSPFv3ルーティングをイネーブルにし、**router ospfv3** コンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40文字未満の英数字です。

### ステップ3 **router-id { router-id }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPFプロセスのルータIDを設定します。

(注) 固定IPアドレスをルータIDとして使用することを推奨します。

ステップ4 ステップ5またはステップ6、または両方のステップを実行して、同じLSAが送受信される間隔を制御します。

—

### ステップ5 **timers lsa refresh seconds**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# timers lsa refresh 1800
```

自動送信 LSA をリフレッシュする頻度を秒単位で設定します。

- OSPF および OSPFv3 の両方で、デフォルトは 1800 秒です。

#### ステップ 6 `timers lsa min-arrival seconds`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# timers lsa min-arrival 2
```

フラッディング中に特定の OSPF Version 2 LSA の新しいプロセスが受け入れられる頻度を制限します。

- デフォルト値は 1 秒です。

#### ステップ 7 `timers lsa group-pacing seconds`

例：

```
RP/0/  
/CPU0:router(config-ospf)# timers lsa group-pacing 1000
```

OSPF リンクステート LSA がフラッディングのグループに収集される間隔を変更します。

- デフォルトは 240 秒です。

#### ステップ 8 `commit`

## OSPF の仮想リンクおよび中継エリア

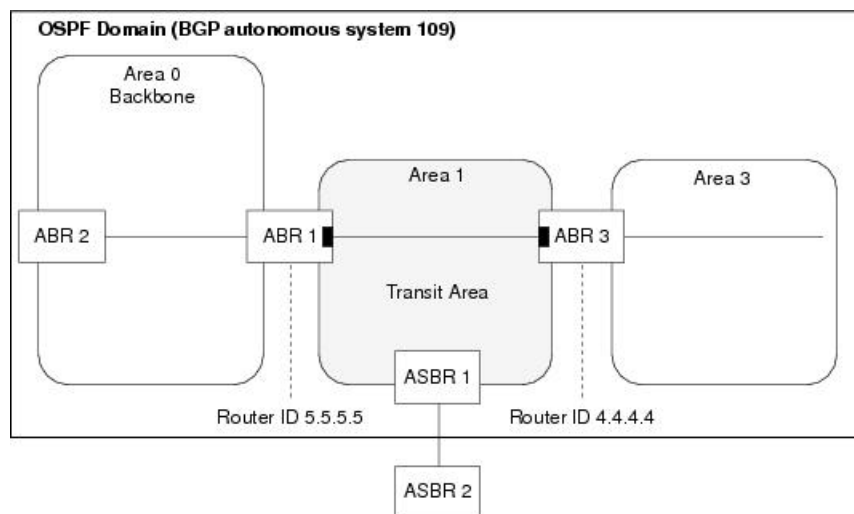
OSPF では、すべてのエリアからのルーティング情報は、ABR によって最初にバックボーンエリアに集約されます。次に、同じ ABR は受信したその情報をアタッチされているエリアに伝播します。このような階層型のルーティング情報の配信では、すべてのエリアがバックボーンエリア（エリア 0）に接続する必要があります。エリアを定義する必要がある場合もありますが、エリア 0 には物理的に接続することはできません。そのような場合の例として、会社で OSPF エリアが含まれる新しい取得を行う場合やエリア 0 自体がパーティション化されている場合が挙げられます。

エリアをエリア 0 に接続できない場合、そのエリアとエリア 0 の間で仮想リンクを設定する必要があります。仮想リンクの 2 つのエンドポイントは ABR であり、仮想リンクは両方のルータで設定する必要があります。2 つのルータが属する、バックボーン以外の共通エリアは中継エリアと呼ばれます。仮想リンクは、他の仮想エンドポイント（他の ABR）の中継エリアとルータ ID を指定します。

仮想リンクはスタブ エリアまたは NSSA から設定することはできません。

図 1: エリア 0 への仮想リンク

この図はエリア 3 からエリア 0 への仮想リンクを示します。



## 仮想リンクの作成

このタスクでは、仮想リンクをバックボーン（エリア 0）に作成して MD5 認証を適用する方法について説明します。説明されている手順は、仮想リンクの各端にある両方の ABR で実行する必要があります。



- (注) 明示的にエリアパラメータ値を設定したら、インターフェイスの値を上書きして明示的に設定しないかぎり、その値はそのエリアにバインドされているすべてのインターフェイスに継承されます。

### 始める前に

MD5 認証が設定された仮想リンクをエリア 0 に作成するには、次の前提条件を満たす必要があります。

- ローカルルータを設定するリンクの反対の隣接ルータのルータ ID が必要です。ルータ ID を取得するためにリモートルータで `show ospf` コマンドまたは `show ospfv3` コマンドを実行できます。
- 仮想リンクが正常に機能するには、仮想リンクの各端に固定ルータ ID が必要です。ルータ ID は変更されないようにします。デフォルトでルータ ID を割り当てると、変更される可能性があります。したがって、仮想リンクを設定する前に、次のいずれかのタスクを実行することをお勧めします。
  - ルータ ID を設定するには、`router-id` コマンドを使用します。この方法を推奨します。

- ルータが安定したルータ ID を持つために、ループバック インターフェイスを設定します。
- OSPF Version 2 の仮想リンクを設定する前に、プレーン テキスト認証、MD5 認証、認証なし（デフォルト）のうち、どの認証を設定するかを決定する必要があります。認証に関連する追加のタスクを実行する必要があるかどうかに応じて決定します。

## 手順

**ステップ 1** 次のいずれかを実行します。

- **show ospf** [ *process-name* ]
- **show ospfv3** [ *process-name* ]

例：

```
RP/0//CPU0:router# show ospf
```

または

```
RP/0//CPU0:router# show ospfv3
```

(任意) OSPF ルーティング プロセスに関する一般情報を表示します。

- 出力にはローカル ルータのルータ ID が表示されます。このルータ ID はリンクの另一端を設定するために必要です。

**ステップ 2** **configure**

**ステップ 3** 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0//CPU0:router(config)# router ospf 1
```

または

```
RP/0//CPU0:router(config)# router ospfv3 1
```

指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。

または

指定したルーティングプロセスに OSPFv3 ルーティングをイネーブルにし、**router ospfv3** コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

**ステップ 4** `router-id { router-id }`

例 :

```
RP/0//CPU0:router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

**ステップ 5** `area area-id`

例 :

```
RP/0//CPU0:router(config-ospf)# area 1
```

エリア コンフィギュレーションモードを開始し、OSPF プロセスのバックボーン以外のエリアを設定します。

- `area-id` 引数は、`area 1000` や `area 0.0.3.232` など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1 つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

**ステップ 6** `virtual-link router-id`

例 :

```
RRP/0//CPU0:router(config-ospf-ar)# virtual-link 10.3.4.5
```

OSPF 仮想リンクを定義します。

- を参照してください。

**ステップ 7** `authentication message-digest`

例 :

```
RP/0//CPU0:router(config-ospf-ar-vl)#authentication message-digest
```

この仮想リンクに対して MD5 認証を選択します。

**ステップ 8** `message-digest-key key-id md5 { key | clear key | encrypted key }`

例 :

```
RP/0//CPU0:router(config-ospf-ar-vl)#message-digest-key 4 md5 yourkey
```

OSPF 仮想リンクを定義します。

- 仮想リンクを理解するには、を参照してください。
- `key-id` 引数は、1 ~ 255 の範囲の数です。 `key` 引数は最大 16 文字の英数字です。仮想リンクの両端のルータには同じキー ID と、OSPF トラフィックをルーティングできるキーが必要です。
- `authentication-key key` コマンドは、OSPFv3 ではサポートされていません。

- キーが暗号化されたら、その暗号化を保持する必要があります。

**ステップ 9** 仮想リンクの反対側にある ABR でこのタスクのすべての手順を繰り返します。このルータで仮想リンクに指定する同じキー ID およびキーを指定します。

**ステップ 10** **commit**

**ステップ 11** 次のいずれかを実行します。

- **show ospf** [ *process-name* ] [ *area-id* ] **virtual-links**
- **show ospfv3** [ *process-name* ] **virtual-links**

例：

```
RP/0//CPU0:router# show ospf 1 2 virtual-links
```

または

```
RP/0//CPU0:router# show ospfv3 1 virtual-links
```

(任意) OSPF 仮想リンクのパラメータと現在の状態を表示します。

仮想リンクの作成：例

**ABR 1 の設定**

**ABR 2 の設定**

次の例では、`show ospfv3 virtual links` コマンドで、OSPFv3 ネイバーへの OSPF\_VL0 仮想リンクが起動しており、仮想リンク インターフェイスの ID が 2 であり、仮想リンク エンドポイントの IPv6 アドレスが 2003:3000::1 であることを検証します。

```
show ospfv3 virtual-links
```

```
Virtual Links for OSPFv3 1
```

```
Virtual Link OSPF_VL0 to router 10.0.0.3 is up
  Interface ID 2, IPv6 address 2003:3000::1
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 0.1.20.255, via interface TenGigE 0/11/0/0 Cost of using 2
  Transmit Delay is 5 sec,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:02
  Adjacency State FULL (Hello suppressed)
  Index 0/2/3, retransmission queue length 0, number of retransmission 1
  First 0(0)/0(0)/0(0) Next 0(0)/0(0)/0(0)
  Last retransmission scan length is 1, maximum is 1
  Last retransmission scan time is 0 msec, maximum is 0 msec
```

```
Check for lines:
```

```
Virtual Link OSPF_VL0 to router 10.0.0.3 is up
```

```
Adjacency State FULL (Hello suppressed)
```

```
State is up and Adjacency State is FULL
```

この例では、エリア 0 と 1 および仮想リンク 10.0.0.217 と 10.0.0.212 で構成される OSPFv3 トポロジのエリア 1 からバックボーンを接続するように仮想リンクを設定する方法を説明します。

```
router ospfv3 1
router-id 10.0.0.217
area 0
interface TenGigE 0/11/0/1
area 1
virtual-link 10.0.0.212
interface TenGigE 0/11/0/0
```

```
router ospfv3 1
router-id 10.0.0.212
area 0
interface TenGigE 0/11/0/1
area 1
virtual-link 10.0.0.217
interface TenGigE 0/11/0/0
```

この例では、ルータ ABR1 のすべてのインターフェイスは MD5 認証を使用します。

```
router ospf ABR1
router-id 10.10.10.10
authentication message-digest
message-digest-key 100 md5 0 cisco
area 0
interface TenGigE 0/11/0/1
interface TenGigE 0/11/0/0
area 1
interface TenGigE 0/11/0/0
virtual-link 10.10.5.5
!
!
```

この例では、ルータ ABR3 のエリア 1 インターフェイスだけが MD5 認証を使用します。

```
router ospf ABR2
router-id 10.10.5.5
area 0
area 1
authentication message-digest
message-digest-key 100 md5 0 cisco
interface TenGigE 0/11/0/1
virtual-link 10.10.10.10
area 3
interface Loopback 0
interface TenGigE 0/11/0/0
!
```



## OSPF ABR でのサブネットワーク LSA の集約

IPアドレスをインターフェイスに割り当てたときに複数のサブネットワークを設定した場合、すべてのサブネットワークが含まれ、ローカルエリアが別のエリアにアドバタイズする1つのLSAにソフトウェアを集約することができます。このようにソフトウェアを集約するとLSAの数を減らすことができるため、ネットワークリソースを節約できます。この集約はエリア間ルート集約と呼ばれます。これは自律システム内のルートに適用されます。再配布によってOSPFに挿入された外部ルートには適用されません。

このタスクでは、一緒にアドバタイズされる範囲に収まるすべてのサブネットワークを指定することによって、サブネットワークを1つのLSAに集約するようにOSPFを設定します。このタスクは1つのABRでのみ実行します。

### 手順

#### ステップ1 **configure**

ステップ2 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスにOSPFルーティングをイネーブルにし、ルータコンフィギュレーションモードでルータを配置します。

または

指定したルーティングプロセスにOSPFv3ルーティングをイネーブルにし、**router ospfv3**コンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40文字未満の英数字です。

#### ステップ3 **router-id** { *router-id* }

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPFプロセスのルータIDを設定します。

(注) 固定IPv4アドレスをルータIDとして使用することを推奨します。

#### ステップ4 **area** *area-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area
```

エリアコンフィギュレーションモードを開始し、OSPFプロセスのバックボーン以外のエリアを設定します。

- **area-id** 引数は、**area 1000** や **area 0.0.3.232** など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

ステップ5 次のいずれかを実行します。

- **range** *ip-address mask* [ **advertise** | **not-advertise** ]
- **range** *ipv6-prefix / prefix-length* [ **advertise** | **not-advertise** ]

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# range 192.168.0.0 255.255.0.0 advertise
```

または

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# range 4004:f000::/32 advertise
```

エリア境界で OSPF ルートを統合および集約します。

- **advertise** キーワードにより、タイプ 3 サマリー LSA のソフトウェアがサブネットワークのアドレス範囲をアドバタイズします。
- **not-advertise** キーワードによって、ソフトウェアがタイプ 3 サマリー LSA に制限され、この範囲内のサブネットワークは他のエリアには見えません。
- 最初の例では、ネットワーク **192.168.0.0** のすべてのサブネットワークが ABR によって集約され、バックボーン外のエリアにアドバタイズされます。
- 2 番目の例では、複数の IPv4 インターフェイスが **192.x.x** のネットワークに対応しています。

#### ステップ6 **interface** *type interface-path-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

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て、1つ以上のインターフェイスをエリアに関連付けます。

#### ステップ7 **commit**

---

### 例

次の例では、エリア1からバックボーンに集約されたプレフィックス範囲2300::/16を示します。

```
router ospfv3 1
router-id 192.168.0.217
area 0
interface TenGigE 0/11/0/0
area 1
range 2300::/16
interface TenGigE 0/11/0/0
```

## OSPFのルート再配布

再配布により、異なるルーティングプロトコルを使用してルーティング情報を交換できます。この手法を使用すると、複数のルーティングプロトコルに接続を広げることができます。redistribute コマンドでは、OSPFからの再配布ではなく、OSPFプロセスへの再配布が制御されることに注意することが重要です。

## OSPFへのルートの再配布

このタスクでは、IGP（別のOSPFプロセスでも可）からOSPFにルートを再配布します。

### 手順

#### ステップ1 configure

ステップ2 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスにOSPFルーティングをイネーブルにし、ルータコンフィギュレーションモードでルータを配置します。

または

指定したルーティングプロセスにOSPFv3ルーティングをイネーブルにし、router ospfv3 コンフィギュレーションモードでルータを配置します。

(注) `process-name` 引数は、40 文字未満の英数字です。

### ステップ3 `router-id { router-id }`

例：

```
RRP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

### ステップ4 `redistribute protocol [process-id] { level-1 | level-1-2 | level-2 } [metric metric-value] [metric-type type-value] [match { external [ 1 | 2 ] } [tag tag-value] [route-policy policy-name]`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# redistribute bgp 100
```

または

```
RP/0/RP0/cpu 0: router(config-router)# redistribute bgp 110
```

1 つのルーティング ドメインから別のルーティング ドメインへの OSPF ルートの再配布

または

あるルーティング ドメインから別のルーティング ドメインへ OSPFv3 ルートを再配布します。

- このコマンドを実行すると、定義上ルータが ASBR になります。
- OSPF は再配布によって学習したすべてのルートを `external` とタグ付けします。
- プロトコルとそのプロセス ID (設定されている場合) は、OSPF に再配布されるプロトコルを示します。
- メトリックは外部ルートに割り当てるコストです。すべてのプロトコルでデフォルトは 20 です。ただし、BGP のデフォルトのメトリックは 1 です。
- OSPF の例では、BGP 自律システム 1、レベル 1 のルートを OSPF にタイプ 2 外部ルートとして再配布します。
- OSPFv3 の例では、BGP 自律システム 1、レベル 1 および 2 のルートを OSPF に再配布します。OSPFv3 ルーティング ドメインにアドバタイズされるデフォルトルートに関連付けられている外部リンク タイプは、タイプ 1 の外部ルートです。

(注) OSPFv3 では RPL はサポートされていません。

ステップ5 次のいずれかを実行します。

- `summary-prefix address mask [ not-advertise ] [ tag tag ]`
- `summary-prefix ipv6-prefix / prefix-length [ not-advertise ] [ tag tag ]`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# summary-prefix 10.1.0.0 255.255.0.0
```

または

```
RP/0/RP0/cpu 0: router(config-router)# summary-prefix 2010:11:22::/32
```

(任意) OSPF の集約アドレスを作成します。

または

(任意) OSPFv3 の集約アドレスを作成します。

- このコマンドは、非 OSPF ルートの外部ルート集約を行います。
- 集約される外部範囲は隣接している必要があります。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。
- このコマンドはオプションです。これを指定しないと、各ルートはリンクステートデータベースに含まれ、LSA にアドバタイズされます。
- OSPFv2 の例では、集約アドレス 10.1.0.0 にアドレス 10.1.1.0、10.1.2.0、10.1.3.0 などが含まれています。外部の LSA では、アドレス 10.1.0.0 だけがアドバタイズされます。
- OSPFv3 の例では、集約アドレス 2010:11:22::/32 には 2010:11:22:0:1000::1、2010:11:22:0:2000:679:1、などのアドレスがあります。外部 LSA にはアドレス 2010:11:22::/32 だけがアドバタイズされます。

## ステップ 6 commit

### 例

次の例では、プレフィックスリストを使用して、他のプロトコルから再配布されるルートを制限します。

上位 32 ビットの 9898:1000 および 32 から 64 のプレフィックス長を持つルートだけが BGP 42 から再配布されます。このパターンに一致しないルートだけが BGP 1956 から再配布されます。

```
ipv6 prefix-list list1
 seq 10 permit 9898:1000::/32 ge 32 le 64
ipv6 prefix-list list2
 seq 10 deny 9898:1000::/32 ge 32 le 64
 seq 20 permit ::/0 le 128
router ospfv3 1
 router-id 10.0.0.217
 redistribute bgp 42
 redistribute bgp 1956
 distribute-list prefix-list list1 out bgp 42
 distribute-list prefix-list list2 out bgp 1956
 area 1
 interface TenGigE 0/0/0/0
```

## OSPF Version 2 のノンストップ フォワーディング

OSPF Version 2 の NSF では、フェールオーバー後にルーティングプロトコル情報を保存しながら、既知のルートを通してデータパケットの転送が継続されるようにできます。NSFを使用すると、ピアネットワークングデバイスで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、データトラフィックはインテリジェントラインカードを介して転送されますが、スタンバイルートプロセッサ (RP) では、障害が発生した RP からの制御と見なされず、フェールオーバー中にラインカードのアップ状態が維持され、アクティブ RP の転送情報ベース (FIB) が最新状態に維持される機能が、NSF の動作にとって非常に重要です。

OSPF などのルーティングプロトコルは、アクティブ RP または DRP 上でのみ実行され、隣接ルータからルーティングアップデートを受信します。OSPF NSF 対応ルータが RP のフェールオーバーを実行する場合、リンクステータスデータベースを OSPF ネイバーと再同期するために、次の2つの処理を実行する必要があります。まず、ネイバー関係をリセットせずに、ネットワーク上の使用可能な OSPF ネイバーを再学習します。次に、ルータはネットワークのリンクステータスデータベースのコンテンツを再取得します。

RP フェールオーバーの後できるだけ早く、NSF 対応ルータは OSPF NSF 信号を隣接する NSF 対応デバイスに送信します。この信号はフェールオーバールータで生成されたリンクローカル LSA の形式になります。ネイバーネットワークングデバイスは、この信号をこのルータとのネイバー関係がリセットされるべきでないことを示す指示として認識します。NSF 対応ルータがネットワーク上の他のルータから信号を受信すると、ネイバーリストの再構築を始めます。

ネイバー関係が再構築されると、NSF 対応ルータはすべての NSF 認識ネイバーとデータベースの再同期化を始めます。この時点でルーティング情報は OSPF ネイバーの間で交換されます。交換が完了すると、NSF 対応デバイスはルーティング情報を使用して、失効ルートを削除し、RIB を更新して、新しい転送情報で FIB を更新します。ルータおよび OSPF ネイバー上の OSPF が完全にコンバージされるようになりました。

## Cisco for OSPF Version 2 固有のノンストップ フォワーディングの設定

このタスクでは、NSF 対応ルータの Cisco に専用の OSPF NSF を設定する方法を説明します。このタスクはオプションです。

### 始める前に

OSPF NSF では、すべてのネイバーネットワークングデバイスが NSF 対応である必要があります。ルータにソフトウェアイメージをインストールすると自動的に NSF 対応になります。NSF 対応ルータが特定のネットワークセグメントで NSF 非認識ネイバーを検出すると、そのセグメントで NSF 機能をディセーブルにします。NSF 対応または NSF 認識ルータで完全に構成された他のネットワークセグメントに対しては、継続して NSF 機能を提供します。



(注) ノンストップ フォワーディングの設定では、次の制約事項が適用されます。

- 仮想リンク用 Cisco OSPF NSF はサポートされません。
- ネイバーは NSF 対応である必要があります。

## 手順

### ステップ 1 **configure**

#### ステップ 2 **router ospf** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティング プロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

#### ステップ 3 **router-id** { *router-id* }

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

#### ステップ 4 次のいずれかを実行します。

- **nsf cisco**
- **nsf cisco enforce global**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# nsf cisco enforce global
```

OSPF プロセスの Cisco NSF 操作をイネーブルにします。

- オプションの **enforce** および **global** キーワードを指定せずに **nsf cisco** コマンドを使用して、検出された NSF 以外のネイバーのインターフェイスで NSF 再起動メカニズムを中絶し、NSF ネイバーが適切に機能できるようにします。
- 再起動中にルータが NSF を実行するようにする場合は、オプションの **enforce** および **global** キーワードを指定して **nsf cisco** コマンドを使用します。ただし、NSF 以外のネイバーが検出されると、OSPF プロセス全体で NSF 再起動はキャンセルされます。

**ステップ5 nsf interval seconds**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# nsf interval 120
```

NSF 再起動の試行間隔の最小時間を設定します。

(注) このコマンドを使用する場合、OSPF が NSF 再起動実行を試みる前の OSPF プロセスを、最小でも 90 秒に設定する必要があります。

**ステップ6 nsfflush-delay-timesseconds**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)#nsf flush-delay-time 1000
```

外部ルートの学習に許可される最大時間を秒単位で設定します。

**ステップ7 nsflifetimeseconds**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)#nsf lifetime 90
```

再起動に続く NSF のルートの最大有効期間を秒単位で設定します。

**ステップ8 nsfietf**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)#nsf ietf
```

ietf グレースフル リスタートをイネーブルにします。

**ステップ9 commit**

## OSPF Shortest Path First スロットリング

OSPF SPF スロットリングにより、SPF スケジューリングをミリ秒間隔で設定して、ネットワークが不安定な場合に SPF 計算を遅らせることができます。トポロジ変化が発生した場合、Shortest Path Tree (SPT) を再計算するように SPF がスケジューリングされます。SPF が 1 回実行されると、複数のトポロジ変化イベントが発生します。

SPF 計算の実行間隔は、ネットワークのトポロジ変化の頻度に応じて動的に選択されます。ユーザ指定値の範囲内で、間隔は選択されます。ネットワークトポロジが不安定な場合、トポロジが安定するまで、SPF スロットリング機能は SPF スケジューリング間隔を長目に計算します。



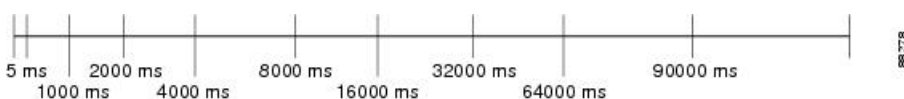
SPF の計算は、`timers throttle spf` コマンドで設定した間隔で実行されます。待機期間とは、次の SPF 計算が実行されるまで待機する時間のことです。計算を行うたびに、待機期間はその前の期間の 2 倍の長さになり、指定された最大待機期間に達するまでそれが行われます。

SPF タイミングについて、例を使用して説明します。この例では、開始時の間隔は 5 ミリ秒 (ms)、初回待機時間は 1000 ミリ秒、最大待機期間は 90,000 ミリ秒に設定されます。

```
timers spf 5 1000 90000
```

図 2: `timers spf` コマンドで設定される SPF の計算間隔

次の図に、ある待機期間中に少なくとも 1 回のトポロジ変化イベントを受信する場合の、SPF 計算の実行間隔を示します。

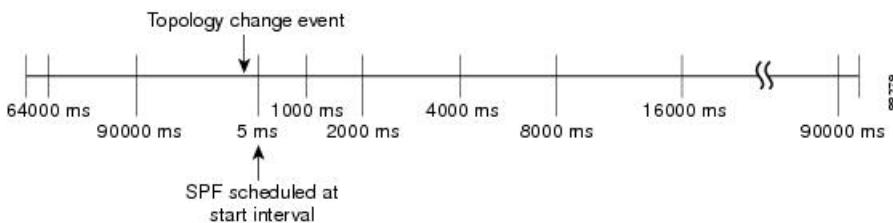


前の待機期間中に少なくとも 1 回のトポロジ変化イベントを受信すると、SPF 計算の待機期間が 2 倍になることに注意してください。最大待機期間に達すると、トポロジが安定し、待機期間中にイベントを受信しなくなるまで、待機期間が変化しなくなります。

現在の待機期間の経過後に、最初のトポロジ変化イベントを受信した場合は、開始時待機期間として指定されている時間だけ SPF 計算が遅延されます。その後の待機期間は、動的パターンに従います。

最大待機期間の開始後に、最初のトポロジ変化イベントが発生した場合、SPF 計算は開始時待機期間で再びスケジュールリングされ、その後の待機期間は `timers throttle spf` コマンドで指定されたパラメータに従ってリセットされます。図 3: トポロジ変化イベント後のタイマー間隔のリセット (33 ページ) では、最大待機期間の開始後にトポロジ変化イベントを受信して、SPF 間隔がリセットされることに注意してください。

図 3: トポロジ変化イベント後のタイマー間隔のリセット



## OSPF Shortest Path First スロットリングの設定

このタスクでは、SPF スケジュールリングをミリ秒間隔で設定し、ネットワークが不安定な場合に SPF 計算を遅らせる方法について説明します。このタスクはオプションです。

## 手順

**ステップ1 configure**

**ステップ2** 次のいずれかを実行します。

- **router ospf** *process-name*
- **router ospfv3** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 1
```

指定したルーティングプロセスにOSPFルーティングをイネーブルにし、ルータ コンフィギュレーションモードでルータを配置します。

または

指定したルーティングプロセスにOSPFv3ルーティングをイネーブルにし、**router ospfv3** コンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40文字未満の英数字です。

**ステップ3 router-id { router-id }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router-id 192.168.4.3
```

OSPFプロセスのルータIDを設定します。

(注) 固定IPv4アドレスをルータIDとして使用することを推奨します。

**ステップ4 timers throttle spf spf-start spf-hold spf-max-wait**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# timers throttle spf 10 4800 90000
```

SPFスロットリングタイマーを設定します。

**ステップ5 area area-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 0
```

エリアコンフィギュレーションモードを開始し、バックボーンエリアを設定します。

- *area-id* 引数は、`area 1000` や `area 0.0.3.232` など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

#### ステップ6 `interface type interface-path-id`

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

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て、1つ以上のインターフェイスをエリアに関連付けます。

#### ステップ7 `commit`

ステップ8 次のいずれかを実行します。

- `show ospf [ process-name ]`
- `show ospfv3 [ process-name ]`

例：

```
RP/0/RP0/cpu 0: router# show ospf 1
```

または

```
RP/0/RP0/cpu 0: router# RP/0/RP0/CPU0:router# show ospfv3 2
```

(任意) SPF スロットリング タイマーを表示します。

## OSPFv3のグレースフルリスタート

OSPFv3 グレースフルシャットダウン機能により、次の状況でもデータプレーン機能を維持することができます。

- ソフトウェアのアップグレードまたはダウングレードに伴う再起動などの、計画された OSPFv3 プロセスの再起動
- プロセスのクラッシュに伴う再起動などの、予期しない OSPFv3 プロセスの再起動

また、プロセッサが使用可能なメモリで非常に低いことを示す重大なメモリイベントが `sysmon` のウォッチドッグプロセスから受信された場合、OSPFv3 は一方的にシャットダウンするか、または終了状態に入ります。

この機能を使うと、OSPFv3 ルーティングプロトコルが再起動している間に、確立されているルートでノンストップデータ転送が行われます。そのため、この機能により IPv6 転送の可用性が向上します。

## OSPFv3 グレースフル リスタートの設定

このタスクでは、OSPFv3 プロセスのグレースフル リスタートを設定する方法を説明します。  
このタスクはオプションです。

### 手順

---

#### ステップ 1 **configure**

#### ステップ 2 **router ospfv3 process-name**

例：

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 test
```

OSPFv3 のルータ コンフィギュレーション モードを開始します。プロセス名は OSPF ルーティング プロセスを一意に識別する 1 つの単語です。プロセス名はスペースを含まない 40 文字以内の任意の英数字ストリングです。

#### ステップ 3 **graceful-restart**

例：

```
RP/0/RP0/cpu 0: router(config-ospfv3)#graceful-restart
```

現行ルータでグレースフル リスタートをイネーブルにします。

#### ステップ 4 **graceful-restart lifetime**

例：

```
RP/0/RP0/cpu 0: router(config-ospfv3)# graceful-restart lifetime 120
```

グレースフル リスタートの最大時間を指定します。

- デフォルトは 95 秒です。
- 値の範囲は 90 ~ 3600 秒です。

#### ステップ 5 **graceful-restart interval seconds**

例：

```
RP/0/RP0/cpu 0: router(config-ospfv3)# graceful-restart interval 120
```

現行ルータのグレースフル リスタートの間隔（最小時間）を指定します。

- 間隔のデフォルト値は 90 秒です。
- 値の範囲は 90 ~ 3600 秒です。

#### ステップ 6 **graceful-restart helper disable**

例：

```
RP/0/RP0/cpu 0: router(config-ospfv3)# graceful-restart helper disable
```

ヘルパー機能をディセーブルにします。

**ステップ7 commit**

**ステップ8 show ospfv3 [ process-name [ area-id ]] database grace**

例 :

```
RP/0/RP0/cpu 0: router# show ospfv3 1 database grace
```

グレースフル リスタート リンクのステータスを表示します。

## グレースフル リスタートに関する情報の表示

ここでは、グレースフルリスタートに関する情報を表示するために使用できるタスクについて説明します。

- 機能がイネーブルかどうかや、グレースフルリスタートが最後に実行された時間を確認するには、`show ospf` コマンドを使用します。OSPFv3 インスタンスの詳細を参照するには、`show ospfv3 process-name [ area-id ] database grace` コマンドを使用します。

### グレースフル リスタート機能のステータスの表示

次の画面出力は、ローカルルータのグレースフル リスタート機能の状態を示しています。

```
RP/0/RP0/cpu 0: router# show ospfv3 1 database grace

Routing Process "ospfv3 1" with ID 2.2.2.2
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF 10000 msec
Maximum wait time between two consecutive SPF 10000 msec
Initial LSA throttle delay 0 msec
Minimum hold time for LSA throttle 5000 msec
Maximum wait time for LSA throttle 5000 msec
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Maximum number of configured interfaces 255
Number of external LSA 0. Checksum Sum 00000000
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Graceful Restart enabled, last GR 11:12:26 ago (took 6 secs)
Area BACKBONE(0)
Number of interfaces in this area is 1
SPF algorithm executed 1 times
Number of LSA 6. Checksum Sum 0x0268a7
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0
```

# OSPF Version 2 のウォームスタンバイとノンストップルーティング

OSPFv2 ウォームスタンバイは、RP のスイッチオーバー全体でハイアベイラビリティを実現します。ウォームスタンバイ拡張機能により、アクティブ RP で実行されているプロセスごとに、スタンバイ RP で開始された、対応するスタンバイプロセスがあります。スタンバイ OSPF プロセスは、アクティブな OSPF プロセスにパフォーマンスへ影響を与えることなく、OSPF パケットを送受信できます。

ノンストップルーティング (NSR) によって、RP フェールオーバー、プロセスの再起動、またはインサービスアップグレードはピアルータから見えなくなり、パフォーマンスまたは処理への影響が最小限になります。ルーティングプロトコルはルータ間でやり取りされるため、NSR の影響を受けません。NSR はウォームスタンバイ拡張機能によって構築されます。NSR を使用すると Cisco NSF および IETF グレースフルリスタートプロトコル拡張機能の要件が緩和されます。

NSR は OSPF ではデフォルトで有効になっています。NSR を無効にするには、OSPF コンフィギュレーションモードで `nsr disable` コマンドを使用します。



- (注) Hello 間隔値はデフォルトの 10 秒に設定することをお勧めします。設定された Hello 間隔値がデフォルト値より小さい場合、スイッチオーバー中に OSPF セッションがフラップすることがあります。

## OSPFv2 のノンストップルーティングのイネーブル化

このオプションのタスクでは、OSPFv2 プロセスのノンストップルーティング (NSR) をイネーブルにする方法を説明します。NSR はデフォルトでディセーブルになっています。NSR をイネーブルにすると、アクティブな RP の OSPF プロセスは必要なすべてのデータと状態をスタンバイ RP の OSPF プロセスと同期します。スイッチオーバーが発生すると、新たにアクティブになった RP の OSPF プロセスには実行の継続に必要なすべてのデータと状態が含まれており、ネイバーからの援助は必要ありません。

### 手順

ステップ 1 **configure**

ステップ 2 **router ospf instance-id**

例 :

```
RP/0/RP0/cpu 0: router(config)# router ospf isp
```

指定したルーティングプロセスに OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードでルータを配置します。この例では、OSPF インスタンスは `isp` と呼ばれます。

### ステップ 3 nsr

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# nsr
```

OSPFv2 プロセスの NSR をイネーブルにします。

### ステップ 4 commit

## OSPFVersion3のウォームスタンバイとノンストップルーティング

この機能を使うと、フェールオーバー（FO）の前に OSPFv3 が自動で初期化され、障害が発生する前に機能する準備が整います。また、スイッチオーバー中のダウンタイムを減らすことができます。デフォルトでは、ルータは `hello` パケットを 40 秒ごとに送信します。

各 OSPF プロセスのウォームスタンバイ プロセスが、アクティブルートプロセッサで実行されている場合、対応する OSPF プロセスはスタンバイ RP で開始する必要があります。この機能のためにコンフィギュレーションを変更する必要はありません。

ウォームスタンバイは常にイネーブルです。この機能は、IGP として OSPFv3 を実行しているシステムが RP フェールオーバーを実行するときに有利です。

## OSPFv3 のノンストップルーティングのイネーブル化

このタスクでは、OSPFv3 プロセスのノンストップルーティング（NSR）をイネーブルにする方法を説明します。NSR はデフォルトでディセーブルになっています。NSR をイネーブルにすると、アクティブな RP の OSPF プロセスは必要なすべてのデータと状態をスタンバイ RP の OSPF プロセスと同期します。スイッチオーバーが発生すると、新たにアクティブになった RP の OSPF プロセスには実行の継続に必要なすべてのデータと状態が含まれており、ネイバーからの援助は必要ありません。

手順

### ステップ 1 configure

### ステップ 2 router ospfv3 instance-id

例：

```
RP/0/RP0/cpu 0: router(config)# router ospfv3 isp
```

指定したルーティングプロセスにOSPFルーティングをイネーブルにし、ルータ コンフィギュレーションモードでルータを配置します。この例では、OSPF インスタンスは `isp` と呼ばれます。

### ステップ3 nsr

例：

```
RP/0/RP0/cpu 0: router(config-ospfv3)# nsr
```

OSPFv3 プロセスのNSR をイネーブルにします。

### ステップ4 commit

## OSPFv2OSPF SPF プレフィックスのプライオリティ設定

OSPFv2OSPF SPF のプレフィックスのプライオリティ設定機能によって、ルートのインストール中に、高速モードで、管理者が重要なプレフィックスを収束できます。

多くのプレフィックスがルーティング情報ベース（RIB）および転送情報ベース（FIB）にインストールされる必要がある場合、SPF 中の、最初のプレフィックスから最後のプレフィックスまでの更新期間が、かなりの長さになることがあります。

時間依存のトラフィック（VoIP など）が他のトラフィック フローとともに同じルータを通過する可能性があるネットワークでは、SPF 中の、これらの時間に依存するプレフィックスの RIB および FIB アップデートを優先することが重要です。

OSPFv2OSPF SPF のプレフィックスのプライオリティ設定機能によって、SPF 計算中に RIB にインストールされる重要なプレフィックスに、管理者が優先順位を付けることが可能になります。重要なプレフィックスは、領域ごとに同じルートタイプのプレフィックス内で高速で収束します。RIB および FIB のインストール前に、ルートとプレフィックスは指定したルートポリシーに基づいて OSPF ローカル RIB のさまざまなプライオリティ バッチ キューに割り当てられます。RIB プライオリティ バッチ キューはプライオリティの高い順から「critical」、 「high」、 「medium」、 「low」に分類されます。

イネーブルの場合、次のプレフィックス プライオリティで RIB 更新シーケンスが変更されます。

**Critical > High > Medium > Low**

プレフィックスプライオリティが設定されると、デフォルトでは /32 プレフィックスは優先されなくなり、より高いプライオリティポリシーに一致しない場合は、low プライオリティキューに配置されます。ルート ポリシーは、/32 が高いプライオリティのキュー（High プライオリティ、または Medium プライオリティ）に保持されるように考案する必要があります。

プライオリティはルート ポリシーを使用して指定されます。このルート ポリシーは、IP アドレスまたはルート タグに基づいて照会することができます。SPF 中に、指定したルートポリシーに対してプレフィックスがチェックされ、適切な RIB バッチ プライオリティ キューに割り当てられます。



これらは、このシナリオの例です。

- **high** プライオリティ ルート ポリシーだけを指定した場合は、**medium** プライオリティに対してルート ポリシーは設定されません。
  - 許可されたプレフィックスは、**high** プライオリティ キューに配置されます。
  - /32 を含む一致しないプレフィックスは、**low** プライオリティ キューに配置されます。
- **high** プライオリティと **medium** プライオリティの両方のルート ポリシーが指定され、**critical** プライオリティにマップが指定されない場合
  - **high** プライオリティのルート ポリシーに一致する許可されたプレフィックスは、**high** プライオリティ キューに配置されます。
  - **medium** プライオリティのルート ポリシーに一致する許可されたプレフィックスは、**medium** プライオリティ キューに配置されます。
  - /32 を含む一致しないプレフィックスは、**low** プライオリティ キューに移動されます。
- **critical** プライオリティと **high** プライオリティの両方のルート ポリシーが指定されており、**medium** プライオリティにマップが指定されていない場合
  - **critical** プライオリティのルート ポリシーに一致する許可されたプレフィックスは、**critical** プライオリティ キューに配置されます。
  - **high** プライオリティのルート ポリシーに一致する許可されたプレフィックスは、**high** プライオリティ キューに配置されます。
  - /32 を含む一致しないプレフィックスは、**low** プライオリティ キューに配置されます。
- **medium** プライオリティ ルート ポリシーだけが指定され、**high** プライオリティまたは **critical** プライオリティにマップが指定されていない場合
  - **medium** プライオリティのルート ポリシーに一致する許可されたプレフィックスは、**medium** プライオリティ キューに割り当てられます。
  - /32 を含む一致しないプレフィックスは、**low** プライオリティ キューに配置されます。

**[no] spf prefix-priority route-policy *rpl*** コマンドを使用して、SPF 中に OSPFv2OSPF プレフィックス インストールのプライオリティをグローバル RIB で設定します。

SPF プレフィックスのプライオリティ設定は、デフォルトではディセーブルです。ディセーブルモードでは、/32 プレフィックスは他のプレフィックスよりも前にグローバル RIB にインストールされます。SPF プライオリティ設定がイネーブルの場合、ルートは route-policy 基準に対して照会され、SPF プライオリティ セットに基づいて適切なプライオリティ キューに割り当てられます。/32 を含む一致しないプレフィックスは、**low** プライオリティのキューに配置されます。

すべての /32 を **high** プライオリティ キューまたは **medium** プライオリティ キューで処理する必要がある場合、次の 1 つのルート マップを設定します。

```
prefix-set ospf-medium-prefixes
 0.0.0.0/0 ge 32
end-set
```

## OSPFv2 OSPF SPF プレフィックス プライオリティの設定

このタスクを実行して、OSPFv2 OSPF SPF (Shortest Path First) プレフィックスプライオリティを設定します。

### 手順

#### ステップ1 **configure**

#### ステップ2 **prefix-set** *prefix-set name*

例：

```
RP/0/RP0/cpu 0: router(config)#prefix-set ospf-critical-prefixes
RP/0/RP0/cpu 0: router(config-pfx)#66.0.0.0/16
RP/0/RP0/cpu 0: router(config-pfx)#end-set
```

プレフィックスセットを設定します。

#### ステップ3 **route-policy** *route-policy name* **if destination in** *prefix-set name* **then set spf-priority** {critical | high | medium} **endif**

例：

```
RP/0/RP0/cpu 0: router#route-policy ospf-spf-priority
RP/0/RP0/cpu 0: router(config-rpl)#if destination in ospf-critical-prefixes then
  set spf-priority critical
endif
RP/0/RP0/cpu 0: router(config-rpl)#end-policy
```

ルートポリシーと OSPF SPF プライオリティを設定します。

#### ステップ4 次のいずれかのコマンドを使用します。

- **router ospf** *ospf-name*
- **router ospfv3** *ospfv3-name*

例：

```
RP/0/RP0/cpu 0: router# router ospf 1
```

または

```
RP/0/RP0/cpu 0: router# router ospfv3 1
```

ルータ OSPF 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 **router ospf** *ospf name*

例：

```
RP/0/RP0/cpu 0: router# router ospf 1
```

ルータ OSPF 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 `spf prefix-priority route-policy route-policy name`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# spf prefix-priority route-policy ospf-spf-priority
```

または

```
RP/0/RP0/cpu 0: router(config-ospfv3)#spf prefix-priority route-policy ospf3-spf-priority
```

定義されているルート ポリシーの SPF プレフィックス プライオリティを設定します。

(注) OSPF ルータで `spf prefix-priority` コマンドを設定します。

#### ステップ7 `commit`

#### ステップ8 `show rpl route-policy route-policy name detail`

例：

```
RP/0/RP0/cpu 0: router#show rpl route-policy ospf-spf-priority detail
  prefix-set ospf-critical-prefixes
    66.0.0.0/16
  end-set
  !
  route-policy ospf-spf-priority
    if destination in ospf-critical-prefixes then
      set spf-priority critical
    endif
  end-policy
  !
```

SPF のプレフィックス プライオリティのセットを表示します。

---

## OSPFv2

### OSPFv3

この例では、/32 プレフィックスを一般的に **medium** プライオリティに設定し、一部の /32 および /24 プレフィックスを **critical** プライオリティおよび **high** プライオリティ キューに設定する方法を示します。

```
prefix-set ospf-critical-prefixes
  192.41.5.41/32,
  11.1.3.0/24,
  192.168.0.44/32
end-set
!
prefix-set ospf-high-prefixes
  44.4.10.0/24,
```

```
192.41.4.41/32,
41.4.41.41/32
end-set
!
prefix-set ospf-medium-prefixes
0.0.0.0/0 ge 32
end-set
!

route-policy ospf-priority
  if destination in ospf-high-prefixes then
    set spf-priority high
  else
    if destination in ospf-critical-prefixes then
      set spf-priority critical
    else
      if destination in ospf-medium-prefixes then
        set spf-priority medium
      endif
    endif
  endif
end-policy

router ospf 1
  spf prefix-priority route-policy ospf-priority
  area 0
    interface TenGigE 0/0/0/1
    !
  !
  area 3
    interface TenGigE 0/0/0/0
    !
  !
  area 8
    interface TenGigE 0/0/0/0

router ospfv3 1
  spf prefix-priority route-policy ospf-priority
  area 0
    interface TenGigE 0/0/0/1
    !
  !
  area 3
    interface TenGigE 0/0/0/0
    !
  !
  area 8
    interface TenGigE 0/0/0/0
```

# プロバイダーエッジからカスタマーエッジ (PE-CE) プロトコルとしての OSPF の設定

## 手順

ステップ1 **configure**

ステップ2 **router ospf process-name**

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティングプロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

ステップ3 **vrf vrf-name**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# vrf vrf1
```

VRF インスタンスを作成し、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4 **router-id { router-id }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# router-id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

ステップ5 **redistribute protocol [process-id] { level-1 | level-1-2 | level-2 } [ metric metric-value ] [ metric-type type-value ] [ match { external [ 1 | 2 ] } ] [ tag tag-value ] route-policy policy-name**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# redistribute bgp 1 level-1
```

1つのルーティングドメインから別のルーティングドメインへの OSPF ルートの再配布

- このコマンドを実行すると、定義上ルータが ASBR になります。
- OSPF は再配布によって学習したすべてのルートを **external** とタグ付けします。
- プロトコルとそのプロセス ID (設定されている場合) は、OSPF に再配布されるプロトコルを示します。

- メトリックは外部ルートに割り当てるコストです。すべてのプロトコルでデフォルトは20です。ただし、BGP のデフォルトのメトリックは1です。
- この例では、BGP 自律システム 1、レベル 1 のルートを OSPF にタイプ 2 の外部ルートとして再配布します。

#### ステップ 6 **area** *area-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# area 0
```

エリア コンフィギュレーション モードを開始し、OSPF プロセスのエリアを設定します。

- *area-id* 引数は、area 1000 や area 0.0.3.232 など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。

#### ステップ 7 **interface** *type interface-path-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、1つ以上のインターフェイスを VRF に関連付けます。

#### ステップ 8 **exit**

例：

```
RP/0/RP0/cpu 0: router(config-if)# exit
```

インターフェイス コンフィギュレーション モードを終了します。

#### ステップ 9 **domain-id** [*secondary*] **type** {**0005** | **0105** | **0205** | **8005**} **value** *value*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# domain-id type 0105 value 1AF234
```

OSPF VRF のドメイン ID を指定します。

- *value* 引数は 6 オクテットの 16 進数です。

#### ステップ 10 **domain-tag** *tag*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# domain-tag 234
```

OSPF VRF ドメイン タグを指定します。

- *tag* の有効範囲は、0 ~ 4294967295 です。

#### ステップ 11 **disable-dn-bit-check**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# disable-dn-bit-check
```

ダウンビットを無視するように設定します。

#### ステップ 12 commit

---

## 複数の OSPF インスタンスの作成 (OSPF プロセスおよび VRF)

このタスクでは、複数の OSPF インスタンスの作成方法を説明します。この場合、インスタンスは通常の OSPF インスタンスと VRF インスタンスです。

手順

---

#### ステップ 1 configure

#### ステップ 2 router ospf process-name

例 :

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティングプロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

#### ステップ 3 area area-id

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)# area 0
```

エリア コンフィギュレーションモードを開始し、バックボーンエリアを設定します。

- *area-id* 引数は、*area 1000* や *area 0.0.3.232* など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

#### ステップ 4 interface type interface-path-id

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

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て、1つ以上のインターフェイスをエリアに関連付けます。

#### ステップ5 **exit**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# exit
```

OSPF 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 vrf-name**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# vrf vrf1
```

VRF インスタンスを作成し、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7 **area area-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# area 0
```

エリアコンフィギュレーションモードを開始して、OSPF プロセスで VRF インスタンスのエリアを設定します。

- *area-id* 引数は、*area 1000* や *area 0.0.3.232* など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。

#### ステップ8 **interface type interface-path-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# interface TenGigE 0/0/0/0
```

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て、1つ以上のインターフェイスを VRF に関連付けます。

#### ステップ9 **commit**

---

## OSPF のラベル配布プロトコル IGP 自動設定

ラベル配布プロトコル (LDP) 内部ゲートウェイプロトコル (IGP) 自動設定を使うと、OSPF などの IGP インスタンスに使用されているインターフェイスのセットで LDP をイネーブルにする手順を簡略化できます。LDP IGP 自動設定は、多数のインターフェイス（転送に LDP がコアで使用される場合など）および複数の OSPF インスタンスで同時に使用できます。

この機能は、デフォルトの VPN ルーティングおよび転送 (VRF) インスタンスとして IPv4 ユニキャストアドレスファミリをサポートします。



LDP IGP 自動設定は、LDP の個々のインターフェイスベースで `igp auto-config disable` コマンドを使用して明示的にディセーブルにすることもできます。これにより、明示的にディセーブルにしたインターフェイスを除くすべての OSPF インターフェイスを LDP で受信できます。

## OSPF のラベル配布プロトコル IGP 自動設定の設定

このタスクでは、OSPF インスタンスに対する LDP 自動設定を設定する方法について説明します。

オプションで、OSPF インスタンスのエリアにこの機能を設定できます。

### 手順

---

#### ステップ 1 `configure`

#### ステップ 2 `router ospf process-name`

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティング プロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーション モードでルータを配置します。

(注) `process-name` 引数は、40 文字未満の英数字です。

#### ステップ 3 `mpls ldp auto-config`

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# mpls ldp auto-config
```

OSPF インスタンスの LDP IGP インターフェイスの自動設定をイネーブルにします。

- 任意で、このコマンドを OSPF インスタンスのエリアに設定されます。

#### ステップ 4 `commit`

---

## LDP IGP 同期の設定：OSPF

このタスクを実行して、OSPF で LDP IGP 同期を設定します。



---

(注) デフォルトでは、LDP と IGP 間の同期は行われません。

---

## 手順

ステップ1 **configure**ステップ2 **router ospf process-name**

例 :

```
RP/0/RP0/cpu 0: router(config)# router ospf 100
```

OSPF ルーティング プロセスを識別し、OSPF 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 (任意) **vrf vrf-name**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)# vrf red
```

デフォルト以外の VRF を指定します。

## ステップ4 次のいずれかのコマンドを使用します。

- **mpls ldp sync**
- **area area-id mpls ldp sync**
- **area area-id interface name mpls ldp sync**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf)# mpls ldp sync
```

インターフェイスで LDP IGP 同期をイネーブルにします。

## ステップ5 (任意) 次のいずれかのコマンドを使用します。

- **mpls ldp sync**
- **area area-id mpls ldp sync**
- **area area-id interface name mpls ldp sync**

例 :

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# mpls ldp sync
```

```
RP/0/RP0/cpu 0: router(config-ospf-vrf)# area 1 mpls ldp sync
```

指定された VRF のインターフェイス上で LDP IGP 同期を有効にします。

ステップ6 **commit**ステップ7 (任意) **show mpls ldp vrf vrf-name ipv4 igp sync**

例 :

```
RP/0/RP0/cpu 0: router# show mpls ldp vrf red ipv4 igp sync
```

アドレスファミリ IPv4 の指定された VRF に関する LDP IGP 同期情報を表示します。

**ステップ 8** (任意) **show mpls ldp vrf all ipv4 igp sync**

例 :

```
RP/0/RP0/cpu 0: router# show mpls ldp vrf all ipv4 igp sync
```

アドレスファミリ IPv4 のすべての VRF に関する LDP IGP 同期情報を表示します。

**ステップ 9** (任意) **show mpls ldp { ipv4 | ipv6 } igp sync**

例 :

```
RP/0/RP0/cpu 0: router# show mpls ldp ipv4 igp sync
```

```
RP/0/RP0/cpu 0: router# show mpls ldp ipv6 igp sync
```

IPv4 または IPv6 アドレスファミリの LDP IGP 同期情報を表示します。

---

**例**

次に、OSPF の LDP IGP 同期を設定する例を示します。

```
router ospf 100
mpls ldp sync
!
mpls ldp
  igp sync delay 30
!
```

## OSPF 認証のメッセージダイジェスト管理

すべての OSPF ルーティングプロトコル交換は認証されます。使用される方法は、認証が設定される方法によって異なります。暗号認証を使用する場合、OSPF ルーティングプロトコルは、Message Digest 5 (MD5) 認証アルゴリズムを使用してネットワーク内のネイバー間で送信されたパケットを認証します。各 OSPF プロトコルパケットでは、キーを使用して、OSPF パケットの最後に付加されるメッセージダイジェストを生成および検証します。メッセージダイジェストは OSPF プロトコルパケットおよび秘密キーの単方向機能です。各キーは使用されるインターフェイスとキー ID の組み合わせで識別されます。インターフェイスでは、複数のキーが常にアクティブになっています。

キーのロールオーバーを管理し、OSPF の MD5 認証を拡張するには、キーチェーンと呼ばれるキーのコンテナを設定できます。この各キーは、生成/受け取り時間、キー ID、認証アルゴリズムの属性で構成されます。

## OSPFの認証メッセージダイジェスト管理の設定

このタスクでは、OSPF インターフェイスのキーチェーンの認証を管理する方法を説明します。

### 始める前に

このタスクを実行するには、有効なキーチェーンを設定する必要があります。

### 手順

#### ステップ1 **configure**

#### ステップ2 **router ospf** *process-name*

例：

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティング プロセスに OSPF ルーティングを有効にし、ルータ コンフィギュレーション モードでルータを配置します。

(注) *process-name* 引数は、40 文字未満の英数字です。

#### ステップ3 **router-id** { *router-id* }

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router id 192.168.4.3
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

#### ステップ4 **area** *area-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 1
```

エリア コンフィギュレーション モードを開始します。

*area-id* 引数は、*area 1000* や *area 0.0.3.232* など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

#### ステップ5 **interface** *type interface-path-id*

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE 0/0/0/0
```

インターフェイス コンフィギュレーション モードを開始して、1つ以上のインターフェイスをエリアに関連付けます。

**ステップ6 authentication [ message-digest keychain | keychain keychain | null]**

MD5キーチェーンを設定します。キーチェーンは、次の3つのレベルの認証で設定できます。

- ルータレベルの認証
- エリアレベルの認証
- インターフェイスレベルの認証

例：

次に、キーチェーン認証の設定例を示します。

```
RP/0/RP0/CPU0:router(config-ospf)# authentication keychain test_chain -----□ Router-level
authentication
RP/0/RP0/CPU0:router(config-ospf)# router-id 1.1.1.1
RP/0/RP0/CPU0:router(config-ospf)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-ospf)# area 1
RP/0/RP0/CPU0:router(config-ospf-ar)# authentication keychain test_chain -----□ Area-level
authentication
RP/0/RP0/CPU0:router(config-ospf-ar)# interface TenGigE 0/0/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# authentication keychain test_chain ---□
Interface-level authentication
RP/0/RP0/CPU0:router(config-ospf-ar-if)# commit
```

例：

次に、message-digest 認証の設定例を示します。

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# authentication message-digest keychain ospf_int1
```

(注) 上記の例では、この手順を実行する前に *ospf\_int1* キーチェーンを設定する必要があります。

**ステップ7 commit**

例

次の例は、5つのキーIDを持つキーチェーン *ospf\_intf\_1* の設定方法を示します。各キーIDは異なる **send-lifetime** 値で設定されます。ただし、すべてのキーIDはキーに同じテキスト文字列を指定します。

```
key chain ospf_intf_1
key 1
send-lifetime 11:30:30 May 1 2007 duration 600
cryptographic-algorithm MD5T
key-string clear ospf_intf_1
key 2
send-lifetime 11:40:30 May 1 2007 duration 600
cryptographic-algorithm MD5
key-string clear ospf_intf_1
key 3
send-lifetime 11:50:30 May 1 2007 duration 600
cryptographic-algorithm MD5
```

```

key-string clear ospf_intf_1
key 4
send-lifetime 12:00:30 May 1 2007 duration 600
cryptographic-algorithm MD5
key-string clear ospf_intf_1
key 5
send-lifetime 12:10:30 May 1 2007 duration 600
cryptographic-algorithm MD5
key-string clear ospf_intf_1

```

次の例は、TenGigE 0/0/0/0 インターフェイスでキーチェーン認証がイネーブルであることを示します。

```
show ospf 1 interface TenGigE 0/0/0/0
```

```

TenGigE 0/0/0/0 is up, line protocol is up
Internet Address 100.10.10.2/24, Area 0
Process ID 1, Router ID 2.2.2.1, Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 2.2.2.1, Interface address 100.10.10.2
Backup Designated router (ID) 1.1.1.1, Interface address 100.10.10.1
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:02
Index 3/3, flood queue length 0
Next 0(0)/0(0)
Last flood scan length is 2, maximum is 16
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 1.1.1.1 (Backup Designated Router)
Suppress hello for 0 neighbor(s)
Keychain-based authentication enabled
  Key id used is 3
Multi-area interface Count is 0

```

次に、アクティブに設定されたキーの出力の例を示します。

```
show key chain ospf_intf_1
```

```

Key-chain: ospf_intf_1/ -

Key 1 -- text "0700325C4836100B0314345D"
  cryptographic-algorithm -- MD5
  Send lifetime: 11:30:30, 01 May 2007 - (Duration) 600
  Accept lifetime: Not configured
Key 2 -- text "10411A0903281B051802157A"
  cryptographic-algorithm -- MD5
  Send lifetime: 11:40:30, 01 May 2007 - (Duration) 600
  Accept lifetime: Not configured
Key 3 -- text "06091C314A71001711112D5A"
  cryptographic-algorithm -- MD5
  Send lifetime: 11:50:30, 01 May 2007 - (Duration) 600 [Valid now]
  Accept lifetime: Not configured
Key 4 -- text "151D181C0215222A3C350A73"
  cryptographic-algorithm -- MD5
  Send lifetime: 12:00:30, 01 May 2007 - (Duration) 600
  Accept lifetime: Not configured
Key 5 -- text "151D181C0215222A3C350A73"
  cryptographic-algorithm -- MD5
  Send lifetime: 12:10:30, 01 May 2007 - (Duration) 600

```

```
Accept lifetime: Not configured
```

## OSPFのGTSM TTLセキュリティメカニズム

OSPFは、ネイバーに対するネットワーク、フラッディングリンクステートアドバタイズメント (LSA) アップデートで、トポロジの変更を検出し、トポロジの新しいビュー上ですばやくコンバージするためにネットワークングデバイスを必要とするリンクステートプロトコルです。ただし、ネイバーからのLSAの受信動作中は、ネットワーク攻撃が発生する可能性があります。これは、ユニキャストパケットが仮想リンクの1ホップまたは複数ホップ向こう側に配置されているネイバーから送信されているという確認ができないためです。

仮想リンクについては、OSPFパケットはネットワーク全体の複数ホップを通過して送信されます。したがって、TTL値は複数回にわたり減少していく可能性があります。このようなリンクの種類では、最小TTL値が複数ホップパケットで許可され受け入れられなければなりません。

複数ホップを通過して送信される無効なソースから発生するネットワーク攻撃をフィルタリングするには、一般TTLセキュリティメカニズム (GTSM) のRFC 3682を使用して、攻撃を防止します。GTSMはリンクローカルアドレスをフィルタリングして、TTL値255のコンフィギュレーションの1ホップネイバーとなる隣接関係だけを許可します。IPヘッダーのTTL値はOSPFパケットが生成されるときに255に設定され、受信されたOSPFパケットでデフォルトのGTSM TTL値255またはユーザ設定されたGTSM TTL値に対してチェックされます。このようにして、TTLホップを超える不正なOSPFパケットをブロックします。

## OSPFの一般TTLセキュリティメカニズム (GTSM) の設定

このタスクでは、GTSMのインターフェイスにおけるセキュリティの存続可能時間メカニズムの設定方法を説明します。

### 手順

- ステップ1 **configure**
- ステップ2 **router ospf *process-name***

例:

```
RP/0/RP0/cpu 0: router(config)# router ospf 1
```

指定したルーティングプロセスにOSPFルーティングを有効にし、ルータコンフィギュレーションモードでルータを配置します。

(注) *process-name* 引数は、40文字未満の英数字です。

- ステップ3 **router-id { *router-id* }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# router id 10.10.10.100
```

OSPF プロセスのルータ ID を設定します。

(注) 固定 IPv4 アドレスをルータ ID として使用することを推奨します。

#### ステップ 4 **log adjacency changes [ detail | disable ]**

例：

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# log adjacency changes detail
```

(任意) ネイバー変更の通知を要求します。

- デフォルトでは、この機能はイネーブルです。
- ネイバー変更によって生成されたメッセージは通知と見なされます。このメッセージは `logging console` コマンドで重大度レベル 5 に分類されます。 `logging console` コマンドではどの重大度レベルのメッセージをコンソールに送信するかを制御します。デフォルトでは、すべての重大度レベルのメッセージが送信されます。

#### ステップ 5 **nsf { cisco [ enforce global ] | ietf [ helper disable ] }**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# nsf ietf
```

(任意) NSF OSPF プロトコルを設定します。

この例ではグレースフル リスタートをイネーブルにします。

#### ステップ 6 **timers throttle spf spf-start spf-hold spf-max-wait**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# timers throttle spf 500 500 10000
```

(任意) SPF スロットリング タイマーを設定します。

#### ステップ 7 **area area-id**

例：

```
RP/0/RP0/cpu 0: router(config-ospf)# area 1
```

エリア コンフィギュレーション モードを開始します。

`area-id` 引数は、`area 1000` や `area 0.0.3.232` など、ドット付き 10 進表記または IPv4 アドレス形式で入力できます。ただし、1つのエリアでは同じ形式を選択する必要があります。IPv4 アドレス形式を使用することを推奨します。

#### ステップ 8 **interface type interface-path-id**

例：



```
RP/0/RP0/cpu 0: router(config-ospf-ar)# interface TenGigE0/0/0/0
```

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て、1つ以上のインターフェイスをエリアに関連付けます。

**ステップ9** `security ttl [ disable | hops hop-count ]`

例:

```
RP/0/RP0/cpu 0: router(config-ospf-ar-if)# security ttl hops 2
```

OSPF パケットの IP ヘッダーのセキュリティ TTL 値を設定します。

**ステップ10** `commit`

**ステップ11** `show ospf [ process-name ][ area-id ] interface [ type interface-path-id ]`

例:

```
RP/0/RP0/cpu 0: router# show ospf 1 interface TenGigE0/0/0/0
```

OSPF インターフェイス情報を表示します。

---

**例**

OSPF インターフェイスに設定されている GTSM セキュリティ TTL 値を表示する出力例を次に示します。

```
show ospf 1 interface TenGigE0/5/0/0
```

```
TenGigE0/0/0/0 is up, line protocol is up
  Internet Address 120.10.10.1/24, Area 0
  Process ID 1, Router ID 100.100.100.100, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State BDR, Priority 1
  TTL security enabled, hop count 2
  Designated Router (ID) 102.102.102.102, Interface address 120.10.10.3
  Backup Designated router (ID) 100.100.100.100, Interface address 120.10.10.1
  Flush timer for old DR LSA due in 00:02:36
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Index 1/1, flood queue length 0
  Next 0(0)/0(0)
  Last flood scan length is 1, maximum is 4
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 102.102.102.102 (Designated Router)
  Suppress hello for 0 neighbor(s)
  Multi-area interface Count is 0
```

## OSPFの参照

OSPFを実装するには、次の概念を理解する必要があります。

### OSPF機能の概要

OSPFは、IP用のルーティングプロトコルです。これは、ディスタンスベクトルプロトコルではなく、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は、送信元マシンと宛先マシンを接続するリンクの状態に基づいて、ルーティングの決定を行います。リンクステートは、インターフェイスと、その隣接ネットワークングデバイスとの関係を説明するものです。インターフェイス情報には、インターフェイスのIPアドレス、ネットワークマスク、接続されているネットワークの種類、そのネットワークに接続されているルータなどがあります。この情報は、さまざまなタイプのリンクステートアドバタイズメント（LSA）によって伝播します。

ルータは受信したLSAデータの集まりをリンクステートデータベースに格納します。このデータベースにはこのルータのリンクのLSAデータが含まれます。ダイクストラアルゴリズムが採用されている場合、データベースの内容からデータが抽出されてOSPFルーティングテーブルが作成されます。データベースとルーティングテーブルの違いは、データベースにはすべてのrawデータが含まれており、ルーティングテーブルには特定のルータインターフェイスポートを介した既知の宛先への最短パスのリストが含まれていることです。

OSPFは大規模ネットワークにまで拡張できるため、IGPとして適しています。エリアを使用してネットワークをより管理しやすい大きさに分割するとともに、ネットワークに階層を導入します。ルータはネットワークの1つのエリアまたは複数のエリアに接続されます。エリア内のすべてのネットワークングデバイスは、デバイスが属するエリア内のみのリンクステートがすべて揃った、同じデータベース情報を維持します。ネットワーク内のすべてのリンクステートについての情報は持ちません。エリア内のルータ間におけるデータベース情報の合意はコンバージェンスと呼ばれます。

ドメイン内レベルで、OSPFはIntermediate System-to-Intermediate System（IS-IS）を使用して取得したルートを取り込むことができます。OSPFルートをIS-ISに伝達することもできます。ドメイン間レベルで、OSPFはボーダーゲートウェイプロトコル（BGP）を使用して取得したルートを取り込むことができます。OSPFルートをBGPに伝達することもできます。

Routing Information Protocol（RIP）とは異なり、OSPFは定期的なルーティングアップデートを送りません。OSPFルータはネイバーになると、データベースを交換および同期することによって隣接関係を確立します。その後、変更されたルーティング情報だけが伝播されます。エリア内のすべてのルータは自分のリンクのコストとステートをアドバタイズします。この情報はLSA内で送られます。このステート情報は、1ホップ先のすべてのOSPFネイバーに送られます。その後すべてのOSPFネイバーは、ステート情報を変更せずに送信します。このフラッディングプロセスは、エリア内のすべてのデバイスが同じリンクステートデータベースを持つまで続けられます。

宛先への最適なルートを決めるために、宛先へのルートに含まれるリンクのすべてのコストがソフトウェアによって合計されます。各ルータが別のネットワークングデバイスからルー

ティング情報を受信した後で、Shortest Path First (SPF) アルゴリズムが実行されて、データベース内の各宛先ネットワークへの最適なパスが計算されます。

OSPF を実行しているネットワーキング デバイスは、ネットワーク内のトポロジの変化を検出して、リンクステート アップデートをネイバーにフラッディングし、新しいトポロジビューをすぐに収束させます。ネットワーク内の各 OSPF ルータは、すぐに再び同じトポロジビューを持ちます。OSPF は、同じ宛先に対する複数の等コストのパスを許容します。すべてのリンクステート情報がフラッディングされて SPF 計算に使用されるため、複数の等コストパスが計算されてルーティングに使用されることがあります。

ブロードキャスト ネットワークおよび非ブロードキャスト マルチアクセス (NBMA) ネットワークでは、指定ルータ (DR) またはバックアップ DR が LSA フラッディングを実行します。

OSPF は直接 IP の上で実行され、TCP やユーザ データグラム プロトコル (UDP) を使用しません。OSPF はパケット ヘッダーおよび LSA のチェックサムを使用してそれ自体でエラー訂正を実行します。

OSPFv3 は、基本概念は OSPF Version 2 と同じですが、IPv6 の拡大されたアドレス サイズのサポートが追加されています。IPv6 のアドレスとプレフィックスを伝送するために新しい LSA タイプが作成され、個々の IP サブネットベースではなく、個々のリンク ベースでプロトコルが実行されます。

OSPF は通常多くの内部ルータ間の調整を必要とします。このようなルータには、複数のエリアに接続されたエリア境界ルータ (ABR) や、他のソース (IS-IS、BGP、静的ルートなど) からの再ルーティングを OSPF トポロジに伝達する自律システム境界ルータ (ASBR) があります。OSPF ベースのルータまたはアクセス サーバの最小設定では、すべてのデフォルトパラメータ値、およびエリアに割り当てられたインターフェイスが使用され、認証は行われません。環境をカスタマイズする場合は、すべてのルータの調和が取れた設定が必要です。

## Cisco IOS XR ソフトウェアの OSPFv3 と OSPFv2 の比較

OSPFv3 プロトコルの大半は OSPFv2 と同じです。OSPFv3 は RFC 2740 に記載されています。

Cisco IOS XR ソフトウェアの OSPFv3 プロトコルと OSPFv2 プロトコルの主な相違点は、次のとおりです。

- OSPFv2 を拡張した OSPFv3 では、IPv6 ルーティング プレフィッ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。
- NBMA インターフェイスを OSPFv3 で使用する場合、ユーザは、ネイバーのリストを使用してルータを手動で設定する必要があります。隣接ルータはネイバーに接続されたインターフェイスのリンク ローカルアドレスによって識別されます。
- OSPFv2 とは異なり、複数の OSPFv3 プロセスをリンク上で実行できます。
- OSPFv3 の LSA は、「アドレスとマスク」ではなく、「プレフィックスとプレフィックス長」として表現されます。
- ルータ ID は IPv6 アドレスとは無関係な 32 ビットの数値です。

## OSPFの階層 CLI および CLI 継承

階層 CLI とは、定義された階層レベル（ルータ レベル、エリア レベル、インターフェイス レベルなど）で、ネットワーク コンポーネント情報がグループ化されたものです。階層 CLI を使うと、OSPF の設定、メンテナンス、トラブルシューティングをより簡単に行えます。コンフィギュレーションコマンドと一緒に階層コンテキストに表示されると、視覚的な検査が簡単になります。階層 CLI はサポートされる CLI 継承自体に備わっています。

CLI 継承を使うと、エリアやインターフェイスのパラメータを明示的に設定する必要がありません。ソフトウェアでは、同じエリアのインターフェイスのパラメータだけを1つのコマンドで設定できます。また、エリア コンフィギュレーション レベルやルータ OSPF コンフィギュレーション レベルなどの高い階層レベルからパラメータ値を継承できます。

たとえば、インターフェイスの `hello interval` 値は、IF ステートメントの優先順位によって次のように決まります。

インターフェイス コンフィギュレーション レベルで `hello interval` コマンドが設定されている場合は、インターフェイスに設定されている値を使用します。

エリア コンフィギュレーション レベルで `hello interval` コマンドが設定されている場合は、エリアに設定されている値を使用します。

ルータ コンフィギュレーションレベルで `hello interval` コマンドが設定されている場合は、ルータ設定されている値を使用します。

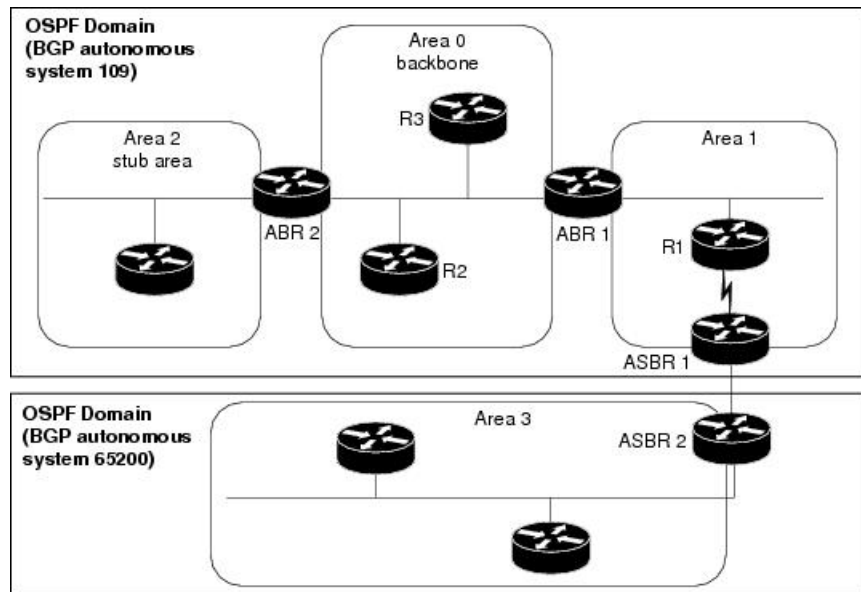
その他の場合は、コマンドのデフォルト値を使用します。

## OSPF ルーティング コンポーネント

OSPF を実装する前に、ルーティング コンポーネントの概要とその使用目的を把握する必要があります。これらは自律システム、エリア タイプ、内部ルータ、ABR、および ASBR で構成されます。

図 4: OSPF ルーティング コンポーネント

次の図に、OSPF ネットワークのトポロジのルーティング コンポーネントを示します。



88721

## 自律システム

自律システムは、同じ管理制御下で、相互にルーティング情報を共有するネットワークの集合です。自律システムは、ルーティングドメインとも呼ばれます。図1：OSPFルーティングコンポーネントには、109と65200の2つの自律システムが示されています。自律システムは1つまたは複数のOSPFエリアで構成されます。

## エリア

エリアでは、自律システムをより小さく管理しやすいネットワークや隣接ネットワークのセットに再分割できます。図1：OSPFルーティングコンポーネントに示すように、自律システム109はエリア0、エリア1、エリア2の3つのエリアから構成されます。

OSPFは1つのエリアのトポロジをその他の自律システムから見えないようにします。1つのエリアのネットワークトポロジはそのエリア内のルータにのみ認識されます。OSPFルーティングがエリア内にある場合、そのルーティングはエリア内ルーティングと呼ばれます。このルーティングは、ネットワークにフラディングするリンクステート情報量を制限して、ルーティングトラフィックを少なくします。各ルータのトポロジ情報のサイズも小さくし、各ルータの処理と必要なメモリを節約します。

また、エリア内のルータはエリア外の詳細なネットワークトポロジを見ることはできません。このようにトポロジ情報の開示が制限されているため、自律システム全体が1つのルーティングドメインであるときに、エリア間のトラフィックフローを制御して、ルーティングトラフィックを少なくすることができます。

## バックボーンエリア

バックボーンエリアは、自律システムの複数エリア間でルーティング情報を配布する役割を担当します。エリアの外で発生するOSPFルーティングをエリア間ルーティングと呼びます。

エリアのプロパティはすべてバックボーン自体にあります。これは、バックボーンだけにある ABR、ルータ、ネットワークで構成されます。図 1 : OSPF ルーティング コンポーネントに示すように、エリア 0 は OSPF バックボーン エリアです。すべての OSPF バックボーン エリアでは、0.0.0.0 の ID が予約されています。

## ルータ

OSPF ネットワークは ABR、ASBR、内部ルータで構成されます。

### エリア境界ルータ

エリア境界ルータ (ABR) は複数のエリアのネットワークに直接接続する複数のインターフェイスを持つルータです。ABR は OSPF アルゴリズムのコピーを個別に実行し、バックボーン エリアを含む、アタッチされる各エリアに対する個別のルーティングデータを保持します。また、ABR はアタッチされたエリアの設定の集約をバックボーン エリアに送り、バックボーン エリアではこの情報を自律システム内の他の OSPF エリアに配布します。図 1 : OSPF ルーティング コンポーネント セクションには、2 つの ABR があります。ABR 1 はバックボーン エリアに対するエリア 1 のインターフェイスとなります。ABR 2 はスタブ エリアであるエリア 2 に対するバックボーン エリア 0 のインターフェイスとなります。

### 自律システム境界ルータ (ASBR)

自律システム境界ルータ (ASBR) を使用すると、1 つの自律システムから別のシステムに接続できるようになります。ASBR は自律システムルーティング情報を他の自律システムの境界ルータと交換します。自律システム内のすべてのルータは、その自律システムの境界ルータに到達する方法を情報として保有しています。

ASBR は、BGP などの他のプロトコルから外部ルーティング情報をインポートして、それらをネットワークに AS-External (ASE) タイプ 5 LSA として再配布できます。Cisco IOS XR ルータが ASBR の場合、コンテンツの VIP アドレスを自律システムの外部ルートとしてアドバタイズするようにルータを設定できます。このようにして、ASBR は OSPF ネットワーク内のルータに外部ネットワークに関する情報をフラッドします。

ASBR ルートは、タイプ 1 またはタイプ 2 の ASE としてアドバタイズできます。タイプ 1 とタイプ 2 ではコストの計算方法が異なります。タイプ 2 ASE では、同じ宛先への複数パスを比較するとき、外部コスト (メトリック) のみが考慮されます。タイプ 1 ASE では、外部コストと ASBR に到達するためのコストの組み合わせが使用されます。タイプ 2 の外部コストがデフォルトであり、常に OSPF ルートよりコストがかかるため、OSPF ルートが存在しない場合にのみ使用されます。

### 内部ルータ

内部ルータ (図 1 : OSPF ルーティング コンポーネントの R1 など) は 1 つの領域に接続されています (たとえば、同一の領域に存在するすべてのインターフェイス) 。

## OSPF プロセスおよびルータ ID

OSPF プロセスは、物理ルータで OSPF を実行している論理ルーティング エンティティです。システム管理者が物理ボックスをパーティションで個別のルータに区切ることができる論理ルーティング機能がありますが、その機能とこの論理ルーティングエンティティを混同しないでください。

物理ルータは複数の OSPF プロセスを実行できます。ただし、複数のプロセスを実行するのは、複数の OSPF ドメインに接続する場合のみです。各プロセスにはそれぞれのリンクステートデータベースがあります。ルーティング テーブルのルートはリンクステートデータベースから計算されます。ルートが再配布されないかぎり、1 つの OSPF プロセスは別の OSPF プロセスとルートを共有しません。

各 OSPF プロセスは、ルータ ID で識別されます。ルータ ID はルーティング ドメイン全体で一意である必要があります。OSPF はルータ ID を優先度の高い順に次の送信元から取得します。

- デフォルトでは、OSPF プロセスが初期化されると、チェックポイント データベースに `router-id` があるかどうかをチェックします。
- ルータ コンフィギュレーション モードで `OSPF router-id` コマンドで指定された 32 ビット数値。（この値には任意の 32 ビット値を指定できます。このルータのインターフェイスに割り当てられた IPv4 アドレス以外のアドレスを設定できます。また、ルーティング可能な IPv4 アドレスでなくてもかまいません）。
- ITAL が選択した `router-id`。
- OSPF プロセスが実行されているインターフェイスのプライマリ IPv4 アドレス。OSPF インターフェイスの最初のインターフェイス アドレスが選択されます。

ルータ コンフィギュレーション モードで `router-id` コマンドを使用してルータ ID を設定することを推奨します。個別の OSPF プロセスは同じルータ ID を共有できますが、その場合、それらのプロセスは同じ OSPF ルーティング ドメインには存在できません。

## サポート対象 OSPF ネットワーク タイプ

OSPF は異なるメディアを次のタイプのネットワークに分類します。

- NBMA ネットワーク
- ブロードキャスト ネットワーク

ブロードキャストまたは NBMA ネットワークとしてネットワークを設定できます。たとえば、ユーザのネットワークにあるルータでマルチキャストアドレッシングがサポートされない場合に、この機能を使用してブロードキャスト ネットワークを NBMA ネットワークとして設定できます。

## OSPFのルート認証方法

OSPF Version 2 は 2 種類の認証（プレーンテキスト認証と MD5 認証）をサポートします。デフォルトでは、認証はイネーブルになっていません（RFC 2178 ではヌル認証と呼ばれます）。

OSPF Version 3 では、キーロールオーバーを除くすべてのタイプの認証がサポートされています。

### プレーンテキスト認証

プレーンテキスト認証（タイプ1認証とも呼ばれる）では、物理メディアを移動するパスワードを使用します。この認証は、アクセス権限を持たないユーザや、ネットワークに接続するパスワードを使用できないユーザでも簡単に見ることができます。そのため、プレーンテキスト認証はセキュリティで保護されません。プレーンテキスト認証は OSPF インターフェイスの誤った実装や設定ミスにより、間違った OSPF パケットが送信されることを防止できる場合があります。

### MD5 認証

MD5 認証はセキュリティで保護されます。パスワードは物理メディアに移動されません。その代わりに、ルータでは MD5 を使用して、OSPF パケットとキーのメッセージダイジェストが生成され、このメッセージダイジェストが物理メディアに送信されます。MD5 認証を使用すると、未認証または悪意のあるルーティングアップデートをルータで受け取らないようにできますが、トラフィックを迂回させることによってネットワークセキュリティが危険にさらされる可能性があります。



(注) MD5 認証では複数のキーがサポートされています。キー番号をキーに関連付ける必要があります。「OSPF 認証のメッセージダイジェスト管理」を参照してください。

### キーロールオーバー

OSPF 隣接関係（およびトポロジ）を中断することなく、操作ネットワークで MD5 キーを変更するために、キーロールオーバーメカニズムがサポートされています。ネットワーク管理者が新しいキーを複数のネットワークデバイスに設定するとき、異なるデバイスで新しいキーと古いキーの両方が使用されていることがあります。インターフェイスに新しいキーが設定されている場合、ソフトウェアから2つの同じパケットのコピーが送信されます。それぞれのパケットは古いキーと新しいキーによって認証されます。ソフトウェアではどのデバイスが新しいキーの使用を開始したかを追跡し、すべてのネイバーで新しいキーが使用されていることを検出すると、重複パケットの送信を停止します。次に、ソフトウェアでは古いキーを廃棄します。ネットワーク管理者は、各ルータの各コンフィギュレーションファイルから古いキーを削除する必要があります。



## OSPF FIB ダウンロード通知

OSPF FIB ダウンロード通知によって、ラインカードのリロード後の長期間にわたり入力トラフィックのドロップが最小化されます。

Open Shortest Path First (OSPF) は、ITAL を介してルーティング情報ベース (RIB) に登録され、すべてのルートが転送情報ベース (FIB) にダウンロードされるまで、インターフェイスがダウン状態のままになります。OSPF は、リロードされたラインカード上のすべてのルートが RIB/FIB を介してダウンロードされると、インターフェイス アップ通知を取得します。

RIB は、以下の場合に登録クライアントに通知を提供します。

- ノードが失われた。
- ノードが作成された。
- ノードの FIB アップロードが完了した。

## OSPF の指定ルータ (DR)

OSPF は、1つのルータを DR に、もう1つのルータを BDR に選択することで、ブロードキャストセグメントまたは NBMA セグメント上でのみ、セグメント上で交換される情報量を最小化します。このため、セグメント上のルータには、情報交換のための中央接続ポイントがあります。各ルータは、セグメント上の他の各ルータとルーティングアップデートを交換するのではなく、DR および BDR と情報を交換します。DR および BDR は、情報を他のルータに中継します。

ソフトウェアによってセグメント上の各ルータのプライオリティが確認され、DR および BDR となるルータが決定されます。最も高いプライオリティのルータが DR として選択されます。プライオリティが同じ場合、よりの高位ルータ ID を持つルータが優先されます。DR が選択されると、BDR も同様の方法で選択されます。プライオリティが 0 に設定されているルータは、DR または BDR になる資格がありません。

## OSPF のデフォルト ルート

タイプ 5 (ASE) LSA が生成され、スタブエリアを除くすべてのエリアにフラッドされます。スタブエリアにあるルータから、スタブエリア外の宛先にパケットをルーティングできるようにするために、スタブエリアにアタッチされている ABR によってデフォルトルートが挿入されます。

デフォルトルートのコストは 1 です (デフォルト)。または、default-cost コマンドに指定されている値によって決まります。

## OSPF Version 2 のリンクステート アドバタイズメント タイプ

次の各 LSA タイプには、個別の目的があります。

- ルータ LSA (タイプ 1) : 1つのエリア内にルータが持つリンクと各リンクのコストを表します。これらのLSAは、エリア内でのみフラッドされます。LSAは、QoS (Quality of Service) に基づいてルータがパスを計算できるかどうか、ルータが ABR または ASBR のどちらであるか、ルータが仮想リンクの一端であるかどうかを示します。また、タイプ 1 の LSA は、スタブ ネットワークへのアドバタイズにも使用されます。
- ネットワーク LSA (タイプ 2) : マルチアクセス ネットワーク セグメントにアタッチされているすべてのルータに関するリンク ステートとコストの情報を表します。この LSA ではネットワークセグメントにアタッチされているインターフェイスを持つすべてのルータを一覧にします。この LSA のコンテンツを生成して追跡するのは、ネットワーク セグメントの指定ルータの仕事です。
- ABR のサマリー LSA (タイプ 3) : 他のエリア内のルータ (エリア間ルート) に内部ネットワークをアドバタイズします。タイプ 3 の LSA は、1つのネットワークを表すことも、1つのプレフィックスに集約された一連のネットワークを表すこともあります。サマリー LSA を生成するのは ABR だけです。
- ASBR のサマリー LSA (タイプ 4) : ASBR および ASBR に到達するまでのコストをアドバタイズします。外部ネットワークにアクセスしようとするルータは、これらのアドバタイズメントを使用して、ネクストホップへの最適パスを決定します。ABR はタイプ 4 LSA を生成します。
- 自律システム外部 LSA (タイプ 5) : 別の自律システムからルートのを再配布します。通常は別のルーティングプロトコルから OSPF に再配布します。
- 自律システム外部 LSA (タイプ 7) : 外部ルート情報を NSSA 内で伝搬するために提供されます。タイプ 7 LSA は NSSA で生成およびアドバタイズできます。NSSA はタイプ 5 LSA を受信または生成しません。タイプ 7 LSA は 1つの NSSA 内でのみアドバタイズされます。境界ルータによってバックボーンエリアや他のエリアにフラッドされることはありません。
- 内部エリアプレフィックス LSA (タイプ 9) : ルータは各ルータまたは中継ネットワークに複数の内部エリアプレフィックス LSA を生成できます。それぞれの内部エリアプレフィックス LSA には固有のリンクステート ID があります。それぞれの内部エリアプレフィックス LSA のリンクステート ID には、ルータ LSA またはネットワーク LSA に対する関係と、スタブおよび中継ネットワークのプレフィックスが記されています。
- エリアローカルスコープ (タイプ 10) : Opaque LSA は関連付けられているエリアの境界を越えてフラッドされません。
- リンクステート (タイプ 11) : LSA は AS を通してフラッドされます。タイプ 11 LSA のフラッドスコープは、AS-External (タイプ 5) LSA のフラッドスコープと同じです。タイプ 5 LSA と同様、タイプ 11 Opaque LSA がスタブエリア内の隣接ルータからスタブエリアに受信されると、LSA は拒否されます。タイプ 11 Opaque LSA には、次のような属性があります。
  - LSA はすべての中継エリアを超えてフラッドされます。
  - LSA はバックボーンからのスタブ エリアにはフラッドされません

- LSA はルータから、ルータが接続されたスタブ エリアには発信されません。

## OSPFv3 のリンクステートアドバタイズメントタイプ

次の各 LSA タイプには、個別の目的があります。

- ルータ LSA (タイプ 1) : リンク ステートおよびエリアに対するルータ リンクのコストを表します。これらの LSA は、エリア内でのみフラッドされます。LSA は、ルータが ABR または ASBR のどちらであるか、および仮想リンクの一端であるかどうかを示します。また、タイプ 1 の LSA は、スタブ ネットワークへのアドバタイズにも使用されます。OSPFv3 では、これらの LSA はアドレス情報を持たず、ネットワーク プロトコルに依存しません。OSPFv3 では、ルータ インターフェイス情報は複数のルータ LSA 間で拡散されます。受信者は、SPF 計算を実行する前に、特定のルータから発信されたすべてのルータ LSA を連結する必要があります。
- ネットワーク LSA (タイプ 2) : マルチアクセス ネットワーク セグメントにアタッチされているすべてのルータに関するリンク ステートとコストの情報を表します。この LSA ではネットワークセグメントにアタッチされているインターフェイスを持つすべての OSPF ルータを一覧にします。ネットワークセグメントに選択された指定ルータだけが、セグメントのネットワーク LSA を生成して追跡できます。OSPFv3 では、ネットワーク LSA はアドレス情報を持たず、ネットワーク プロトコルに依存しません。
- ABR のエリア間プレフィックス LSA (タイプ 3) : 他のエリア内のルータ (エリア間ルート) に内部ネットワークがアドバタイズされます。タイプ 3 の LSA は、1 つのネットワークを表すことも、1 つのプレフィックスとして集約された一連のネットワークを表すこともあります。ABR はタイプ 3 LSA だけを生成します。OSPFv3 では、これらの LSA のアドレスは「address および mask」ではなく「prefix および prefix length」で表されます。デフォルトルートは、長さが 0 のプレフィックスとして表現されます。
- ASBR のエリア間ルータ LSA (タイプ 4) : ASBR および ASBR に到達するまでのコストをアドバタイズします。外部ネットワークにアクセスしようとするルータは、これらのアドバタイズメントを使用して、ネクスト ホップへの最適パスを決定します。ABR はタイプ 4 LSA を生成します。
- 自律システム外部 LSA (タイプ 5) : 別の自律システムからルートを再配布します。通常は別のルーティングプロトコルから OSPF に再配布します。OSPFv3 では、これらの LSA のアドレスは「address および mask」ではなく「prefix および prefix length」で表されます。デフォルトルートは、長さが 0 のプレフィックスとして表現されます。
- 自律システム外部 LSA (タイプ 7) : 外部ルート情報を NSSA 内で伝搬するために提供されます。タイプ 7 LSA は NSSA で生成およびアドバタイズできます。NSSA はタイプ 5 LSA を受信または生成しません。タイプ 7 LSA は 1 つの NSSA 内でのみアドバタイズされます。境界ルータによってバックボーンエリアや他のエリアにフラッドされることはありません。
- リンク LSA (タイプ 8) : リンクローカルフラッド スコープを持ち、関連付けられているリンクを超えてフラッドすることはありません。リンク LSA は、リンク

またはネットワークセグメントに接続されている他のすべてのルータに対してルータのリンクローカルアドレスを提供し、リンクに接続されている他のルータに、そのリンクに関連付ける IPv6 プレフィックスのリストを通知します。また、ルータが Options ビットの集まりをアサートして、リンクの起点となるネットワーク LSA と関連付けできるようにします。

- 内部エリアプレフィックス LSA (タイプ 9) : ルータは各ルータまたは中継ネットワークに複数の内部エリアプレフィックス LSA を生成できます。それぞれの内部エリアプレフィックス LSA には固有のリンクステート ID があります。それぞれの内部エリアプレフィックス LSA のリンクステート ID には、ルータ LSA またはネットワーク LSA に対する関係と、スタブおよび中継ネットワークのプレフィックスが記されています。

新しく定義された LSA のほとんどすべてに、アドレスプレフィックスが存在します。プレフィックスは、Prefix Length、Prefix Options、および Address Prefix の 3 つのフィールドで表現されます。OSPFv3 では、これらの LSA のアドレスは「address および mask」ではなく「prefix および prefix length」で表されます。デフォルトルートは、長さが 0 のプレフィックスとして表現されます。

エリア間プレフィックス LSA およびエリア内プレフィックス LSA では、すべての IPv6 プレフィックス情報が伝送されます。IPv4 ではこの情報はルータ LSA およびネットワーク LSA に含まれます。特定の LSA (ルータ LSA、ネットワーク LSA、エリア間ルータ LSA、およびリンク LSA) の Options フィールドは、IPv6 で OSPF をサポートするために 24 ビットに拡張されています。

OSPFv3 では、エリア間プレフィックス LSA、エリア間ルータ LSA、および自律システム外部 LSA のリンクステート ID の機能は、リンクステートデータベースの個々の部分を識別することだけです。OSPF Version 2 ではリンクステート ID で表されたアドレスまたはルータ ID はすべて、OSPFv3 では LSA の本体で伝送されます。

## パッシブインターフェイス

パッシブとしてインターフェイスを設定すると、ネイバーへのルーティングアップデートの送信が無効になるため、隣接関係は OSPF で形成されません。ただし、特定のサブネットは OSPF ネイバーに引き続きアドバタイズされます。インターフェイスでの OSPF プロトコル動作の送信を抑制するには、適切なモードで **passive** コマンドを使用します。

ホストを持つ LAN セグメントをネットワークの残りに接続しているが、ルータ間のトランジットリンクになるように作られていないインターフェイスでは、パッシブ設定を使用することを推奨します。

## グレースフル リスタート操作のモード

ルータがこの機能に使用できる動作モードは、再起動モードとヘルパーモード再起動モード、ヘルパーモード、およびプロトコルシャットダウンモードです。再起動モードは、OSPFv3 プロセスがグレースフルリスタートを実行しているときに開始されます。ヘルパーモードでは、OSPFv3 が隣接ルータで再起動している間に、確立されている OSPFv3 ルートでトラフィックを転送し続けている隣接ルータを参照します。

## リスタートモード

OSPFv3 プロセスが開始されたときに、グレースフルリスタートを試行する必要があるかどうかを決定します。決定は、グレースフルリスタートがそれまでにイネーブルされているかどうかに基づきます。（OSPFv3 は、ルータの初回の起動時にグレースフルリスタートを試行しません）。OSPFv3 グレースフルリスタートを有効にすると、RIB の消去タイマーがゼロ以外の値に変わります。

グレースフルリスタート中、ルータは OSPFv3 ルートを RIB に入力しません。ルータは再起動前に OSPFv3 が保有していた完全に隣接するネイバーとの完全な隣接関係を立ち上げようとします。最終的に、OSPFv3 プロセスは、（何らかの理由により）グレースフルリスタートを終了するため、または、グレースフルリスタートを終了したため、プロセスがコンバージされたことを RIB に示します。

最後の再起動から間を空けずに OSPFv3 が再起動を試みると、OSPFv3 プロセスは頻繁に繰り返しくラッシュするようになり、新しいグレースフルリスタートの実行が停止します。グレースフルリスタートの許可間隔を制御するには、`graceful-restart interval` コマンドを使用します。起動する最初のインターフェイスで OSPFv3 がグレースフルリスタートを開始すると、グレースフルリスタートの期間（有効期間）を制限するためにタイマーが起動します。`graceful-restart lifetime` コマンドを使用して、この期間を設定できます。起動する各インターフェイスで *grace LSA*（タイプ 11）がフラッディングされ、このルータがグレースフルリスタートを試みていることを隣接ルータに示します。ネイバーはヘルパーモードを開始します。再起動中のネイバーから受信した *hello* パケットの指定ルータとバックアップ指定ルータパケットの指定ルータチェックは正しくないため、バイパスされます。

## ヘルパーモード

ヘルパーモードは、デフォルトでイネーブルになっています。グレースフルリスタートを試みているルータから（ヘルパー）ルータが *grace LSA*（タイプ 11）を受け取ると、次のイベントが発生します。

- `graceful-restart helper disable` コマンドによりヘルパーモードがディセーブルされている場合、ルータは *LSA* パケットをドロップします。
- ヘルパーモードがイネーブルの場合、次の条件がすべて満たされると、ルータはヘルパーモードを開始します。
  - ローカルルータ自体がグレースフルリスタートを試みていない。
  - ローカル（ヘルパー）ルータに送信先ネイバーとの完全な隣接関係がある。
  - 受信した *LSA* の *lsage*（リンクステートの経過時間）の値が、要求された猶予期間よりも短い。
  - *grace LSA* の送信元が *grace LSA* の生成元と同じである。
- ヘルパーモードを開始すると、ルータは一定期間そのヘルパー機能を実行します。この期間は再起動モードにあるルータの有効期間の値から、受信した *grace LSA* の *lsage* の値を引いた値です。グレースフルリスタートが時間内に成功すると、ヘルパータイマーが期

限切れになる前に停止します。ヘルパー タイマーの期限が切れた場合、再起動しているルータへの隣接関係がダウンし、通常の OSPFv3 機能が再開します。

- デッド タイマーはヘルパー モードにあるルータでは使用できません。
- 次のいずれかの場合に、ヘルパー モードにあるルータはヘルパー機能の実行を停止します。
  - ヘルパー ルータが再起動中のルータとの完全な隣接関係を起動できる。
  - ヘルパー機能のローカル タイマーの有効期限が切れている。

## プロトコル シャットダウンモード

このモードでは、OSPFv3 操作は完全に無効になっています。これは、自己生成リンク ステートアドバタイズメント (LSA) をフラッシュすることで達成され、ローカルの OSPFv3 対応インターフェイスが即座に停止し、リンクステートデータベース (LSDB) がクリアされます。ローカル以外の LSDB エントリは OSPFv3 によって削除され、フラッシュ (MaxAged) されません。

プロトコルシャットダウンモードは、**protocol shutdown** コマンドを使用 (プロトコルインスタンスが無効になります) して手動で起動できます。または OSPFv3 プロセスのメモリが不足すると起動します。次のイベントは、プロトコルシャットダウンが実行されると発生します。

- ローカルルータ LSA およびすべてのローカルリンク LSA がフラッシュされます。他の LSA はすべて、ドメイン内の他の OSPFv3 ルータによって最終的にエージアウトされます。
- まだローカルルータとフル状態になっていない OSPFv3 ネイバーは、Kill\_Nbr イベントとともに停止します。
- 3 秒の遅延後、空の Hello パケットはアクティブな隣接関係がある各ネイバーに即座に送信されます。
  - 空の Hello パケットは、dead\_interval が経過するまで定期的に送信されます。
  - dead\_interval が経過すると、Hello パケットは送信されなくなります。

Dead Hello インターバルの遅延 (4 X Hello インターバル) 後、次のイベントが実行されます。

- その OSPFv3 インスタンスからの LSA データベースがクリアされます。
- OSPFv3 によってインストールされた RIB からのすべてのルートが消去されます。

ルータは、プロトコルシャットダウン状態時にネイバーから受信するいずれの OSPF 制御パケットにも応答しません。

## プロトコルの復元

プロトコルを復元する方法は、シャットダウンを最初に引き起こしたトリガーに依存します。OSPFv3 が **protocol shutdown** コマンドを使用してシャットダウンされた場合、OSPFv3 を通常

の動作に復元するには **no protocol shutdown** コマンドを使用します。OSPFv3 が **sysmon** からの重要なメモリ メッセージによってシャットダウンされた場合は、十分なメモリがプロセッサに復元されたことを示す **sysmon** からの通常のメモリ メッセージによって OSPFv3 プロトコルが復元され、通常の動作が再開されます。OSPFv3 が重要なメモリ トリガーによってシャットダウンされた場合は、通常のメモリ レベルがルート プロセッサで復元された際に、手動で再起動する必要があります。これは自動的に復元されません。

次のイベントは、OSPFv3 が復元されると発生します。

1. すべての OSPFv3 インターフェイスが、Hello パケットとデータベース交換を使用してバックアップされます。
2. ローカル ルータおよびリンク LSA が再作成され、アドバタイズされます。
3. ルータは、ネイバーから受信したすべての OSPFv3 制御メッセージに正常に応答します。
4. 他の OSPFv3 ルータから学習されたルートが RIB にインストールされます。

## OSPF Version 2 および OSPFv3 でのロード バランシング

ルータは、複数のルーティング プロセス（またはルーティング プロトコル）を使用して特定のネットワークへの複数のルートを確認すると、最短のアドミニストレーティブ ディスタンスを持つルートを選択してルーティング テーブルにインストールします。同じアドミニストレーティブ ディスタンスを持つ同じルーティング プロセスを使用して認識された多数のルートから、1つのルートを選択する必要があることもあります。この場合、ルータはその宛先へのコスト（またはメトリック）が最も小さいパスを選択します。各ルーティング プロセスはコストをそれぞれの方法で計算します。コストは、ロード バランシングを実現するために処理が必要なこともあります。

OSPF では、自動的にロード バランシングが実行されます。OSPF により、複数のインターフェイスを通して宛先に到達できること、および各パスのコストが同じであることが検出された場合は、ルーティング テーブルに各パスがインストールされます。同じ宛先へのパスの数は、**maximum-paths** (OSPF) コマンドを指定しない限り、制限されません。

最大パスの範囲は 1 から 8 です。デフォルトの最大パスの数は 8 です。

## OSPFv2 のパス 計算要素

PCE はネットワーク パスやルートをネットワーク図に基づいて計算し、計算上の制限を適用する機能を持つエンティティ（コンポーネント、アプリケーション、ネットワーク ノード）です。

PCE は、PCE アドレスおよびクライアントが MPLS-TE に設定されると実行されます。PCE はその PCE アドレスおよび機能を OSPF に通信して、OSPF はこの情報を PCE ディスカバリ Type-Length-Value (TLV) (タイプ 2) にパッケージ化し、RI LSA を再発信します。OSPF には、すべての RI LSA でルータ機能 TLV (タイプ 1) も含まれます。PCE ディスカバリ TLV には PCE アドレス サブ TLV (タイプ 1) およびパス スコープ サブ TLV (タイプ 2) が含まれません。

PCE アドレス サブ TLV では PCE に到達するために使用される必要がある IP アドレスを指定します。このアドレスは常に到達可能なループバックアドレスにする必要があります。この TLV は必須であり、PCE ディスカバリ TLV 内に存在する必要があります。パス スコープ サブ TLV は、PCE パス計算スコープを示します。これは、PCE 機能を参照して計算したり、エリア内ルート、エリア間、AS 間、またはレイヤ TE 間 LSP の計算に参加したりします。

OSPFv2 への PCE 拡張機能には、ルータ情報リンク ステート アドバタイズメント (RI LSA) のサポートが含まれます。OSPFv2 は、すべてのエリアの範囲 (LSA タイプ 9、10、および 11) を受信するように拡張されます。ただし、OSPFv2 はエリア範囲タイプ 10 のみを発信します。

パス計算要素機能の詳細については、『*MPLS Configuration guide*』の「*Implementing MPLS Traffic Engineering*」モジュール、および次の IETF ドラフトを参照してください。

- draft-ietf-ospf-cap-09
- draft-ietf-pce-disco-proto-ospf-00

## OSPFv3 の管理情報ベース (MIB)

Cisco IOS XR では RFC 5643 に定義されている MIB および OSPFv3 のトラップが完全にサポートされています。RFC 5643 には、IPv6 用の Open Shortest Path First (OSPF) ルーティングプロトコル (OSPF バージョン 3) で使用する管理情報ベース (MIB) のオブジェクトが定義されています。

OSPFv3 MIB の実装は、IETF ドラフト『*Management Information Base for OSPFv3 (draft-ietf-ospf-ospfv3-mib-8)*』に基づきます。RFC 5643 にアップグレードすると、ユーザは新しい MIB をピックアップするように NMS アプリケーションを更新する必要があります。

### 複数の OSPFv3 インスタンス

SNMPv3 は、複数の OSPFv3 インスタンスに MIB ビューを設定するために使用できる「コンテキスト」を同じシステムでサポートします。

## OSPFv2 の VRF-lite サポート

OSPF バージョン 2 (OSPFv2) の VRF Lite 機能は、イネーブルになっています。VRF-Lite は、BGP/MPLS ベースのバックボーンがない状態での仮想ルーティングおよび転送 (VRF) 導入です。VRF-Lite では、個別のプロバイダーエッジ (PE) ルータは VRF インターフェイスを使用して直接接続されています。OSPFv2 の VRF-Lite をイネーブルにするには、VRF コンフィギュレーションモードで **capability vrf-lite** コマンドを設定します。VRF-Lite が設定されている場合、DN ビット処理および自動エリア境界ルータ (ABR) のステータス設定はディセーブルです。



## OSPFv3 タイマーの更新

Open Shortest Path First バージョン 3 (OSPFv3) タイマー リンク ステート アドバタイズメント (LSA)、Shortest Path First (SPF) スロットリングのデフォルト値は次のように更新されます。

- **timers throttle lsa all** : *start-interval* : 50 ミリ秒および *hold-interval* : 200 ミリ秒
- **timers throttle spf** : *spf-start* : 50 ミリ秒、*spf-hold* : 200 ミリ秒、*spf-max-wait* : 5000 ミリ秒

