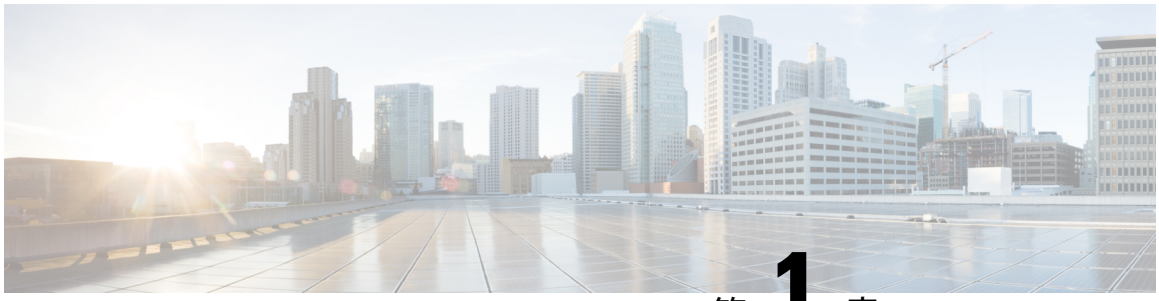
Cisco IOS XE Bengaluru 17.4.x（Catalyst 9300 スイッチ）IP アドレッシング サービス コンフィギュレーション ガイド

初版：2020 年 7 月 31 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

【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、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。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。

目 次



第 1 章

IP アドレッシングサービスの概要

このセクションでは、IP アドレッシングサービスについて説明します。

IPv6 の概要

IPv4 ユーザーは IPv6 に移行することができ、エンドツーエンドのセキュリティ、Quality of Service（QoS）、およびグローバルに一意なアドレスのようなサービスを利用できます。IPv6 アドレス スペースによって、プライベート アドレスの必要性が低下し、ネットワーク エッジの境界ルータで Network Address Translation（NAT; ネットワーク アドレス変換）処理を行う必要性も低下します。

シスコの IPv6 の実装方法については、次の URL を参照してください。

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

IPv6 およびこの章のその他の機能については、

- 『*Cisco IOS IPv6 Configuration Library*』を参照してください。
- Cisco.com の [Search] フィールドを使用して、Cisco IOS ソフトウェア マニュアルを特定します。たとえば、スタティック ルートについての情報が必要な場合は、[Search] フィールドで *Implementing Static Routes for IPv6* と入力すると、スタティック ルートについて調べられます。

IPv6 アドレス

スイッチがサポートするのは、IPv6ユニキャストアドレスのみです。サイトローカルユニキャスト アドレスおよびマルチキャスト アドレスはサポートされません。

IPv6 の 128 ビット アドレスは、コロンで区切られた一連の 8 つの 16 進フィールド（n:n:n:n:n:n:n:n. の形式）で表されます。次に、IPv6 アドレスの例を示します。

2031:0000:130F:0000:0000:09C0:080F:130B

実装を容易にするために、各フィールドの先行ゼロは省略可能です。上記アドレスは、先行ゼロを省略した次のアドレスと同じです。

2031:0:130F:0:0:9C0:80F:130B

2 つのコロン（::）を使用して、ゼロが連続する 16 進フィールドを表すことができます。ただし、この短縮形を使用できるのは、各アドレス内で 1 回のみです。

2031:0:130F::09C0:080F:130B

IPv6 アドレス形式、アドレス タイプ、および IPv6 パケット ヘッダーの詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-3e/ip6b-xe-3e-book.html を参照してください。

- IPv6 アドレス形式
- IPv6 アドレス タイプ：マルチキャスト
- Ipv6 アドレス 出力表示
- 簡易 IPv6 パケット ヘッダー

128 ビット幅のユニキャスト アドレス

スイッチは集約可能なグローバル ユニキャスト アドレスおよびリンク ローカル ユニキャスト アドレスをサポートします。サイト ローカル ユニキャスト アドレスはサポートされていません。

- 集約可能なグローバル ユニキャスト アドレスは、集約可能グローバル ユニキャスト プレフィックスの付いた IPv6 アドレスです。このアドレス構造を使用すると、ルーティング プレフィックスを厳格に集約することができ、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 エントリ数が制限されます。これらのアドレスは、組織を経由して最終的にインターネット サービス プロバイダに至る集約リンク上で使用されます。

  これらのアドレスはグローバル ルーティング プレフィックス、サブネット ID、およびインターフェイス ID によって定義されます。現在のグローバル ユニキャスト アドレス割り当てには、バイナリ値 001（2000::/3）で開始するアドレス範囲が使用されます。プレフィックスが 2000::/3（001）～ E000::/3（111）のアドレスには、Extended Unique Identifier（EUI）64 フォーマットの 64 ビット インターフェイス ID を設定する必要があります。

- リンク ローカル ユニキャスト アドレスをすべてのインターフェイスに自動的に設定するには、修飾 EUI フォーマット内で、リンク ローカル プレフィックス FE80::/10（1111 1110 10）およびインターフェイスID を使用します。ネイバー探索プロトコル（NDP）およびステートレス自動設定プロセスでは、リンクローカルアドレスが使用されます。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ます。通信する場合に、グローバルに一意なアドレスは不要です。IPv6 ルータは、リンクローカルの送信元または宛先アドレスを持つパケットをその他のリンクに転送しません。

詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library* 』の「Implementing IPv6 Addressing and Basic Connectivity」の章にある IPv6 ユニキャスト アドレスに関する項を参照してください。

IPv6 の DNS

IPv6 は、ドメイン ネーム システム（DNS）のレコード タイプを、DNS 名前/アドレスおよびアドレス/名前の検索プロセスでサポートします。DNS AAAA リソース レコード タイプは IPv6 アドレスをサポートし、IPv4 の A アドレス レコードと同等です。スイッチは IPv4 および IPv6 の DNS 解決をサポートします。

IPv6 のステートレス自動設定および重複アドレス検出

スイッチではステートレス自動設定が使用されているため、ホストやモバイル IP アドレスの管理のような、リンク、サブネット、およびサイトアドレス指定の変更を管理することができます。ホストは独自のリンクローカルアドレスを自動的に設定します。起動元ノードはルータに送信請求を送信して、インターフェイス設定をアドバタイズするようルータに要求します。

Cisco IOS XE Gibraltar 16.11.1 以降、自動設定された IPv6 アドレスには、RFC5453 で指定されている予約済みインターフェイス識別子の範囲に含まれないインターフェイス識別子が含まれるようになります。

自動設定および重複アドレス検出の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing IPv6 Addressing and Basic Connectivity」の章を参照してください。

IPv6 アプリケーション

スイッチは、次のアプリケーションについて IPv6 をサポートします。

- ping、Traceroute、Telnet、および Trivial File Transfer Protocol（TFTP）
- IPv6 トランスポートによるセキュア シェル（SSH）
- IPv6 トランスポートによる HTTP サーバー アクセス
- IPv4 トランスポートによる AAAA の DNS レゾルバ

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

これらのアプリケーションの管理に関する詳細については、Cisco.com の『*Cisco IOS IPv6 Configuration Library*』を参照してください。

DHCP for IPv6 アドレスの割り当て

DHCPv6 を使用すると、DHCP サーバーは IPv6 ネットワーク アドレスなどの設定パラメータを IPv6 クライアントに渡すことができます。このアドレス割り当て機能により、ホストが接続するネットワークに基づいて、適切なプレフィックス内での重複しないアドレス割り当てが管理されます。アドレスは、1 つまたは複数のプレフィックス プールから割り当てることができます。デフォルトのドメインおよび DNS ネーム サーバー アドレスなど、その他のオプションは、クライアントに戻すことができます。アドレス プールは、特定のインターフェイス、複数のインターフェイス上で使用する場合に割り当てられます。または、サーバーが自動的に適切なプールを検出できます。

DHCP for IPv6 の設定については、「*DHCP for IPv6* アドレス割り当ての設定」のセクションを参照してください。

DHCPv6 クライアント、サーバー、またはリレーエージェント機能の設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』を参照してください。

HTTP(S) Over IPv6

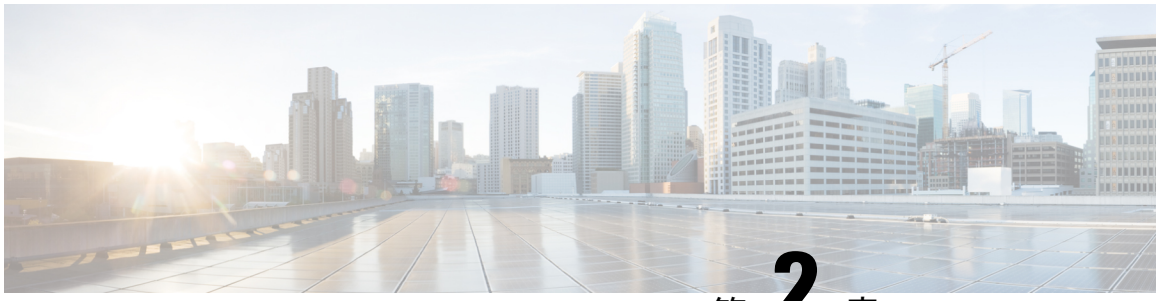
HTTP クライアントは要求を IPv4 HTTP サーバーと IPv6 HTTP サーバーの両方に送信し、これらのサーバーは IPv4 HTTP クライアントと IPv6 HTTP クライアントの両方からの要求に応答します。IPv6 アドレスを含む URL は、16 ビット値をコロンで区切った 16 進数で指定する必要があります。

受信ソケット コールは、IPv4 アドレス ファミリまたは IPv6 アドレス ファミリを選択します。受信ソケットは、IPv4 ソケットまたは IPv6 ソケットのいずれかです。リスニング ソケットは、接続を示す IPvv4 と IPv6 の両方の信号を待ち受け続けます。IPv6 リスニング ソケットは、IPv6 ワイルドカード アドレスにバインドされています。

基本 TCP/IP スタックは、デュアル スタック環境をサポートします。HTTP には、TCP/IP スタック、およびネットワーク層相互作用を処理するためのソケットが必要です。

HTTP 接続を確立するには、基本ネットワーク接続（**ping**）がクライアントとサーバーホストとの間に存在する必要があります。

詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Managing Cisco IOS Applications over IPv6」の章を参照してください。



第 2 章

IPv6 クライアントの IP アドレス ラーニング

IPv6 クライアント アドレス ラーニングの前提条件

IPv6 クライアント アドレス ラーニングを設定する前に、IPv6 をサポートするように クライアントを設定します。

IPv6 クライアント アドレス ラーニングについて

クライアント アドレス ラーニングは、関連付け、再関連付け、認証解除、タイムアウトの際に、クライアントの IPv4 および IPv6 アドレス、デバイスによって保持されるクライアント変換の状態について学習するために、デバイスで設定されます。

IPv6 クライアントで IPv6 アドレスを取得するには、次の 3 つの方法があります。

- ステートレス アドレス自動設定（SLACC）
- ステートフル DHCPv6
- 静的設定

これらの方法のいずれの場合も、IPv6 クライアントは常にネイバー送信要求 DAD（重複アドレス検出）要求を送信して、ネットワークに重複する IP アドレスがないようにします。デバイスは、クライアントのネイバー探索プロトコル（NDP）およびDHCPv6パケットをスヌーピングして、そのクライアント IP アドレスについて学習します。

重複する IPv6 アドレスが設定されると、DAD は重複するアドレスを検出し、ルータアドバタイズメント（RA）でアドバタイズします。重複するアドレスは、システムから手動で削除できます。削除すると、接続されたアドレスに表示されず、RA プレフィックスにアドバタイズされません。

SLAAC アドレス割り当て

IPv6 クライアント アドレス割り当て用の最も一般的な方法は、ステートレス アドレス自動設定（SLAAC）です。SLAAC はクライアントが IPv6 プレフィクスに基づいてアドレスを自己割り当てするシンプルなプラグアンドプレイ接続を提供します。このプロセスが実現しました。

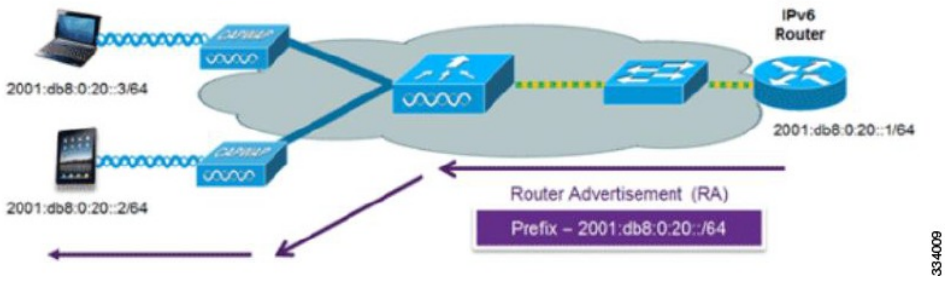
次のように、ステートレス アドレス自動設定（SLAAC）は設定されています。

- ホストは、ルータ送信要求メッセージを送信します。
- ホストは、ルータ アドバタイズメント メッセージを待機します。
- ホストは、ルータ アドバタイズメント メッセージから IPv6 プレフィックスの最初の 64 ビットを取得し、これを 64 ビット EUI-64 アドレス（イーサネットの場合、MAC アドレスから作成されます）と組み合わせて、グローバル ユニキャスト メッセージを作成します。ホストは、デフォルト ゲートウェイとして、ルータ アドバタイズメント メッセージの IP ヘッダーに含まれる送信元 IP アドレスも使用します。
- 重複アドレス検出は、選択されるランダムアドレスが他のクライアントと重複しないように、IPv6 クライアントによって実行されます。
- アルゴリズムの選択はクライアントに依存し、多くの場合は設定できます。

次の 2 種類のアルゴリズムに基づいて IPv6 アドレスの最後の 64 ビットが学習可能です。

- インターフェイスの MAC アドレスに基づく EUI-64、または
- ランダムに生成されるプライベート アドレス。

図 *1 : SLAAC* アドレス割り当て



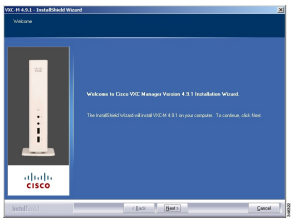
Cisco 対応 IPv6 ルータからの次の Cisco IOS コンフィギュレーション コマンドを使用して、SLAAC のアドレッシングとルータ アドバタイズメントをイネーブルにします。

```
ipv6 unicast-routing
interface Vlan20
description IPv6-SLAAC
```

```
ip address 192.168.20.1 255.255.255.0
ipv6 address FE80:DB8:0:20::1 linklocal
ipv6 address 2001:DB8:0:20::1/64
ipv6 enable
end
```

ステートフル DHCPv6 アドレス割り当て

図 *2 :* ステートフル *DHCPv6* アドレス割り当て



DHCPv6 の使用は、SLAAC がすでに導入されている場合は、IPv6 クライアント接続で要求されません。DHCPv6 にはステートレスおよびステートフルという 2 種類の動作モードがあります。

DHCPv6 ステートレス モードは、ルータ アドバタイズメントで使用できない追加のネットワーク情報をクライアントに提供するために使用しますが、これは IPv6 アドレスではありません。すでに SLAAC によって提供されているためです。この情報には DNS ドメイン名、DNS サーバー、その他の DHCP ベンダー固有オプションを含めることができます。このインターフェイス設定は、SLAAC をイネーブルにしてステートレス DHCPv6 を実装する Cisco IOS IPv6 ルータ用です。

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateless
ip address 192.168.20.1 255.255.255.0
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
ipv6 address 2001:DB8:0:20::1/64
end
```

マネージド モードとも呼ばれる DHCPv6 ステートフル オプションは、DHCPv4 に対して同じように動作します。つまり固有のアドレスを、SLAAC のとおりにアドレスの最後の 64 ビットを生成するクライアントではなく、それぞれのクライアントに割り当てます。このインターフェイス設定は、ローカル デバイスのステートフル DHCPv6 を実装している Cisco IOS IPv6 ルータ用です。

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateful
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
```

```
ipv6 nd prefix 2001:DB8:0:20::/64 no-advertise
ipv6 nd managed-config-flag
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
end
```

次のインターフェイス設定は、外部 DHCP サーバーのステートフル DHCPv6 を実装している Cisco IOS IPv6 ルータ用です。

```
ipv6 unicast-routing
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateful
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
ipv6 nd prefix 2001:DB8:0:20::/64 no-advertise
ipv6 nd managed-config-flag
ipv6 nd other-config-flag
ipv6 dhcp_relay destination 2001:DB8:0:20::2
end
```

静的 IP アドレス割り当て

クライアントにスタティックに設定されたアドレス。

ルータ要求

ルータ要求メッセージは、ローカルルーティングに関する情報を入手できる、またはステートレス自動設定を設定できるルータアドバタイズメントを送信するようにローカルルータを促すために、ホストによって発行されます。ルータ アドバタイズメントは定期的に送信され、起動時または再起動操作後などに、ホストはルータ送信要求を使用して即時ルータ アドバタイズメントを要求します。

ルータ アドバタイズメント

ルータ アドバタイズメント メッセージは、ルータから定期的に送信されるか、ホストからのルータ送信要求メッセージへの応答として送信されます。これらのメッセージに含まれる情報は、ホストでステートレス自動設定を実行し、ルーティング テーブルを変更するために使用されます。

ネイバー探索

IPv6 ネイバー ディスカバリとは、近隣のノード間の関係を決定するメッセージとプロセスのことです。ネイバー ディスカバリは、IPv4 で使用されていた ARP、ICMP ルータ探索、および ICMP リダイレクトに代わるものです。

信頼できるバインディング テーブル データベースを構築するために、IPv6 ネイバー ディスカバリ検査によってネイバー ディスカバリ メッセージが分析され、準拠しない IPv6 ネイバー ディスカバリ パケットはドロップされます。スイッチのネイバー バインディング テーブルで

は、各 IPv6 アドレスと、関連付けられている MAC アドレスが追跡されます。クライアントは、ネイバー バインディング タイマーに従って、テーブルから消去されます。

ネイバー探索抑制

クライアントの IPv6 アドレスは、デバイスによってキャッシュされます。デバイスが IPv6 アドレスを検索する NS マルチキャストを受信したときに、デバイスによって特定された目的のアドレスがクライアントのいずれかに属している場合、デバイスはクライアントに代わって NA メッセージで応答します。このプロセスによって IPv4 のアドレス解決プロトコル（ARP）テーブルと同等のテーブルが生成されますが、より効率的であり、たいていの場合、使用されるメッセージは少なくなります。

（注） デバイスがプロキシのように動作し NA で応答するのは、**ipv6 nd suppress** コマンドが設定されている場合だけです。

デバイスにクライアントの IPv6 アドレスがない場合、デバイスは NA で応答せず、NS パケットを転送します。この問題を解決するために、NS マルチキャスト フォワーディング ノブが用意されています。このノブが有効になっている場合、デバイスは、把握していない（キャッシュ欠落）IPv6 アドレスの NS パケットを取得して転送します。このパケットは目的のクライアントに到達し、クライアントは NA で応答します。

このキャッシュ ミス シナリオが発生するのはまれで、完全な IPv6 スタックが実装されていないクライアントが、NDP 時にそれらの IPv6 アドレスをアドバタイズしない可能性はほとんどありません。

RA ガード

IPv6 クライアントは、IPv6 アドレスを設定し、IPv6 ルータ アドバタイズメント（RA）パケットに基づいてルータ テーブルにデータを入力します。RA ガード機能は、有線ネットワークの RA ガード機能に類似しています。RA ガードは、クライアントから発信される不要または不正な RA パケットをドロップすることによって、IPv6 ネットワークのセキュリティを強化します。この機能が設定されていないと、悪意のある IPv6 クライアントが、多くの場合は高い優先順位で、それ自体をネットワークのルータとして通知する可能性があり、結果としてそのクライアントが正規の IPv6 ルータよりも優先されることになります。

また、RA ガードは、着信 RA を調べて、メッセージまたはスイッチ設定で検出された情報のみに基づいて、それらをスイッチするかブロックするかを決定します。受信したフレームで使用できる情報は、RA の検証に有用です。

- フレームが受信されるポート
- IPv6 送信元アドレス
- プレフィックス リスト

スイッチで作成された次の設定情報は、受信した RA フレームで検出された情報に対して検証するときに RA ガードで使用できます。

- RA ガード メッセージの受信用に信頼できる/信頼できないポート
- RA 送信者の信頼できる/信頼できない送信元 IPv6 アドレス
- 信頼できる/信頼できないプレフィックス リストおよびプレフィックス範囲
- ルータ プリファレンス

RA ガードはデバイスに適用されます。デバイスで RA メッセージをドロップするようにデバイスを設定できます。すべての IPv6 RA メッセージがドロップされ、その結果、他のクライアントおよびアップストリーム有線ネットワークが悪意のある IPv6 クライアントから保護されます。

```
//Create a policy for RA Guard//
ipv6 nd raguard policy raguard-router
trusted-port
device-role router

//Applying the RA Guard Policy on port/interface//
interface tengigabitethernet1/0/1 (Katana)
interface gigabitethernet1/0/1 (Edison)

ipv6 nd raguard attach-policy raguard-router
```

IPv6 クライアント アドレス ラーニングの設定方法

ここでは、IPv6 クライアント アドレス ラーニングに関する設定情報について説明します。

IPv6 ユニキャストの設定

IPv6 ユニキャストはスイッチで常に有効にしておく必要があります。IPv6 ユニキャスト ルーティングはディセーブルに設定されています。

IPv6 ユニキャストを設定するには、次の手順を実行します。

始める前に

IPv6ユニキャストデータグラムの転送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 **ipv6 unicast-routing** コマンドを使用します。IPv6 ユニキャスト データグラムの転送をディセーブルにするには、このコマンドの **no** 形式を使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 unicast routing** 例： `Device(config)# ipv6 unicast routing`	IPv6 ユニキャスト データグラムの転送をイネーブルにします。

RA ガード ポリシーの設定

IPv6 クライアントアドレスを追加し、IPv6 ルータ アドバタイズメント パケットに基づいてルータテーブルに入力するには、デバイスで RA ガードポリシーを設定します。

RA ガードポリシーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 nd raguard policy raguard-router** 例： `Device(config)# ipv6 nd raguard policy raguard-router`	RA ガード ポリシー名を定義して、RA ガード ポリシー 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	**trustedport** 例： `Device(config-ra-guard)# trustedport`	（任意）このポリシーが信頼できるポートに適用されることを指定します。

	コマンドまたはアクション	目的
ステップ 5	**device-role router** 例： `Device(config-ra-guard)# device-role router`	ポートに接続されているデバイスの役割を指定します。
ステップ 6	**exit** 例： `Device(config-ra-guard)# exit`	RA ガード ポリシー コンフィギュレーション モードを終了してグローバル コンフィギュレーション モードに戻ります。

RA ガードポリシーの適用

デバイスで RA ガードポリシーを適用すると、すべての信頼できない RA がブロックされます。

RA ガードポリシーを適用するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface tengigabitethernet 1/0/1** 例： `Device(config)# interface tengigabitethernet 1/0/1`	インターフェイスのタイプと番号を指定し、デバイスをインターフェイス コンフィギュレーション モードにします。
ステップ 4	**ipv6 nd raguard attach-policy raguard-router** 例： `Device(config-if)# ipv6 nd raguard attach-policy raguard-router`	指定したインターフェイスに IPv6 RA ガード機能を適用します。
ステップ 5	**exit** 例：	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。

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

IPv6 スヌーピングの設定

（注） IPv6 スヌーピングのレガシー設定ではなく、SISF ベースのデバイス追跡設定を設定することをお勧めします。詳細については、『セキュリティ コンフィギュレーション ガイド』の「*SISF* ベースのデバイス追跡の設定」の項を参照してください。

スイッチで IPv6 スヌーピングを常に有効にしておく必要があります。

IPv6 スヌーピングを設定するには、次の手順を実行します。

始める前に

クライアント マシンで IPv6 をイネーブルにします。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan configuration 1** 例： `Device(config)# vlan configuration 1`	VLAN 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	**ipv6 snooping** 例： `Device(config-vlan)# ipv6 snooping`	Vlan で IPv6 スヌーピングをイネーブルにします。
ステップ 5	**ipv6 nd suppress** 例：	Vlan で IPv6 ND 抑制をイネーブルにします。

	コマンドまたはアクション	目的
	`Device(config-vlan-config)# ipv6 nd suppress`	
ステップ 6	**exit** 例： `Device(config-vlan-config)# exit`	設定を保存し、Vlan コンフィギュレーション モードを終了します。

IPv6 ND 抑制ポリシーの設定

IPv6 ネイバー探索（ND）マルチキャスト抑制機能では、ドロップする（およびターゲットに代わって送信要求に応答する）、またはユニキャスト トラフィックに変換することで、できるだけ多くの ND マルチキャスト ネイバー送信要求（NS）メッセージを停止します。この機能は、レイヤ2スイッチで実行され、適切なリンクの処理に必要な制御トラフィックの量を減らすために使用されます。

アドレスがバインディング テーブルに挿入されると、マルチキャスト アドレスに送信されたアドレス解決要求が代行受信され、デバイスはアドレスの所有者に代わって応答するか、レイヤ 2 で要求をユニキャスト メッセージに変換して宛先に転送します。

IPv6 ND 抑制ポリシーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 nd suppress policy** *policy_name* 例： `Device(config)# ipv6 nd suppress policy policy1`	ND 制御ポリシー名を定義して ND 制御ポリシー コンフィギュレーション モードを開始します。

VLAN/PortChannel での IPv6 スヌーピングの設定

ネイバー探索（ND）抑制は、VLAN またはスイッチ ポートでイネーブルまたはディセーブルにできます。

VLAN/PortChannel で IPv6 スヌーピングを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan config901** 例： `Device(config)# vlan config901`	VLAN を作成し、VLAN 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	**ipv6 nd suppress** 例： `Device(config-vlan)# ipv6 nd suppress`	VLAN に IPv6 nd 抑制を適用します。
ステップ 5	**end** 例： `Device(config-vlan)# end`	VLAN コンフィギュレーション モードを終了し、グローバルコンフィギュレーション モードを開始します。
ステップ 6	**interface gi1/0/1** 例： `Device(config)# interface gi1/0/1`	ギガビット イーサネット ポート インターフェイスを作成します。
ステップ 7	**ipv6 nd suppress** 例： `Device(config-vlan)# ipv6 nd suppress`	インターフェイスに IPv6 nd 抑制を適用します。
ステップ 8	**end** 例： `Device(config-vlan)# end`	VLAN コンフィギュレーション モードを終了し、グローバルコンフィギュレーション モードを開始します。

スイッチインターフェイスでの IPv6 の設定

インターフェイスで IPv6 を設定するには、次の手順に従います。

始める前に

クライアント上の IPv6 および有線インフラストラクチャ上の IPv6 サポートをイネーブルにします。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface vlan 1** 例： `Device(config)# interface vlan 1`	インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**ip address fe80::1 link-local**
	例：
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		リンクローカル オプションを使用してインターフェイスで IPv6 アドレスを設定します。

ステップ 5	**ipv6 enable** 例： `Device(config)# ipv6 enable`	（任意）インターフェイス上で IPv6 をイネーブルにします。
ステップ 6	**end** 例： `Device(config)# end`	インターフェイス モードを終了します。

スイッチインターフェイスでの DHCP プールの設定

インターフェイスで DHCP プールを設定するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 dhcp pool Vlan21** 例： `Device(config)# ipv6 dhcp pool vlan1`	コンフィギュレーション モードを開始し、VLAN の IPv6 DHCP プールを設定します。
ステップ 4	**address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10** 例： `Device(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10`	コンフィギュレーション DHCP モードを開始し、VLAN のアドレス プールとそのライフタイムを設定します。
ステップ 5	**dns-server 2001:100:0:1::1** 例： `Device(config-dhcpv6)# dns-server 2001:20:21::1`	DHCP プールの DNS サーバーを設定します。
ステップ 6	**domain-name example.com** 例： `Device(config-dhcpv6)# domain-name example.com`	完全な非修飾ホスト名になるようにドメイン名を設定します。
ステップ 7	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DHCP を使用しないステートレス自動アドレスの設定

DHCP を使用しないステートレス自動アドレス設定を指定するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface vlan 1** 例： `Device(config)# interface vlan 1`	インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**ip address fe80::1 link-local**
	例：
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		リンクローカル オプションを使用してインターフェイスで IPv6 アドレスを設定します。

ステップ 5	**ipv6 enable** 例： `Device(config)# ipv6 enable`	（任意）インターフェイス上で IPv6 をイネーブルにします。

ステップ 6	**no ipv6 nd managed-config-flag**
	例：
	`Device(config)# interface vlan 1`
	`Device(config-if)# no ipv6 nd managed-config-flag`
		接続されたホストで、アドレスの取得にステートフル自動設定が使用されないようにします。

ステップ 7	**no ipv6 nd other-config-flag** 例： `Device(config-if)# no ipv6 nd other-config-flag`	接続されたホストで、DHCP からの非アドレス オプションの取得に（ドメインなど）ステートフル自動設定が使用されないようにします。

	コマンドまたはアクション	目的
ステップ 8	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。

DHCP を使用したステートレス自動アドレスの設定

DHCP を使用したステートレス自動アドレス設定を指定するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface vlan 1** 例： `Device(config)# interface vlan 1`	インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。

ステップ 4	**ip address fe80::1 link-local**
	例：
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		リンクローカル オプションを使用してインターフェイスで IPv6 アドレスを設定します。

ステップ 5	**ipv6 enable** 例： `Device(config)# ipv6 enable`	（任意）インターフェイス上で IPv6 をイネーブルにします。

	コマンドまたはアクション	目的

ステップ 6	**no ipv6 nd managed-config-flag**
	例：
	`Device(config)# interface vlan 1`
	`Device(config-if)# no ipv6 nd managed-config-flag`
		接続されたホストで、アドレスの取得にステートフル自動設定が使用されないようにします。

ステップ 7	**ipv6 nd other-config-flag** 例： `Device(config-if)# no ipv6 nd other-config-flag`	接続されたホストで、DHCPからの非アドレス オプションの取得に（ドメインなど）ステートフル自動設定が使用されないようにします。
ステップ 8	**end** 例： `Device(config)# end`	インターフェイスモードを終了します。

ステートフル DHCP のローカル設定

このインターフェイス設定は、ローカルデバイスのステートフルDHCPv6を実装しているCisco IOS Ipv6 ルータ用です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 unicast-routing** 例： `Device(config)# ipv6 unicast-routing`	ユニキャスト用にIPv6を設定します。
ステップ 4	**ipv6 dhcp pool IPv6_DHCPPOOL** 例： `Device(config)# ipv6 dhcp pool IPv6_DHCPPOOL`	コンフィギュレーションモードを開始し、VLAN の IPv6 DHCP プールを設定します。

	コマンドまたはアクション	目的
ステップ 5	**address prefix 2001:DB8:0:1:FFFF:1234::/64** 例： `Device(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64`	プールに入力するアドレス範囲を指定します。
ステップ 6	**dns-server 2001:100:0:1::1** 例： `Device(config-dhcpv6)# dns-server 2001:100:0:1::1`	DHCP クライアントに DNS サーバーのオプションを提供します。
ステップ 7	**domain-name example.com** 例： `Device(config-dhcpv6)# domain-name example.com`	DHCP クライアントにドメイン名オプションを提供します。
ステップ 8	**exit** 例： `Device(config-dhcpv6)# exit`	前のモードに戻ります。
ステップ 9	**interface vlan1** 例： `Device(config)# interface vlan 1`	インターフェイスモードを開始して、ステートフル DHCP を設定します。
ステップ 10	**description IPv6-DHCP-Stateful** 例： `Device(config-if)# description IPv6-DHCP-Stateful`	ステートフル IPv6 DHCP の説明を入力します。
ステップ 11	**ipv6 address 2001:DB8:0:20::1/64** 例： `Device(config-if)# ipv6 address 2001:DB8:0:20::1/64`	ステートフル IPv6 DHCP の IPv6 アドレスを入力します。
ステップ 12	**ip address 192.168.20.1 255.255.255.0** 例： `Device(config-if)# ip address 192.168.20.1 255.255.255.0`	ステートフル IPv6 DHCP の IPv6 アドレスを入力します。
ステップ 13	**ipv6 nd prefix 2001:db8::/64 no-advertise** 例： `Device(config-if)# ipv6 nd prefix 2001:db8::/64 no-advertise`	アドバタイズしてはならない、IPv6 ルーティングプレフィックスアドバタイズメントを設定します。

	コマンドまたはアクション	目的
ステップ 14	**ipv6 nd managed-config-flag** 例： `Device(config-if)# ipv6 nd managed-config-flag`	ホストでアドレス設定に DHCP を使用できるように、IPv6 インターフェイス ネイバー探索を設定します。
ステップ 15	**ipv6 nd other-config-flag** 例： `Device(config-if)# ipv6 nd other-config-flag`	ホストで非アドレス設定に DHCP を使用できるように、IPv6 インターフェイス ネイバー探索を設定します。
ステップ 16	**ipv6 dhcp server IPv6_DHCPPOOL** 例： `Device(config-if)# ipv6 dhcp server IPv6_DHCPPOOL`	インターフェイスに DHCP サーバーを設定します。

ステートフル DHCP の外部設定

このインターフェイス設定は、外部 DHCP サーバーのステートフル DHCPv6 を実装している Cisco IOS IPv6 ルータ用です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 unicast-routing** 例： `Device(config)# ipv6 unicast-routing`	ユニキャスト用に IPv6 を設定します。
ステップ 4	**dns-server 2001:100:0:1::1** 例： `Device(config-dhcpv6)# dns-server 2001:100:0:1::1`	DHCP クライアントに DNS サーバーのオプションを提供します。

	コマンドまたはアクション	目的
ステップ 5	**domain-name example.com** 例： `Device(config-dhcpv6)# domain-name example.com`	DHCP クライアントにドメイン名オプションを提供します。
ステップ 6	**exit** 例： `Device(config-dhcpv6)# exit`	前のモードに戻ります。
ステップ 7	**interface vlan1** 例： `Device(config)# interface vlan 1`	インターフェイスモードを開始して、ステートフル DHCP を設定します。
ステップ 8	**description IPv6-DHCP-Stateful** 例： `Device(config-if)# description IPv6-DHCP-Stateful`	ステートフル IPv6 DHCP の説明を入力します。
ステップ 9	**ipv6 address 2001:DB8:0:20::1/64** 例： `Device(config-if)# ipv6 address 2001:DB8:0:20::1/64`	ステートフル IPv6 DHCP の IPv6 アドレスを入力します。
ステップ 10	**ip address 192.168.20.1 255.255.255.0** 例： `Device(config-if)# ip address 192.168.20.1 255.255.255.0`	ステートフル IPv6 DHCP の IPv6 アドレスを入力します。
ステップ 11	**ipv6 nd prefix 2001:db8::/64 no-advertise** 例： `Device(config-if)# ipv6 nd prefix 2001:db8::/64 no-advertise`	アドバタイズしてはならない、IPv6 ルーティングプレフィックスアドバタイズメントを設定します。
ステップ 12	**ipv6 nd managed-config-flag** 例： `Device(config-if)# ipv6 nd managed-config-flag`	ホストでアドレス設定に DHCP を使用できるように、IPv6 インターフェイス ネイバー探索を設定します。
ステップ 13	**ipv6 nd other-config-flag** 例： `Device(config-if)# ipv6 nd other-config-flag`	ホストで非アドレス設定に DHCP を使用できるように、IPv6 インターフェイス ネイバー探索を設定します。

	コマンドまたはアクション	目的
ステップ 14	**ipv6 dhcp_relaydestination 2001:DB8:0:20::2** 例： `Device(config-if)# ipv6 dhcp_relay destination 2001:DB8:0:20::2`	インターフェイスに DHCP サーバーを設定します。

IPv6 アドレス ラーニング設定の確認

次に、**show ipv6 dhcp pool** コマンドの出力例を示します。このコマンドは、デバイスでの IPv6 サービスの設定を表示します。vlan 21 の設定済みプールの詳細には、プールからアドレスを現在使用している 6 つのクライアントが表示されます。

手順

	コマンドまたはアクション	目的

ステップ 1	**show ipv6 dhcp pool**
	例：
	`Device show ipv6 dhcp pool`
	`DHCPv6 pool: vlan21`
	`Address allocation prefix: 2001:DB8:0:1:FFFF:1234::/64 valid 86400 preferred 86400 (6 in use, 0 conflicts)`
	`DNS server: 2001:100:0:1::1`
	`Domain name: example.com`
	`Active clients: 6`
		デバイスでの IPv6 サービスの設定を表示します。

その他の参考資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)*

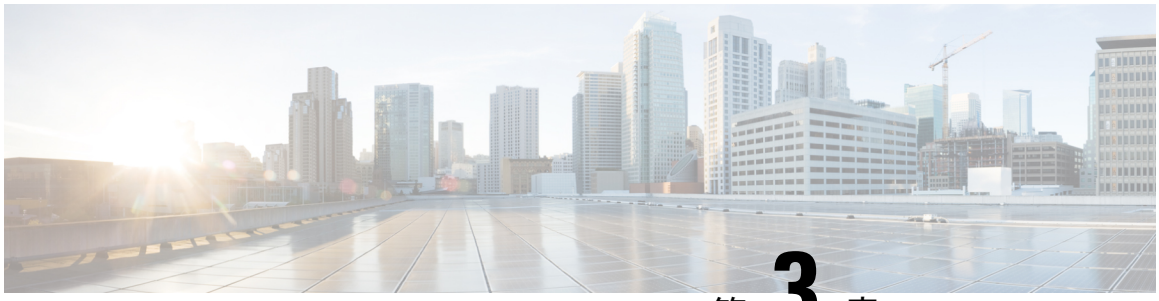
IPv6 クライアント アドレス ラーニングの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IPv6 クライアント アドレス ラーニング機能	クライアントアドレス ラーニングは、関連付け、再関連付け、認証解除、タイムアウトの際に、クライアントの IPv4 および IPv6 アドレス、デバイスによって保持されるクライアント変換の状態について学習するために、デバイスで設定されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 3 章

DHCP の設定

このセクションでは、DHCP の設定について説明します。

DHCP を設定するための前提条件

次の前提条件が DHCP スヌーピングおよびオプション 82 に適用されます。

- DHCP スヌーピングは、スイッチ上でグローバルにイネーブルにする必要があります。
- スイッチ上で DHCP スヌーピングをグローバルにイネーブルにする前に、DHCP サーバーや DHCP リレー エージェントとして機能するデバイスが設定され、イネーブルになっていることを確認してください。
- スイッチを DHCP 要求に応答するようにする場合は、DHCP サーバーとして設定する必要があります。
- スイッチで DHCP スヌーピング情報オプションを設定する前に、DHCP サーバーとして機能するデバイスを設定してください。DHCP サーバーが割り当てたり除外したりできる IP アドレスを指定するか、またはそれらのデバイスの DHCP オプションを設定する必要があります。
- DHCP スヌーピングが正常に機能するには、すべての DHCP サーバを信頼できるインターフェイスを介してスイッチに接続し、信頼できない DHCP メッセージが信頼できるインターフェイスにだけ転送されるようにする必要があります。サービス プロバイダ ネットワークでは、同じネットワーク内のデバイスのポートに接続されたインターフェイスが信頼できるインターフェイスとなります。
- DHCP スヌーピングで Cisco IOS DHCP サーバー バインディング データベースを使用するには、Cisco IOS DHCP サーバー バインディング データベースを使用するようにスイッチを設定する必要があります。

- 信頼できない入力でパケットを受け入れる DHCP スヌーピング オプションを使用するには、スイッチがエッジ スイッチからオプション 82 情報を含むパケットを受信する集約スイッチである必要があります。
- 次の前提条件が DHCP スヌーピング バインディング データベースの設定に適用されます。
  - DHCP スヌーピング用にスイッチを使用するには、DHCP スヌーピング バインディング データベースで宛先を設定する必要があります。
  - NVRAM とフラッシュ メモリは、いずれも記憶容量が限られているため、バインディング ファイルを TFTP サーバーに保存することを推奨します。
  - ネットワーク ベースの URL（TFTP や FTP など）については、スイッチがバインディングをその URL のバインディング ファイルに初めて書き込む前に、設定された URL に空のファイルを作成する必要があります。空のファイルをサーバ上に作成する必要があるかどうかについては、TFTP サーバのマニュアルを参照してください。TFTP サーバによっては、そのように設定できないことがあります。
  - データベースに正しいリース期間が記録されるように、ネットワーク タイム プロトコル（NTP）をイネーブルにし、設定することを推奨します。
  - NTP が設定されている場合、スイッチのシステム クロックが NTP と同期化されたときにだけ、スイッチがバインディングの変更内容をバインディング ファイルに書き込みます。
- スイッチで DHCP リレー エージェントを設定する前に、DHCP サーバーとして機能するデバイスを設定してください。DHCP サーバーが割り当てたり除外したりできる IP アドレスを指定するか、デバイスの DHCP オプションを設定するか、または DHCP データベース エージェントをセットアップする必要があります。
- スイッチが DHCP パケットをリレーするようにする場合は、DHCP サーバーの IP アドレスは DHCP クライアントのスイッチ仮想インターフェイス（SVI）に設定する必要があります。
- スイッチポートが DHCP サーバに接続されている場合は、**ip dhcp snooping trust interface** コンフィギュレーション コマンドを入力して、ポートを信頼できるポートとして設定してください。
- スイッチポートが DHCP クライアントに接続されている場合は、**no ip dhcp snooping trust** インターフェイス コンフィギュレーション コマンドを入力して、ポートを信頼できないポートとして設定してください。

DHCP の設定に関する制限

DHCP スヌーピング、DHCP リレーエージェントをサポートする送信（Tx）スイッチドポートアナライザ（SPAN）または出力 SPAN は使用しないことを推奨します。Tx での SPAN が必要な場合は、DHCP パケットの転送パスに含まれる VLAN ポートを使用しないでください。

DHCP に関する情報

DHCP サーバ

DHCP サーバーは、スイッチまたはルータ上の指定されたアドレス プールから DHCP クライアントにIPアドレスを割り当て、それらのアドレスを管理します。DHCPサーバーがそのデータベースから要求された設定パラメータを取得してDHCPクライアントに渡すことができない場合は、ネットワーク管理者が定義した1つまたは複数のセカンダリ DHCPサーバーに要求を転送します。スイッチは、DHCP サーバーとして機能できます。DHCP サーバーは、要求された設定をクライアントに送信するときに、メッセージを他のサーバーに転送しません。

DHCP リレー エージェント

DHCP リレー エージェントは、クライアントとサーバの間で DHCP パケットを転送するレイヤ3デバイスです。リレーエージェントは、同じ物理サブネット上にないクライアントとサーバの間で要求および応答を転送します。リレー エージェントによる転送は、IP データグラムをネットワーク間で透過的に交換するレイヤ 2 での通常の転送とは異なります。リレー エージェントは、DHCPメッセージを受け取ると、新しいDHCPメッセージを生成して、出力インターフェイス上で送信します。

DHCP スヌーピング

DHCPスヌーピングは、信頼できないDHCPメッセージのフィルタリングとDHCPスヌーピング バインディング データベース（DHCP スヌーピング バインディング テーブルとも呼ばれる）の作成および管理によってネットワーク セキュリティを確保する DHCP セキュリティ機能です。

DHCPスヌーピングは、信頼できないホストとDHCPサーバの間でファイアウォールに似た役割を果たします。DHCPスヌーピングを使用することにより、エンドユーザーに接続された信頼できないインターフェイスとDHCPサーバーまたは別のスイッチに接続された信頼できるインターフェイスを区別できます。

（注） DHCPスヌーピングが正常に機能するには、すべてのDHCPサーバを信頼できるインターフェイスを介してスイッチに接続し、信頼できないDHCPメッセージが信頼できるインターフェイスにだけ転送されるようにする必要があります。

信頼できない DHCP メッセージとは、信頼できないインターフェイス経由で送信されたメッセージのことです。デフォルトでは、スイッチはすべてのインターフェイスを信頼できないものと見なします。そのため、スイッチはいくつかのインターフェイスを信頼してDHCPスヌーピングを使用するように設定する必要があります。サービス プロバイダ環境で DHCP スヌーピングを使用する場合は、カスタマーのスイッチなど、サービス プロバイダ ネットワーク内

には存在しないデバイスから送信されたメッセージが信頼できないメッセージとなります。不明なデバイスから送信されたメッセージは、トラフィック攻撃の原因になりうるため、信頼できません。

DHCP スヌーピング バインディング データベースには、MAC アドレス、IP アドレス、リース期間、バインディングの種類、VLAN 番号、およびスイッチの信頼できないローカルインターフェイスのインターフェイス情報が含まれています。このデータベースには、信頼できるインターフェイスに接続されたホストの情報はありません。

サービス プロバイダー ネットワークでは、信頼できるインターフェイスとして設定できるものの例として、同じネットワーク内のデバイスのポートに接続されたインターフェイスがあります。信頼できないインターフェイスには、ネットワーク内の信頼できないインターフェイスまたはネットワークに属さないデバイスのインターフェイスに接続されたインターフェイスがあります。

スイッチが信頼できないインターフェイスでパケットを受信し、そのインターフェイスが属している VLAN で DHCP スヌーピングがイネーブルに設定されている場合、スイッチは送信元 MAC アドレスと DHCP クライアントのハードウェア アドレスを比較します。アドレスが一致した場合（デフォルト）、スイッチはパケットを転送します。アドレスが一致しない場合、スイッチはパケットをドロップします。

スイッチは、次のいずれかの状況が発生した場合に DHCP パケットをドロップします。

- DHCPOFFER パケット、DHCPACK パケット、DHCPNAK パケット、DHCPLEASEQUERY パケットなど、DHCP サーバからのパケットがネットワークまたはファイアウォールの外側から着信した。
- パケットが信頼できないインターフェイスに着信し、送信元 MAC アドレスと DHCP クライアントのハードウェア アドレスが一致しない。
- スイッチが DHCPRELEASE または DHCPDECLINE ブロードキャスト メッセージを受信し、その MAC アドレスは DHCP スヌーピング バインディング データベースに含まれているが、バインディング データベース内のインターフェイス情報がメッセージを受信したインターフェイスと一致しない。
- DHCP リレー エージェントが 0.0.0.0 以外のリレー エージェント IP アドレスを含む DHCP パケットを転送し、Option 82 情報が含まれないパケットを信頼できないポートに転送する。
- DHCP スヌーピングがイネーブルになっている場合に、最大スヌーピングキューサイズの 1000 を超える。

DHCP スヌーピングをサポートする集約スイッチであり、DHCP オプション 82 情報を挿入するエッジ スイッチに接続されているスイッチは、オプション 82 情報を含むパケットが信頼できないインターフェイスに着信した場合、それらのパケットをドロップします。DHCP スヌーピングがイネーブルに設定されている場合に、パケットが信頼できるポートに着信しても、集約スイッチは接続されたデバイスの DHCP スヌーピング バインディングを認識せず、完全な DHCP スヌーピング バインディング データベースを作成できません。

集約スイッチを信頼できないインターフェイス経由でエッジスイッチに接続できる場合、**ip dhcp snooping information option allow-untrusted** グローバル コンフィギュレーション コマンドを入力すると、集約スイッチはエッジスイッチによって挿入されたオプション 82 情報を含むパケットを受け入れます。集約スイッチは、信頼できないスイッチインターフェイスを介して接続されたホストのバインディングを認識します。集約スイッチで、ダイナミック ARP インスペクションや IP ソース ガードなど、DHCP セキュリティ機能をイネーブルに設定することもできますが、その場合でもスイッチは Option 82 情報を含むパケットをホストが接続されている信頼できない入力インターフェイスで受信します。集約スイッチ上のエッジスイッチとの接続ポートは、信頼できるインターフェイスとして設定する必要があります。

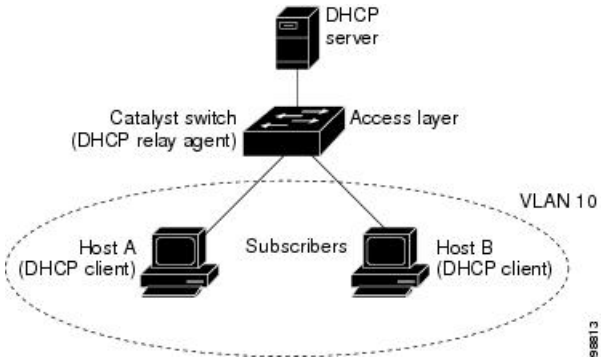
オプション 82 データ挿入

住宅地域にあるメトロポリタン イーサネット アクセス環境では、DHCP は多数の加入者に対し、IP アドレスの割り当てを一元的に管理できます。スイッチで DHCP スヌーピングの Option 82 機能をイネーブルにすると、加入者装置は MAC アドレスだけでなく、その装置をネットワークに接続するスイッチ ポートによっても識別されます。サブスクライバ LAN 上の複数のホストをアクセス スイッチの同じポートに接続できます。これらのホストは一意に識別されます。

（注） DHCP オプション 82 機能は、DHCP スヌーピングがグローバルに有効であり、オプション 82 を使用する加入者装置が割り当てられた VLAN で有効である場合に限りサポートされます。

次の図に、一元的な DHCP サーバーがアクセス レイヤのスイッチに接続された加入者に IP アドレスを割り当てるメトロポリタン イーサネット ネットワークを示します。DHCP クライアントとそれらに関連付けられた DHCP サーバは同じ IP ネットワークまたはサブネット内に存在しないため、DHCP リレー エージェント（Catalyst スイッチ）にヘルパー アドレスを設定することにより、ブロードキャスト転送をイネーブルにし、クライアントとサーバ間で DHCP メッセージを転送します。

図 *3:* メトロポリタン イーサネット ネットワークにおける *DHCP* リレー エージェント



スイッチで DHCP スヌーピング情報 オプション 82 を有効にすると、次のイベントがこの順序で発生します。

- ホスト（DHCP クライアント）は DHCP 要求を生成し、これをネットワーク上にブロードキャストします。
- スイッチは、この DHCP 要求を受信すると、パケットに Option 82 情報を追加します。デフォルトでは、リモート ID サブオプションがスイッチの MAC アドレスで、回線 ID サブオプションはパケットを受信するポート ID（**vlan-mod-port**）です。リモート ID と回線 ID を設定できます。
- リレー エージェントの IP アドレスが設定されている場合、スイッチはこの IP アドレスを DHCP パケットに追加します。
- スイッチは、オプション 82 フィールドを含む DHCP 要求を DHCP サーバーに転送します。
- DHCP サーバはこのパケットを受信します。Option 82 に対応しているサーバであれば、リモート ID と回線 ID のいずれか一方または両方を使用して、IP アドレスを割り当てたり、1 つのリモート ID または回線 ID に割り当てることができる IP アドレスの数を制限するようなポリシーを実装したりできます。次に DHCP サーバは、DHCP 応答内にオプション 82 フィールドをエコーします。
- スイッチによって要求がサーバーにリレーされた場合、DHCP サーバーは応答をスイッチにユニキャストします。スイッチは、リモート ID フィールドと、場合によっては回線 ID フィールドを調べ、Option 82 データが挿入済みであることを確認します。スイッチは Option 82 フィールドを削除してから、DHCP 要求を送信した DHCP クライアントに接続するスイッチ ポートにパケットを転送します。

デフォルトのサブオプション設定では、前述のイベントのシーケンスが発生すると、次のフィールドの値は変化しません（図「サブオプションのパケット形式」を参照）。

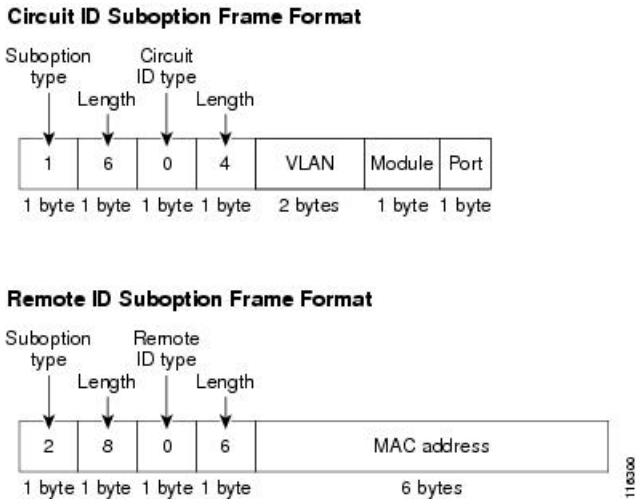
- 回線 ID サブオプション フィールド
  - サブオプション タイプ
  - サブオプション タイプの長さ
  - 回線 ID タイプ
  - 回線 ID タイプの長さ
- リモート ID サブオプション フィールド
  - サブオプション タイプ
  - サブオプション タイプの長さ
  - リモート ID タイプ
  - リモート ID タイプの長さ

回線 ID サブオプションのポート フィールドでは、ポート番号が 3 から始まります。たとえば、24 個の 10/100/1000 ポートおよび 4 つの Small Form-Factor Pluggable（SFP）モジュール ス

ロットを搭載するスイッチでは、ポート 3 がギガビット イーサネット 1/0/1 ポート、ポート 4 がギガビット イーサネット 1/0/2 ポートとなり、以降同様に続きます。ポート 27 は SFP モジュール スロットのギガビット イーサネット 1/0/25 となり、以降同様に続きます。

図「サブオプションのパケット形式」に、デフォルトのサブオプション設定が使用されている場合のリモート ID サブオプションおよび回線 ID サブオプションのパケット形式を示します。回線 ID サブオプションでは、モジュール番号は、スタックにあるスイッチ番号に対応します。スイッチがこれらのパケット形式を使用するのは、DHCP スヌーピングをグローバルに有効にし、ip dhcp snooping information option グローバル コンフィギュレーション コマンドを入力した場合です。

図 *4*: サブオプションのパケット形式

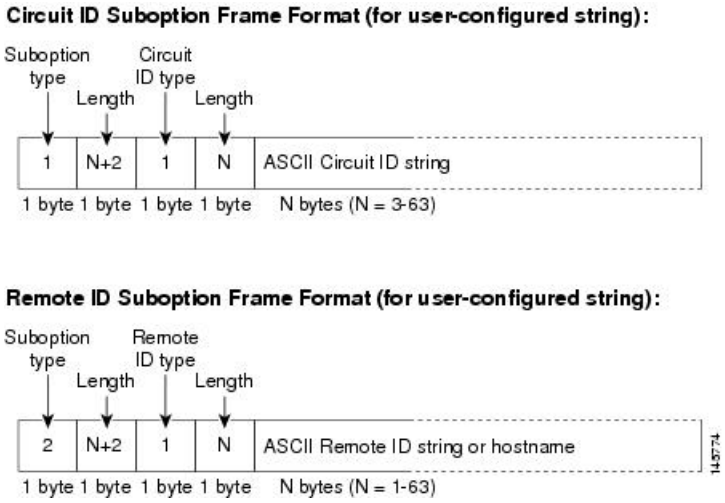


図「ユーザー設定のサブオプションのパケット形式」は、ユーザー設定のリモート ID サブオプション、および回線 ID サブオプションのパケット形式を示しています。スイッチでは、DHCP スヌーピングをグローバルにイネーブルにし、**ip dhcp snooping information option format remote-id** グローバル コンフィギュレーション コマンド、および **ip dhcp snooping vlan information option format-type circuit-id string** インターフェイス コンフィギュレーション コマンドを入力した場合に、これらのパケット形式が使用されます。

パケットでは、リモート ID および回線 ID サブオプションを次のように設定した場合、これらのフィールドの値がデフォルト値から変更されます。

- 回線 ID サブオプション フィールド
  - 回線 ID タイプが 1 である。
  - 設定した文字列の長さに応じて、長さの値が変化する。
- リモート ID サブオプション フィールド
  - リモート ID タイプが 1 である。
  - 設定した文字列の長さに応じて、長さの値が変化する。

図 *5:* ユーザ設定のサブオプションのパケット形式



Cisco IOS DHCP サーバ データベース

DHCP ベースの自動設定プロセスの間、指定 DHCP サーバは Cisco IOS DHCP サーバ データベースを使用します。これには IP アドレス、アドレス バインディング、およびブート ファイルなどの設定パラメータが含まれます。

アドレス バインディングは、Cisco IOS DHCP サーバ データベース内のホストの IP アドレスおよび MAC アドレス間のマッピングです。クライアント IP アドレスを手動で割り当てること、または、DHCP サーバが DHCP アドレス プールから IP アドレスを割り当てることが可能です。

DHCP スヌーピング バインディング データベース

DHCP スヌーピングをイネーブルにすると、スイッチは信頼できないインターフェイスに関する情報を DHCP スヌーピング バインディング データベースに保存します。データベースには、64,000 のバインディングを含めることができます。

各データベース エントリ（バインディング）は、IP アドレス、それに関連付けられた MAC アドレス、リース期間（16 進形式）、バインディングが適用されるインターフェイス、およびインターフェイスが属する VLAN で構成されます。データベース エージェントは、設定された場所のファイルにバインディングを保存します。各エントリの末尾にあるチェックサムは、ファイルの先頭のバイトを含め、エントリに関連付けられたすべてのバイトを対象として計算されます。各エントリは、まず 77 バイトのデータがあり、その後に 1 つのスペースとチェックサム値と EOL 記号が続きます。

スイッチのリロード後もバインディングを保持するには、DHCP スヌーピング データベース エージェントを使用する必要があります。エージェントがディセーブルで、ダイナミック ARP インスペクションまたは IP ソース ガードがイネーブルにされ、DHCP スヌーピング バインディング データベースがダイナミック バインディングされている場合、スイッチは接続を切断されます。このエージェントがディセーブルで、DHCP スヌーピングだけがイネーブルである

場合、スイッチの接続は切断されませんが、DHCP スヌーピングは DHCP スプーフィング攻撃を防止できないことがあります。

リロードすると、スイッチはバインディング ファイルを読み込み、DHCP スヌーピング バインディングデータベースを作成します。スイッチは、データベースに変更が加えられたときにはバインディング ファイルを更新します。

スイッチは、新しいバインディングを認識するか、バインディングを失うと、ただちにデータベース内のエントリを更新します。スイッチはバインディング ファイル内のエントリも更新します。バインディング ファイルの更新頻度は設定可能な遅延時間によって決まり、更新はバッチ処理されます。ファイルが指定された時間内（書き込み遅延および中断タイムアウトの値によって設定される）に更新されない場合、更新は停止します。

バインディングが含まれるファイルの形式は次のとおりです。

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

このファイルの各エントリにはチェックサム値を示すタグが付けられます。スイッチは、ファイルを読み取るときに、このチェックサムを使用してエントリを検証します。最初の行の initial-checksum エントリは、最新のファイル更新に関連するエントリを以前のファイル更新に関連するエントリと区別します。

次に、バインディング ファイルの例を示します。

```
3ebe1518
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
10.1.1.1 512 001.0001.0005 3EBE2881 Gi1/1                                   e5e1e733
10.1.1.1 512 001.0001.0002 3EBE2881 Gi1/1                                   4b3486ec
10.1.1.1 1536 001.0001.0004 3EBE2881 Gi1/1                                  f0e02872
10.1.1.1 1024 001.0001.0003 3EBE2881 Gi1/1                                  ac41adf9
10.1.1.1 1 001.0001.0001 3EBE2881 Gi1/1                                     34b3273e
END
```

スイッチが起動し、計算されたチェックサム値が保存されているチェックサム値と一致した場合、スイッチはバインディング ファイルのエントリを読み取り、バインディングを DHCP スヌーピング バインディングデータベースに追加します。次のいずれかの状況が発生した場合、スイッチはエントリを無視します。

- スイッチがエントリを読み取り、計算されたチェックサム値が保存されているチェックサム値と一致しない。この場合、そのエントリとそれ以降のエントリは無視されます。
- エントリに含まれているリース期間が終了している（スイッチはリース期間の終了時にバインディング エントリを削除しないことがある）。

- エントリに含まれるインターフェイスが現在はシステムに存在しない。
- インターフェイスがルーテッド インターフェイスまたは DHCP スヌーピングにおける信頼できるインターフェイスである。

DHCP スヌーピングおよびスイッチ スタック

DHCP スヌーピングは、アクティブスイッチで管理されます。新しいスイッチは、スタックに追加されると、アクティブスイッチから DHCP スヌーピング設定を受信します。メンバがスタックから除外されると、スイッチに関連付けられているすべての DHCP スヌーピング アドレス バインディングがエージング アウトします。

すべてのスヌーピング統計情報は、アクティブスイッチ上で生成されます。新しいアクティブスイッチが選定された場合、統計カウンタはリセットされます。

スタックのマージが発生し、アクティブスイッチではなくなった場合、アクティブスイッチにあったすべての DHCP スヌーピングバインディングが失われます。スタックパーティションを使用すると、既存のアクティブスイッチは変更されず、パーティション分割されたスイッチに属しているバインディングはエージアウトします。パーティション分割されたスタックの新しいアクティブスイッチで、新たな着信 DHCP パケットの処理が開始されます。

DHCP スヌーピングのデフォルト設定

表 *1 : DHCP* のデフォルト設定

機能	デフォルト設定
DHCP サーバ	Cisco IOS ソフトウェアではイネーブル、設定が必要[1]
DHCP リレー エージェント	イネーブル[2]
DHCP パケット転送アドレス	未設定
リレー エージェント情報の確認	イネーブル（無効なメッセージは廃棄）
DHCP リレー エージェント転送ポリシー	既存のリレー エージェント情報を置換。
DHCP スヌーピングをグローバルにイネーブル	ディセーブル
DHCP スヌーピング情報オプション	イネーブル
パケットを信頼できない入力インターフェイスで受け取る DHCP スヌーピング オプション[3]	ディセーブル
DHCP スヌーピング レート制限	未設定
DHCP スヌーピング信頼状態	信頼できない

機能	デフォルト設定
DHCP スヌーピング VLAN	ディセーブル
DHCP スヌーピングの MAC アドレス検証	イネーブル
Cisco IOS DHCP サーバ バインディング データベース	Cisco IOS ソフトウェアではイネーブル、設定が必要。 （注） スイッチは、DHCP サーバとして設定されているデバイスからだけ、ネットワーク アドレスおよび設定パラメータを取得します。
DHCP スヌーピング バインディング データベース エージェント	Cisco IOS ソフトウェアではイネーブル、設定が必要。この機能は宛先が設定されている場合に限り有効。

[1] スイッチは、DHCP サーバとして設定されている場合に限り DHCP 要求に応答します。
[2] スイッチは、DHCP サーバの IP アドレスが DHCP クライアントの SVI に設定されている場合に限り DHCP パケットをリレーします。
[3] この機能は、スイッチがエッジスイッチによってオプション 82 情報が挿入されたパケットを受信する集約スイッチである場合に使用します。

DHCP スヌーピング設定時の注意事項

- スイッチポートが DHCP サーバに接続されている場合は、**ip dhcp snooping trust interface** コンフィギュレーション コマンドを入力して、ポートを信頼できるポートとして設定してください。
- スイッチポートが DHCP クライアントに接続されている場合は、**no ip dhcp snooping trust** インターフェイス コンフィギュレーション コマンドを入力して、ポートを信頼できないポートとして設定してください。
- **show ip dhcp snooping statistics** ユーザー EXEC コマンドを入力して DHCP スヌーピング統計情報を表示したり、**clear ip dhcp snooping statistics** 特権 EXEC コマンドを入力してスヌーピング統計情報をクリアしたりできるようになりました。

DHCP サーバーとスイッチ スタック

DHCPバインディングデータベースは、アクティブスイッチで管理されます。新しいアクティブスイッチが割り当てられると、新しいアクティブスイッチに、TFTP サーバーで保存されているバインディングデータベースがダウンロードされます。スイッチの切り替えが発生した場合、新しいアクティブスイッチは、SSO機能を使用して以前のアクティブスイッチで同期されたデータベースファイルを使用します。失われたバインディングに関連付けられていた IP ア

ドレスは、解放されます。自動バックアップは、**ip dhcp database** *url* [**timeout** *seconds* | **write-delay** *seconds*] グローバル コンフィギュレーション コマンドを使用して設定する必要があります。

DHCP サーバ ポートベースのアドレス割り当て

DHCP サーバー ポートベースのアドレス割り当ては、接続されたデバイス クライアントの ID またはクライアント ハードウェア アドレスに関係なく、DHCP がイーサネット スイッチ ポートで同じ IP アドレスを維持できるようにする機能です。

ネットワークに導入されたイーサネット スイッチは、直接接続されたデバイスに接続を提供します。工場の作業場など、一部の環境では、あるデバイスで不具合が発生した場合は、それと同時に、そのネットワークで代わりのデバイスが動作を開始しなければなりません。現在の DHCP 実装では、この代わりのデバイスに、DHCP が同じ IP アドレスを提供する保証はありません。コントロールやモニタリングなどを行うソフトウェアは、各デバイスに関連付けられた IP アドレスが一定であることを期待しています。デバイスを交換した場合、DHCP クライアントが変更された場合でも、アドレスの割り当ては一定のままでなければなりません。

DHCP サーバ ポートベースのアドレス割り当て機能が設定されている場合、この機能により、ある接続ポートで受信された DHCP メッセージでクライアント ID やクライアント ハードウェア アドレスが変更されたとしても、同じ接続ポートには常に同じ IP アドレスが提供されることが保証されます。DHCP プロトコルは、DHCP パケットのクライアント ID オプションにより、DHCP クライアントを識別します。クライアント ID オプションを含まないクライアントは、クライアント ハードウェア アドレスにより識別されます。この機能を設定すると、インターフェイスのポート名が、クライアント ID またはハードウェア アドレスよりも優先され、実際の接続ポイントであるスイッチ ポートがクライアント ID になります。

すべてのケースで、同じポートにイーサネット ケーブルを接続することにより、接続されたデバイスに、DHCP 経由で同じ IP アドレスが割り当てられます。

DHCP サーバ ポートベースのアドレス割り当て機能がサポートされているのは、Cisco IOS DHCP サーバだけです。サードパーティ製のサーバではサポートされていません。

ポートベースのアドレス テーブルのデフォルト設定

デフォルトでは、DHCP サーバ ポートベースのアドレス割り当てはディセーブルにされています。

ポートベースのアドレス割り当て設定時の注意事項

- デフォルトでは、DHCP サーバ ポートベースのアドレス割り当てはディセーブルにされています。
- DHCP プールから事前に設定された予約への割り当てを制限する（予約されていないアドレスはクライアントに提供されず、その他のクライアントはプールによるサービスを受けない）ために、**reserved-only** DHCP プール コンフィギュレーション コマンドを入力することができます。

DHCP の設定方法

DHCP サーバの設定

スイッチは、DHCP サーバーとして機能できます。管理ポートを備えた DHCP クライアント用に DHCP サーバーを使用する場合は、管理 VRF を使用して DHCP プールと対応するインターフェイスの両方を設定する必要があります。

DHCP リレー エージェントの設定

スイッチ上で DHCP リレー エージェントをイネーブルにするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**service dhcp** 例： `Device(config)# service dhcp`	スイッチ上で DHCP サーバおよび DHCP リレー エージェントをイネーブルにします。デフォルトでは、この機能はイネーブルです。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

- リレー エージェント情報のチェック（検証）
- リレー エージェント転送ポリシーの設定

パケット転送アドレスの指定

DHCP サーバーおよび DHCP クライアントが異なるネットワークまたはサブネットにある場合、スイッチを **ip helper-address** *address* インターフェイス コンフィギュレーション コマンドで設定する必要があります。一般的なルールは、クライアントに最も近いレイヤ 3 インターフェイス上にコマンドを設定することです。**ip helper-address** コマンドで使用されているアドレスは、特定の DHCP サーバ IP アドレスか、または他の DHCP サーバが宛先ネットワークセグメントにある場合はネットワークアドレスにすることができます。ネットワーク アドレスを使用することで、どの DHCP サーバも要求に応答できるようになります。

パケット転送アドレスを指定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface vlan** *vlan-id* 例： `Device(config)# interface vlan 1`	VLAN ID を入力してスイッチの仮想インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address subnet-mask* 例： `Device(config-if)# ip address 192.108.1.27 255.255.255.0`	インターフェイスに IP アドレスおよび IP サブネットを設定します。
ステップ 5	**ip helper-address** *address* 例： `Device(config-if)# ip helper-address 172.16.1.2`	DHCP パケット転送アドレスを指定します。 • ヘルパー アドレスは特定の DHCP サーバ アドレスにするか、他の DHCP サーバが宛先ネットワーク セグメントにある場合は、ネットワーク アドレスにすることができます。ネットワーク アドレスを使用することで、他のサーバも

	コマンドまたはアクション	目的
		DHCP 要求に応答できるようになります。 • 複数のサーバがある場合、各サーバに 1 つのヘルパー アドレスを設定できます。
ステップ 6	**exit** 例： `Device(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、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	次のいずれかを使用します。 • **interface range** *port-range* • **interface** *interface-id* 例： `Device(config)# interface gigabitethernet 1/0/2`	DHCP クライアントに接続されている複数の物理ポートを設定し、インターフェイス範囲コンフィギュレーション モードを開始します。 または DHCP クライアントに接続されている単一の物理ポートを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**switchport mode access** 例： `Device(config-if)# switchport mode access`	ポートの VLAN メンバーシップ モードを定義します。
ステップ 9	**switchport access vlan** *vlan-id* 例： `Device(config-if)# switchport access vlan 1`	ステップ 2 で設定したのと同じ VLAN をポートに割り当てます。
ステップ 10	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

DHCP for IPv6 アドレス割り当ての設定

DHCPv6 アドレス割り当てのデフォルト設定

デフォルトで、DHCPv6 機能はスイッチに設定されています。

DHCPv6 アドレス割り当ての設定時の注意事項

DHCPv6 アドレス割り当ての設定時には、次の前提条件が適用されます。

- 次の手順では、次に示すレイヤ 3 インターフェイスの 1 つを指定する必要があります。
  - IPv6 アドレスが明示的に設定されていない場合は、**ipv6 enable** コマンドを使用して IPv6 ルーティングを有効にします。
  - レイヤ 3 インターフェイスで DHCPv6 ルーティングが有効になっている必要があります。
  - SVI：**interface vlan** *vlan_id* コマンドを使用して作成された VLAN インターフェイス。
  - レイヤ 3 モードの EtherChannel ポートチャネル：**interface port-channel port-channel-number** コマンドを使用して作成されたポートチャネル論理インターフェイス。
- デバイスは、DHCPv6 クライアント、サーバー、またはリレーエージェントの役割を果たすことが可能です。DHCPv6 クライアント、サーバー、およびリレー機能は、インターフェイスで相互に排他的です。
- Cisco IOS XE Gibraltar 16.11.1 以降、DHCPv6 アドレスには、RFC5453 で指定されている予約済みインターフェイス識別子の範囲に含まれないインターフェイス識別子が含まれるようになります。

DHCPv6 サーバー機能の有効化（CLI）

DHCPv6 プールの特性を変更するには、**no** 形式の DHCP プール コンフィギュレーション モード コマンドを使用します。インターフェイスに対して DHCPv6 サーバー機能を無効にするには、**no ipv6 dhcp server** インターフェイス コンフィギュレーション コマンドを使用します。

インターフェイスで DHCPv6 サーバー機能を有効にするには、次の手順を実行します。

Procedure

	Command or Action	Purpose
ステップ 1	**enable** **Example:** `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **Example:** `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	**ipv6 dhcp pool** *poolname* **Example:** `Device(config)# ipv6 dhcp pool 7`	DHCP プール コンフィギュレーション モードを開始して、IPv6 DHCP プールの名前を定義します。プール名は、記号文字列（Engineering など）または整数（0 など）です。
ステップ 4	**address prefix** *IPv6-prefix* {**lifetime**} {*t1 t1* | **infinite**} **Example:** `Device(config-dhcpv6)# address prefix 2001:1000::0/64 lifetime 3600`	（任意）アドレス割り当て用のアドレス プレフィックスを指定します。 このアドレスは、16 ビット値をコロンで区切った 16 進数で指定する必要があります。 **lifetime** *t1 t1*：IPv6 アドレス プレフィックスが有効な状態を維持するタイムインターバル（秒）を指定します。指定できる範囲は 5 ～ 4294967295 秒です。時間間隔なしの場合は、**infinite** を指定します。
ステップ 5	**link-address** *IPv6-prefix* **Example:** `Device(config-dhcpv6)# link-address 2001:1002::0/64`	（任意）link-address IPv6 プレフィックスを指定します。 着信インターフェイス上のアドレスまたはパケットのリンクアドレスが指定した IPv6 プレフィックスに一致する場合、サーバーは設定情報プールを使用します。 このアドレスは、16 ビット値をコロンで区切った 16 進数で指定する必要があります。
ステップ 6	**vendor-specific** *vendor-id* **Example:** `Device(config-dhcpv6)# vendor-specific 9`	（任意）ベンダー固有のコンフィギュレーション モードを開始して、ベンダー固有の ID 番号を指定します。この番号は、ベンダーの IANA プライベート エンタープライズ番号です。指定できる範囲は 1 ～ 4294967295 です。
ステップ 7	**suboption** *number* { **address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string*} **Example:** `Device(config-dhcpv6-vs)# suboption 1 address 1000:235D::`	（任意）ベンダー固有のサブオプション番号を入力します。指定できる範囲は 1 ～ 65535 です。IPv6 アドレス、ASCII テキスト、または 16 進文字列をサブオプション パラメータで定義されているように入力します。

	Command or Action	Purpose
ステップ 8	**exit** **Example:** `Device(config-dhcpv6-vs)# exit`	DHCP プール コンフィギュレーション モードに戻ります。
ステップ 9	**exit** **Example:** `Device(config-dhcpv6)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	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、設定するインターフェイスを指定します。
ステップ 11	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**] **Example:** `Device(config-if)# ipv6 dhcp server automatic`	インターフェイスに対して DHCPv6 サーバー機能を有効にします。 • *poolname*：（任意）IPv6 DHCP プールのユーザー定義の名前。プール名は、記号文字列（Engineering など）または整数（0 など）です。 • **automatic**：（任意）サーバーが、クライアントにアドレスを割り当てるときに使用するプールを自動的に決定できるようにします。 • **rapid-commit**：（任意）2 つのメッセージを交換する方式を許可します。 • **preference** 値：（任意）サーバーによって送信されるアドバタイズメント メッセージ内のプリファレンス オプションで指定するプリファレンス値を設定します。範囲は 0 ～ 255 です。デフォルトのプリファレンス値は 0 です。 • **allow-hint**：（任意）サーバーが SOLICIT メッセージに含まれるクライアントの提案を考慮するかど

	Command or Action	Purpose
		うかを指定します。デフォルトでは、サーバーはクライアントのヒントを無視します。
ステップ 12	**end** **Example:** `Device(config)# end`	特権 EXEC モードに戻ります。

ステップ 13	次のいずれかを実行します。
	• **show ipv6 dhcp pool**
	• **show ipv6 dhcp interface**
	Example:
	`Device# show ipv6 dhcp pool`
	または
	`Device# show ipv6 dhcp interface`
		• DHCPv6 プール設定を確認します。
		• DHCPv6 サーバー機能がインターフェイス上で有効であることを確認します。

ステップ 14	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

DHCPv6 クライアント機能の有効化

インターフェイスで DHCPv6 クライアントを有効にするには、次の手順を実行します。

Procedure

	Command or Action	Purpose
ステップ 1	**enable** **Example:** `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **Example:** `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	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、設定するインターフェイスを指定します。
ステップ 4	**ipv6 address dhcp** [**rapid-commit**] **Example:** `Device(config-if)# ipv6 address dhcp rapid-commit`	インターフェイスで DHCPv6 サーバーから IPv6 アドレスを取得できるようにします。 **rapid-commit**：（任意）アドレス割り当てに2つのメッセージを交換する方式を許可します。
ステップ 5	**ipv6 dhcp client request** [**vendor-specific**] **Example:** `Device(config-if)# ipv6 dhcp client request vendor-specific`	（任意）インターフェイスでベンダー固有のオプションを要求できるようにします。
ステップ 6	**end** **Example:** `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ipv6 dhcp interface** **Example:** `Device# show ipv6 dhcp interface`	DHCPv6 クライアントがインターフェイスで有効になっていることを確認します。

Cisco IOS DHCP サーバ データベースのイネーブル化

Cisco IOS DHCP サーバーデータベースを有効にして設定する手順については、『Cisco IOS IP Configuration Guide』の「Configuring DHCP」の章にある「DHCP Configuration Task List」のセクションを参照してください。

DHCP スヌーピング バインディング データベース エージェントのイネーブル化

スイッチ上で DHCP スヌーピング バインディング データベース エージェントをイネーブルにし、設定するには、特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp://***user***:***password***@***host*/*filename* | **http://**[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp://***user***@***host*/*filename*}| **tftp://***host*/*filename* 例： `Device(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2`	次のいずれかの形式を使用して、データベース エージェントまたはバインディング ファイルの URL を指定します。 • **flash**[*number*]**:**/*filename* • **ftp://***user***:***password***@***host*/*filename* • **http://**[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** • **rcp://***user***@***host*/*filename* • **tftp://***host*/*filename*
ステップ 4	**ip dhcp snooping database timeout** *seconds* 例： `Device(config)# ip dhcp snooping database timeout 300`	データベース転送プロセスが完了するのを待ち、それまでに完了しない場合はプロセスを停止する時間（秒数）を指定します。 デフォルトは300秒です。指定できる範囲は0～86400です。無期限の期間を定義するには、0を使用します。これは転送を無期限に試行することを意味します。
ステップ 5	**ip dhcp snooping database write-delay** *seconds* 例： `Device(config)# ip dhcp snooping database write-delay 15`	バインディング データベースが変更されてから転送を開始するまでの遅延時間を指定します。指定できる範囲は 15 ～ 86400 秒です。デフォルトは 300 秒（5 分）です。
ステップ 6	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds* 例： `Device# ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gigabitethernet 1/1/0 expiry 1000`	（任意）DHCP スヌーピング バインディング データベースにバインディング エントリを追加します。*vlan-id* に指定できる範囲は 1 ～ 4904 です。*seconds* の範囲は 1 ～ 4294967295 です。 このコマンドは、追加するエントリごとに入力します。 このコマンドは、スイッチをテストまたはデバッグするときに使用します。
ステップ 8	**show ip dhcp snooping database** [**detail**] 例： `Device# show ip dhcp snooping database detail`	DHCP スヌーピング バインディング データベース エージェントのステータスおよび統計情報を表示します。

DHCP スヌーピング情報のモニタリング

表 *2: DHCP* 情報を表示するためのコマンド

show ip dhcp snooping	スイッチの DHCP スヌーピングの設定を表示します。
show ip dhcp snooping binding	DHCP スヌーピング バインディング データベース内の動的に設定さディングだけを表示します。このようなバインディングは、バインブルとも呼ばれます。
show ip dhcp snooping database	DHCP スヌーピング バインディング データベースのステータスおよを表示します。
show ip dhcp snooping statistics	DHCP スヌーピングの統計情報を要約または詳細形式で表示します
show ip source binding	動的および静的に設定されたバインディングを表示します。

（注） DHCP スヌーピングがイネーブルでインターフェイスがダウン ステートに変更された場合、静的に設定されたバインディングは削除されません。

DHCP サーバ ポートベースのアドレス割り当てのイネーブル化

ポートベースのアドレス割り当てをグローバルにイネーブル化し、インターフェイス上で加入者 ID を自動的に生成するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip dhcp use subscriber-id client-id** 例： `Device(config)# ip dhcp use subscriber-id client-id`	すべての着信 DHCP メッセージで、加入者 ID がクライアント ID としてグローバルに使用されるように DHCP サーバを設定します。
ステップ 4	**ip dhcp subscriber-id interface-name** 例： `Device(config)# ip dhcp subscriber-id interface-name`	インターフェイスの短い名前に基づいて、加入者 ID を自動的に生成します。 特定のインターフェイスで設定された加入者 ID は、このコマンドで優先されます。
ステップ 5	**interface** *interface-type interface-number* 例： `Device(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、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	**ip dhcp server use subscriber-id client-id** 例： `Device(config-if)# ip dhcp server use subscriber-id client-id`	インターフェイス上ですべての着信 DHCP メッセージで、加入者 ID がクライアント ID として使用されるように DHCP サーバを設定します。
ステップ 7	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

次のタスク

スイッチ上での DHCP ポートベースのアドレス割り当てをイネーブルにした後で、**ip dhcp pool** グローバル コンフィギュレーション コマンドを使用して、IP アドレスの事前割り当てと、クライアントへの関連付けを行います。

DHCP サーバ ポートベースのアドレス割り当てのモニタリング

表 *3 : DHCP* ポートベースのアドレス割り当て情報を表示するためのコマンド

コマンド	目的
show interface *interface id*	特定のインターフェイスのステータスおよび設定を表示します。
show ip dhcp pool	DHCP アドレス プールを表示します。
show ip dhcp binding	Cisco IOS DHCP サーバのアドレス バインディングを表示します。

DHCP の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

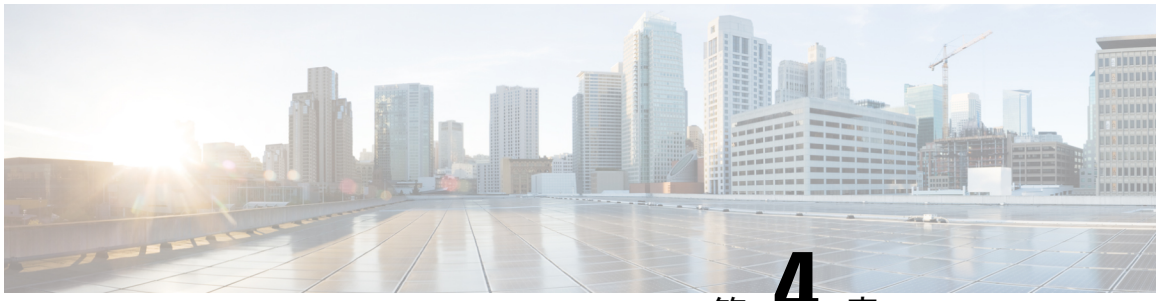
これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

表 *4 :* 新しい機能の履歴

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	DHCP	DHCP はインターネット ホストに設定パラメータを提供します。DHCP は 2 つのコンポーネントで構成されます。1 つはホスト固有の設定パラメータを DHCP サーバーからホストに配信するためのプロトコルで、もう 1 つはホストにネットワーク アドレスを割り当てるためのメカニズムです。DHCP はクライアント/サーバー モデルに基づいています。指定された DHCP サーバー ホストが、ダイナミックに設定されるホストに対して、ネットワーク アドレスを割り当て、設定パラメータを提供します。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	DHCP クライアントオプション 12	DHCP クライアントオプション 12 機能により、クライアントのホスト名が指定されます。Dynamic Host Configuration Protocol（DHCP）サーバーからインターフェイスの IP アドレスを取得する際に、クライアントデバイスが応答内の DHCP Hostname オプションを受信すると、このオプションのホスト名が設定されます。DHCP は、IP ネットワークにおける動作のための設定情報を取得するために DHCP クライアントによって使用されます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 4 章

DHCP オプションのサポート

DHCP オプションサポートに関する制約事項

プライマリ VLAN に対して DHCP スヌーピングが設定されている場合は、いずれのセカンダリ VLAN に対しても、異なる設定を持つスヌーピングを設定できません。関連付けられているすべての VLAN 用の DHCP スヌーピングをプライマリ VLAN に対して設定する必要があります。プライマリ VLAN に対して DHCP スヌーピングが設定されていないときに、セカンダリ VLAN、たとえば VLAN 200 に対して設定しようとすると、次のメッセージが表示されます。

```
2w5d:%DHCP_SNOOPING-4-DHCP_SNOOPING_PVLAN_WARNING:DHCP Snooping configuration may not
take effect
on secondary vlan 200. DHCP Snooping configuration on secondary vlan is derived from its
 primary vlan.
```

show ip dhcp snooping コマンドを使用すると、プライマリかセカンダリかを問わず、DHCP スヌーピングが有効にされているすべての VLAN が表示されます。

DHCP オプションのサポートに関する情報

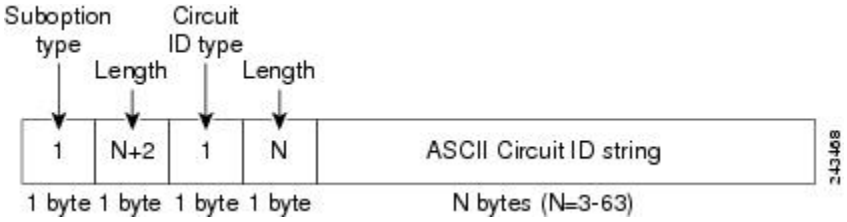
DHCP Option 82 の設定が可能な回線 ID およびリモート ID

DHCP Option 82 設定可能な回線 ID およびリモート ID 機能では、Option 82 リモート ID サブオプションおよび Option 82 回線 ID サブオプションで提供する情報を指定できるため、検証セキュリティが強化されます。

DHCP スヌーピングはプライベート VLAN 上でイネーブルにできます。DHCP スヌーピングがイネーブルの場合、設定はプライマリ VLAN およびそれに関連付けられているセカンダリ VLAN の両方に伝播します。プライマリ VLAN で DHCP スヌーピングがイネーブルの場合は、セカンダリ VLAN でもイネーブルにされます。

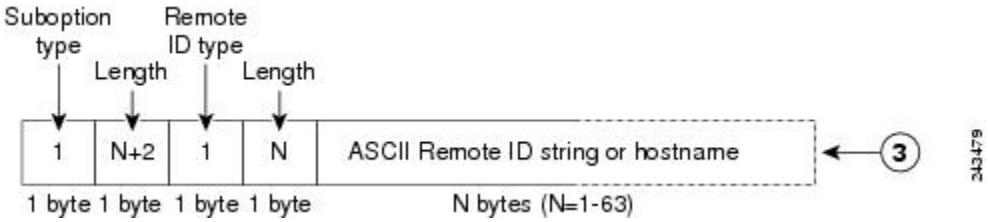
次の図に、DHCP スヌーピングがグローバルに有効になっており、回線 ID サブオプションを指定して **ip dhcp snooping information option** グローバル コンフィギュレーション コマンドを入力した場合に使用されるパケットフォーマットを示します。

図 *6:* 回線 *ID* を指定した場合のサブオプション パケット フォーマット



次の図に、DHCP スヌーピングがグローバルに有効になっており、リモート ID サブオプションを指定して **ip dhcp snooping information option** グローバル コンフィギュレーション コマンドを入力した場合に使用されるパケットフォーマットを示します。

図 *7:* リモート *ID* を指定した場合のサブオプション パケット フォーマット



DHCP クライアントオプション 12

DHCP クライアントオプション 12 機能により、クライアントのホスト名が指定されます。Dynamic Host Configuration Protocol（DHCP）サーバーからインターフェイスの IP アドレスを取得する際に、クライアントデバイスが応答内の DHCP Hostname オプションを受信すると、このオプションのホスト名が設定されます。DHCP は、IP ネットワークにおける動作のための設定情報を取得するために DHCP クライアントによって使用されます。

設定パラメータやその他の制御情報は、DHCP メッセージのオプションフィールドに格納されたタグ付きデータ項目で伝送されます。DHCP クライアントに対してオプション 12 を設定できるため、DHCP クライアントには柔軟性があります。

オプション 12 により、クライアントの名前が指定されます。この名前は、ローカルドメインで修飾される場合と修飾されない場合があります。

プライベート VLAN に対する DHCP スヌーピングの設定

プライベートのプライマリ VLAN およびセカンダリ VLAN に対して DHCP スヌーピングを設定するには、次の作業を実行してください。

- プライベートのプライマリ VLAN を設定します。
- 独立 VLAN をこのプライマリ VLAN に関連付けます。
- プライマリ VLAN 用の SVI インターフェイスを作成し、適切なループバック IP およびヘルパー アドレスをインターフェイスに関連付けます。
- プライマリ VLAN で DHCP スヌーピングをイネーブルにします。その結果、関連付けられている VLAN でも DHCP スヌーピングがイネーブルになります。

（注） スヌーピングに実効性を持たせるには、IP アドレス、DHCP プール、およびリレー ルートを割り当てるサーバーを設定する必要もあります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan** *vlan-id* 例： `Device(config)# vlan 70`	指定したプライベート VLAN の VLAN 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	**private-vlan primary** 例： `Device(config-vlan)# private-vlan primary`	VLAN をプライマリ PVLAN として指定します。

	コマンドまたはアクション	目的
ステップ 5	**private-vlan association** *secondary-vlan-list* 例： `Device(config-vlan)# private-vlan association 7`	プライベート VLAN（PVLAN）の設定および PVLAN とセカンダリ VLAN とのアソシエーションの設定を行います。
ステップ 6	**exit** 例： `Device(ocnfig-vlan)# exit`	VLAN コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**vlan** *vlan_ID* 例： `Device(config)# vlan 7`	指定したプライベート VLAN の VLAN コンフィギュレーション モードを開始します。 • この例では、関連付けられるセカンダリ VLAN は vlan 7 です。
ステップ 8	**private-vlan isolated** 例： `Device(config-vlan)# private-vlan isolated`	この VLAN を独立プライベート VLAN として指定します。
ステップ 9	**exit** 例： `Device(config-vlan)# exit`	VLAN コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface vlan** *primary-vlan_id* 例： `Device(config)# interface vlan 70`	プライマリ VLAN でダイナミック スイッチ仮想インターフェイス（SVI）を作成して、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 11	**ip unnumbered loopback** 例： `Device(config-if)# ip unnumbered loopback1`	IP アンナンバード ループバックを指定します。
ステップ 12	**private-vlan mapping** [*secondary-vlan-list* | **add** *secondary-vlan-list*| **remove** *secondary-vlan-list*] 例：	プライマリ VLAN とセカンダリ VLAN のマッピングを作成して、それらに同じプライマリ VLAN SVI を共有させます。

	コマンドまたはアクション	目的
	`Device(config-if)# private-vlan mapping 7`	
ステップ 13	**exit** 例： `Device(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、グローバル コンフィギュレーション モードに戻ります。
ステップ 14	**ip dhcp snooping vlan** *primary-vlan_id* 例： `Device(config)# ip dhcp snooping vlan 70`	プライマリ VLAN および関連付けられた VLAN で DHCP スヌーピングをイネーブルにします。
ステップ 15	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例：プライベート VLAN 関連付けのマッピング

次のインターフェイス コンフィギュレーションの例は、プライベート VLAN アソシエーションのマッピング方法を示します。ユーザー設定可能な回線 ID「aabb11」がセカンダリ VLAN である vlan 7 に挿入されます。

```
Device> enable
Device# configure terminal
Device(config-if)# interface GigabitEthernet 9/0/1
Device(config-if)# switchport
Device(config-if)# switchport private-vlan host-association 70 7
Device(config-if)# switchport mode private-vlan host
Device(config-if)# no mls qos trust
Device(config-if)# spanning-tree portfast
Device(config-if)# exit
Device(config)# ip dhcp snooping vlan 7 information option format-type circuit-id string
 aabb11
Device(config)# end
```

次の例は、DHCP クラス「C1」を定義し、このインターフェイス コンフィギュレーションの例で入力された回線 ID 値と一致する 16 進文字列を使用して、サーバーで対応するクラスの 16 進文字列を指定する方法を示しています。つまり、16 進文字列 00000000000000000000000000000000616162623131 マスク ffffffffffffffffffffffffffffffff00000000000000 は、回線 ID aabb11 と一致します。

```
Device> enable
Device# configure terminal
Device(config)# ip dhcp class C1
Device(config-dhcp-class)# relay agent information
Device(config-dhcp-class-relayinfo)# relay-information hex
```

```
00000000000000000000000000000066616162623131
mask ffffffffffffffffffffffffffffffff0000000000000
Device(config-dhcp-class-relayinfo)# end
```

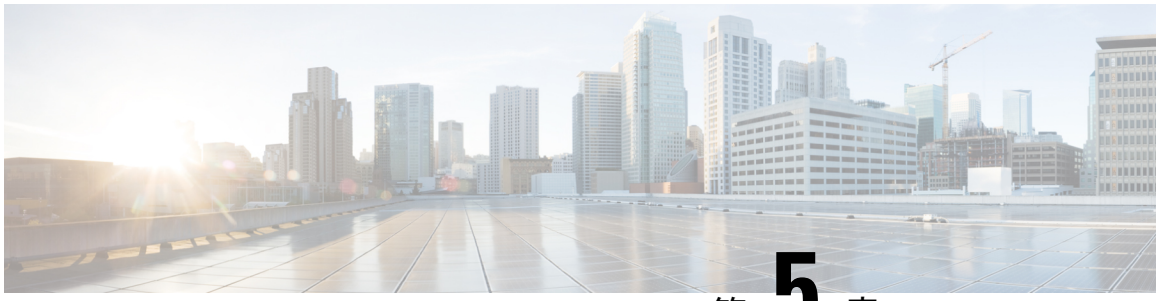
DHCP オプションサポートの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	DHCP クライアントオプション 12	DHCP クライアントオプション 12 機能により、クライアントのホスト名が指定されます。Dynamic Host Configuration Protocol（DHCP）サーバーからインターフェイスの IP アドレスを取得する際に、クライアントデバイスが応答内の DHCP Hostname オプションを受信すると、このオプションのホスト名が設定されます。DHCP は、IP ネットワークにおける動作のための設定情報を取得するために DHCP クライアントによって使用されます。
	DHCP Option 82 設定可能な回線 ID およびリモート ID	Option 82 リモート ID サブオプションおよび Option 82 回線 ID サブオプションでの命名の選択肢を規定します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 5 章

DHCPv6 オプションのサポート

DHCPv6 オプションのサポートに関する情報

CAPWAP アクセスコントローラ DHCPv6 オプション

Control And Provisioning of Wireless Access Points（CAPWAP）プロトコルでは、中央管理型アクセスポイントが接続可能なワイヤレスコントローラをDHCPを使用して検出できます。CAPWAPは標準の相互運用プロトコルであり、コントローラによるワイヤレスアクセスポイントの集合の管理を可能にします。

ワイヤレスアクセスポイントは、プライマリ、セカンダリ、およびターシャリ ワイヤレス コントローラの IPv6 管理インターフェイスアドレスを提供する DHCPv6 オプション 52（RFC 5417）を使用します。

ステートレスとステートフル両方のDHCPv6アドレッシングモードがサポートされています。ステートレスモードでは、アクセスポイントがステートレスアドレス自動設定（SLAAC）を使用して IPv6 アドレスを取得する一方で、（ルータアドバタイズメントから取得されない）その他のネットワーク情報はDHCPv6サーバーから取得されます。ステートフルモードでは、アクセスポイントが IPv6 アドレスと他のネットワーク情報の両方を DHCPv6 サーバーのみから取得します。どちらのモードでも、DHCPv6を使用してワイヤレスコントローラを検出する必要がある場合、オプション 52 を可能にするには DHCPv6 サーバーが必要です。

MAX_PACKET_SIZE が 15 を超えており、オプション 52 が設定されている場合、DHCPv6 サーバーは DHCP パケットを送信しません。

DNS 検索リストのオプション

DNS 検索リスト（DNSSL）は、ドメインネームシステム（DNS）サフィックスドメイン名のリストであり、IPv6 ホストで短い、修飾子を持たないドメイン名に対する DNS クエリ検索を実行する際に使用されます。DNSSL オプションには、1 つ以上のドメイン名が含まれます。すべてのドメイン名が同じライフタイム値を共有します。ライフタイム値とは、DNSSL を使用できる最大時間を秒単位で示したものです。異なるライフタイム値が必要な場合は、複数の DNSSL オプションを使用できます。最大 5 つの DNSSL を設定できます。

長い DNSSL 名を持つ DHCP メッセージは、デバイスによって破棄されます。

（注） 複数のルータアドバタイズメント（RA）や DHCP から DNS 情報を入手できる場合、ホストはこの DNS 情報の順序付きリストを保持する必要があります。

RFC 6106 は、拡張 DNS 設定のため、IPv6 ルータが IPv6 ホストに DNS 検索リスト（DNSSL）をアドバタイズできるようする IPv6 ルータアドバタイズメント（RA）オプションを指定しています。

DNS ライフタイムの範囲は、次の例に示すように、最大 RA 間隔の値と最大 RA 間隔を 2 倍にした値の間に設定する必要があります。

```
(max ra interval) <= dns lifetime <= (2*(max ra interval))
```

最大 RA 間隔の値は 4 ー 1800 秒の間で指定できます（デフォルトは 240 秒）。次の例は、範囲外のライフタイムを示しています。

```
Device(config-if)# ipv6 nd ra dns-search-list sss.com 3600
! Lifetime configured out of range for the interface that has the  default maximum RA
interval.!
```

DHCPv6 クライアントのリンク層アドレスオプション

DHCPv6 クライアントのリンク層アドレスオプション（RFC 6939）は、ファーストホップ DHCPv6 リレーエージェント（クライアントと同じリンクに接続されたリレーエージェント）がサーバーに送信されている DHCPv6 メッセージでクライアントのリンク層アドレスを提供できるようにするための、オプションのメカニズムと関連 DHCPv6 オプションを定義します。

クライアントのリンク層アドレスオプションは、リレーエージェントとサーバー間でのみ交換されます。DHCPv6 クライアントは、クライアントのリンク層アドレスオプションの使用を認識しません。DHCPv6 クライアントは、クライアントのリンク層アドレスオプションを送信してはならず、クライアントのリンク層アドレスオプションを無視する必要があります。

各 DHCPv6 クライアントとサーバーは、DHCP 固有識別子（DUID）によって識別されます。DUID は、クライアント識別子およびサーバー識別子オプションで伝送されます。DUID はすべての DHCP クライアントとサーバーで一意であり、特定のクライアントまたはサーバーに固

定されます。DHCPv6 では、クライアントとサーバーの両方の識別子にリンク層アドレスに基づく DUID を使用します。デバイスは、最も小さい番号のインターフェイスの MAC アドレスを使用して DUID を形成します。ネットワーク インターフェイスは、デバイスに永続的に接続されていると見なされます。

DHCP リレー エージェント

DHCP リレー エージェントは、クライアントとサーバの間で DHCP パケットを転送するレイヤ 3 デバイスです。リレー エージェントは、同じ物理サブネット上にないクライアントとサーバの間で要求および応答を転送します。リレー エージェントによる転送は、IP データグラムをネットワーク間で透過的に交換するレイヤ 2 での通常の転送とは異なります。リレー エージェントは、DHCP メッセージを受け取ると、新しい DHCP メッセージを生成して、出力インターフェイス上で送信します。

DHCPv6 オプションサポートの設定方法

このセクションでは、DHCPv6 オプションサポートを設定する方法について説明します。

CAPWAP アクセスポイントの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 dhcp pool** *poolname* 例： `Device(config)# ipv6 dhcp pool pool1`	DHCPv6 サーバー設定情報プールを設定し、DHCPv6 プール コンフィギュレーション モードを開始します。
ステップ 4	**capwap-ac address** *ipv6-address* 例： `Device(config-dhcpv6)# capwap-ac address 2001:DB8::1`	CAPWAP アクセス コントローラ アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Device(config-dhcpv6)# end`	DHCPv6 プール コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

IPv6 ルータ アドバタイズメント オプションを使用した DNS 検索リストの設定

IPv6 ルータ アドバタイズメント オプションを使用して DNS 検索リストを設定するには、次のタスクを実行します。

（注） ドメイン名の設定は、RFC 1035 に従って行う必要があります。そうでない場合、設定が拒否されます。たとえば、次のドメイン名の設定はエラーになります。

```
Device(config-if)# ipv6 nd ra dns-search-list domain example.example.com infinite-lifetime
```

（注） **ipv6 nd ra dns-search-list domain** コマンドは、レイヤ 3 モードでルーテッドポートとして設定されている物理インターフェイスのみで設定できます。この設定は、インターフェイス コンフィギュレーション モードで **no switchport** コマンドを使用することにより実行できます。

インターフェイスで単一の DNS 検索リストを削除するには、インターフェイス コンフィギュレーション モードで **no ipv6 nd ra dns-search-list domain** *domain-name* コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-type interface-number* 例： `Device(config)# interface GigabitEthernet 0/2/0`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**no switchport** 例： `Device(config-if)# no switchport`	物理ポートに限り、レイヤ3モードを開始します。
ステップ 5	**ipv6 nd prefix** *ipv6-prefix/prefix-length* 例： `Device(config-if)# ipv6 nd prefix 2001:DB8::1/64 1111 222`	IPv6 ネイバー探索（ND）ルータアドバタイズメントに含める IPv6 プレフィックスを設定します。
ステップ 6	**ipv6 nd ra lifetime** *seconds* 例： `Device(config-if)# ipv6 nd ra lifetime 9000`	インターフェイス上の IPv6 ルータアドバタイズメントに含まれるデバイスのライフタイム値を設定します。
ステップ 7	**ipv6 nd ra dns-search-list domain** *domain-name* [**lifetime** [*lifetime-value* | **infinite**]] 例： `Device(config-if)# ipv6 nd ra dns-search-list domain example.example.com lifetime infinite`	DNS 検索リストを設定します。検索リストのライフタイムを指定できます。 （注） Cisco IOS XE Giraltar 16.12.1 よりも前のリリースの場合、このコマンドは **ipv6 nd ra dns search list** *list-name***infinite-lifetime** として存在します。
ステップ 8	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

例：CAPWAP アクセスポイントの設定

次に、CAPWAP アクセスポイントの設定方法の例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ipv6 dhcp pool pool1
Device(config-dhcpv6)# capwap-ac address 2001:DB8::1
Device(config-dhcpv6)# end
Device#
```

DHCPv6 オプションサポートの確認

オプション 52 サポートの確認

次に、**show ipv6 dhcp pool** コマンドの出力例として DHCPv6 設定プールの情報を表示します。

```
Device# show ipv6 dhcp pool

DHCPv6 pool: svr-p1
  Static bindings:
    Binding for client 000300010002FCA5C01C
      IA PD: IA ID 00040002,
        Prefix: 2001:db8::3/72
                preferred lifetime 604800, valid lifetime 2592000
      IA PD: IA ID not specified; being used by 00040001
        Prefix: 2001:db8::1/72
                preferred lifetime 240, valid lifetime 54321
        Prefix: 2001:db8::2/72
                preferred lifetime 300, valid lifetime 54333
        Prefix: 2001:db8::3/72
                preferred lifetime 280, valid lifetime 51111
  Prefix from pool: local-p1, Valid lifetime 12345, Preferred lifetime 180
  DNS server: 1001::1
  DNS server: 1001::2
  CAPWAP-AC Controller address: 2001:DB8::1
  Domain name: example1.com
  Domain name: example2.com
  Domain name: example3.com
 Active clients: 2
```

次に、DHCPv6 のデバッグを有効にする例を示します。

```
Device# debug ipv6 dhcp detail

IPv6 DHCP debugging is on (detailed)
```

DHCPv6 オプションのサポートに関する追加情報

標準および RFC

標準/RFC	Title
RFC 6106	DNS 設定の IPv6 ルータ アドバタイズメント オプション
RFC 54171	Control And Provisioning of Wireless Access Points（CAPWAP）アクセスコントローラ DHCP オプション
RFC 6939	DHCPv6 のクライアントリンク層アドレスオプション

DHCPv6 オプションサポートの機能履歴

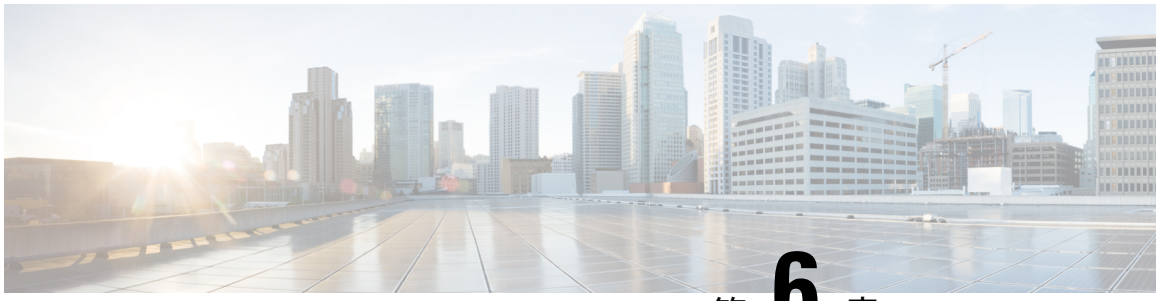
次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	CAPWAP アクセスコントローラ DHCPv6 オプション 52	CAPWAP プロトコルでは、中央管理型アクセスポイントの接続先ワイヤレスコントローラを DHCPv6 を使用して検出できます。CAPWAP は標準の相互運用プロトコルであり、コントローラによるワイヤレスアクセスポイントの集合の管理を可能にします。
	DHCPv6 クライアントのリンク層アドレスオプション	DHCPv6 クライアントのリンク層アドレスオプション（RFC6939）は、ファーストホップ DHCPv6 リレーエージェント（クライアントと同じリンクに接続されたリレーエージェント）がサーバーに送信されている DHCPv6 メッセージでクライアントのリンク層アドレスを提供できるようにするための、オプションのメカニズムと関連DHCPv6 オプションを定義します。
	DNS 検索リスト	DNS 検索リスト（DNSSL）は、ドメインネームシステム（DNS）サフィックスドメイン名のリストであり、IPv6ホストで短い、修飾子を持たないドメイン名に対する DNS クエリ検索を実行する際に使用されます。DNSSL オプションには、1 つ以上のドメイン名が含まれます。

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	DHCPv6 リレーチェーニングおよびルート挿入	DHCPv6 リレーチェーニングおよびルート挿入機能により、DHCPv6 メッセージを複数のリレーエージェントでリレーできます。
	DHCPv6 クライアントのリンク層アドレスオプション：コマンド変更	ipv6 nd ra dns search list コマンドの構文が ipv6 nd ra dns-search-list domain に変更されました。show ipv6 nd ra dns-search-list コマンドが導入されました。
	RFC 6106 および RFC 5417 の IPv6 サポート	IPv6 のサポートは、DNS 設定の IPv6 ルータ アドバタイズメント オプション（RFC 6106）、および Control And Provisioning of Wireless Access Points（CAPWAP）アクセスコントローラ DHCP オプション（RFC 5417）で導入されました。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 6 章

DHCPv6 リレー ソース設定

DHCPv6 リレー送信元の設定の制限事項

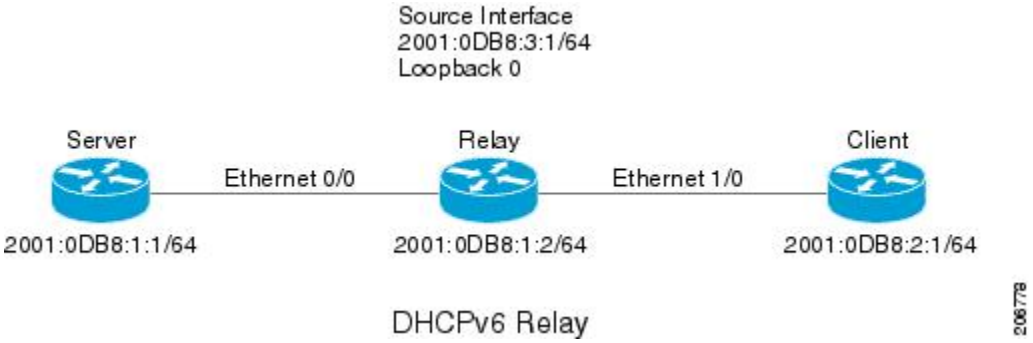
- 設定済みのインターフェイスがシャットダウンされた場合、またはその IPv6 アドレスのすべてが削除された場合、リレーは標準の動作に戻ります。
- IPv6 アドレスが設定されていないインターフェイスを指定しようとすると、コマンドライン インターフェイス（CLI）によってエラーが報告されます。
- インターフェイス コンフィギュレーションとグローバル コンフィギュレーションの両方が設定されている場合、インターフェイス コンフィギュレーションが優先されます。

DHCPv6 リレー送信元の設定に関する情報

DHCPv6 サーバーは、応答を中継されたメッセージの送信元アドレスに送信します。通常、DHCPv6 リレーは、メッセージ送信に使用されたサーバー方向インターフェイスのアドレスを送信元として使用します。ただし、一部のネットワークでは、より安定したアドレス（ループバック インターフェイスなど）を設定し、そのインターフェイスを中継されたメッセージの送信元アドレスとしてリレーで使用することが望ましい場合があります。DHCPv6 リレー送信元設定機能には、この機能が用意されています。

次の図に、単一のクライアント、リレー、およびサーバーで構成される簡単なネットワークを示します。リレーとサーバーは 2001:DB8:1::/64 を介して通信し、リレーには 2001:DB8:2::/64 に対するクライアント方向インターフェイスがあります。リレーには、アドレス 2001:DB8:3:1/64 が設定されたループバック インターフェイスもあります。

図 *8 : DHCPv6* リレー送信元設定 - 簡単なネットワーク



リレーはクライアントから要求を受信すると、クライアント方向インターフェイス（イーサネット 1/0）のアドレスを relay-forward メッセージの link-address フィールドに含めます。このアドレスは、サーバーによってアドレス プールの選択に使用されます。その後、リレーは relay-forward メッセージをサーバーに送信します。デフォルトでは、サーバー方向（イーサネット 0/0）インターフェイスのアドレスが IPv6 送信元として使用され、サーバーはそのアドレスに応答を送信します。

リレーの送信元インターフェイスが明示的に設定されている場合、リレーはそのインターフェイスのプライマリ IPv6 アドレスを、転送するメッセージの IPv6 送信元として使用します。たとえば、ループバック 0 を送信元として設定すると、リレーは、サーバーに中継されるメッセージの IPv6 送信元アドレスとして 2001:DB8:3:1/64 を使用します。

DHCPv6 リレー送信元の設定方法

DHCPv6 リレー送信元の設定

DHCPv6 リレー送信元を設定するには、次の作業を実行します。

インターフェイスに対する DHCPv6 リレー送信元の設定

メッセージの中継時に送信元として使用するインターフェイスを設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface loopback 0`	インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 dhcp relay source-interface** *interface-type interface-number* 例： `Device(config-if)# ipv6 dhcp relay source-interface loopback 0`	このインターフェイスで受信したメッセージの中継時に送信元として使用するインターフェイスを設定します。
ステップ 5	**end** 例： `Device(config-if)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

DHCPv6 リレー送信元のグローバルな設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 dhcp-relay source-interface** *interface-type interface-number* 例： `Device(config)# ipv6 dhcp-relay source-interface loopback 0`	メッセージの中継時に送信元として使用するインターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

DHCPv6 リレー送信元の設定例

例：インターフェイスに対する DHCPv6 リレー送信元の設定

次の例で、リレーの送信元として使用するループバック 0 インターフェイスの設定方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface loopback 0
Device(config-if)# ipv6 dhcp relay source-interface loopback 0
Device(config-if)# end
```

DHCPv6 リレー送信元の設定に関する追加情報

標準および RFC

標準/RFC	タイトル
IPv6 に関する RFC	*IPv6 RFCs*

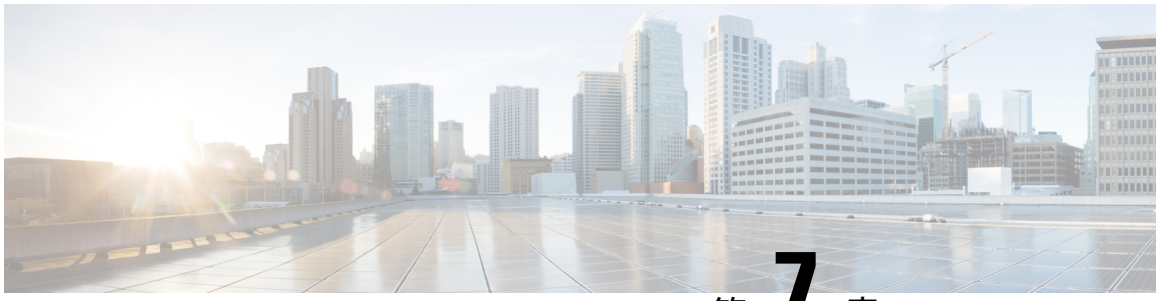
DHCPv6 リレー送信元の設定に関する機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	DHCPv6 リレー ソース設定	DHCPv6 を使用する一部のネットワークでは、より安定したアドレス（ループバックインターフェイスなど）を設定し、そのインターフェイスを中継されたメッセージの送信元アドレスとしてリレーで使用することが望ましい場合があります。DHCPv6 リレー送信元設定機能には、この機能が用意されています。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 7 章

IPv4 GRE トンネルを介した IPv6 の設定

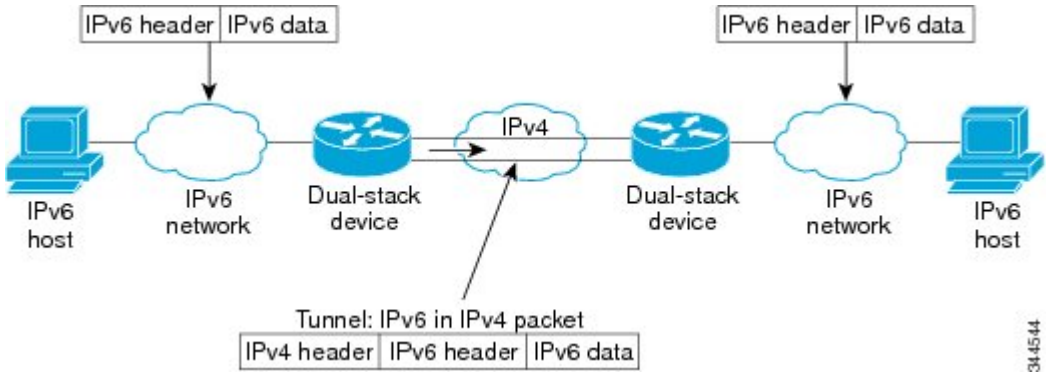
IPv4 GRE トンネルを介した IPv6 の設定に関する情報

続くセクションでは、IPv4 GRE トンネルを介した IPv6 の設定について説明します。

Ipv6 用オーバーレイ トンネル

オーバーレイ トンネリングでは、IPv4 パケット内で IPv6 パケットをカプセル化して、IPv4 インフラストラクチャ（コア ネットワークまたは以下の図）へ伝送します。オーバーレイ トンネルを使用することで、孤立した IPv6 ネットワークと通信できます。このとき、孤立した複数の IPv6 ネットワーク間にある IPv4 インフラストラクチャをアップグレードする必要はありません。オーバーレイ トンネルは、境界デバイス間、または境界デバイスとホスト間に設定できますが、両方のエンドポイントが IPv4 プロトコル スタックと IPv6 プロトコル スタックの両方をサポートしている必要があります。

図 *9*：オーバーレイ トンネル



（注） オーバーレイ トンネルにより、インターフェイスの最大伝送単位（MTU）が 20 オクテット減少します（ただし、基本 IPv4 パケット ヘッダーにオプション フィールドが含まれていないことを前提とします）。オーバーレイ トンネルを使用するネットワークは、トラブルシューティングが困難です。したがって、独立した IPv6 ネットワークに接続するオーバーレイ トンネルは、最終的な IPv6 ネットワーク アーキテクチャと見なしてはいけません。オーバーレイ トンネルの使用は、IPv4 と IPv6 の両方のプロトコル スタック、または IPv6 プロトコル スタックだけをサポートするネットワークへの移行方法と見なす必要があります。

IPv6 は、GRE タイプのオーバーレイトンネリングをサポートします。IPv4 GRE トンネルを介した IPv6 は、IPv6、Connectionless Network Service（CLNS）など、さまざまなタイプのパケットを伝送できます。

IPv6 トラフィック用の GRE IPv4 トンネル サポート

IPv6 トラフィックは、標準的なポイントツーポイントのカプセル化スキームの実装にサービスを提供するように設計されている標準 GRE トンネリング技術を使用して、IPv4 GRE トンネルを介して伝送できます。GRE トンネルは、手動で設定された IPv6 トンネルと同様、リンクごとに個別のトンネルが設定された 2 つのポイント間のリンクです。これらのトンネルは、特定のパッセンジャまたはトランスポート プロトコルに結合されていませんが、この場合、GRE を使用するパッセンジャ プロトコルとして IPv6 を伝送し、トランスポート プロトコルとして IPv4 または IPv6 を伝送します。

GRE トンネルは、2 つのエッジ デバイス間またはエッジ デバイスとエンド システム間に定期的でセキュアな通信を必要とする安定した接続のために主に使用されます。エッジデバイスとエンド システムは、デュアル スタック実装である必要があります。

GRE IPv6 トンネルの設定

IPv6 ネットワーク上で GRE トンネルを設定するには、次の作業を実行します。GRE トンネルは、IPv6 ネットワーク層上で実行し、IPv6 トンネルの IPv6 パケットおよび IPv6 トンネルの IPv4 パケットを転送するように設定できます。

GRE IPv6トンネルを設定するには、次の手順を実行します。

始める前に

GRE IPv6 トンネルが設定されている場合、IPv6 アドレスは、トンネル送信元およびトンネル宛先に割り当てられます。トンネル インターフェイスは、割り当て済みの IPv4 アドレスまたは IPv6 アドレスを持つことができます（ここでは説明していません）。設定されたトンネルの両端にあるホストまたはルータは、IPv4 プロトコル スタックと IPv6 プロトコル スタックの両方をサポートしている必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface tunnel** *tunnel-number* 例： `Device(config)# interface tunnel 0`	トンネル インターフェイスおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 address** *ipv6-prefix* / *prefix-length* [**eui-64**] 例： `Device(config-if)# ipv6 address 3ffe:b00:c18:1::3/127`	インターフェイスに割り当てられている IPv6 ネットワークを指定し、インターフェイスで IPv6 処理をイネーブルにします。
ステップ 5	**tunnel source** {*ip-address* | *ipv6-address* | *interface-type interface-number*} 例： `Device(config-if)# tunnel source ethernet 0`	トンネルインターフェイスの送信元IPv4アドレスまたは送信元インターフェイスタイプと番号を指定します。 • インターフェイスが指定されている場合、そのインターフェイスはIPv4アドレスを使用して設定されている必要があります。
ステップ 6	**tunnel destination** {*host-name* | *ip-address* | *ipv6-address*} 例： `Device(config-if)# tunnel destination 2001:DB8:1111:2222::1/64`	宛先 IPv6 アドレスまたはトンネル インターフェイスのホスト名を指定します。
ステップ 7	**tunnel mode** {**aurp** | **cayman** | **dvmrp** | **eon** | **gre**| **gre multipoint** | **gre ipv6** | **ipip** [**decapsulate-any**] | **iptalk** | **ipv6** | **mpls** | **nos** 例： `Device(config-if)# tunnel mode gre ipv6`	GRE IPv6 トンネルを指定します。 （注） **tunnel mode gre ipv6** コマンドは、トンネルのカプセル化プロトコルとしてGREを指定します。

設定例：IPv6 トンネルのトンネル宛先アドレス

```
Device> enable
Device# configure terminal
Device(config)# interface Tunnel 0
Device(config-if)# ipv6 address 2001:1:1::1/48
Device(config-if)# tunnel source GigabitEthernet 0/0/0
Device(config-if)# tunnel destination 10.0.0.2
Device(config-if)# tunnel mode gre ipv6
Device(config-if)# exit
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.0.0.1 255.255.255.0
Device(config-if)# exit
!
Device(config)# ipv6 unicast-routing
Device(config)# router isis
Device(config-router)# net 49.0000.0000.000a.00
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)*

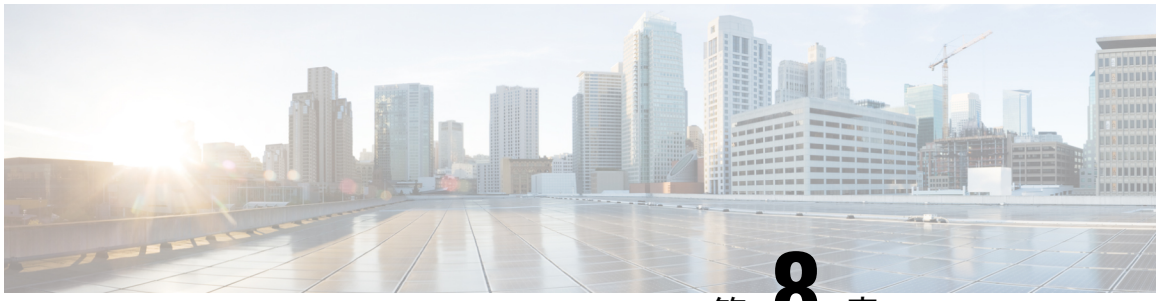
IPv4 GRE トンネルを介した IPv6 の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	IPv4 GRE トンネルを介する IPv6	GRE トンネルは、2 つのポイント間のリンクであり、リンクごとに個別のトンネルがあります。これらのトンネルは、特定のパッセンジャまたはトランスポート プロトコルに結合されていませんが、この場合、GRE を使用するパッセンジャ プロトコルとして IPv6 を伝送し、トランスポート プロトコルとして IPv4 または IPv6 を伝送します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 8 章

GLBP の設定

GLBP の制限事項

拡張オブジェクト トラッキング（EOT）はステートフル スイッチオーバー（SSO）を認識しないため、SSO モードで GLBP と併用することはできません。

GLBP の前提条件

GLBP を設定する前に、デバイスが物理インターフェイス上の複数の MAC アドレスをサポートできることを確認してください。設定している GLBP フォワーダごとに、追加の MAC アドレスが使用されます。

GLBP に関する情報

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 ホストの自動デバイス バックアップを行います。LAN 上の複数のファーストホップ デバイスを連結し、IP パケットの転送負荷を共有しながら単一の仮想ファーストホップ IP デバイスを提供します。LAN 上にあるその他のデバイスは、冗長化された GLBP デバイスとして動作でき

ます。このデバイスは、既存のフォワーディングデバイスが機能しなくなった場合にアクティブになります。

GLBPは、ユーザーに対してはHSRPやVRRPと同様の機能を実行します。HSRPおよびVRRPは、仮想 IP アドレスを指定して設定された仮想デバイス グループに、複数のデバイスを参加させます。グループの仮想 IP アドレスに送信されたパケットを転送するアクティブ デバイスとして、1つのメンバが選択されます。グループ内の他のデバイスは、アクティブデバイスで障害が発生するまでは冗長デバイスです。これらのスタンバイ デバイスには、プロトコルによって使用されていない未使用帯域幅があります。同じデバイスセットに対して複数の仮想デバイス グループを設定できますが、ホストは異なるデフォルト ゲートウェイに対して設定する必要があります。その結果、管理上の負担が大きくなります。GLBPには、単一の仮想IPアドレスと複数の仮想 MAC アドレスを使用して、複数のデバイス（ゲートウェイ）上でのロードバランシングを提供するというメリットがあります。転送負荷は、GLBPグループ内のすべてのデバイス間に分散されるため、単一のデバイスだけが処理して残りのデバイスがアイドルのままになるようなことはありません。各ホストは、同じ仮想 IP アドレスで設定され、仮想デバイス グループ内のすべてのデバイスが参加してパケットの転送を行います。GLBPメンバは、Hello メッセージを使用して相互に通信します。このメッセージは 3 秒ごとにマルチキャスト アドレス 224.0.0.102、UDP ポート 3222（送信元と宛先）に送信されます。

GLBP パケット タイプ

GLBP は実行に 3 つの異なるパケット タイプを使用します。そのパケット タイプは、Hello、要求、および応答です。Hello パケットはプロトコル情報をアドバタイズするために使用されます。Hello パケットはマルチキャストで、仮想ゲートウェイまたはバーチャル フォワーダが Speak、Standby、Active のいずれかの状態のときに送信されます。要求パケットと応答パケットは、仮想 MAC アドレスの割り当てに使用されます。これらはどちらもアクティブ仮想ゲートウェイ（AVG）間のユニキャスト メッセージです。

GLBP アクティブ仮想ゲートウェイ

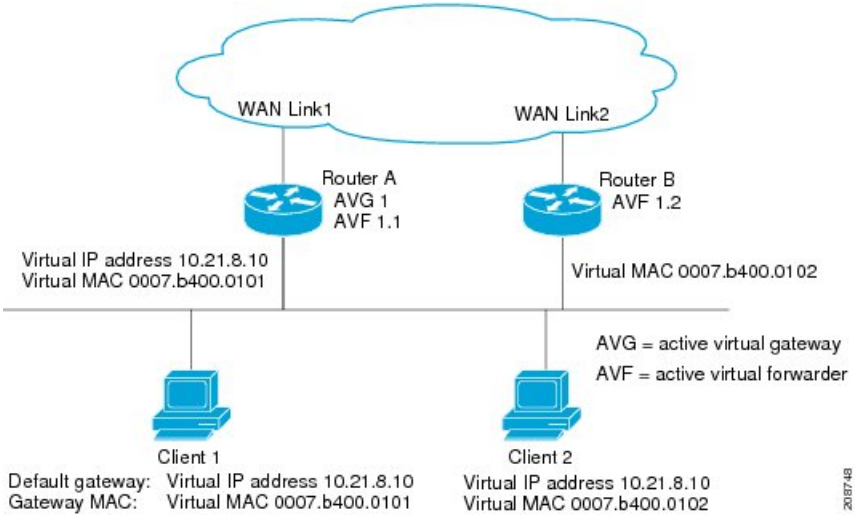
GLBP グループのメンバは、1 つのゲートウェイをそのグループのアクティブ仮想ゲートウェイ（AVG）として選択します。他のグループ メンバは、AVG が使用できなくなった場合のバックアップとなります。AVGはGLBPグループの各メンバに仮想MACアドレスを割り当てます。各ゲートウェイは、AVGによって割り当てられている仮想MACアドレスに送信されたパケットを転送する役割を引き継ぎます。これらのゲートウェイは、仮想 MAC アドレスのアクティブ仮想フォワーダ（AVF）と呼ばれます。

AVG は、仮想 IP アドレスのアドレス解決プロトコル（ARP）要求への応答も行います。ロード シェアリングは、AVG が異なる仮想 MAC で ARP 要求に応答することによって行われます。

no glbp load-balancing コマンドが設定されているときに、AVG が AVF を備えていない場合、先頭の仮想フォワーダ（VF）の MAC アドレスで ARP 要求に応答します。そのため、その VF が現在の AVG に戻るまでは、トラフィックが別のゲートウェイ経由でルーティングされる可能性があります。

下の図では、ルータ A（またはデバイス A）は GLBP グループの AVG で、仮想 IP アドレス 10.21.8.10 に関する処理を行います。ルータ A は、仮想 MAC アドレス 0007.b400.0101 の AVF でもあります。ルータ B（またはデバイス B）は同じ GLBP グループのメンバであり、仮想 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 のデフォルト ゲートウェイ IP アドレスは 10.21.8.10、ゲートウェイ MAC アドレスは 0007.b400.0101 です。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ため、ゲートウェイ MAC アドレス 0007.b400.0102 が与えられます。

図 *10 : GLBP* トポロジ



ルータ A が使用できなくなった場合でも、クライアント 1 は WAN にアクセスできます。これは、ルータ B がルータ A の仮想 MAC アドレスに送信されたパケットの転送を引き継ぎ、ルータ B 自身の仮想 MAC アドレスに送信されたパケットに応答するからです。ルータ B は、GLBP グループ全体の AVG の役割も引き継ぎます。GLBP グループ内のデバイスで障害が発生しても、GLBP メンバの通信は継続されます。

GLBP 仮想 MAC アドレスの割り当て

GLBP グループごとに最大 4 つの仮想 MAC アドレスを設定できます。AVG は、仮想 MAC アドレスをグループの各メンバに割り当てます。他のグループ メンバは、hello メッセージを通じて AVG を検出したあとで仮想 MAC アドレスを要求します。ゲートウェイには、シーケンスにおける次の MAC アドレスが割り当てられます。AVG によって仮想 MAC アドレスが割り当てられた仮想フォワーダは、プライマリ仮想フォワーダと呼ばれます。GLBP グループの他のメンバは、hello メッセージから仮想 MAC アドレスを学習します。仮想 MAC アドレスを学習した仮想フォワーダは、セカンダリ仮想フォワーダと呼ばれます。

GLBP 仮想ゲートウェイの冗長性

GLBP では、HSRP と同じ方法で仮想ゲートウェイの冗長性が実現されます。1 つのゲートウェイが AVG として選択され、もう 1 つのゲートウェイがスタンバイ仮想ゲートウェイとして選択されます。残りのゲートウェイはリッスン状態になります。

AVG の機能が停止すると、スタンバイ仮想ゲートウェイが該当する仮想 IP アドレスの処理を担当します。その後、リッスン状態のゲートウェイから新しいスタンバイ仮想ゲートウェイが選択されます。

GLBP 仮想フォワーダの冗長性

仮想フォワーダの冗長化は、AVF で使用する仮想ゲートウェイの冗長化に類似しています。AVF で障害が発生すると、リッスン状態のセカンダリ仮想フォワーダの 1 つが仮想 MAC アドレスの役割を引き継ぎます。

新しい AVF は、別のフォワーダ番号のプライマリ仮想フォワーダでもあります。GLBP は、ゲートウェイがアクティブ仮想フォワーダ状態に変わるとすぐに始動する 2 つのタイマーを使用して、古いフォワーダ番号からホストを移行します。GLBP は hello メッセージを使用してタイマーの現在の状態を通信します。

リダイレクト時間は、AVG がホストを古い仮想フォワーダ MAC アドレスにリダイレクトし続ける時間です。リダイレクト時間が経過すると、仮想フォワーダが、古い仮想フォワーダ MAC アドレスに送信されたパケットを転送し続けても、AVG は、ARP 応答で古い仮想フォワーダ MAC アドレスの使用を停止します。

仮想フォワーダが有効である時間は、セカンダリ ホールド時間になります。セカンダリ ホールド時間が経過すると、GLBP グループのすべてのゲートウェイから仮想フォワーダが削除されます。期限切れになった仮想フォワーダ番号は、AVG による再割り当てが可能になります。

GLBP ゲートウェイのプライオリティ

各 GLBP ゲートウェイが果たすロールと、AVG の機能が停止したときにどのようなことが発生するかについては、GLBP ゲートウェイ プライオリティによって決まります。

また、GLBP デバイスがバックアップ仮想ゲートウェイとして機能するかどうか、および現在の AVG で障害が発生した場合に AVG になる順番も決まります。各バックアップ仮想ゲートウェイの優先順位には、**glbp priority** コマンドを使用して 1 ～ 255 の値を設定できます。

「GLBP トポロジ」の図では、LAN トポロジ内の AVG であるルータ A（またはデバイス A）で障害が発生すると、選択プロセスが実行され、処理を引き継ぐバックアップ仮想ゲートウェイが決定されます。この例では、ルータ B（またはデバイス B）がグループ内の唯一の他のメンバであるため、ルータ B（またはデバイス B）が自動的に新しい AVG になります。同じ GLBP グループ内にプライオリティの高い別のデバイスが存在していた場合は、そのプライオリティの高いデバイスが選択されます。両方のデバイスのプライオリティが同じである場合は、IP アドレスが大きい方のバックアップ仮想ゲートウェイが選択され、アクティブ仮想ゲートウェイになります。

デフォルトでは、GLBP 仮想ゲートウェイのプリエンプティブ方式はディセーブルになっています。バックアップ仮想ゲートウェイが AVG になるのは、仮想ゲートウェイに割り当てられているプライオリティにかかわらず、現在の AVG で障害が発生した場合だけです。**glbp preempt** コマンドを使用すると、GLBP 仮想ゲートウェイのプリエンプティブスキームを有効にすることができます。プリエンプションを使用すると、バックアップ仮想ゲートウェイに現在の AVG よりも高いプライオリティが割り当てられている場合に、そのバックアップ仮想ゲートウェイを AVG にすることができます。

GLBP ゲートウェイの重み付けとトラッキング

GLBP では、重み付けによって GLBP グループ内の各デバイスの転送容量を決定します。GLBP グループ内のデバイスに割り当てられた重み付けを使用して、そのルータがパケットを転送するかどうか、転送する場合はパケットを転送する LAN 内のホストの比率を決定できます。しきい値は、GLBP の重み付けが一定の値を下回ったときに転送を無効化し、別のしきい値を上回ったときには自動的に転送を再度有効化にするように設定できます。

GLBP グループの重み付けは、デバイス内のインターフェイス状態のトラッキングによって自動的に調整できます。追跡対象のインターフェイスがダウンした場合、GLBP グループの重み付けは指定された値だけ小さくなります。GLBP の重み付けの減少値は、追跡対象のインターフェイスごとに変えることができます。

デフォルトでは、GLBP 仮想フォワーダのプリエンプティブ方式はイネーブルになっており、遅延は 30 秒です。現在の AVF の重み付けが下限しきい値を下回り、その状態で 30 秒経過すると、バックアップ仮想フォワーダが AVF になります。**no glbp forwarder preempt** コマンドを使用して GLBP フォワーダのプリエンプティブスキームを無効化するか、**glbp forwarder preempt delay minimum** コマンドを使用して遅延を変更することができます。

GLBP MD5 認証

GLBP MD5 認証は、信頼性とセキュリティを向上させるために業界標準の MD5 アルゴリズムを採用しています。MD5 認証を使用すると、別のプレーン テキスト認証方式よりもセキュリティを強化でき、スプーフィング ソフトウェアから保護できます。

MD5 認証では、各 GLBP グループ メンバが秘密キーを使用して、発信パケットに含まれるキー付き MD5 ハッシュを生成できます。着信パケットのキー付きハッシュが生成され、着信パケット内のハッシュが生成されたハッシュに一致しない場合、そのパケットは無視されます。

MD5 ハッシュのキーは、キーストリングを使用して設定で直接指定するか、またはキーチェーンを使用して間接的に指定できます。キーストリングは、100 文字の長さを超えることはできません。

デバイスは、GLBP グループに対する認証設定と異なる設定を持つデバイスからの着信 GLBP パケットを無視します。GLBP には、次の 3 つの認証方式があります。

- 認証なし
- プレーン テキスト認証
- MD5 認証

GLBP パケットは、次のいずれかの場合に拒否されます。

- 認証方式がデバイスと着信パケットの間で異なっている。
- MD5 ダイジェストがデバイスと着信パケットで異なる。
- テキスト認証文字列がデバイスと着信パケットで異なる。

ISSU-GLBP

GLBP は In Service Software Upgrade（ISSU）をサポートします。ISSU を使用すると、アクティブおよびスタンバイのルート プロセッサ（RP）またはライン カード上で異なるバージョンの Cisco IOS ソフトウェアが実行されている場合でも、ハイアベイラビリティ（HA）システムをステートフル スイッチオーバー（SSO）モードで実行できるようになります。

ISSU は、サポートされる Cisco IOS Release から別のリリースへアップグレードまたはダウングレードする機能を提供します。この場合、パケット転送は継続して行われ、セッションは維持されるため、予定されるシステムの停止時間を短くすることができます。アップグレードまたはダウングレードする機能は、アクティブ RP およびスタンバイ RP 上で異なるバージョンのソフトウェアを実行することで実現します。これにより、RP 間でステート情報を維持する時間が短くなります。この機能により、システムをアップグレード対象（またはダウングレード対象）のソフトウェアを実行するセカンダリ RP に切り替えることができ、セッションを切断することなく、またパケットの損失も最小限に抑えながら、継続してパケットを転送できます。この機能は、デフォルトでイネーブルにされています。

GLBP SSO

GLBP SSO 機能が導入されたため、GLBP はステートフル スイッチオーバー（SSO）を認識するようになりました。GLBP は、デバイスがセカンダリ ルータ プロセッサ（RP）にフェールオーバーしたことを検出し、グループの現在の状態を継続することができます。

SSO は、デュアル RP をサポートするネットワーキング デバイス（通常はエッジ デバイス）で機能します。1 台の RP をアクティブ プロセッサとして設定し、他の RP をスタンバイ プロセッサとして設定することで、RP 冗長化を実現します。また、RP 間の重要なステート情報を同期するため、ネットワーク ステート情報は RP 間でダイナミックに維持されます。

SSO を認識せずに RP が冗長化されたデバイスに GLBP を展開した場合、アクティブ RP とスタンバイ RP 間のロールがスイッチオーバーされると、デバイスの GLBP グループ メンバとしてのアクティビティは破棄され、デバイスはリロードされた場合と同様にグループに再び参加することになります。GLBP SSO 機能により、スイッチオーバーが行われても、GLBP は継続してグループ メンバとしてのアクティビティを継続できます。冗長化された RP 間の GLBP ステート情報は維持されるため、スタンバイ RP はスイッチオーバーの実行中も実行後も GLBP 内で引き続きデバイスのアクティビティを実行できます。

この機能は、デフォルトでイネーブルにされています。この機能を無効化するには、グローバル コンフィギュレーション モードで **no glbp sso** コマンドを使用します。

GLBP の利点

ロード シェアリング

LAN クライアントからのトラフィックを複数のデバイスで共有するように GLBP を設定できるため、利用可能なデバイス間でより公平にトラフィックの負荷を共有できます。

複数の仮想デバイス

GLBP では、デバイスの各物理インターフェイス上に最大 1024 台の仮想デバイス（GLBP グループ）とグループごとに最大 4 つの仮想フォワーダがサポートされます。

プリエンプション

GLBP の冗長性スキームにより、使用可能になっているプライオリティの高いバックアップ仮想ゲートウェイをアクティブ仮想ゲートウェイ（AVG）にすることができます。フォワーダ プリエンプションも同じように機能しますが、フォワーダ プリエンプションはプライオリティの代わりに重み付けを使用し、デフォルトでイネーブルになっている点が異なります。

認証

GLBP は、信頼性やセキュリティを向上させて GLBP スプーフィング ソフトウェアからの保護を強化するための業界標準のメッセージ ダイジェスト 5（MD5）アルゴリズムをサポートしています。GLBP グループ内のデバイスの認証文字列が他のデバイスとは異なる場合、そのデバイスは他のグループ メンバによって無視されます。GLBP グループ メンバ間で簡単なテキスト パスワード認証方式を使用して、設定エラーを検出することもできます。

GLBP の設定方法

GLBP のカスタマイズ

GLBP 動作のカスタマイズは任意です。GLBP グループをイネーブルにすると、そのグループはすぐに動作します。GLBP グループをイネーブルにしてから GLBP をカスタマイズすると、機能のカスタマイズを完了する前にデバイス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。したがって、GLBP をカスタマイズする場合は、GLBP をイネーブルにする前に行うことを推奨します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address mask* [**secondary**] 例： `Device(config-if)# ip address 10.21.8.32 255.255.255.0`	インターフェイスのプライマリ IP アドレスまたはセカンダリ IP アドレスを指定します。
ステップ 5	**glbp** *group* **timers** [**msec**] *hellotime* [**msec**] *holdtime* 例： `Device(config-if)# glbp 10 timers 5 18`	GLBP グループ内の AVG によって連続的に送信される hello パケットの間隔を設定します。 • *holdtime* 引数には、hello パケット内の仮想ゲートウェイと仮想フォワーダの情報が無効と見なされるまでの時間を秒数で指定します。 • オプションの **msec** キーワードは、その後に続く引数がデフォルトの秒単位ではなくミリ秒単位で表されることを指定します。
ステップ 6	**glbp** *group* **timers redirect** *redirect timeout* 例： `Device(config-if)# glbp 10 timers redirect 1800 28800`	AVG がクライアントを AVF にリダイレクトし続ける時間を設定します。デフォルトは 600 秒（10 分）です。 • *timeout* 引数には、セカンダリ仮想フォワーダが無効になるまでの時間を秒数で指定します。デフォルトは 14,400 秒（4 時間）です。

	コマンドまたはアクション	目的
		（注） *redirect* 引数のゼロ（0）値は、指定できる値の範囲から除外することはできません。Cisco IOS ソフトウェアの事前設定でゼロ（0）値を使用しているため、アップグレードに悪影響を及ぼすことになります。ただし、ゼロ（0）値に設定することは推奨しません。この値を使用すると、リダイレクトタイマーが期限切れになりません。リダイレクトタイマーが期限切れにならず、デバイスに障害が発生すると、新しいホストがバックアップへリダイレクトされずに、障害が発生したデバイスに引き続き割り当てられます。
ステップ 7	**glbp** *group* **load-balancing** [**host-dependent** | **round-robin** | **weighted**] 例： `Device(config-if)# glbp 10 load-balancing host-dependent`	GLBP AVG で使用するロードバランシングの方式を指定します。
ステップ 8	**glbp** *group* **priority** *level* 例： `Device(config-if)# glbp 10 priority 254`	GLBP グループ内のゲートウェイのプライオリティ レベルを設定します。 • デフォルト値は 100 です。
ステップ 9	**glbp** *group* **preempt** [**delay minimum** *seconds*] 例： `Device(config-if)# glbp 10 preempt delay minimum 60`	デバイスのプライオリティが現在の AVG よりも高い場合に、GLBP グループの AVG として処理を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になっています。 • AVG のプリエンプションが行われるまでの最小遅延時間を秒数で指定するには、オプションの **delay** キーワードと **minimum** キーワー

	コマンドまたはアクション	目的
		ドおよび *seconds* 引数を使用します。
ステップ 10	**glbp** *group* **client-cache maximum** *number* [**timeout** *minutes*] 例： `Device(config-if)# glbp 10 client-cache maximum 1200 timeout 245`	（任意）GLBP クライアント キャッシュをイネーブルにします。 • このコマンドは、デフォルトでディセーブルになっています。 • *number* 引数を使用して、キャッシュがこの GLBP グループのためにホールドするクライアントの最大数を指定します。範囲は 8 ～ 2000 です。 • オプションの **timeout** *minutes* キーワードと引数のペアを使用して、クライアント情報が最後に更新された後、クライアントエントリが GLBP クライアントキャッシュに保存される最大時間を設定します。範囲は、1 ～ 1440 分（1 日）です。 （注） IPv4 ネットワークには、予測されるエンド ホストの Address Resolution Protocol（ARP）キャッシュの最大タイムアウト値よりも若干長い GLBP クライアント キャッシュのタイムアウト値を設定することを推奨します。
ステップ 11	**glbp** *group* **name** *redundancy-name* 例： `Device(config-if)# glbp 10 name abc123`	GLBP グループに名前を割り当てることによって、IP 冗長性をイネーブルにします。 • 冗長クライアントと GLBP グループを接続できるように、GLBP 冗長クライアントに同じ GLBP グループ名を設定する必要があります。

	コマンドまたはアクション	目的
ステップ 12	**exit** 例： `Device(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、デバイスをグローバル コンフィギュレーション モードに戻します。
ステップ 13	**no glbp sso** 例： `Device(config)# no glbp sso`	（任意）SSO の GLBP サポートをディセーブルにします。

キー ストリングを使用した GLBP MD5 認証の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address mask* [**secondary**] 例： `Device(config-if)# ip address 10.0.0.1 255.255.255.0`	インターフェイスのプライマリ IP アドレスまたはセカンダリ IP アドレスを指定します。
ステップ 5	**glbp** *group-number* **authentication md5 key-string** [**0** | **7**] *key* 例： `Device(config-if)# glbp 1 authentication md5 key-string d00b4r987654321a`	GLBP MD5 認証の認証キーを設定します。 • キー ストリングは、100 文字の長さを超えることはできません。 • *key* 引数にプレフィックスを指定しない場合や、**0** を指定した場合、

	コマンドまたはアクション	目的
		キーが暗号化されないことを意味します。 • **7** を指定した場合、キーが暗号化されることを意味します。**service password-encryption** グローバル コンフィギュレーション コマンドが有効になっている場合、key-string 認証キーは自動的に暗号化されます。
ステップ 6	**glbp** *group-number* **ip** [*ip-address* [**secondary**]] 例： `Device(config-if)# glbp 1 ip 10.0.0.10`	インターフェイス上でGLBPを設定し、仮想ゲートウェイのプライマリ IP アドレスを指定します。
ステップ 7	通信する各デバイスに対してステップ 1 ～ 6 を繰り返します。	—
ステップ 8	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show glbp** 例： `Device# show glbp`	（任意）GLBP の情報を表示します。 • このコマンドを使用して、設定を確認します。設定されている場合はキー ストリングと認証タイプが表示されます。

キー チェーンを使用した GLBP MD5 認証の設定

キーチェーンを使用したGLBPMD5認証を設定するには、次の作業を実行します。キーチェーンを使用すると、キー チェーン設定に従って異なる時点で異なるキー ストリングを使用できます。GLBP は、適切なキー チェーンを照会して、指定されたキー チェーンの現在アクティブなキーとキー ID を取得します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。

	コマンドまたはアクション	目的
	`Device> enable`	• パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**key chain** *name-of-chain* 例： `Device(config)# key chain glbp2`	ルーティング プロトコルの認証をイネーブルにし、認証キーのグループを識別し、キー チェーン キー 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	**key** *key-id* 例： `Device(config-keychain)# key 100`	キー チェーンの認証キーを識別します。 • *key-id* 引数の値には数値を指定する必要があります。
ステップ 5	**key-string** *string* 例： `Device(config-keychain-key)# key-string abc123`	キーの認証文字列を指定し、キー チェーン キー コンフィギュレーション モードを開始します。 • *string* 引数の値は、1 ～ 80 文字の大文字または小文字の英数字を指定できます。最初の文字には数字を使用できません。
ステップ 6	**exit** 例： `Device(config-keychain-key)# exit`	キー チェーン キー コンフィギュレーション モードに戻ります。
ステップ 7	**exit** 例： `Device(config-keychain)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 9	**ip address** *ip-address mask* [**secondary**] 例： `Device(config-if)# ip address 10.21.0.1 255.255.255.0`	インターフェイスのプライマリ IP アドレスまたはセカンダリ IP アドレスを指定します。
ステップ 10	**glbp** *group-number* **authentication md5 key-chain** *name-of-chain* 例： `Device(config-if)# glbp 1 authentication md5 key-chain glbp2`	GLBP MD5 認証の認証 MD5 キーチェーンを設定します。 • キー チェーン名は、ステップ 3 で指定した名前に一致する必要があります。
ステップ 11	**glbp** *group-number* **ip** [*ip-address* [**secondary**]] 例： `Device(config-if)# glbp 1 ip 10.21.0.12`	インターフェイス上で GLBP を設定し、仮想ゲートウェイのプライマリ IP アドレスを指定します。
ステップ 12	通信する各デバイスに対してステップ 1 ～ 10 を繰り返します。	—
ステップ 13	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 14	**show glbp** 例： `Device# show glbp`	（任意）GLBP の情報を表示します。 • このコマンドを使用して、設定を確認します。設定されている場合はキーチェーンと認証タイプが表示されます。
ステップ 15	**show key chain** 例： `Device# show key chain`	（任意）認証キー情報を表示します。

GLBP テキスト認証の設定

テキスト認証は最小限のセキュリティを提供します。セキュリティが必須の場合は、MD5 認証を使用してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address mask* [**secondary**] 例： `Device(config-if)# ip address 10.0.0.1 255.255.255.0`	インターフェイスのプライマリ IP アドレスまたはセカンダリ IP アドレスを指定します。
ステップ 5	**glbp** *group-number* **authentication text** *string* 例： `Device(config-if)# glbp 10 authentication text stringxyz`	グループ内の他のデバイスから受信した GLBP パケットを認証します。 • 認証を設定する場合は、GLBP グループ内のすべてのデバイスで同じ認証文字列を使用する必要があります。
ステップ 6	**glbp** *group-number* **ip** [*ip-address* [**secondary**]] 例： `Device(config-if)# glbp 1 ip 10.0.0.10`	インターフェイス上で GLBP を設定し、仮想ゲートウェイのプライマリ IP アドレスを指定します。
ステップ 7	通信する各デバイスに対してステップ 1 ～ 6 を繰り返します。	—
ステップ 8	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	**show glbp** 例： `Device# show glbp`	（任意）GLBP の情報を表示します。 • このコマンドを使用して、設定を確認します。

GLBP の重み付けの値とオブジェクト トラッキング

GLBP 重み付けにより、GLBP グループが仮想フォワーダとして動作できるかどうかが決定されます。重み付けの初期値を設定したり、オプションのしきい値を指定したりできます。インターフェイスの状態を追跡し、インターフェイスがダウンした場合に重み付けの値を減らすための減少値を設定できます。GLBP グループの重み付けが指定の値を下回ると、グループはアクティブ仮想フォワーダでなくなります。重み付けが指定の値を上回ると、グループは再びアクティブ仮想フォワーダとしてのロールを実行できるようになります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number* **interface** *type number* {**line-protocol** |{**ip** | **ipv6**} **routing**} 例： `Device(config)# track 2 interface GigabitEthernet 1/0/1 ip routing`	GLBP ゲートウェイの重み付けに影響する状態変化を追跡するインターフェイスを設定し、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。 • このコマンドは、**glbp weighting track** コマンドで使用されるインターフェイスと対応するオブジェクトの数を設定します。 • **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状態かどうかが追跡されます。**ip routing** キーワードを指定すると、インターフェイス上でIPルーティングが有効になっているかどうか、およびIPアドレスが設定され

	コマンドまたはアクション	目的
		ているかどうかもチェックされます。
ステップ 4	**exit** 例： `Device(config-track)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 1/0/1`	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 6	**glbp** *group* **weighting** *maximum* [**lower** *lower*] [**upper** *upper*] 例： `Device(config-if)# glbp 10 weighting 110 lower 95 upper 105`	GLBP ゲートウェイの重み付けの初期値、上限しきい値、および下限しきい値を指定します。
ステップ 7	**glbp** *group* **weighting track** *object-number* [**decrement** *value*] 例： `Device(config-if)# glbp 10 weighting track 2 decrement 5`	GLBP ゲートウェイの重み付けに影響する、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付けを減らす量を指定します。
ステップ 8	**glbp** *group* **forwarder preempt** [**delay minimum** *seconds*] 例： `Device(config-if)# glbp 10 forwarder preempt delay minimum 60`	GLBP グループの現在の AVF の値が重みしきい値よりも低くなった場合に、GLBP グループの AVF としてのロールを引き継ぐデバイスを設定します。 • このコマンドは、デフォルトでイネーブルになっており、遅延は 30 秒です。 • AVF のプリエンプションが行われるまでの最小遅延時間を秒数で指定するには、オプションの **delay** キーワードと **minimum** キーワードおよび *seconds* 引数を使用します。

	コマンドまたはアクション	目的
ステップ 9	**exit** 例： `Device(config-if)# exit`	特権 EXEC モードに戻ります。
ステップ 10	**show track** [*object-number* | **brief**] [**interface** [**brief**] | **ip route** [**brief**] | **resolution** | **timers**] 例： `Device# show track 2`	トラッキング情報を表示します。

GLBP のトラブルシューティング

GLBP には、GLBP 動作に関する各種イベントに関連する診断出力を可視化する 5 つの特権 EXEC モード コマンドが導入されています。**debug condition glbp**、**debug glbp errors**、**debug glbp events**、 **debug glbp packets**、**debug glbp terse** コマンドは、使用時にソフトウェアが生成する出力の量によってデバイスの性能が著しく低下するため、トラブルシューティング専用となります。**debug glbp** コマンドを使用した場合の影響を最小限に抑えるには、次の作業を実行します。

この手順により、コンソールポートが文字ごとにプロセッサ割り込みを行わなくなるため、**debug condition glbp** コマンドまたは **debug glbp** コマンドを使用することでデバイスにかかる負荷が最小限に抑えられます。直接コンソールに接続できない場合は、ターミナルサーバーを介してこの手順を実行できます。ただし、Telnet 接続を切断しなければならない場合は、デバッグ出力の生成でプロセッサに負荷がかかりデバイスが応答できないことに起因して、再接続できないことがあります。

始める前に

この作業では、コンソールに直接接続された GLBP を実行しているデバイスが必要です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**no logging console** 例： `Device(config)# no logging console`	コンソール端末へのすべてのロギングをディセーブルにします。 • コンソールへのロギングを再度有効にするには、グローバル コンフィギュレーション モードで **logging console** コマンドを使用します。
ステップ 4	Telnet を使用してデバイス ポートにアクセスし、ステップ 1 と 2 を繰り返します。	再帰 Telnet セッションでグローバル コンフィギュレーション モードを開始します。これにより、出力をコンソール ポートからリダイレクトできます。
ステップ 5	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**terminal monitor** 例： `Device# terminal monitor`	仮想端末でのロギング出力をイネーブルにします。
ステップ 7	**debug condition glbp** *interface-type interface-number group* [*forwarder*] 例： `Device# debug condition glbp GigabitEthernet 0/0/0 1`	GLBP状態に関するデバッグ メッセージを表示します。 • 特定の **debug condition glbp** または **debug glbp** コマンドだけを入力して特定のサブコンポーネントへの出力を分離し、プロセッサの負荷を最小化します。適切な引数とキーワードを使用して、指定したサブコンポーネント上に詳細なデバッグ情報を生成します。 • 終了したら、特定の **no debug condition glbp** または **no debug glbp** コマンドを入力します。
ステップ 8	**terminal no monitor** 例： `Device# terminal no monitor`	仮想端末でのロギングをディセーブルにします。

GLBP の設定例

例：GLBP 設定のカスタマイズ

```
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# ip address 10.21.8.32 255.255.255.0
Device(config-if)# glbp 10 timers 5 18
Device(config-if)# glbp 10 timers redirect 1800 28800
Device(config-if)# glbp 10 load-balancing host-dependent
Device(config-if)# glbp 10 priority 254
Device(config-if)# glbp 10 preempt delay minimum 60

Device(config-if)# glbp 10 client-cache maximum 1200 timeout 245
```

例：キー ストリングを使用した GLBP MD5 認証の設定

次に、キー ストリングを使用して GLBP MD5 認証を設定する例を示します。

```
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# ip address 10.0.0.1 255.255.255.0
Device(config-if)# glbp 2 authentication md5 key-string ThisStringIsTheSecretKey
Device(config-if)# glbp 2 ip 10.0.0.10
```

例：キー チェーンを使用した GLBP MD5 認証の設定

次に、GLBP がキー チェーン「AuthenticateGLBP」を照会して、指定されたキー チェーンの現在アクティブなキーとキー ID を取得する例を示します。

```
Device(config)# key chain AuthenticateGLBP
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string ThisIsASecretKey
Device(config-keychain-key)# exit
Device(config-keychain)# exit
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# ip address 10.0.0.1 255.255.255.0
Device(config-if)# glbp 2 authentication md5 key-chain AuthenticateGLBP
Device(config-if)# glbp 2 ip 10.0.0.10
```

例：GLBP テキスト認証の設定

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 10.21.8.32 255.255.255.0
Device(config-if)# glbp 10 authentication text stringxyz
Device(config-if)# glbp 10 ip 10.21.8.10
```

例：GLBP 重み付けの設定

次に、デバイスを POS インターフェイス 5/0/0 と 6/0/0 の IP ルーティング状態を追跡するように設定し、GLBP の重み付けの初期値、上限しきい値、下限しきい値、および重み付けの減少値 10 を設定する例を示します。POS インターフェイス 5/0/0 と 6/0/0 がダウンすると、デバイスの重み付けの値が小さくなります。

```
Device(config)# track 1 interface GigabitEthernet 1/0/1 line-protocol
Device(config)# track 2 interface GigabitEthernet 1/0/3 line-protocol
Device(config)# interface TenGigabitEthernet 0/0/1
Device(config-if)# ip address 10.21.8.32 255.255.255.0
Device(config-if)# glbp 10 weighting 110 lower 95 upper 105
Device(config-if)# glbp 10 weighting track 1 decrement 10
Device(config-if)# glbp 10 weighting track 2 decrement 10
```

例：GLBP 設定のイネーブル化

次の例では、デバイスは GLBP をイネーブルにするように設定されています。GLBP グループ 10 には、仮想 IP アドレス 10.21.8.10 が指定されています。

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 10.21.8.32 255.255.255.0
Device(config-if)# glbp 10 ip 10.21.8.10
```

GLBP に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	の「IP マルチキャスト ルーティングのコマンド」の項を参照してください。*Command Reference (Catalyst 9300 Series Switches)*

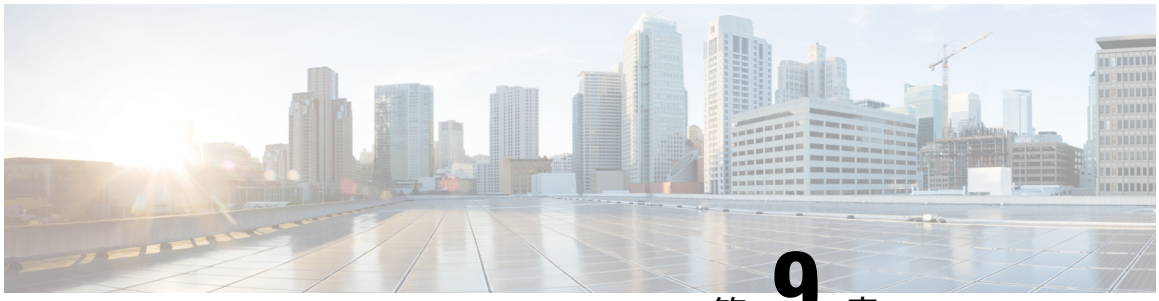
GLBP の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.12.1	Gateway Load Balancing Protocol	GLBP は、冗長化されたルータ グループ間でパケットのロード シェアリングを行う一方、機能を停止したルータや回路（HSRP や VRRP など）からのデータ トラフィックを保護します。
	GLBP MD5 認証	MD5 認証を使用すると、別のプレーンテキスト認証方式よりもセキュリティを強化できます。MD5 認証では、各 GLBP グループ メンバが秘密キーを使用して、発信パケットに含まれるキー付き MD5 ハッシュを生成できます。着信パケットのキー付きハッシュが生成され、着信パケット内のハッシュが生成されたハッシュに一致しない場合、そのパケットは無視されます。
	SSO：GLBP	GLBP が SSO を認識するようになりました。GLBP は、ルータがセカンダリ RP にフェールオーバーしたことを検出し、GLBP グループの現在の状態を継続することができます。 別の RP がインストールされ、プライマリ RP が機能を停止した場合にはその処理を引き継ぐように設定されても、SSO を認識する前であるときは GLBP はこれを認識できません。プライマリが機能を停止すると、GLBP デバイスは GLBP グループに参加しなくなります。また、そのロールに応じて、グループ内の他のルータにアクティブ ルータとしてのロールが引き継がれます。このように機能が強化され、GLBP がセカンダリ RP に対するフェールオーバーを検出できるようになったため、GLBP グループに何ら変化は生じません。セカンダリ RP が機能を停止した場合、プライマリ RP が以前として利用できない状態であると、GLBP グループはこの状態を検出して新たなアクティブ GLBP ルータを再度選定します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 9 章

HSRP の設定

ホット スタンバイ ルータ プロトコルに関する情報

ここでは、Hot Standby Router Protocol（HSRP）について説明します。

HSRP の概要

HSRP は、デフォルト ゲートウェイ IP アドレスが設定された IEEE 802 LAN 上の IP ホストにファーストホップ冗長性を確保することでネットワークのアベイラビリティを高めるシスコの標準方式です。HSRP を使用すると、特定のルータのアベイラビリティに依存せず IP トラフィックをルーティングできます。また、一連のルータ インターフェイスを組み合わせることで、1 台の仮想ルータ、または LAN 上のホストへのデフォルト ゲートウェイのように機能させることができます。ネットワークまたはセグメント上に HSRP を設定すると、仮想 MAC（メディア アクセス コントロール）アドレス、および設定されたルータ グループ間で共有される IP アドレスを使用できるようになり HSRP が設定された複数のルータは、仮想ルータの MAC アドレスおよび IP ネットワーク アドレスを使用できるようになります。仮想ルータは、実際には存在しません。仮想ルータは、相互にバックアップ機能を提供するように設定されている複数のルータの共通のターゲットを表します。1 台のルータがアクティブなルータとして、もう 1 台のルータがスタンバイ ルータとして選択されます。スタンバイ ルータは、指定されたアクティブ ルータが故障した場合に、グループの MAC アドレスおよび IP アドレスを制御するルータです。

（注） HSRP グループ内のルータには、ルーテッド ポート、スイッチ仮想インターフェイス（SVI）など、HSRP をサポートする任意のルータ インターフェイスを指定できます。

HSRPは、ネットワーク上のホストからのIPトラフィックに冗長性を提供することで、ネットワークのアベイラビリティを高めます。アクティブ ルータは、ルータ インターフェイスのグループ内でパケットのルーティングを実行するために選択されたルータです。スタンバイ ルータは、アクティブ ルータが故障した場合、または事前に設定した条件が満たされた場合に、ルーティング作業を引き継ぐルータです。

HSRPは、ホストがルータ ディスカバリ プロトコルをサポートしておらず、選択されたルータのリロードや電源故障時に新しいルータに切り替えることができない場合に有効です。HSRPをネットワーク セグメントに設定すると、HSRP は仮想 MAC アドレスと IP アドレスを 1 つずつ提供します。このアドレスは、HSRPが動作するルータ インターフェイス グループ内のルータ インターフェイス間で共有できます。プロトコルによってアクティブ ルータとして選択されたルータは、グループの MAC アドレス宛てのパケットを受信し、ルーティングします。n台のルータでHSRP が稼働している場合、n+1 個の IP アドレスおよびMACアドレスが割り当てられます。

指定されたアクティブ ルータの故障を HSRP が検出すると、選択されているスタンバイ ルータがホットスタンバイ グループの MAC アドレスおよび IP アドレスの制御を引き継ぎます。この時点で新しいスタンバイ ルータも選択されます。HSRPが稼働しているデバイスは、マルチキャスト UDP ベースの hello パケットを送受信することにより、ルータ障害の検出、アクティブ ルータおよびスタンバイ ルータの指定を行います。インターフェイスに HSRP が設定されている場合、そのインターフェイスではインターネット制御メッセージプロトコル（ICMP）のリダイレクト メッセージが自動的にイネーブルになっています。

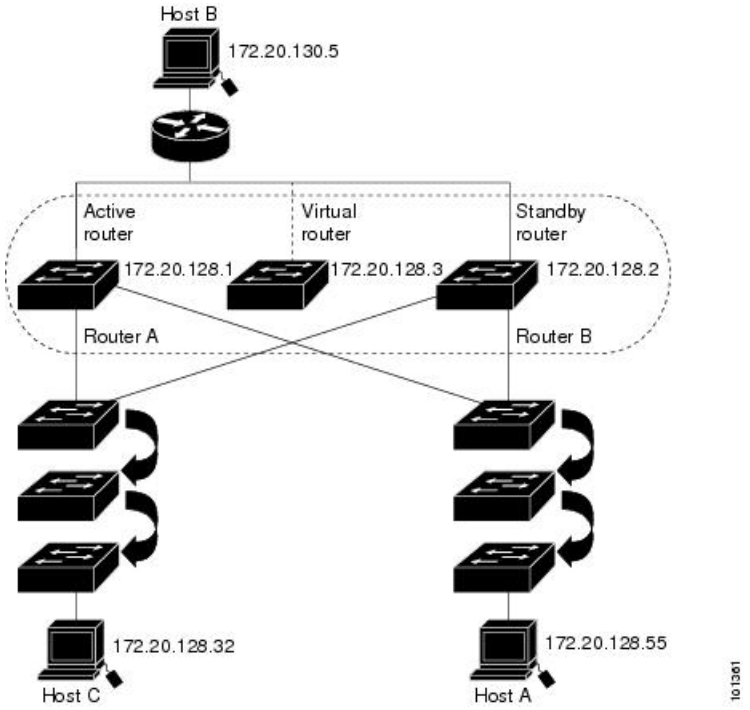
レイヤ 3 で動作するスイッチおよびスイッチ スタック間で複数のホット スタンバイ グループを設定すると、冗長ルータをさらに活用できます。

そのためには、インターフェイスに設定するホットスタンバイ コマンド グループごとにグループ番号を指定します。たとえば、スイッチ1のインターフェイスをアクティブ ルータ、スイッチ2のインターフェイスをスタンバイ ルータとして設定できます。また、スイッチ2の別のインターフェイスをアクティブ ルータ、スイッチ 1 の別のインターフェイスをスタンバイ ルータとして設定することもできます。

次の図に、HSRP 用に設定されたネットワークのセグメントを示します。各ルータには、仮想ルータの MAC アドレスおよび IP ネットワーク アドレスが設定されています。ルータ A の IP アドレスをネットワーク上のホストに設定する代わりに、デフォルト ルータとして仮想ルータの IP アドレスを設定します。ホスト C からホスト B にパケットが送信される場合、ホスト C は仮想ルータの MAC アドレスにパケットを送信します。何らかの理由により、ルータ A がパケットの転送を停止すると、ルータ B が仮想 IP アドレスおよび仮想 MAC アドレスに応答してアクティブ ルータとなり、アクティブ ルータの作業を行います。ホスト C は引き続き仮想ルータの IP アドレスを使用し、ホスト B 宛のパケットをアドレッシングします。ルータ B はそのパケットを受信し、ホスト B に送信します。ルータ B は HSRP の機能を使用し、ルータ A が動作を再開するまで、ホスト B のセグメント上のユーザーと通信する必要があるホスト C

のセグメント上のユーザーに連続的にサービスを提供します。また、ホストAセグメントとホストB の間で、引き続き通常のパケット処理機能を実行します。

図 *11 : HSRP* の一般的な構成



HSRP のバージョン

Cisco IOS XE Everest 16.5.1a 以降のスイッチでサポートされている Hot Standby Router Protocol（HSRP）のバージョンは次のとおりです。

スイッチでは、次の HSRP バージョンがサポートされます。

- HSRPv1：HSRP のバージョン 1（デフォルトのバージョン）。次の機能があります。
  - HSRP グループ番号は 0 ～ 255 まで使用できます。
  - HSRPv1 は 224.0.0.2 のマルチキャスト アドレスを使用して hello パケットを送信しますが、これは Cisco Group Management Protocol（CGMP）の脱退処理と競合します。HSRPv1 と CGMP は相互に排他的なため、同時には使用できません。
- HSRPv2：HSRP のバージョン 2。このバージョンには次の機能があります。
  - HSRPv2 は 224.0.0.102 のマルチキャスト アドレスを使用して hello パケットを送信します。HSRPv2 と CGMP 脱退処理は相互に排他的ではありません。同時に使用できます。
  - HSRPv2 のパケット形式は、HSRPv1 とは異なります。

HSRPv1 を実行しているスイッチは、ルータの送信元 MAC アドレスが仮想 MAC アドレスのため、hello パケットを送信した物理的なルータを特定できません。

HSRPv2 のパケット形式は、HSRPv1 とは異なります。HSRPv2 パケットは、パケットを送信した物理ルータの MAC アドレスを格納できる 6 バイトの識別子フィールドを持った、Type Length Value（TLV）形式を使用します。

HSRPv1 を実行しているインターフェイスが HSRPv2 パケットを取得した場合、このタイプフィールドは無視されます。

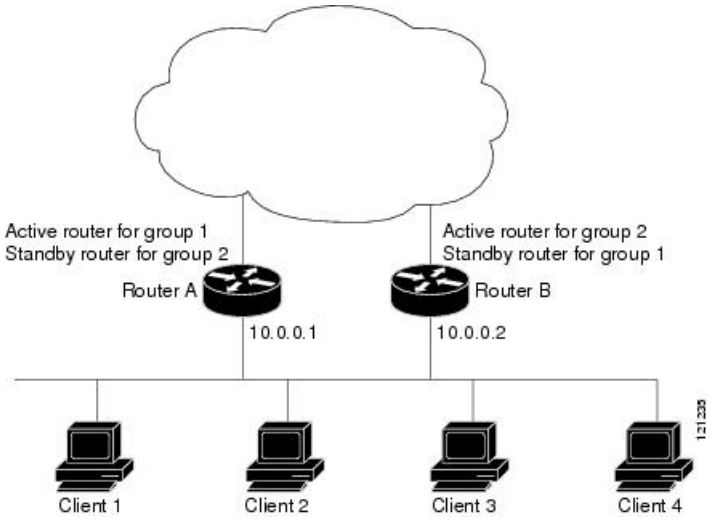
MHSRP

スイッチは、Multiple HSRP（MHSRP）をサポートします。MHSRP は HSRP の拡張版で、複数の HSRP グループ間でのロード シェアリングが可能です。ホスト ネットワークからサーバー ネットワークまで、ロードバランシングを実現して複数のスタンバイ グループ（およびパス）を使用するために、MHSRP を設定できます。

下の図では、半分のクライアントがルータ A に設定されており、もう半分はルータ B に設定されています。ルータ A およびルータ B の設定により、合計 2 つの HSRP グループが確立されています。グループ 1 では、ルータ A に最高のプライオリティが割り当てられているので、ルータ A がデフォルトのアクティブ ルータになり、ルータ B がスタンバイ ルータとなります。グループ 2 では、ルータ B に最も高いプライオリティが割り当てられているため、ルータ B がデフォルトのアクティブ ルータであり、ルータ A がスタンバイ ルータです。通常の運用では、2 つのルータが IP トラフィック負荷を分散します。いずれかのルータが使用できなくなると、もう一方のルータがアクティブになり、使用できないルータのパケット転送機能を引き継ぎます。

（注） MHSRP では、ルータに障害が発生して正常に戻った場合にプリエンプションによりロード シェアリングを復元するために、**standby preempt** インターフェイス コンフィギュレーション コマンドを HSRP インターフェイスで入力する必要があります。

図 *12 : MHSRP* ロード シェアリング



SSO HSRP

SSO HSRP は、冗長なルート プロセッサ（RP）を装備したデバイスがステートフル スイッチオーバー（SSO）冗長モード用に設定されているときの HSRP の動作を変更します。ある RP がアクティブで、もう一方の RP がスタンバイになっているとき、アクティブ RP に障害が発生すると、SSO は処理を引き継ぐスタンバイ RP をイネーブルにします。

この機能を使用すると、HSRP の SSO 情報がスタンバイ RP に同期されるため、HSRP 仮想 IP アドレスを使用して送信されるトラフィックをスイッチオーバー中も引き続き転送できるほか、データの損失やパスの変更も発生しません。さらに、HSRP アクティブ デバイスの両方の RP に障害が発生しても、スタンバイ状態の HSRP デバイスが HSRP アクティブ デバイスとして処理を引き継ぎます。

この機能は、動作の冗長モードが SSO に設定されている場合にデフォルトでイネーブルになっています。

HSRP およびスイッチ スタック

HSRP の hello メッセージは、アクティブなスイッチで生成されます。アクティブなスイッチの HSRP に障害が発生すると、HSRP アクティブ状態のフラッピングが生じることがあります。これは、新規のアクティブなスイッチが選択および初期化されている間に HSRP hello メッセージが生成されず、アクティブなスイッチが故障した後でないとスタンバイルータがアクティブにならない可能性があるためです。

IPv6 の HSRP の設定

Network Advantage ライセンスを実行中のスイッチは、IPv6 の Hot Standby Router Protocol（HSRP）をサポートします。HSRP は、任意の単一のルータのアベイラビリティに依存せず、ルーティ

ング IPv6 トラフィックにルーティング冗長性を提供します。IPv6 ホストは、IPv6 ネイバー探索ルータのアドバタイズメント メッセージによって使用可能なルータを学習します。これらのメッセージは定期的にマルチキャストされるか、ホストにより送信請求されます。

HSRP IPv6 グループには、HSRP グループ番号に基づく仮想 MAC アドレス、およびデフォルトで HSRP 仮想 MAC アドレスに基づく HSRP の仮想 IPv6 リンクローカル アドレスがあります。

HSRP グループがアクティブな場合、定期的なメッセージが HSRP 仮想 IPv6 リンクローカル アドレスに送信されます。グループがアクティブ ステートでなくなった場合、これらのメッセージは最後のメッセージが送信されたあとで停止します。

（注） IPv6 の HSRP を設定する場合、インターフェイス上で HSRP version 2（HSRPv2）をイネーブルにする必要があります。

HSRP IPv6 仮想 MAC アドレスの範囲

HSRP IPv6 では、次に示すように、HSRP for IP とは異なる仮想 MAC アドレス ブロックを使用します。

0005.73A0.0000 through 0005.73A0.0FFF（4096 のアドレス）

HSRP IPv6 UDP ポート番号

HSRP IPv6 には、ポート番号 2029 が割り当てられています。

ホット スタンバイ ルータ プロトコルの設定方法

ここでは、HSRP に関する設定情報について説明します。

HSRP のデフォルト設定

表 *5: HSRP* のデフォルト設定

機能	デフォルト設定
HSRP バージョン	バージョン 1
HSRP グループ	未設定
スタンバイ グループ番号	0
スタンバイ MAC アドレス	0000.0c07.acXX に指定されたシステム。XX は、HSRP グループ番号

機能	デフォルト設定
スタンバイ プライオリティ	100
スタンバイ遅延	0（遅延なし）
スタンバイでのインターフェイス プライオリティの追跡	10
スタンバイ hello 時間	3 秒
スタンバイ ホールドタイム	10 秒

HSRP 設定時の注意事項

- HSRPv2 および HSRPv1 は相互に排他的です。HSRPv2 は、同じインターフェイス上で HSRPv1 と一緒には動作しません（その逆も同様）。
- 以下の手順では、次に示すレイヤ 3 インターフェイスの 1 つを指定する必要があります。
  - ルーテッドポート：インターフェイス コンフィギュレーション モードで **no switchport** コマンドを入力することにより、レイヤ 3 ポートとして設定された物理ポート。
  - SVI：グローバル コンフィギュレーション モードで **interface vlan** *vlan_id* を使用して作成された VLAN インターフェイス。デフォルトではレイヤ 3 インターフェイスです。
  - レイヤ 3 モードの Etherchannel ポートチャネル：グローバル コンフィギュレーション モードで **interface port-channel** *port-channel-number* を使用し、イーサネット インターフェイスをチャネルグループにバインドして作成されたポートチャネル論理インターフェイス。
- すべてのレイヤ 3 インターフェイスに IP アドレスを割り当てる必要があります。
- HSRP のミリ秒タイマーはサポートされません。

HSRP のイネーブル化

standby ip インターフェイス コンフィギュレーション コマンドは、設定されているインターフェイスで HSRP をアクティブにします。IP アドレスを指定した場合は、IP アドレスがホットスタンバイ グループの指定アドレスとして使用されます。IP アドレスを指定しなかった場合は、スタンバイ機能によってアドレスが学習されます。指定アドレスを使用し、LAN 上に少なくとも 1 つのレイヤ 3 ポートを設定する必要があります。IP アドレスを設定すると、常に、現在使用されている別の指定アドレスが、設定した IP アドレスに変更されます。

standby ip コマンドがインターフェイス上で有効にされており、プロキシ ARP が有効な場合、インターフェイスのホットスタンバイ状態がアクティブになると、プロキシ ARP 要求に対する応答は、ホットスタンバイグループの MAC アドレスを使用して実行されます。インターフェイスが別のステートの場合、プロキシ ARP の応答は抑制されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch(config)# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface gigabitethernet1/0/1`	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、HSRP をイネーブルにするレイヤ 3 インターフェイスを入力します。
ステップ 3	**standby version** ｛ **1** ｜ **2** ｝ 例： `Switch(config-if)# standby version 1`	（任意）インターフェイスに HSRP バージョンを設定します。 • 1：HSRPv1 を選択します。 • 2：HSRPv2 を選択します。 このコマンドを入力しない場合、またはキーワードを指定しない場合、インターフェイスはデフォルトの HSRP バージョンである HSRPv1 を実行します。
ステップ 4	**standby** ［*group-number*］ **ip** ［*ip-address* ［**secondary**］］ 例： `Switch(config-if)# standby 1 ip`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成（またはイネーブルに）します。 • （任意）group-number：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）ip-address：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入

	コマンドまたはアクション	目的
		力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 5	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります
ステップ 6	**show standby** [*interface-id* [*group*]] 例： `Switch # show standby`	スタンバイ グループの設定を確認します。
ステップ 7	**copy running-config startup-config** 例： `Switch# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 用 HSRP グループの動作のイネーブル化と確認

この作業では、**standby ipv6** コマンドを入力すると、リンクローカルプレフィクスからリンクローカルアドレスが生成され、変更後の EUI-64 形式のインターフェイス識別子が生成されます。EUI-64 インターフェイス識別子は、関連する HSRP 仮想 MAC アドレスからこの形式で作成されます。

リンクローカル アドレスは、リンクローカル プレフィックス FE80::/10（1111 1110 10）と変更された EUI-64 形式のインターフェイス識別子を使用するすべてのインターフェイスを自動的に設定できる 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、ステートレス自動設定プロセスで使用されます。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

使用して通信できます。ノードの通信にサイトローカルアドレスまたはグローバルに一意のアドレスは不要です。

IPv6 では、リンク上のデバイスが RA メッセージでサイトローカル プレフィックスやグローバル プレフィックス、およびリンクのデフォルト デバイスとして動作することをアドバタイズします。RA メッセージは、定期的に送信される場合と、システム始動時にホストから送信されるルータ送信要求メッセージに対する応答として送信される場合があります。

リンク上のノードは、RA メッセージに含まれるプレフィックス（64 ビット）にそのインターフェイス ID（64 ビット）を付加して、自動的にサイトローカルアドレスとグローバル IPv6 アドレスを設定できます。ノードによって設定された 128 ビットの IPv6 アドレスは、重複アドレス検出の対象となり、リンク上での一意性が確保されます。RA メッセージでアドバタイズされたプレフィックスがグローバルに一意である場合、ノードによって設定された IPv6 アドレスもグローバルに一意になります。ICMP パケット ヘッダーのタイプ フィールドの値が 133 であるルータ送信要求メッセージは、システム始動時にホストによって送信されるため、ホストは次のスケジュールされた RA メッセージを待機することなくすぐに自動設定できます。

IPv6 の HSRP グループを有効にして確認するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 unicast-routing** 例： `Device(config)# ipv6 unicast-routing`	IPv6 ユニキャスト データグラムの転送を有効にします。 • HSRP for IPv6 を機能させるには、**ipv6 unicast-routing** コマンドを有効にする必要があります。
ステップ 4	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 0/0/0`	インターフェイスのタイプと番号を指定し、デバイスをインターフェイス コンフィギュレーション モードにします。
ステップ 5	**standby** [*group-number*] **ipv6** {*link-local-address* | **autoconfig**} 例：	IPv6 の HSRP をアクティブにします。

	コマンドまたはアクション	目的
	`Device(config-if)# standby 1 ipv6 autoconfig`	
ステップ 6	**standby** [*group-number*] **preempt** [**delay minimum** *seconds* | **reload** *seconds* | **sync** *seconds*] 例： `Device(config-if)# standby 1 preempt`	HSRP プリエンプションとプリエンプション遅延を設定します。
ステップ 7	**standby** [*group-number*] **priority** *priority* 例： `Device(config-if)# standby 1 priority 110`	HSRP プライオリティを設定します。
ステップ 8	**exit** 例： `Device(config-if)# exit`	デバイスを特権 EXEC モードに戻します。
ステップ 9	**show standby** [*type number* [*group*]] [**all** | **brief**] 例： `Device# show standby`	HSRP 情報を表示します。
ステップ 10	**show ipv6 interface** [**brief**] [*interface-type interface-number*] [**prefix**] 例： `Device# show ipv6 interface GigabitEthernet 0/0/0`	IPv6 向けに設定されたインターフェイスの使用状況を表示します。

HSRP のプライオリティの設定

standby priority, **standby preempt**、および **standby track** インターフェイス コンフィギュレーション コマンドはいずれも、アクティブ ルータとスタンバイ ルータを検索するための特性、および新しいアクティブ ルータが処理を引き継いだ場合の動作を設定するために使用できます。

HSRP プライオリティを設定する場合の注意事項は、次のとおりです。

- プライオリティを割り当てておくと、アクティブ ルータおよびスタンバイ ルータを選択できます。プリエンプションがイネーブルの場合は、プライオリティが最高のルータがア

クティブルータになります。プライオリティが等しい場合は、現在アクティブなルータに変更はありません。

- 最大の値（1 ～ 255）が、最高のプライオリティ（アクティブ ルータになる確率が最も高い）を表します。
- プライオリティ、プリエンプト、またはその両方を設定するときは、少なくとも 1 つのキーワード（**priority**、**preempt**、または両方）を指定する必要があります。
- インターフェイスが **standby track** コマンドによって設定されている場合、ルータ上の別のインターフェイスがダウンすると、デバイスのプライオリティが動的に変更されることもあります。
- **standby track** インターフェイス コンフィギュレーション コマンドを実行すると、ルータのホットスタンバイ プライオリティとインターフェイスのアベイラビリティが関連付けられます。この機能は、HSRP 用に設定されていないインターフェイスを追跡する場合に有効です。追跡対象のインターフェイスが故障すると、トラッキングが設定されているデバイスのホットスタンバイ プライオリティが 10 減少します。追跡対象でないインターフェイスの場合は、そのステートが変わっても、設定済みデバイスのホットスタンバイ プライオリティは変わりません。ホットスタンバイ用に設定されたインターフェイスごとに、追跡するインターフェイスのリストを個別に設定できます。
- **standby track** *interface-priority* インターフェイス コンフィギュレーション コマンドを実行すると、追跡対象のインターフェイスがダウンした場合のホットスタンバイ優先順位の減少幅を指定できます。インターフェイスが稼働状態に戻ると、プライオリティは同じ分だけ増加します。
- *interface-priority* 値が設定されている場合に、複数の追跡対象インターフェイスがダウンすると、設定済みプライオリティの減少幅が累積されます。プライオリティ値が設定されていない追跡対象インターフェイスが故障した場合、デフォルトの減少幅は 10 です。この値は累積されません。
- インターフェイスに対してルーティングを最初にイネーブルにした時点で、完全なルーティング テーブルは存在しません。このインターフェイスがプリエンプトに設定されている場合はアクティブ ルータになりますが、十分なルーティング処理はできません。この問題を解決するには、ルータがルーティング テーブルを更新できるように遅延時間を設定します。

インターフェイスに HSRP プライオリティ特性を設定するには、特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface gigabitethernet1/0/1`	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、プライオリティを設定する HSRP インターフェイスを入力します。
ステップ 3	**standby** ［*group-number*］ **priority***priority* 例： `Switch(config-if)# standby 120 priority 50`	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 • （任意）group-number：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 4	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum***seconds*］ ［**reload***seconds*］ ［**sync***seconds*］］ 例： `Switch(config-if)# standby 1 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引

	コマンドまたはアクション	目的
		き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 5	**standby** ［*group-number*］ **track** *type number*［*interface-priority*］ 例： `Switch(config-if)# standby track interface gigabitethernet1/1/1`	他のインターフェイスを追跡するようにインターフェイスを設定します。この設定により、他のインターフェイスの 1 つがダウンした場合は、そのデバイスのホットスタンバイ プライオリティが減少します。 • （任意）group-number：コマンドが適用されるグループ番号です。 • type：追跡対象のインターフェイス タイプを（インターフェイス番号とともに）入力します。 • number：追跡対象のインターフェイス番号を（インターフェイス タイプとともに）入力します。 • （任意）interface-priority：インターフェイスがダウンした場合、または稼働状態に戻った場合に、ルータのホットスタンバイ プライオリティを減少または増加させる幅を入力します。デフォルト値は 10 です。
ステップ 6	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 8	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MHSRP の設定

MHSRP およびロード バランシングをイネーブルにするには、MHSRP の項の *MHSRP* ロード シェアリングの図に示したように、グループのアクティブ ルータとして 2 つのルータを設定し、スタンバイ ルータとして仮想ルータを設定します。ルータに障害が発生して正常に戻った場合、プリエンプションを発生させてロードバランシングを復元するために、**standby preempt** インターフェイス コンフィギュレーション コマンドをそれぞれの HSRP インターフェイスで入力する必要があります。

ルータ A はグループ 1 のアクティブ ルータとして、ルータ B はグループ 2 のアクティブ ルータとして設定されています。ルータ A の HSRP インターフェイスの IP アドレスは 10.0.0.1、グループ 1 のスタンバイ プライオリティは 110（デフォルトは 100）です。ルータ B の HSRP インターフェイスの IP アドレスは 10.0.0.2、グループ 2 のスタンバイ プライオリティは 110 です。

グループ 1 は仮想 IP アドレス 10.0.0.3 を使用し、グループ 2 は仮想 IP アドレス 10.0.0.4 を使用します。

ルータ A の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type number* 例： `Switch (config)# interface gigabitethernet1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**no switchport** 例： `Switch (config)# no switchport`	レイヤ 2 モードになっているインターフェイスを、レイヤ 3 設定用にレイヤ 3 モードに切り替えます。
ステップ 4	**ip address** *ip-address mask* 例： `Switch (config-if)# ip address 10.0.0.1 255.255.255.0`	インターフェイスの IP アドレスを指定します。
ステップ 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 1 ip 10.0.0.3`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイス

	コマンドまたはアクション	目的
		のグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想IPアドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想IPアドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IPアドレスがセカンダリ ホットスタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IPアドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IPアドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 6	**standby** ［*group-number*］ **priority** *priority* 例： `Switch(config-if)# standby 1 priority 110`	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 • （任意）*group-number*：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 7	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ

	コマンドまたはアクション	目的
	例： `Switch(config-if)# standby 1 preempt delay 300`	ルータよりも高い場合は、アクティブルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカルルータがアクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカルルータがリロードの後アクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません） 。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカルルータがアクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 8	**standby** ［*group-number*］**ip** ［*ip-address* ［**secondary**］］ 例： `Switch (config-if)# standby 2 ip 10.0.0.4`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グルー

	コマンドまたはアクション	目的
		プが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想IPアドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想IPアドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 9	**standby** ［*group-number*］**preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： `Switch(config-if)# standby 2 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600

	コマンドまたはアクション	目的
		（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 10	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 12	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ルータ B の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type number* 例： `Switch (config)# interface gigabitethernet1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**no switchport** 例： `Switch (config)# no switchport`	レイヤ 2 モードになっているインターフェイスを、レイヤ 3 設定用にレイヤ 3 モードに切り替えます。

	コマンドまたはアクション	目的
ステップ 4	**ip address** *ip-address mask* 例： `Switch (config-if)# ip address 10.0.0.2 255.255.255.0`	インターフェイスの IP アドレスを指定します。
ステップ 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 1 ip 10.0.0.3`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 6	**standby** [*group-number*] **priority** *priority* 例：	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大

	コマンドまたはアクション	目的
	Switch(config-if)# **standby 2 priority 110**	の値が、最高のプライオリティを表します。 • （任意）*group-number*：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 7	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： Switch(config-if)# **standby 1 preempt delay 300**	ルータを **preempt** に設定し、ローカルルータのプライオリティがアクティブルータよりも高い場合は、アクティブルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカルルータがアクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカルルータがリロードの後アクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカルルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。

	コマンドまたはアクション	目的
ステップ 8	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 2 ip 10.0.0.4`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホットスタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 9	**standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] 例： `Switch(config-if)# standby 2 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役

	コマンドまたはアクション	目的
		割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません） 。 • （任意）**delay reload**：ローカルルータがリロードの後アクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカルルータがアクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 10	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 12	**copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

HSRP 認証およびタイマーの設定

HSRP 認証ストリングを設定したり、hello タイムインターバルやホールドタイムを変更することもできます。

これらの属性を設定する場合の注意事項は次のとおりです。

- 認証ストリングはすべての HSRP メッセージで暗号化されずに送信されます。相互運用できるように、接続されたすべてのルータおよびアクセス サーバーに同じ認証ストリングを設定する必要があります。認証ストリングが一致しないと、HSRP によって設定された他のルータから、指定されたホットスタンバイ IP アドレスおよびタイマー値を学習できません。
- スタンバイ タイマー値が設定されていないルータまたはアクセス サーバーは、アクティブ ルータまたはスタンバイ ルータからタイマー値を学習できます。アクティブ ルータに設定されたタイマーは、常に他のタイマー設定よりも優先されます。
- ホットスタンバイ グループのすべてのルータで、同じタイマー値を使用する必要があります。通常、*holdtime* は *hellotime* の 3 倍以上です。

インターフェイスに HSRP の認証とタイマーを設定するには、特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config) # interface gigabitethernet1/0/1`	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、プライオリティを設定する HSRP インターフェイスを入力します。
ステップ 3	**standby** [*group-number*] **authentication** *string* 例： `Switch(config-if) # standby 1 authentication word`	（任意）**authentication** *string*：すべての HSRP メッセージで伝達されるストリングを入力します。認証ストリングには 8 文字までを指定できます。デフォルトのストリングは **cisco** です。 （任意）*group-number*：コマンドが適用されるグループ番号です。
ステップ 4	**standby** [*group-number*] **timers** *hellotime holdtime* 例： `Switch(config-if) # standby 1 timers 5 15`	（任意）hello パケット間隔、およびアクティブ ルータのダウンを他のルータが宣言するまでの時間を設定します。 • *group-number*：コマンドが適用されるグループ番号です。 • *hellotime*：連続する hello パケット間のインターバルを秒単位で設定します。範囲は、1 ～ 255 秒です。デフォルトは 3 です。

	コマンドまたはアクション	目的
		• *holdtime*：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は0～3600（1時間）で、デフォルトは0です（リロードの後、引き継ぐ前の遅延はありません）。
ステップ5	**end** 例： `Switch(config-if) # end`	特権 EXEC モードに戻ります。
ステップ6	**show running-config**	スタンバイ グループの設定を確認します。
ステップ7	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ICMP リダイレクト メッセージの HSRP サポートのイネーブル化

HSRP が設定されたインターフェイスでは、ICMP リダイレクト メッセージが自動的にイネーブルになります。ICMP は、エラーをレポートするためのメッセージ パケットや IP 処理に関連する他の情報を提供する、ネットワーク層インターネット プロトコルです。ICMP には、ホストへのエラー パケットの方向付けや送信などの診断機能があります。この機能は、HSRP を介した発信 ICMP リダイレクト メッセージをフィルタリングします。HSRP では、ネクストホップ IP アドレスが HSRP 仮想 IP アドレスに変更される可能性があります。詳細については、『Cisco IOS IP Configuration Guide, Release 12.4』を参照してください。

HSRP グループおよびクラスタリングの設定

デバイスが HSRP スタンバイ ルーティングに参加し、クラスタリングがイネーブルの場合は、同じスタンバイ グループを使用して、コマンド スイッチの冗長性および HSRP の冗長性を確保できます。同じ HSRP スタンバイ グループをイネーブルにし、コマンド スイッチおよびルーティングの冗長性を確保するには、**cluster standby-group** *HSRP-group-name* [**routing-redundancy**] グローバル コンフィギュレーション コマンドを使用します。**routing-redundancy** キーワードを指定せずに同じ HSRP スタンバイ グループ名でクラスタを作成すると、そのグループに対する HSRP スタンバイ ルーティングはディセーブルになります。

HSRP コンフィギュレーションの確認

HSRP 設定を表示するには、次の特権 EXEC モードで次のコマンドを使用します。

show standby ［*interface-id*［*group*］］［**brief**］［**detail**］

スイッチ全体、特定のインターフェイス、HSRP グループ、またはインターフェイスの HSRP グループに関する HSRP 情報を表示できます。HSRP 情報の概要または詳細のいずれを表示するかを指定することもできます。デフォルトの表示は **detail** です。多数の HSRP グループがある場合に、修飾子を指定しないで **show standby** コマンドを使用すると、正確に表示されないことがあります。

例

```
Switch #show standby
VLAN1 - Group 1
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.182
Hot standby IP address is 172.20.128.3 configured
Active router is 172.20.128.1 expires in 00:00:09
Standby router is local
Standby virtual mac address is 0000.0c07.ac01
Name is bbb

VLAN1 - Group 100
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.262
Hot standby IP address is 172.20.138.51 configured
Active router is 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

ホット スタンバイ ルータ プロトコルの設定例

ここでは、HSRP のさまざまな設定例について説明します。

HSRP のイネーブル化：例

次に、インターフェイスのグループ 1 で HSRP をアクティブにする例を示します。ホットスタンバイ グループで使用される IP アドレスは、HSRP を使用して学習されます。

（注） これは、HSRP をイネーブルにするために必要な最小限の手順です。その他の設定は任意です。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch # show standby
```

例：HSRP グループの設定と確認

次に、デバイス 1 とデバイス 2 で構成される IPv6 用 HSRP グループの設定および確認の例を示します。デバイスの設定を確認するため、各デバイスに対して **show standby** コマンドが発行されます。

デバイス 1 の設定

```
interface FastEthernet0/0.100
description DATA VLAN for PCs
encapsulation dot1Q 100
ipv6 address 2001:DB8:CAFE:2100::BAD1:1010/64
standby version 2
standby 101 priority 120
standby 101 preempt delay minimum 30
standby 101 authentication ese
standby 101 track Serial0/1/0.17 90
standby 201 ipv6 autoconfig
standby 201 priority 120
standby 201 preempt delay minimum 30
standby 201 authentication ese
standby 201 track Serial0/1/0.17 90
Device1# show standby
FastEthernet0/0.100 - Group 101 (version 2)
State is Active
2 state changes, last state change 5w5d
Active virtual MAC address is 0000.0c9f.f065
Local virtual MAC address is 0000.0c9f.f065 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 2.296 secs
Authentication text "ese"
Preemption enabled, delay min 30 secs
Active router is local
Priority 120 (configured 120)
Track interface Serial0/1/0.17 state Up decrement 90
IP redundancy name is "hsrp-Fa0/0.100-101" (default)
FastEthernet0/0.100 - Group 201 (version 2)
State is Active
2 state changes, last state change 5w5d
Virtual IP address is FE80::5:73FF:FEA0:C9
Active virtual MAC address is 0005.73a0.00c9
Local virtual MAC address is 0005.73a0.00c9 (v2 IPv6 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 2.428 secs
Authentication text "ese"
Preemption enabled, delay min 30 secs
Active router is local
Standby router is FE80::20F:8FFF:FE37:3B70, priority 100 (expires in 7.856 sec)
Priority 120 (configured 120)
Track interface Serial0/1/0.17 state Up decrement 90
IP redundancy name is "hsrp-Fa0/0.100-201" (default)
```

デバイス 2 の設定

```
interface FastEthernet0/0.100
description DATA VLAN for Computers
encapsulation dot1Q 100
ipv6 address 2001:DB8:CAFE:2100::BAD1:1020/64
standby version 2
standby 101 preempt
standby 101 authentication ese
standby 201 ipv6 autoconfig
standby 201 preempt
standby 201 authentication ese
Device2# show standby
FastEthernet0/0.100 - Group 101 (version 2)
State is Standby
7 state changes, last state change 5w5d
Active virtual MAC address is 0000.0c9f.f065
Local virtual MAC address is 0000.0c9f.f065 (v2 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 0.936 secs
Authentication text "ese"
Preemption enabled
MAC address is 0012.7fc6.8f0c
Standby router is local
Priority 100 (default 100)
IP redundancy name is "hsrp-Fa0/0.100-101" (default)
FastEthernet0/0.100 - Group 201 (version 2)
State is Standby
7 state changes, last state change 5w5d
Virtual IP address is FE80::5:73FF:FEA0:C9
Active virtual MAC address is 0005.73a0.00c9
Local virtual MAC address is 0005.73a0.00c9 (v2 IPv6 default)
Hello time 3 sec, hold time 10 sec
Next hello sent in 0.936 secs
Authentication text "ese"
Preemption enabled
Active router is FE80::212:7FFF:FEC6:8F0C, priority 120 (expires in 7.548 sec)
MAC address is 0012.7fc6.8f0c
Standby router is local
Priority 100 (default 100)
IP redundancy name is "hsrp-Fa0/0.100-201" (default)
```

HSRP のプライオリティの設定：例

次に、ポートをアクティブにして、IP アドレスおよびプライオリティ 120（デフォルト値よりも高いプライオリティ）を設定して、アクティブ ルータになるまで 300 秒（5 分間）待機する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby ip 172.20.128.3
Switch(config-if)# standby priority 120 preempt delay 300
Switch(config-if)# end
Switch # show standby
```

MHSRP の設定：例

次に、*MHSRP* ロード シェアリングの図で示した MHSRP 設定をイネーブルにする例を示します。

ルータ A の設定

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.1 255.255.255.0
Switch(config-if)# standby ip 10.0.0.3
Switch(config-if)# standby 1 priority 110
Switch(config-if)#  standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)#  standby 2 preempt
Switch(config-if)# end
```

ルータ B の設定

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.2 255.255.255.0
Switch(config-if)# standby ip 10.0.0.3
Switch(config-if)#  standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 2 priority 110
Switch(config-if)#  standby 2 preempt
Switch(config-if)# end
```

HSRP 認証およびタイマーの設定：例

次に、グループ 1 のホットスタンバイ ルータを相互運用させるために必要な認証ストリングとして、word を設定する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 authentication word
Switch(config-if)# end
```

次に、hello パケット間隔が 5 秒、ルータがダウンしたと見なされるまでの時間が 15 秒となるように、スタンバイ グループ 1 のタイマーを設定する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

HSRP グループおよびクラスタリングの設定：例

次に、スタンバイ グループ my_hsrp をクラスタにバインドし、同じ HSRP グループをイネーブルにしてコマンド スイッチおよびルータの冗長性に使用する例を示します。このコマンドを実行できるのは、コマンド スイッチに対してだけです。スタンバイ グループの名前または番号が存在しない場合、またはスイッチがクラスタ メンバー スイッチである場合は、エラー メッセージが表示されます。

```
Switch # configure terminal
Switch(config) # cluster standby-group my_hsrp routing-redundancy
Switch(config-if)# end
```

HSRP の設定に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	*Command Reference (Catalyst 9300 Series Switches)*

標準および RFC

標準/RFC	タイトル
RFC 2281	『Cisco Hot Standby Router Protocol』

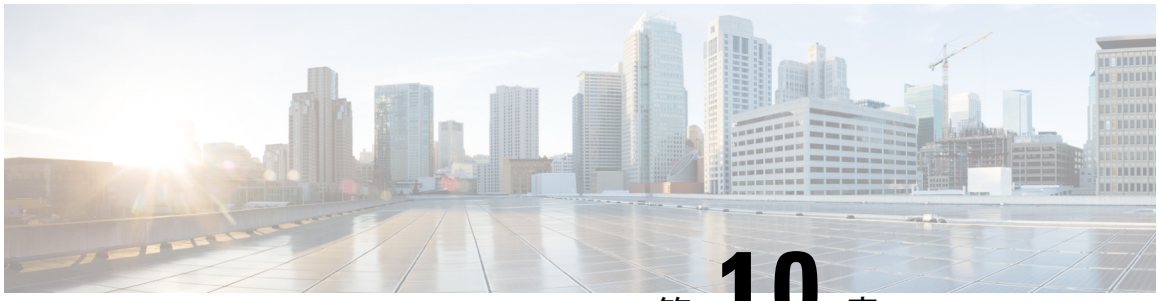
HSRP の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	HSRP	HSRP は、デフォルト ゲートウェイ IP アドレスが設定された IEEE 802 LAN 上の IP ホストにファースト ホップ冗長性を確保することでネットワークのアベイラビリティを高めるシスコの標準方式です。
Cisco IOS XE Fuji 16.8.1a	HSRP for IPv6	HSRP は、ファーストホップ IPv6 ルータの透過的なフェールオーバーを可能にする FHRP です。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 10 章

NHRP の設定

Next Hop Resolution Protocol に関する情報

ここでは、Next Hop Resolution Protocol（NHRP）について説明します。

NHRP および NBMA のネットワークの相互作用

WAN ネットワークのほとんどは、ポイントツーポイント リンクの集まりです。仮想トンネル ネットワーク（総称ルーティング カプセル化（GRE）トンネルなど）もまた、ポイントツーポイント リンクの集まりです。これらのポイントツーポイント リンクの接続を効率的にスケーリングするために、通常は、単一またはマルチレイヤのハブアンドスポーク ネットワークにグループ化します。マルチポイント インターフェイス（GRE トンネル インターフェイスなど）を使用して、このようなネットワークのハブ ルータの設定を減らすことができます。その結果として生じるネットワークが NBMA ネットワークです。

単一のマルチポイント インターフェイスを通って到達可能なトンネル エンドポイントが複数あるため、この NBMA ネットワークを介してトンネル インターフェイスからパケットを転送するには、論理トンネル エンドポイントの IP アドレスから物理トンネル エンドポイントの IP アドレスへのマッピングが必要です。このマッピングはスタティックに設定することが可能ですが、これは、マッピングがダイナミックに検出または学習できる場合に推奨します。

NHRP は、これらの NBMA ネットワークの問題を軽減する ARP と同様のプロトコルです。NHRP を使用すると、NBMA ネットワークに接続されているシステムは、ネットワークの一部である他のシステムの NBMA アドレスをダイナミックに学習します。このため、これらのシステムは、トラフィックに中間ホップを使用せずに直接通信できるようになります。

ルータ、アクセス サーバ、およびホストは、NHRP を使用して、NBMA ネットワークに接続された他のルータおよびホストのアドレスを検出できます。部分メッシュ NBMA ネットワー

クには通常、NBMA ネットワークの背後に複数の論理ネットワークがあります。このような構成において、NBMA ネットワークを通るパケットは、出口ルータ（宛先ネットワークに最も近いルータ）に到着するまでに、NBMA ネットワーク上で複数のホップを発生させる必要がある場合があります。

NHRP 登録によって、これらの NBMA ネットワークのサポートが可能になります。

- NHRP 登録：NHRP を使用して、ネクスト ホップ クライアント（NHC）がネクスト ホップ サーバ（NHS）にダイナミックに登録されます。この登録機能により、特に、NHC がダイナミック物理 IP アドレスを持つか、物理 IP アドレスをダイナミックに変更するネットワーク アドレス変換（NAT）ルータの背後にある場合には、NHS で設定を変更しなくても、NHC が NBMA ネットワークに参加できるようになります。この場合、NHC の論理（VPN IP アドレス）と物理（NBMA IP）のマッピングを NHS で事前に設定することができません。

ダイナミックに構築されたハブアンドスポーク ネットワーク

NHRP により、NBMA ネットワークは最初、スポークの NHC とハブの NHS から複数の階層レイヤを構成できるハブアンドスポーク ネットワークとして配置されます。NHC は、NHS に到達するためのスタティック マッピング情報を使用して設定され、NHS に接続して NHRP 登録を NHS に送信します。この設定により、NHS はスポークのマッピング情報をダイナミックに学習できるため、ハブで必要な設定が減り、さらにスポークでダイナミック NBMA（物理）IP アドレスを取得できるようになります。

Next Hop Resolution Protocol の設定方法

ここでは、NHRP に関する設定情報について説明します。

インターフェイス上での NHRP のイネーブル化

スイッチ上のインターフェイスに対して NHRP をイネーブルにするには、次の作業を行います。一般に、論理 NBMA ネットワーク内のすべての NHRP ステーションは、同じネットワーク ID を使用して設定する必要があります。

2 つ以上の NHRP ドメイン（GRE トンネル インターフェイス）が同じ NHRP ノード（スイッチ）で使用可能な場合は、NHRP ネットワーク ID を使用して、NHRP インターフェイスの NHRP ドメインを定義し、複数の NHRP ドメイン間またはネットワーク間で区別します。NHRP ネットワーク ID を使用すると、2 つの NHRP ネットワーク（クラウド）を同じスイッチ上に設定する場合に、それぞれを分けるのに役立ちます。

NHRP ネットワーク ID はローカル専用のパラメータです。これは、ローカル スイッチだけに対して意味があり、NHRP パケットで他の NHRP ノードに送信されることはありません。この理由から、2 台のスイッチが同じ NHRP ドメインに存在する場合、スイッチで設定される NHRP ネットワーク ID の実際の値は、もう一方のスイッチの NHRP ネットワーク ID と一致する必要はありません。NHRP パケットが GRE インターフェイス上に到着すると、そのインターフェ

イスで設定されている NHRP ネットワーク ID のローカル NHRP ドメインに割り当てられます。

同じ NHRP ネットワークに存在するすべてのスイッチ上の GRE インターフェイスでは、同じ NHRP ネットワーク ID を使用することを推奨します。こうすると、どの GRE インターフェイスがどの NHRP ネットワークのメンバであるかを追跡しやすくなります。

NHRP ドメイン（ネットワーク ID）は、スイッチ上の各 GRE トンネル インターフェイスで固有に設定できます。NHRP ドメインは、ルート上の GRE トンネル インターフェイス間をまたぐことができます。この場合、GRE トンネル インターフェイスで同じ NHRP ネットワーク ID を使用する効果は、2 つの GRE インターフェイスが単一の NHRP ネットワークに統合されることです。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Switch(config)# interface tunnel 100`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address network-mask* 例： `Switch(config-if)# ip address 10.0.0.1 255.255.255.0`	IP をイネーブルにし、インターフェイスに IP アドレスを提供します。
ステップ 5	**ip nhrp network-id** *number* 例： `Switch(config-if)# ip nhrp network-id 1`	インターフェイスで NHRP を有効にします。
ステップ 6	**end** 例： `Switch(config)# end`	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、特権 EXEC モードに戻ります。

マルチポイント動作のための GRE トンネルの設定

マルチポイント（NMBA）動作のためのGREトンネルを設定するには、次の作業を行います。

マルチポイント トンネル インターフェイスのトンネル ネットワークは、NBMA ネットワークと見なすことができます。同じスイッチ上で複数の GRE トンネルを設定する場合は、固有のトンネル ID キーまたは固有のトンネル送信元アドレスのいずれかを持っている必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Switch(config)# interface tunnel 100`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address* 例： `Switch(config-if)# ip address 172.16.1.1 255.255.255.0`	インターフェイスにIPアドレスを設定します。
ステップ 5	**ip mtu** *bytes* 例： `Switch(config-if)# ip mtu 1400`	各インターフェイスにおいて送信されるIPパケットの最大伝送単位（MTU）サイズを設定します。
ステップ 6	**ip pim sparse-dense-mode** 例： `Switch(config-if)# ip pim sparse-dense-mode`	インターフェイスでProtocol Independent Multicast（PIM）をイネーブルにし、マルチキャスト グループの動作モードに応じて、インターフェイスをスパース モード動作またはデンス モード動作で処理します。
ステップ 7	**ip nhrp map** *ip-address nbma-address* 例：	非ブロードキャスト マルチアクセス（NBMA）ネットワークに接続する宛

	コマンドまたはアクション	目的
	`Switch(config-if)# ip nhrp map 172.16.1.2 10.10.10.2`	先 IP アドレスの IP/NBMA アドレス マッピングをスタティックに設定します。 • *ip-address*：NBMA ネットワークを介して到達可能な宛先の IP アドレス。このアドレスは、NBMA アドレスにマッピングされます。 • *nbma-address*：NBMA ネットワークを介して直接到達可能な NBMA アドレス。アドレスの形式は、使用しているメディアによって異なります。たとえば、ATM はネットワーク サービス アクセス ポイント（NSAP）アドレスを所有し、イーサネットは MAC アドレスを所有し、Switched Multimegabit Data Service（SMDS）は E.164 アドレスを所有しています。このアドレスは、IP アドレスにマッピングされます。
ステップ 8	**ip nhrp map multicast** *nbma-address* 例： `Switch(config-if)# ip nhrp map multicast 10.10.10.2`	ブロードキャストの接続先として、またはトンネルネットワークを介して送信されるマルチキャスト パケットとして使用されるノンブロードキャスト マルチアクセス（NBMA）アドレスを設定します。
ステップ 9	**ip nhrp network-id** *number* 例： `Switch(config-if)# ip nhrp network-id 1`	インターフェイスで Next Hop Resolution Protocol（NHRP）を有効にします。 • *number*：非ブロードキャスト マルチアクセス（NBMA）ネットワークからの、グローバルに一意である 32 ビットのネットワーク ID。範囲は 1 ～ 4294967295 です。
ステップ 10	**ip nhrp nhs** *nhs-address* 例： `Switch(config-if)# ip nhrp nhs 172.16.1.2`	1 つ以上の NHRP サーバのアドレスを指定します。 • *nhs-address*：指定したネクストホップ サーバのアドレス。

	コマンドまたはアクション	目的
ステップ 11	**tunnel source vlan** *interface-number* 例： `Switch(config-if)# tunnel source vlan 1`	トンネルインターフェイスの送信元アドレスを設定します。
ステップ 12	**tunnel destination** *ip-address* 例： `Switch(config-if)# tunnel destination 10.10.10.2`	トンネルインターフェイスの宛先アドレスを設定します。
ステップ 13	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

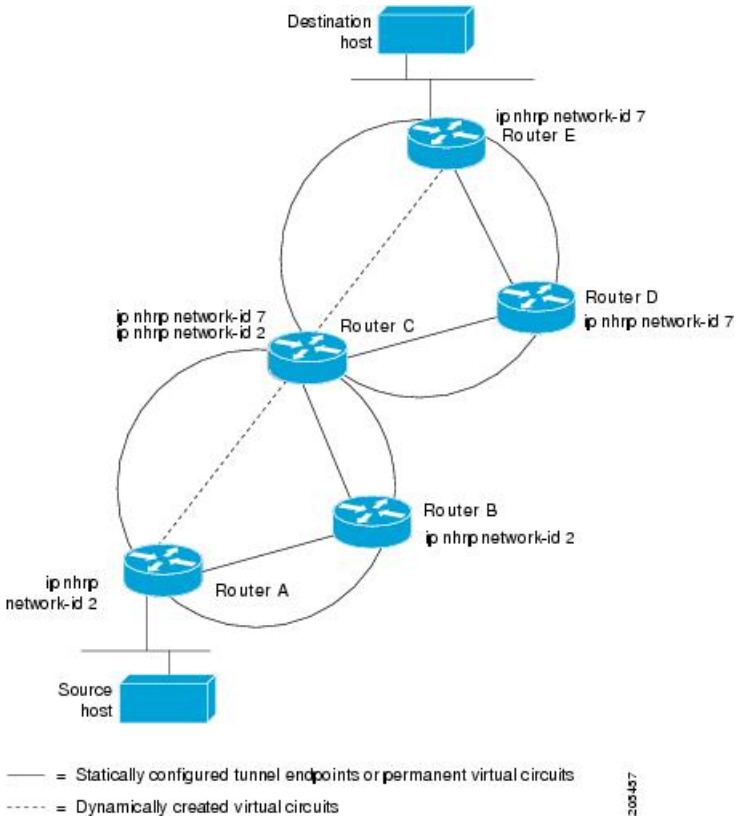
Next Hop Resolution Protocol の設定例

ここでは、NHRP のさまざまな設定例について説明します。

論理 NBMA の物理ネットワーク設計の例

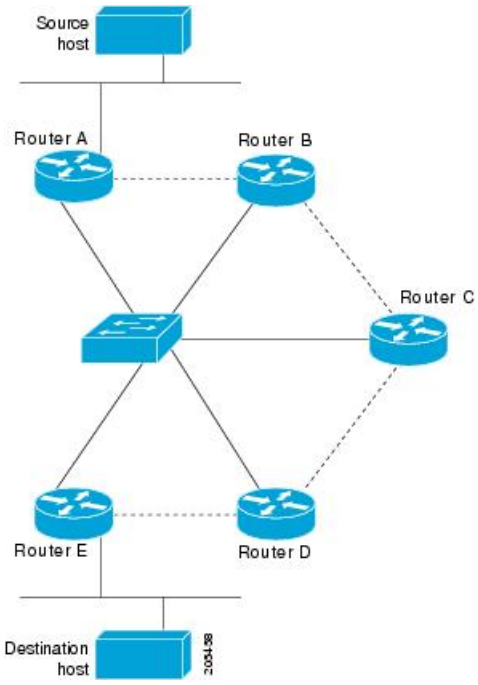
論理 NBMA ネットワークは、NHRP に参加し、同じネットワーク ID を持つインターフェイスおよびホストのグループと考えられます。次の図に、単一の物理 NBMA ネットワーク上に設定された（円で示される）2 つの論理 NBMA ネットワークを示します。ルータ A はルータ B およびルータ C と通信できます。それらが同じネットワーク ID（2）を共有するためです。また、ルータ C はルータ D およびルータ E とも通信できます。それらがネットワーク ID 7 を共有するためです。アドレス解決が完了した後、点線で示すように、ルータ A は IP パケットをホップ 1 回でルータ C に送信でき、ルータ C はそれをホップ 1 回でルータ E に送信できます。

図 *13 : 1* つの物理 *NBMA* ネットワーク上の *2* つの論理 *NBMA* ネットワーク



上図の 5 台のルータによる物理構成は、実際には下図のような構成である場合もあります。送信元ホストはルータ A に接続されており、宛先ホストはルータ E に接続されています。同じスイッチが 5 つのすべてのルータにサービスを提供し、1 つの物理 NBMA ネットワークを構成しています。

図 *14 : NBMA* ネットワーク例の物理構成



ここでも、上の最初の図を参照してください。最初、送信元ホストから宛先ホストへの IP パケットは、NHRP が NBMA アドレスでも解決するようになるまで、スイッチに接続された 5 台すべてのルータを通過して宛先に到達します。ルータ A は、IP パケットを初めて宛先ホストに向けて転送したときに、宛先ホストの IP アドレスに対する NHRP 要求も生成します。その要求がルータ C に転送され、応答が生成されます。2 つの論理 NBMA ネットワーク間の出力ルータであるため、ルータ C が応答します。

同様に、ルータ C は独自の NHRP 要求を生成し、これに対して、ルータ E が応答します。この例でも、送信元と宛先の間に発生する IP トラフィックが NBMA ネットワークを通過するためには、2 回のホップが必要です。これは、2 つの論理 NBMA ネットワーク間で IP トラフィックを転送する必要があるためです。NBMA ネットワークが論理的に分かれていなければ、必要なホップは 1 回だけです。

例：マルチポイント動作のための GRE トンネル

マルチポイント トンネルを使用すると、単一のトンネル インターフェイスを複数のネイバー スイッチに接続できます。ポイントツーポイント トンネルとは異なり、トンネルの宛先を設定する必要がありません。実際に、設定したとしても、トンネルの宛先は IP マルチキャスト アドレスに対応させる必要があります。

次の例では、スイッチ A とルータ B がイーサネット セグメントを共有しています。マルチポイント トンネル ネットワーク上で最小の接続が設定されるため、部分メッシュ NBMA ネットワークとして扱うことができるネットワークが作成されます。スタティック NHRP マップ エントリにより、スイッチ A はスイッチ B への到達方法を理解していて、その逆も同様です。

次に、GRE マルチポイント トンネルを設定する例を示します。

スイッチ A の設定

```
Switch(config)# interface tunnel 100 !Tunnel interface configured for PIM traffic
Switch(config-if)# no ip redirects
Switch(config-if)# ip address 192.168.24.1 255.255.255.252
Switch(config-if)# ip mtu 1400
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip nhrp map 192.168.24.3 172.16.0.1 !NHRP may optionally be configured
 to dynamically discover tunnel end points.
Switch(config-if)# ip nhrp map multicast 172.16.0.1
Switch(config-if)# ip nhrp network-id 1
Switch(config-if)# ip nhrp nhs 192.168.24.3
Switch(config-if)# tunnel source vlan 1
Switch(config-if)# tunnel destination 172.16.0.1
Switch(config-if)# end
```

スイッチ B の設定

```
Switch(config)# interface tunnel 100
Switch(config-if)# no ip redirects
Switch(config-if)# ip address 192.168.24.2 255.255.255.252
Switch(config-if)# ip mtu 1400
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip nhrp map 192.168.24.4 10.10.0.3
Switch(config-if)# ip nhrp map multicast 10.10.10.3
Switch(config-if)# ip nhrp network-id 1
Switch(config-if)# ip nhrp nhs 192.168.24.4
Switch(config-if)# tunnel source vlan 1
Switch(config-if)# tunnel destination 10.10.10.3
Switch(config-if)# end
```

NHRP の設定に関する追加情報

RFC

RFC	タイトル
RFC 2332	*『NBMA Next Hop Resolution Protocol (NHRP)』*

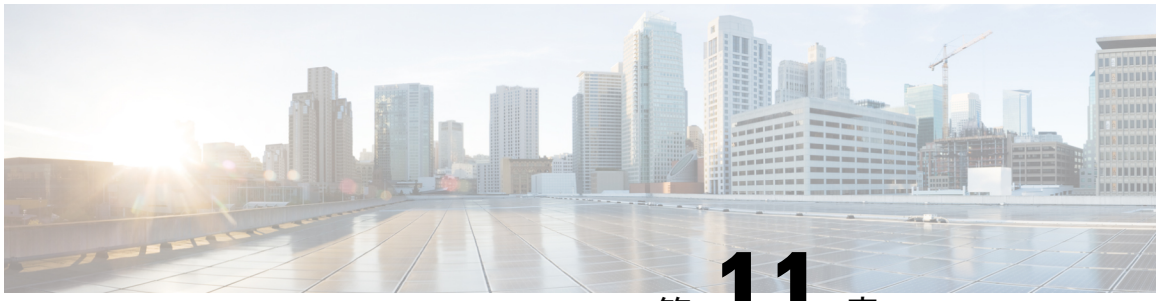
Next Hop Resolution Protocol の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	Next Hop Resolution Protocol：ネクストホップ リソリューション プロトコル	NHRP は、すべてのトンネルエンドポイントを手動で設定する代わりに、NBMA ネットワークを動的にマッピングする ARP のようなプロトコルです。NHRP を使用すると、NBMA ネットワークに接続されたシステムは、そのネットワークに参加している他のシステムの NBMA（物理）アドレスをダイナミックに学習でき、これらのシステムが直接通信できるようになります。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 11 章

ネットワーク アドレス変換の設定

ネットワークアドレス変換に関する情報

ここでは、ネットワークアドレス変換（NAT）について説明します。

Network Address Translation（NAT）

ネットワーク アドレス変換（NAT）は、IP アドレスの節約を目的として設計されています。NAT によって、未登録 IP アドレスを使用するプライベート IP ネットワークをインターネットに接続できます。NAT はデバイス（通常、2 つのネットワークを接続するもの）上で動作し、別のネットワークにパケットを転送する前に、内部ネットワークのプライベート（グローバルに一意ではない）アドレスをグローバルにルート可能なアドレスに変換します。

NAT では、外部にアドバタイズするアドレスをネットワーク全体で 1 つだけにする機能を備えています。この機能により、そのアドレスの後ろにある内部ネットワーク全体を効果的に隠すことができ、セキュリティが強化されます。NAT には、セキュリティおよびアドレス節約の二重の機能性があり、一般的にリモート アクセス環境で実装されます。

NAT は、エンタープライズ エッジでも使用され、内部ユーザーのインターネットへのアクセスを許可し、メール サーバーなど内部デバイスへのインターネット アクセスを許可します。

Cisco Catalyst 9300 シリーズ スイッチはスタックをサポートしており、NAT はスタック設定でサポートされます。

NAT の設定の利点

- IP が枯渇する問題を解決します。

組織が NAT を使用すると、既存のネットワークを持っていてインターネットにアクセスする必要がある場合に、IP アドレスが枯渇する問題を解決できます。ネットワーク インフォメーションセンター（NIC）に登録されたIPアドレスをまだ持っていないサイトは、IP アドレスを取得する必要があります。255 以上のクライアントが存在する、またはそのような環境を予定している場合は、Class Bアドレスの不足が深刻な問題になります。NAT はこのような問題に対応するために、隠された数千の内部アドレスを、取得の容易なClass C アドレスの範囲にマップします。

- クライアント IP アドレスを外部ネットワークから隠すことで、セキュリティ レイヤも提供します。

  内部ネットワークのクライアントの IP アドレスをすでに登録しているサイトでも、ハッカーがクライアントを直接攻撃できないように、これらのアドレスをインターネットから隠すことができます。クライアントアドレスを隠すことにより、セキュリティがさらに強化されます。NAT により LAN 管理者は、インターネット割り当て番号局の予備プールを利用して、Class A アドレスを自由に拡張することができます。Class A アドレスの拡張は組織内で行われ、LAN またはインターネット インターフェイスでアドレッシングの変更に配慮する必要はありません。

- Cisco ソフトウェアは、選択的、またはダイナミックに NAT を実行できます。この柔軟性により、ネットワーク管理者は RFC 1918 アドレスまたは登録したアドレスを使用することができます。

- NAT は、IP アドレスの簡略化や節約のためにさまざまなデバイス上で使用できるように設計されています。また、NAT により、変換に使用できる内部ホストを選択することもできます。

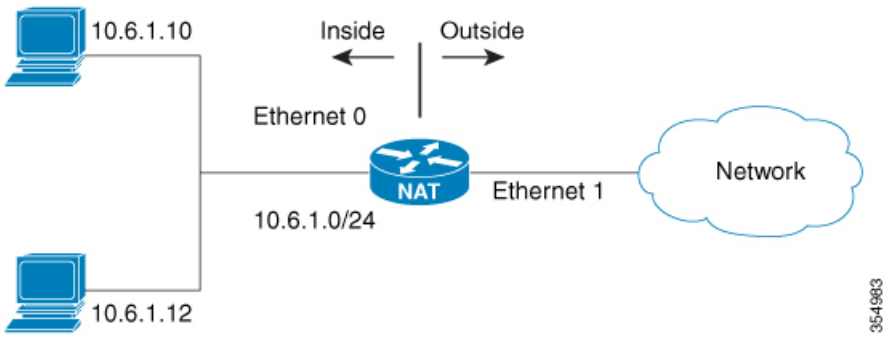
- NAT は、NAT を設定する若干のデバイス以外には、何ら変更を加えずに設定できるという大きな利点があります。

NAT の機能

NAT が設定されたデバイスには、内部ネットワークと外部ネットワークのそれぞれに接続するインターフェイスが少なくとも 1 つずつあります。標準的な環境では、NAT はスタブ ドメインとバックボーン間の出口デバイスに設定されます。パケットがドメインから出て行くとき、NAT はローカルで意味のある送信元 アドレスをグローバルで一意の アドレスに変換します。パケットがドメインに入ってくる際は、NAT はグローバルに一意な宛先アドレスをローカルアドレスに変換します。複数の内部ネットワークをデバイスに接続でき、同様にデバイスから外部ネットワークへと複数の終了ポイントが存在する場合があります。アドレスが足りなくなって、パケットにアドレスを割り当てられなくなった場合、NATはそのパケットをドロップし、Internet Control Message Protocol（ICMP）ホスト到達不能パケットをその接続先に送信します。

変換および転送は、ハードウェアのスイッチング プレーンで実行され、全体的なスループットのパフォーマンスが改善されます。パフォーマンスの詳細については、「NAT のパフォーマンスとスケール数」を参照してください。

図 *15 : NAT*



NAT の用途

NAT は次のような場合に使用できます。

- ホストのごく少数しかグローバルな一意の IP アドレスを持っていない状況でインターネットに接続する場合。

  NAT はスタブ ドメイン（内部ネットワーク）と、インターネットなどのパブリック ネットワーク（外部ネットワーク）との境界にあるデバイス上に設定されます。NAT はパケットを外部ネットワークに送信する前に、内部のローカル アドレスをグローバルに一意の IP アドレスに変換します。接続性の問題への解決策として NAT が役立つのは、スタブ ドメイン内の比較的少数のホストが同時にドメインの外部と通信する場合のみです。この場合、外部との通信が必要なときに、このドメインにある IP アドレスのごく一部をグローバルに一意な IP アドレスに変換する必要があります。また、これらのアドレスは再利用できます。

- 番号付け直し：

  内部アドレスの変更には相当の工数がかかるため、変更する代わりに NAT を使用して変換することができます。

NAT の内部アドレスおよび外部アドレス

NAT において、内部という用語は、変換が必要な組織が所有するネットワークを表します。NAT が設定されている場合、このネットワーク内のホストは、別の空間（グローバル アドレス空間として知られている）にあるものとしてネットワークの外側に現れる 1 つ空間（ローカル アドレス空間として知られている）内のアドレスを持つことになります。

同様に、外部という用語は、スタブ ネットワークの接続先で、通常、その組織の制御下にはないネットワークを表します。外部ネットワーク内のホストを変換の対象にすることもできるため、これらのホストはローカル アドレスとグローバル アドレスを持つことができます。

NAT では、次の定義が使用されます。

- 内部ローカル アドレス：内部ネットワーク上のホストに割り当てられた IP アドレス。このアドレスは、多くの場合、NIC やサービス プロバイダーにより割り当てられたルート可能な IP アドレスではありません。
- 内部グローバル アドレス：外部に向けて、1 つ以上の内部ローカル IP アドレスを表すグローバルなルート可能な IP アドレス（NIC またはサービス プロバイダーにより割り当てられたもの）。
- 外部ローカル アドレス：内部ネットワークから見た外部ホストの IP アドレス。必ずしもルート可能な IP アドレスでありません。内部でルート可能なアドレス空間から割り当てられたものです。
- 外部グローバルアドレス：外部ネットワークに存在するホストに対して、ホストの所有者により割り当てられた IP アドレス。このアドレスは、グローバルにルート可能なアドレス、またはネットワーク空間から割り当てられます。
- 内部送信元アドレス変換：内部ローカル アドレスを内部グローバル アドレスに変換します。
- 外部送信元アドレス変換：外部グローバル アドレスを外部ローカル アドレスに変換します。
- スタティック ポート変換：内部/外部ローカル アドレスの IP アドレスとポート番号を、対応する内部/外部グローバル アドレスの IP アドレスとポート番号に変換します。
- 特定のサブネットのスタティック変換：指定された内部/外部ローカル アドレスの範囲のサブネットを対応する内部/外部グローバル アドレスに変換します。
- ハーフ エントリ：ローカルおよびグローバル アドレス/ポート間のマッピングを表し、NAT モジュールの変換データベースで維持されます。ハーフ エントリは、設定されている NAT ルールに基づいて、静的または動的に作成できます。
- フル エントリ/フロー エントリ：特定のセッションに対応する一意のフローを表します。ローカルからグローバルへのマッピングに加えて、指定したフローを完全修飾する接続先情報も維持されます。フル エントリは常に動的に作成されて NAT モジュールの変換データベースで維持されます。

VRF 対応 NAT

NAT は通常、デフォルトまたはグローバル ルーティング ドメインで動作するように設定されます。この機能により、内部および外部の NAT ドメインはデフォルトの VRF スペースに関連付けられ、適宜変換が行われます。ただし、NAT が VRF 設定で実行される必要がある特定のシナリオがあります。一般的なシナリオの 1 つには、重複するアドレス空間を持つプライベートネットワークの共有サービスアクセスを有効にすることが含まれます。こうしたシナリオでは、特定のプライベートネットワークを異なる VRF に配置し、重複するプライベートアドレスを一意のグローバルアドレスにマッピングする VRF 対応 NAT ルールを設定することで、グローバルサービスアクセスを実現できます。VRF が認識されることにより、プライベートネッ

トワークの VRF が考慮に含められ、結果として NAT がアドレスおよびポートの変換を実行できるようになります。

VRF 対応 NAT では、次の変換シナリオがサポートされます。

- VRF からグローバルへの変換：この変換により、内部ドメインを特定の非デフォルト VRF に関連付けることができます。外部ドメインは暗黙的にデフォルトまたはグローバル VRF に存在すると見なされます。

NAT のタイプ

ネットワーク全体を表す 1 つのアドレスのみを外部にアドバタイズするように NAT を設定できます。これにより、内部ネットワークを外部から効果的に隠すことができるため、セキュリティがさらに強化されます。

NAT には次のタイプがあります。

- スタティック アドレス変換（スタティック NAT）：ローカル アドレスとグローバル アドレスを 1 対 1 マッピングします。
- ダイナミック アドレス変換（ダイナミック NAT）：未登録の IP アドレスを、登録済み IP アドレスのプールから取得した登録済み IP アドレスにマップします。
- オーバーロード/PAT：複数の未登録 IP アドレスを、複数の異なるレイヤ 4 ポートを使用して、1 つの登録済み IP アドレスにマップ（多対 1）します。この方法は、ポート アドレス変換（PAT）とも呼ばれます。オーバーロードを使用することにより、使用できる正規のグローバル IP アドレスが 1 つのみでも、数千のユーザーをインターネットに接続することができます。

NAT による外部ネットワークへのパケットのルーティング（内部送信元アドレス変換）

自分が属するネットワークの外部と通信するときに、未登録の IP アドレスをグローバルで一意な IP アドレスに変換できます。

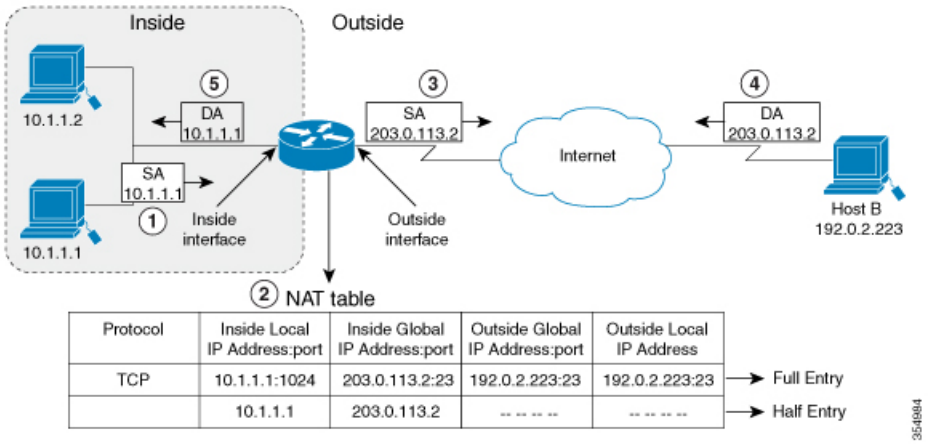
スタティックまたはダイナミック内部送信元アドレス変換は、次のようにして設定できます。

- スタティック変換は、内部ローカル アドレスと内部グローバル アドレスの間に 1 対 1 のマッピングを設定します。外部から固定アドレスを使って内部のホストにアクセスする必要がある場合には、スタティック変換が便利です。スタティック変換は、内部送信元アドレスのスタティック変換の設定 （155 ページ）で説明されているように、スタティック NAT ルールを設定して有効にできます。
- ダイナミック変換は、内部ローカル アドレスとグローバル アドレスのプールの間にマッピングを動的に設定します。ダイナミック変換は、ダイナミック NAT ルールを設定することで有効にできます。マッピングは、設定されているルールをランタイム時に評価した結果に基づいて設定されます。内部ローカルアドレスの指定には、標準と拡張の両方のアクセス コントロール リスト（ACL）を使用できます。内部グローバル アドレスはアドレ

スプールまたはインターフェイスから指定できます。ダイナミック変換は、内部送信元アドレスのダイナミック変換の設定（157ページ）のセクションで説明されているようにダイナミックルールを設定して有効にできます。

次の図には、ネットワーク内の送信元アドレスを、ネットワーク外の送信元アドレスに変換するデバイスが示されています。

図 *16 : NAT* 内部送信元変換



次のプロセスは、上の図に示す内部送信元アドレス変換について示します。

1. ホスト 10.1.1.1 のユーザーは、外部ネットワークのホスト B との接続を開きます。
2. NAT モジュールは、対応するパケットをインターセプトし、パケットを変換しようとします。

   一致する NAT ルールの有無に基づいて、次のシナリオが考えられます。

   - 一致するスタティック変換ルールが存在する場合、パケットは対応する内部グローバルアドレスに変換されます。存在しない場合、パケットはダイナミック変換ルールに対して照合され、一致した場合は対応する内部グローバルアドレスに変換されます。NAT モジュールは、変換したパケットに対応する完全修飾フローエントリを変換データベースに挿入します。これにより、このフローに対応するパケットの高速変換および転送が双方向で促進されます。
   - 一致するルールがない場合、パケットはアドレス変換を行わずに転送されます。
   - 有効な内部グローバルアドレスを取得できない場合は、たとえ一致するルールがあってもパケットはドロップされます。

（注） ダイナミック変換に ACL が使用される場合、NAT は ACL を評価し、特定の ACL で許可されているパケットのみが変換の対象になるようにします。

3. デバイスは、ホスト 10.1.1.1 の内部ローカル送信元アドレスをこの変換の内部グローバルアドレス 203.0.113.2 で置き換えて（パケットに関連したチェックサムのみが更新され、パケットの他のフィールドはすべて変更されません）、パケットを転送します。
4. NAT モジュールは、変換されたパケットフローに対応する完全修飾フローエントリを変換データベースに挿入します。その結果、このフローに対応するパケットの高速変換および転送が双方向で促進されます。
5. ホスト B はこのパケットを受信し、内部グローバル IP 宛先アドレス（DA）203.0.113.2 を使用して、ホスト 10.1.1.1 に応答します。
6. ホスト B からの応答パケットは、内部グローパル アドレスに送られます。NAT モジュールはこのパケットをインターセプトし、変換データベースにセットアップされているフロー エントリを使って対応する内部ローカル アドレスに変換し直します。

ホスト 10.1.1.1 はパケットを受信し、会話を続けます。デバイスは、受信する各パケットについて手順 2 ～ 5 を実行します。

外部送信元アドレス変換

ネットワークの内部から外部に移動する IP パケットの送信元アドレスを変換できます。通常、このタイプの変換は、重複しているネットワークを相互接続するために、内部送信元アドレスの変換と組み合わせて使用されます。

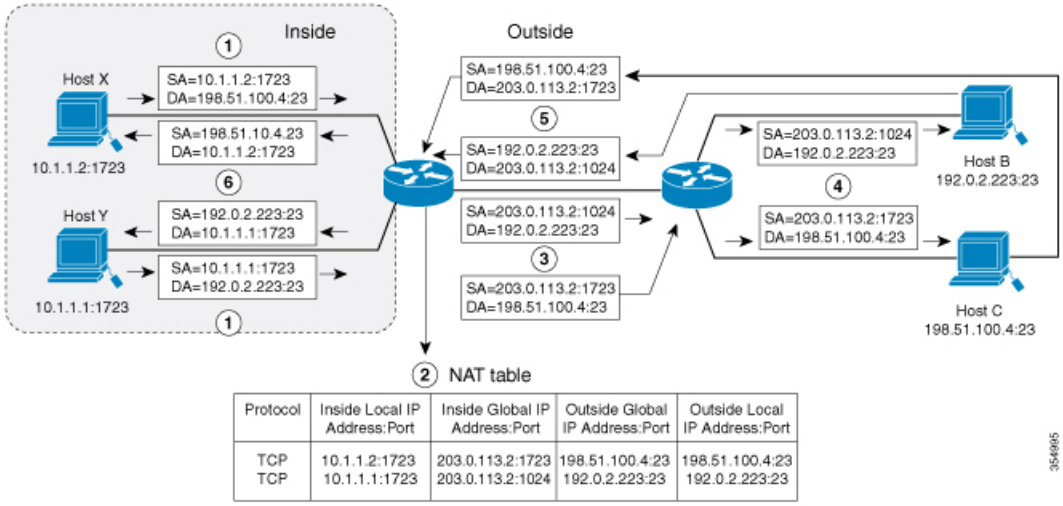
このプロセスについては、セクションで説明します。オーバーラップするネットワークの変換の設定 （164 ページ）

ポート アドレス変換（PAT）

デバイスが複数のローカル アドレスに対して 1 つのグローバル アドレスを使用できるようにすることで、内部グローバル アドレス プール内のアドレスを節約できます。このようなタイプの NAT の設定はオーバーロード、またはポート アドレス変換と呼ばれます。オーバーロードが設定されている場合、デバイスは、より高いレベルのプロトコルから十分な情報（たとえば、TCP または UDP ポート番号）を保持して、グローバル アドレスを正しいローカル アドレスに戻します。複数のローカル アドレスが 1 つのグローバル アドレスにマッピングされる場合、各内部ホストの TCP または UDP ポート番号によりローカル アドレスが区別されます。

次の図は、1 つの内部グローバル アドレスが複数の内部ローカル アドレスを表すときの NAT の動作を示しています。区別は、TCP ポート番号により行われます。

図 *17 :* 内部グローバル アドレスをオーバーロードする *PAT/NAT*



このデバイスは、上の図に示すように、内部グローバルアドレスのオーバーロードで次の処理を行います。ホスト B およびホスト C はいずれも、アドレス 203.0.113.2 にある 1 つのホストと通信していると信じています。しかし、実際には、異なるホストと通信しています。区別にはポート番号が使用されます。つまり、多数の内部ホストは、複数のポート番号を使用して、内部グローバル IP アドレスを共有することができます。

1. ホスト Y のユーザーはホスト B への接続を開き、ホスト X のユーザーはホスト C への接続を開きます。
2. NAT モジュールは、対応するパケットをインターセプトし、パケットの変換を試みます。

   一致する NAT ルールの有無に基づいて、次のシナリオが考えられます。

   - 一致するスタティック変換ルールが存在する場合はそのルールが優先され、パケットは対応するグローバルアドレスに変換されます。存在しない場合、パケットはダイナミック変換ルールに対して照合され、一致した場合は対応するグローバルアドレスに変換されます。NAT モジュールは、変換したパケットに対応する完全修飾フロー エントリを変換データベースに挿入し、このフローに対応するパケットの高速変換および転送を双方向で促進します。
   - 一致するルールがない場合、パケットはアドレス変換を行わずに転送されます。
   - 有効な内部グローバルアドレスを取得できない場合は、一致するルールがあってもパケットはドロップされます。
   - これは PAT 設定であるため、トランスポート ポートにより複数のフローを 1 つのグローバルアドレスに変換できます。（送信元アドレスに加えて送信元ポートも変換されるため、関連付けられているフローエントリは対応する変換マッピングを維持します。）

3. デバイスは、内部ローカル送信元アドレス/ポート 10.1.1.1/1723 および 10.1.1.2/1723 を対応する選択されたグローバル アドレス/ポート 203.0.113.2/1024 および 203.0.113.2/1723 にそれぞれ置き換えてパケットを転送します。
4. ホスト B はこのパケットを受信し、ポート 1024 で内部グローバル IP アドレス 203.0.113.2 を使用してホスト Y に応答します。ホスト C はこのパケットを受信し、ポート 1723 で内部グローバル IP アドレス 203.0.113.2 を使用してホスト X に応答します。
5. デバイスは、内部グローバル IP アドレスを持つパケットを受信すると、内部グローバルアドレスとポート、および外部アドレスとポートをキーとして NAT テーブル検索を実行します。次に、アドレスを内部ローカルアドレス 10.1.1.1:1723/10.1.1.2:1723 に変換し、パケットをホスト Y および X にそれぞれ転送します。

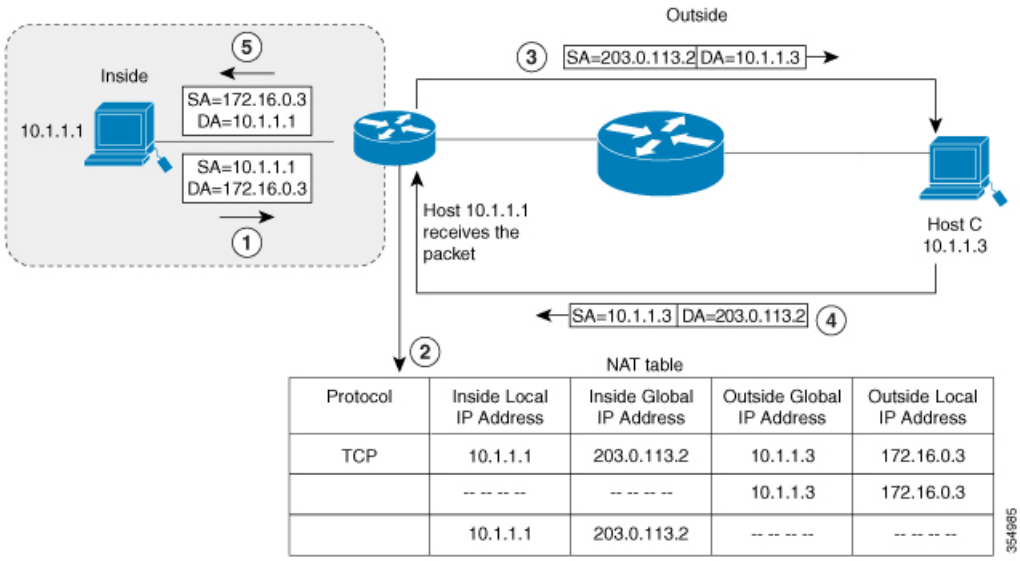
ホスト Y および X はパケットを受信し、通信を続行します。デバイスは、受信する各パケットについて手順 2 ～ 5 を実行します。

オーバーラップ ネットワーク

使用する IP アドレスが合法でない、または正式に割り当てられていない場合、IP アドレスを変換するために NAT を使用します。すでに合法的に所有されインターネットまたは外部ネットワーク上のデバイスに割り当てられている IP アドレスを、独自のネットワーク上の別のデバイスに割り当てると、ネットワークのオーバーラッピングが発生します。

次の図はオーバーラップしたネットワークを示しています。内部ネットワークと外部ネットワークの両方のローカル IP アドレスが同じです（10.1.1.x）。1 台の NAT デバイスを使用している場合、リモート ピアのアドレス（10.1.1.3）を内部から見た別のアドレスに変換するには、そのようにオーバーラップしているアドレス空間の間のネットワーク接続を確立する必要があります。

図 *18 : NAT* によるオーバーラップするアドレスの変換



Protocol	Inside Local IP Address	Inside Global IP Address	Outside Global IP Address	Outside Local IP Address
TCP	10.1.1.1	203.0.113.2	10.1.1.3	172.16.0.3
	-----	-----	10.1.1.3	172.16.0.3
	10.1.1.1	203.0.113.2	-----	-----

内部ローカルアドレス（10.1.1.1）および外部グローバルアドレス（10.1.1.3）が同じサブネットにあることに注意してください。オーバーラップするアドレスを変換するために、まず、内部送信元アドレスの変換によって内部ローカル アドレスが 203.0.113.2 に変換され、NAT テーブルにハーフ エントリが作成されます。受信側では、外部送信元アドレスが 172.16.0.3 に変換され、ハーフ エントリがもう 1 つ作成されます。すべての変換を完了し、NAT テーブルがフル エントリで更新されます。

次の手順は、オーバーラップするアドレスをデバイスが変換する方法を示します。

1. ホスト 10.1.1.1 は 172.16.0.3 への接続を開きます。
2. NAT モジュールは、内部ローカル アドレスと内部グローバル アドレスを相互に、また外部グローバルアドレスと外部ローカルアドレスを相互にマップする変換マッピングをセットアップします。
3. 送信元アドレス（SA）は、内部グローバルアドレスで置き換えられ、宛先アドレス（DA）は外部グローバル アドレスで置き換えられます。
4. ホスト C はパケットを受信し、会話を続けます。
5. デバイスは NAT テーブルの検索を行い、DA を内部ローカル アドレスで、SA を外部ローカル アドレスで置き換えます。
6. この変換プロセスを使用して、パケットがホスト 10.1.1.1 により受信され、会話が続けられます。

NAT の制限事項

- NAT の動作によっては、ハードウェアデータプレーンで現在サポートされていません。比較的遅いソフトウェア データ プレーンで実行される動作は次のとおりです。
  - Internet Control Message Protocol（ICMP）パケットの変換。
  - アプリケーション レイヤ ゲートウェイ（ALG）処理を必要とするパケットの変換。
  - 内側と外側の両方で変換が必要なパケット。
- 理想的な設定のハードウェアで変換および転送できるセッションの最大数は、Cisco Catalyst 9500シリーズ スイッチでは 2500。変換が必要なその他のフローは、スループットを下げてソフトウェア データ プレーンで処理されます。

（注） 変換ごとに TCAM の 2 つのエントリが使用されます。

- 設定されている NAT ルールは、リソースの制約のためにハードウェアにプログラムできない場合があります。これにより、特定のルールに該当するパケットが変換されずに転送されることがあります。

- ALG のサポートは、FTP、TFTP、および ICMP プロトコルに現在制限されています。また、TCP SYN、TCP FIN、および TCP RST は ALG トラフィックの一部ではありませんが、ALG トラフィックの一部として処理されます。
- ダイナミックに作成された NAT フローは、非アクティブな状態が一定期間続くとエージアウトします。
- ポリシーベースルーティング（PBR）と NAT は、同じインターフェイスではサポートされていません。PBR と NAT は、異なるインターフェイス上に設定されている場合にのみ連携します。
- ポート チャネルは、NAT の設定でサポートされていません。
- NAT は、断片化されたパケットの変換をサポートしていません。
- Bidirectional Forwarding Detection（BFD）は、NAT 設定ではサポートされていません。
- NAT は、ステートフル スイッチオーバー（SSO）をサポートしていません。動的に作成された NAT の状態は、アクティブデバイスとスタンバイデバイスの間で同期されません。
- 等コスト マルチパス ルーティング（ECMP）は、NAT でサポートされていません。
- ルートマップを設定された NAT はサポートされていないため、ルートマップを使用せずに NAT 設定を行う必要があります。
- NAT ACL の明示的な拒否アクセス制御エントリ（ACE）はサポートされていません。明示的な許可 ACE のみがサポートされます。

NAT のパフォーマンスとスケール数

NAT モジュールは、転送情報と書き換え情報を使用して関連したハードウェアテーブルをプログラミングすることで、ハードウェアの変換と転送をラインレートで実行できます。NAT のスループットを向上させるために、NAT 重視のリソース割り当てスキームを設定できます。

より良いパフォーマンスとスケール数が NAT で得られるように SDM テンプレートを設定します。次を参照してください。 スイッチ データベース管理（SDM）テンプレートの設定 （168 ページ）

ハードウェアで使用可能な TCAM フローの最大数は 5000 です。

（注） アドレスのみの変換を使用すると、フローの処理が最適化され、NAT 機能のスケールが拡張されます。

アドレスのみの変換

アドレスのみの変換（AOT）機能は、トランスポートポートではなくアドレスフィールドのみを変換する必要がある状況で使用できます。そのような状況で AOT 機能を有効にすると、ハー

ドウェアにおいてラインレートで変換および転送できるフローの数が大幅に増加します。この改善は、変換および転送に関連したさまざまなハードウェアリソースの使用を最適化することによって実現されます。一般的な NAT 集中型リソース割り当てスキームでは、ハードウェア変換を実行するために 5000の TCAM エントリが確保されます。その結果、ラインレートで変換および転送できるフローの数に厳密な上限が設定されます。AOT スキームでは、TCAM リソースの使用が高度に最適化されるため、TCAMテーブルでより多くのフローに対応できるようになり、ハードウェア変換および転送の規模が大幅に拡大します。AOT は、フローの大部分が単一または少数の宛先に送信される場合に非常に効果的です。そのような良好な条件下では、AOT により、特定のエンドポイントから発信されるすべてのフローのラインレート変換および転送が有効になる可能性があります。AOT 機能は、デフォルトでは無効になっています。**no ip nat create flow-entries** コマンドを使用して有効にできます。既存のダイナミックフローは、**clear ip nat translation** コマンドを使用してクリアできます。AOT 機能は、**ip nat create flow-entries** コマンドを使用して無効にできます。

アドレスのみの変換の制限事項

- AOT 機能は、単純な内部スタティックルールおよび内部ダイナミックルールに対応する変換シナリオでのみ正しく機能すると想定されています。単純なスタティックルールのタイプは **ip nat inside source static** *local-ip global-ip* で、ダイナミックルールのタイプは **ip nat inside source list** *access-list* **pool** *name* である必要があります。
- AOT が有効になっている場合、**show ip nat translation** コマンドを使用しても、変換および転送されるすべての NAT フローの可視性が実現することはありません。

NAT でのアプリケーション レベル ゲートウェイの使用

NAT は、アプリケーション データ ストリームで送信元および宛先 IP アドレスを伝送しない TCP/UDP トラフィックにおいて変換サービスを実行します。送信元および宛先 IP アドレスを伝送しないプロトコルには、HTTP、TFTP、telnet、archie、finger、Network Time Protocol（NTP）、ネットワーク ファイル システム（NFS）、リモート ログイン（rlogin）、リモート シェル（rsh）protocol、およびリモート コピー（rcp）があります。

アドレス/ポート情報をペイロードで搬送するアプリケーションは、NAT アプリケーション レベル ゲートウェイ（ALG）により、NAT ドメイン全体で正しく機能できます。パケット ヘッダ内のアドレス/ポートの通常の変換に加えて、ALG はペイロードに存在するアドレス/ポートの変換も処理し、一時マッピングを設定します。

NAT の設定のベスト プラクティス

- スタティック ルールとダイナミック ルールの両方が設定されている場合は、ルールに指定されているローカルアドレスがオーバーラップしていないことを確認してください。このようなオーバーラップの可能性がある場合は、スタティック ルールが使用するアドレスをダイナミック ルールに関連付けられている ACL で除外してください。同様に、グローバルアドレス間のオーバーラップもなくす必要があります。オーバーラップしていると、望ましくない動作が生じることがあります。

- VRF からグローバルへの変換機能では、NAT 外部インターフェイスがデフォルトまたはグローバル VRF に関連付けられていると見なされます。したがって、VRF 対応 NAT の実行中は、NAT 外部インターフェイスをデフォルト以外の VRF に配置することは推奨されません。
- NAT ルールに関連付けられている ACL では、**permit ip any any** などのあいまいなフィルタリングを使用しないでください。このようなフィルタリングは、必要のないパケットを変換することがあります。
- 複数の NAT ルールでアドレス プールを共有しないでください。
- スタティック NAT とダイナミック プールで同じ内部グローバル アドレスを定義しないでください。これを行うと、望ましくない結果を招くことがあります。
- NAT に関連付けられているデフォルトのタイムアウト値を変更する場合は、慎重に行ってください。タイムアウト値を短くすると、CPU の使用率が高くなることがあります。
- 変換エントリを手動でクリアする場合は、アプリケーションセッションが中断されることがあるため、慎重に行ってください。

NAT の設定

このセクションで説明するタスクを使用して、NAT を効果的に設定できます。設定によっては、複数の作業を実行する必要があります。

内部送信元アドレスのスタティック変換の設定

内部ローカル アドレスと内部グローバル アドレス間の 1 対 1 マッピングを可能にするには、内部送信元アドレスのスタティック変換を設定します。外部から固定アドレスを使って内部のホストにアクセスする必要がある場合には、スタティック変換が便利です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** **例**： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **例**： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	要件に応じて次のコマンドのいずれかを使用します。	• 内部ローカル アドレスと内部グローバルアドレス間のスタティック変換を設定します。

	コマンドまたはアクション	目的

	• **ip nat inside source static** *local-ip global-ip*
	`Switch(config)# ip nat inside source static 10.10.10.1 172.16.131.1`
	• **ip nat inside source static** *protocol local-ip port global-ip port*
	`Switch(config)# ip nat inside source static tcp 10.10.10.1 1234 172.16.131.1 5467`
	• **ip nat inside source static network** *local-ip global-ip* { **prefix_len** *len* | **subnet** *subnet-mask*}
	`Switch(config)# ip nat inside source static network 10.10.10.1 172.16.131.1 prefix_len 24`
	• **ip nat inside source static** *local-ip global-ip* **vrf** *vrf-name*
	`Switch(config)# ip nat inside source static 10.10.10.1 172.16.131.1 vrf vrf1`
		• 内部ローカル アドレスと内部グローバルアドレス間のスタティック ポート変換を設定します。
		• 複数の個別変換ルールを指定せずに、サブネット全体のスタティック変換マッピングを許可します。目的のサブネットの変換マッピングを指定できます。実際の変換は、アドレスのネットワーク部を変換することによって実行されます。ホスト部は変更されません。
		• スタティック変換 VRF を認識させ、特定のルールを指定された VRF に関連付けます。

ステップ 4	**interface** *type number* 例： `Switch(config)# interface ethernet 1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)# ip address 10.114.11.39 255.255.255.0`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 6	**ip nat inside** 例： `Switch(config-if)# ip nat inside`	NAT の対象である内部ネットワークにインターフェイスを接続します。
ステップ 7	**exit** 例： `Switch(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 8	**interface** *type number* 例： `Switch(config)# interface gigabitethernet 0/0/0`	異なるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 9	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)# ip address 172.31.232.182 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 10	**ip nat outside** 例： `Switch(config-if)# ip nat outside`	外部ネットワークにインターフェイスを接続します。
ステップ 11	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

内部送信元アドレスのダイナミック変換の設定

ダイナミック変換は、内部ローカル アドレスとグローバル アドレスのプールの間にマッピングを動的に設定します。ダイナミック変換は、ダイナミック NAT ルールを設定することで有効にできます。マッピングは、設定されているルールをランタイム時に評価した結果に基づいて設定されます。内部ローカル アドレスの指定には ACL を使用できます。また、内部グローバル アドレスは、アドレス プール、またはインターフェイスから指定できます。

プライベート ネットワークに存在する複数のユーザーがインターネットへのアクセスを必要としている場合には、ダイナミック変換が便利です。ダイナミックに設定されたプール IP アドレスは必要に応じて使用でき、インターネットへのアクセスが必要なくなったときは別のユーザーが使用できるようにリリースできます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip nat pool** *name start-ip end-ip* **netmask** *netmask* | **prefix-length** *prefix-length* 例： `Switch(config)# ip nat pool net-208 172.16.233.208 172.16.233.223 prefix-length 28`	必要に応じて割り当てられるグローバル アドレスのプールを定義します。

	コマンドまたはアクション	目的
ステップ 4	**access-list** *access-list-number* **permit** *source [source-wildcard]* 例： `Switch(config)# access-list 1 permit 192.168.34.0 0.0.0.255`	変換されるアドレスを許可する標準アクセス リストを定義します。
ステップ 5	**ip nat inside source list** *access-list-number* **pool** *name* **vrf** *vrf-name* 例： `Switch(config)# ip nat inside source list 1 pool net-208`	ステップ 4 で定義したアクセス リストを指定して、ダイナミック送信元変換を設定します。 **vrf** キーワードを使用すると、ダイナミック変換 VRF が認識され、特定のルールが指定された VRF に関連付けられます。
ステップ 6	**interface** *type number* 例： `Switch(config)# interface ethernet 1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**ip address** *ip-address mask* 例： `Switch(config-if)# ip address 10.114.11.39 255.255.255.0`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 8	**ip nat inside** 例： `Switch(config-if)# ip nat inside`	NAT の対象である内部ネットワークにインターフェイスを接続します。
ステップ 9	**exit** 例： `Switch(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、グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface** *type number* 例： `Switch(config)# interface ethernet 0`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 11	**ip address** *ip-address mask* 例： `Switch(config-if)# ip address 172.16.232.182 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 12	**ip nat outside** 例： `Switch(config-if)# ip nat outside`	外部ネットワークにインターフェイスを接続します。

	コマンドまたはアクション	目的
ステップ 13	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

PAT の設定

グローバルアドレスのオーバーロードによる PAT の設定

NAT モジュールは、次のタスクで説明するように、アドレスプールとインターフェイスを介してダイナミック PAT 設定をサポートします。

グローバルアドレスのオーバーロードを使用して、内部ユーザーにインターネットへのアクセスを許可し、内部グローバルアドレスプールのアドレスを節約するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip nat pool** *name start-ip end-ip* **netmask** *netmask* | **prefix-length** *prefix-length* 例： `Switch(config)# ip nat pool net-208 192.168.202.129 192.168.202.158 netmask 255.255.255.224`	必要に応じて割り当てられるグローバル アドレスのプールを定義します。
ステップ 4	**access-list** *access-list-number* **permit** *source [source-wildcard]* 例： `Switch(config)# access-list 1 permit 192.168.201.30 0.0.0.255`	変換されるアドレスを許可する標準アクセス リストを定義します。 アクセスリストは、変換されるアドレスだけを許可する必要があります（各アクセス リストの最後に暗黙の「deny all」ステートメントが存在することに注意してください）。許可が多すぎるアクセスリストを使用すると、予測困難な結果を招くことがあります。

	コマンドまたはアクション	目的
ステップ 5	**ip nat inside source list** *access-list-number* **pool** *name* [**vrf** *vrf-name*] **overload** 例： `Switch(config)# ip nat inside source list 1 pool net-208 overload`	手順 4 で定義されたアクセス リストを指定して、ダイナミック送信元変換を設定します。 **vrf** キーワードを使用すると、ダイナミック変換 VRF が認識され、特定のルールが指定された VRF に関連付けられます。
ステップ 6	**interface** *type number* 例： `Switch(config)# interface ethernet 1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)# ip address 192.168.201.1 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 8	**ip nat inside** 例： `Switch(config-if)# ip nat inside`	NAT の対象である内部ネットワークにインターフェイスを接続します。
ステップ 9	**exit** 例： `Switch(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 10	**interface** *type number* 例： `Switch(config)# interface ethernet 0`	異なるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 11	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)#  ip address 192.168.201.29 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 12	**ip nat outside** 例： `Switch(config-if)# ip nat outside`	外部ネットワークにインターフェイスを接続します。
ステップ 13	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

インターフェイスのオーバーロードによる PAT の設定

インターフェイスのオーバーロードにより、内部ユーザーにインターネットへのアクセスを許可し、内部グローバルアドレスプールのアドレスを節約するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* **permit** *source [source-wildcard]* 例： `Switch(config)# access-list 1 permit 192.168.201.30 0.0.0.255`	変換されるアドレスを許可する標準アクセス リストを定義します。 アクセスリストは、変換されるアドレスだけを許可する必要があります（各アクセスリストの最後に暗黙の「deny all」ステートメントが存在することに注意してください）。許可が多すぎるアクセスリストを使用すると、予測困難な結果を招くことがあります。
ステップ 4	**ip nat inside source list** *access-list-number* **interface** *name* **overload** 例： `Switch(config)# ip nat inside source list 1 interface gigabitethernet0/0/2 overload`	手順 3 で定義されたアクセス リストを指定して、ダイナミック送信元変換を設定します。 （注） オーバーロード用に、有効な IP アドレスを持つ、動作状態にある任意のレイヤ 3 インターフェイスを選択できます。
ステップ 5	**interface** *type number* 例： `Switch(config)# interface gigabitethernet0/0/1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)# ip address 192.168.201.1 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 7	**ip nat inside** 例： `Switch(config-if)# ip nat inside`	NAT の対象である内部ネットワークにインターフェイスを接続します。
ステップ 8	**exit** 例： `Switch(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 9	**interface** *type number* 例： `Switch(config)# interface gigabitethernet0/0/2`	異なるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 10	**ip address** *ip-address mask* **[secondary]** 例： `Switch(config-if)#  ip address 192.168.201.29 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 11	**ip nat outside** 例： `Switch(config-if)# ip nat outside`	外部ネットワークにインターフェイスを接続します。
ステップ 12	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

外部 IP アドレスのみの NAT の設定

デフォルトで NAT は、NAT でのアプリケーション レベル ゲートウェイの使用 （154 ページ）で説明されているように、パケットのペイロードに埋め込まれているアドレスを変換します。埋め込みアドレスを変換することが望ましくない場合は、外部の IP アドレスのみを変換するように NAT を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**ip nat inside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* [**overload**] | **static network** *local-ip global-ip* [**no-payload**]} 例： `Device(config)# ip nat inside source static network 10.1.1.1 192.168.251.0/24 no-payload`	内部ホストデバイスでのネットワークパケット変換を無効化します。
ステップ 4	**ip nat inside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* [**overload**] | **static** {**tcp** | **udp**} *local-ip local-port global-ip global-port* [**no-payload**]} 例： `Device(config)# ip nat inside source static tcp 10.1.1.1 2000 192.168.1.1 2000 no-payload`	内部ホストデバイスでのポートパケット変換を無効化します。
ステップ 5	**ip nat inside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* [**overload**] | **static** [**network**] *local-network-mask global-network-mask* [**no-payload**]} 例： `Device(config)# ip nat inside source static 10.1.1.1 192.168.1.1 no-payload`	内部ホストルータでのパケット変換を無効化します。
ステップ 6	**ip nat outside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* | **static** *local-ip global-ip* [**no-payload**]} 例： `Device(config)# ip nat outside source static 10.1.1.1 192.168.1.1 no-payload`	外部ホストルータでのパケット変換を無効化します。
ステップ 7	**ip nat outside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* | **static** {**tcp** | **udp**} *local-ip local-port global-ip global-port* [**no-payload**]} 例： `Device(config)# ip nat outside source static tcp 10.1.1.1 20000 192.168.1.1 20000 no-payload`	外部ホストデバイスでのポートパケット変換を無効化します。

	コマンドまたはアクション	目的
ステップ 8	**ip nat outside source** {**list** {*access-list-number* | *access-list-name*} **pool** *pool-name* | **static** [**network**] *local-network-mask global-network-mask* [**no-payload**]} 例： `Device(config)# ip nat outside source static network 10.1.1.1 192.168.251.0/24 no-payload`	外部ホストデバイスでのネットワークパケット変換を無効化します。
ステップ 9	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 10	**show ip nat translations** [**verbose**] 例： `Device# show ip nat translations`	アクティブな NAT を表示します。

オーバーラップするネットワークの変換の設定

スタブ ネットワーク内の IP アドレスが別のネットワークに属する正式な IP アドレスであるときに、スタティック変換を使用して、これらのホストやルータと通信する必要がある場合は、オーバーラップするネットワークのスタティック変換を設定します。

（注） NAT 外部変換を成功させるためには、デバイスに外部ローカルアドレスのルートを設定する必要があります。ルートは手動で、または **ip nat outside source** {**static** | **list**} コマンドと関連付けられた **add-route** オプションを使用して設定できます。ルートの自動作成を有効にする **add-route** オプションを使用することを推奨します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	

	コマンドまたはアクション	目的
ステップ 3	**ip nat inside source static** *local-ip global-ip* 例： `Switch(config)# ip nat inside source static 10.1.1.1 203.0.113.2`	内部ローカルアドレスと内部グローバルアドレス間のスタティック変換を設定します。
ステップ 4	**ip nat outside source static** *local-ip global-ip* 例： `Switch(config)# ip nat outside source static 172.16.0.3 10.1.1.3`	外部ローカルアドレスと外部グローバルアドレス間のスタティック変換を設定します。
ステップ 5	**interface** *type number* 例： `Switch(config)# interface ethernet 1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**ip address** *ip-address mask* 例： `Switch(config-if)# ip address 10.114.11.39 255.255.255.0`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 7	**ip nat inside** 例： `Switch(config-if)# ip nat inside`	内部と接続されることを示すマークをインターフェイスに付けます。
ステップ 8	**exit** 例： `Switch(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 9	**interface** *type number* 例： `Switch(config)# interface ethernet 0`	異なるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 10	**ip address** *ip-address mask* 例： `Switch(config-if)#  ip address 172.16.232.182 255.255.255.240`	インターフェイスのプライマリ IP アドレスを設定します。
ステップ 11	**ip nat outside** 例： `Switch(config-if)# ip nat outside`	外部と接続されることを示すマークをインターフェイスに付けます。

	コマンドまたはアクション	目的
ステップ 12	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

アドレス変換タイムアウトの設定

NAT の設定に基づき、アドレス変換のタイムアウトを設定できます。

デフォルトでは、ダイナミックに作成された変換エントリは、さまざまなリソースを効率的に利用できるようにするために、非アクティブな状態が一定時間続くとタイムアウトします。必要に応じて、タイムアウトのデフォルト値を変更できます。主な変換タイプに関連付けられているデフォルトのタイムアウト設定は、次のとおりです。

- 確立された TCP セッション：24 時間
- UDP フロー：5 分
- ICMP フロー：1 分

デフォルトのタイムアウト値は、ほとんどの展開シナリオでタイムアウト要件を満たすことができます。ただし、これらの値は必要に応じて調整/微調整できます。短いタイムアウト値を設定すると（60 秒未満）、CPU の使用率が高くなることがあるため推奨されません。詳細については、NAT の設定のベスト プラクティス （154 ページ）を参照してください。

この項で説明するタイムアウトは、設定に応じて変更できます。

- ダイナミック設定のためにグローバル IP アドレスを迅速に解放する必要がある場合は、**ip nat translation timeout** コマンドを使用して、デフォルトのタイムアウトよりもタイムアウトを短く設定してください。ただし、次の手順で指定するコマンドで設定した他のタイムアウトよりも長い時間にしてください。
- TCP セッションが両側から受け取る終了（FIN）パケットで正しく終了していない場合、またはリセット時に正しく終了しない場合は、**ip nat translation tcp-timeout** コマンドを使用してデフォルトの TCP タイムアウトを変更してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**ip nat translation** *seconds* 例： `Switch(config)# ip nat translation 300`	（任意）NAT 変換がタイムアウトになるまでの時間を変更します。 デフォルト タイムアウトは 24 時間です。これは、ハーフエントリのエージング タイムに適用されます。
ステップ 4	**ip nat translation udp-timeout** *seconds* 例： `Switch(config)# ip nat translation udp-timeout 300`	（任意）UDP タイムアウト値を変更します。
ステップ 5	**ip nat translation tcp-timeout** *seconds* 例： `Switch(config)# ip nat translation tcp-timeout 2500`	（任意）TCP タイムアウト値を変更します。 デフォルトは 24 時間です。
ステップ 6	**ip nat translation finrst-timeout** *seconds* 例： `Switch(config)# ip nat translation finrst-timeout 45`	（任意）Finish and Reset タイムアウト値を変更します。 finrst-timeout：TCP セッションが finish-in（FIN-IN）要求と finish-out（FIN-OUT）要求の両方を受信した後の、またはTCP セッション リセット後のエージング タイム。
ステップ 7	**ip nat translation icmp-timeout** *seconds* 例： `Switch(config)# ip nat translation icmp-timeout 45`	（任意）ICMP タイムアウト値を変更します。
ステップ 8	**ip nat translation syn-timeout** *seconds* 例： `Switch(config)# ip nat translation syn-timeout 45`	（任意）同期（SYN）タイムアウト値を変更します。 同期タイムアウトまたはエージング タイムは、TCP セッションで SYN 要求が受信された場合にのみ使用されます。同期確認応答（SYNACK）要求が受信されると、タイムアウトが TCP タイムアウトに変更されます。
ステップ 9	**end** 例： `Switch(config-if)# end`	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、特権 EXEC モードに戻ります。

スイッチ データベース管理（SDM）テンプレートの設定

SDM テンプレートを使用し、NAT に合わせてシステム リソースを最適に設定します。

テンプレートを設定してシステムを再起動した後、**show sdm prefer** 特権 EXEC コマンドを使用して、新しいテンプレート設定を確認できます。**reload** 特権 EXEC コマンドを入力する前に、**show sdm prefer** コマンドを入力すると、**show sdm prefer** コマンドにより、現在使用しているテンプレートおよびリロード後にアクティブになるテンプレートが表示されます。

SDM テンプレートを設定して NAT の動作を最適にサポートするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**sdm prefer nat** 例： `Switch(config)# sdm prefer nat`	スイッチで使用する SDM テンプレートを指定します。 Cisco IOS XE Gibraltar 16.12.1 から開始する場合、このテンプレートを利用するには、Network Advantage ライセンスが必要です。以前のすべてのリリースでは、DNA Advantage ライセンスで利用できます。
ステップ 3	**end** 例： `Switch(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**write memory** 例： `Switch# write memory`	リロードする前に現在の構成を保存します。
ステップ 5	**reload** 例： `Switch# reload`	オペレーティング システムをリロードします。

NAT の設定例

例：内部送信元アドレスのスタティック変換の設定

次の例では、10.114.11.0 ネットワークからアドレス指定される内部ホストがグローバルに一意な 172.31.233.208/28 ネットワークにどのように変換されるかを示しています。その後、10.114.11.0 ネットワーク（本物の 10.114.11.0 ネットワーク）の外部ホストから送信されたパケットは、10.0.1.0/24 ネットワーク由来に見えるように変換されます。

```
ip nat pool net-208 172.31.233.208 172.31.233.223 prefix-length 28
ip nat pool net-10 10.0.1.0 10.0.1.255 prefix-length 24
ip nat inside source list 1 pool net-208
ip nat outside source list 1 pool net-10
!
interface gigabitethernet 0/0/0
 ip address 172.31.232.182 255.255.255.240
 ip nat outside
!
interface gigabitethernet 1/1/1
 ip address 10.114.11.39 255.255.255.0
 ip nat inside
!
access-list 1 permit 10.114.11.0 0.0.0.255
```

次に、重複するローカルアドレスを変換する静的 VRF 対応 NAT 設定の例を示します。

```
ip nat inside source static 192.168.121.33 10.2.2.1 vrf vrf1
ip nat inside source static 192.168.121.33.10.2.2.2 vrf vrf2
```

例：内部送信元アドレスのダイナミック変換の設定

次の例では、192.168.1.0 または 192.168.2.0 ネットワークのいずれかのネットワークからアドレス指定される内部ホストがグローバルに一意な 172.31.233.208/28 ネットワークにどのように変換されるかを示しています。

```
ip nat pool net-208 172.31.233.208 172.31.233.223 prefix-length 9
ip nat inside source list 1 pool net-208
!
interface gigabitethernet 0/0/0
 ip address 172.31.232.182 255.255.255.240
 ip nat outside
!
interface gigabitethernet 1/1/1
 ip address 192.168.1.94 255.255.255.0
 ip nat inside
!
access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
!
```

次に、重複するローカルアドレスを変換するダイナミック VRF 対応 NAT 設定の例を示します。

```
ip nat inside source list 1 interface gigabitethernet 0/0/0 vrf vrf1 overload
```

```
!
ip route vrf vrf1 0.0.0.0 0.0.0.0 192.168.1.1
!
access-list 1 permit 10.1.1.1.0 0.0.0.255
!
ip nat inside source list 1 interface gigabitethernet 1/1/1 vrf vrf1 overload
!
ip route vrf vrf1 0.0.0.0 0.0.0.0 172.16.1.1 global
access-list 1 permit 10.1.1.0 0.0.0.255
!
```

NAT のトラブルシューティング

ここでは、NAT のトラブルシューティングと確認のための基本的な手順について説明します

- NAT で実現できることを明確に定義する。
- **show ip nat translation** コマンドを使用して、正しい変換テーブルが存在することを確認する。
- **show ip nat translation vrf***vrf-name* コマンドを使用して、VRF 対応 NAT の正しい変換テーブルが存在することを確認する。
- **show ip nat translation verbose** コマンドを使用して、タイマーの値が正しく設定されていることを確認する。
- **show ip access-list** コマンドを使用して、NAT の ACL 値をチェックする。
- **show ip nat statistics** コマンドを使用して、NAT の全体的な設定をチェックする。
- **clear ip nat translation** コマンドを使用して、タイマーの期限が切れる前に NAT 変換テーブルのエントリをクリアする。
- NAT 設定をデバッグするには、**debug nat ip** および **debug nat ip detailed** コマンドを使用します。
- VRF 対応 NAT に関連した問題をトラブルシューティングするには、**debug ip nat vrf** *vrf-name* コマンドを使用します。

NAT のトラブルシューティングの詳細については、を参照してください。 http://www.cisco.com/c/en/us/support/docs/ip/network-address-translation-nat/8605-13.html

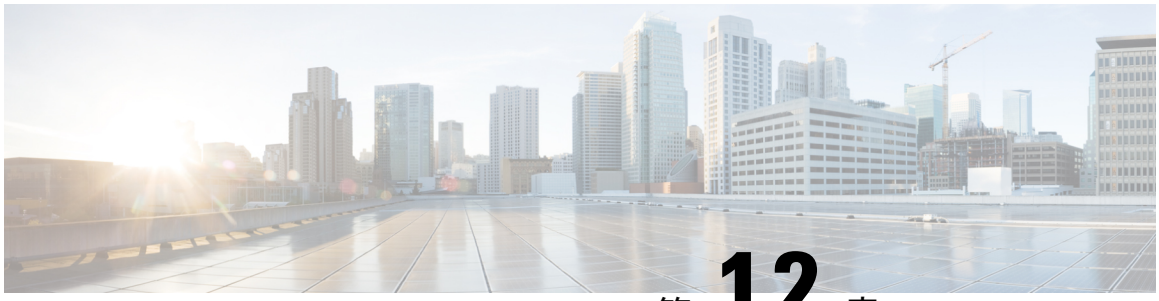
ネットワークアドレス変換の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.10.1	ネットワーク アドレス変換	NATによって、未登録IPアドレスを使用するプライベートIPネットワークをインターネットに接続できます。NATはデバイス（通常、2つのネットワークを接続するもの）上で動作し、別のネットワークにパケットを転送する前に、内部ネットワークのプライベート アドレスをグローバルなルート可能なアドレスに変換します。
	アドレスのみの変換	アドレスのみの変換（AOT）は、ネットワークアドレス変換（NAT）でラインレートで変換および転送できるIPフローの数を増やすことを目的としています。AOTは、TCAMSなどのハードウェアリソースの使用を最適化し、より多くのフローの処理を可能にします。
Cisco IOS XE Gibraltar 16.12.1	NATのライセンスレベル	NATをNetwork Advantageライセンスで使用できるようになりました。以前のすべてのリリースでは、DNA Advantageライセンスで使用できます。
Cisco IOS XE Amsterdam 17.2.1	VRF対応NAT	NATのVRFサポートが導入されました。

Cisco Feature Navigatorを使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigatorには、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 12 章

VRRPv3 プロトコルのサポート

VRRPv3 プロトコルのサポートの制限事項

- VRRPv3 は既存のダイナミック プロトコルの代替にはなりません。VRRPv3 は、マルチアクセス、マルチキャスト、またはブロードキャスト対応イーサネット LAN で使用するために設計されています。
- VRRPv3 は、イーサネット、ファストイーサネット、ブリッジグループ仮想インターフェイス（BVI）、およびギガビット イーサネット インターフェイス、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）、VRF を認識する MPLS VPN、および VLAN 上でサポートされます。
- BVI インターフェイスの初期化に関連して転送遅延が発生するため、VRRPv3 アドバタイズ タイマーの時間は BVI インターフェイスでの転送遅延時間より短く設定する必要があります。VRRPv3 アドバタイズタイマーの時間を BVI インターフェイスでの転送遅延時間以上の値に設定すると、最近初期化された BVI インターフェイス上にある VRRP デバイスが無条件にプライマリロールを引き継ぐことができなくなります。BVI インターフェイスでの転送遅延を設定するには、**bridge forward-time** コマンドを使用します。VRRP アドバタイズメントタイマーを設定するには、**vrrp timers advertise** コマンドを使用します。
- VRRPv3 は、ステートフル スイッチオーバー（SSO）をサポートしていません。
- VRRP が VRRS 経路の冗長インターフェイスと同じネットワーク パス上で動作する場合にのみ、完全なネットワークの冗長性を実現できます。完全な冗長性のために、次の制約事項が適用されます。

- VRRS 経路は、親 VRRP グループと異なる物理インターフェイスを共有したり、親 VRRP グループと異なる物理インターフェイスを持つサブインターフェイス上で設定することはできません。
- VRRS 経路は、関連付けられた VLAN が親 VRRP グループが設定された VLAN と同じトランクを共有していない限り、スイッチ仮想インターフェイス（SVI）に設定することはできません。

VRRPv3 プロトコル サポートについて

ここでは、VRRPv3 プロトコルサポートについて説明します。

VRRPv3 の利点

IPv4 と IPv6 のサポート

VRRPv3 は、VRRPv2 が IPv4 アドレスしかサポートしていないのに対し、IPv4 と IPv6 アドレス ファミリをサポートしています。

（注） VRRPv3 が使用中の場合、VRRPv2 は使用できません。VRRPv3 を設定可能にするには、**fhrp version vrrp v3** コマンドをグローバル コンフィギュレーション モードで使用する必要があります。

冗長性

VRRP により、複数のデバイスをデフォルト ゲートウェイ デバイスとして設定できるようになり、ネットワークに単一障害点が生じる可能性を低減できます。

ロード シェアリング

LAN クライアントとのトラフィックを複数のデバイスで共有するように VRRP を設定できるため、利用可能なデバイス間でより公平にトラフィックの負荷を共有できます。

複数の仮想デバイス

VRRP はデバイスの物理インターフェイス上で（拡張の制限に従って）最大 255 の仮想デバイス（VRRP グループ）をサポートします。複数の仮想デバイスをサポートすることで、LAN トポロジ内で冗長化とロードシェアリングを実装できます。拡張環境では、VRRS 経路は VRRP 制御グループと組み合わせて使用する必要があります。

複数の IP アドレス

仮想デバイスは、セカンダリ IP アドレスを含め複数の IP アドレスを管理できます。そのため、イーサネット インターフェイスに複数のサブネットを設定した場合、サブネットごとに VRRP を設定できます。

（注） VRRP グループでセカンダリ IP アドレスを使用するには、プライマリ アドレスを同じグループで設定する必要があります。

プリエンプション

VRRP の冗長性スキームにより、仮想デバイスバックアップのプリエンプションが可能になり、より高い優先順位が設定された仮想デバイスバックアップが、機能を停止したプライマリ仮想デバイスを引き継ぐことができます。

（注） 優先順位の低いプライマリデバイスのプリエンプションは、オプションの遅延時間を指定して有効にします。

アドバタイズメント プロトコル

VRRP は、VRRP アドバタイズメント専用のインターネット割り当て番号局（IANA）標準マルチキャスト アドレスを使用します。IPv4 では、マルチキャスト アドレスは 224.0.0.18 です。IPv6 では、マルチキャスト アドレスは FF02:0:0:0:0:0:0:12 です。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デバイス数が最小限になり、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 では VRRP に IP プロトコル番号 112 を割り当てていました。

VRRP デバイスのプライオリティおよびプリエンプション

VRRP 冗長性スキームの重要な一面に、VRRP デバイス プライオリティがあります。優先順位により、各 VRRP デバイスが実行する役割と、仮想プライマリデバイスが機能を停止したときにどのようなことが起こるかが決定されます。

特定の VRRP デバイスが仮想デバイスの IP アドレスと物理インターフェイスの IP アドレスのオーナーである場合には、このデバイスが仮想プライマリデバイスとして機能します。

特定の VRRP デバイスが仮想バックアップデバイスとして機能するかどうか、および仮想プライマリデバイスが機能を停止した場合に仮想プライマリデバイスを引き継ぐ順序も、優先順位によって決定されます。各仮想バックアップデバイスの優先順位は、**priority** コマンドを使用して 1 ～ 254 の値に設定できます（**vrrp address-family** コマンドを使用して VRRP 設定モードに入り、**priority** オプションにアクセスします）。

たとえば、LAN トポロジのプライマリ仮想デバイスであるデバイス A が機能を停止した場合、選択プロセスが実行され、仮想デバイスバックアップ B または C が引き継ぐかどうかが決定

されます。デバイス B とデバイス C がそれぞれ優先順位 101 と 100 に設定されている場合、優先順位の高いデバイス B がプライマリ仮想デバイスになります。デバイス B とデバイス C が両方とも優先順位 100 に設定されている場合、IP アドレスが大きい方の仮想デバイスバックアップが選択されてプライマリ仮想デバイスになります。

デフォルトでは、プリエンプティブスキームが有効になっています。この場合、プライマリ仮想デバイスになるように選択されている仮想バックアップデバイスの中で、より高い優先順位が設定されている仮想バックアップデバイスがプライマリ仮想デバイスになります。このプリエンプティブスキームは、**no preempt** コマンドを使用して無効にできます（**vrrp address-family** コマンドを使用して VRRP 設定モードに入り、**no preempt** コマンドを入力します）。プリエンプションが無効になっている場合は、元のプライマリ仮想デバイスが回復して再びプライマリになるまで、プライマリ仮想デバイスになるように選択されている仮想デバイスバックアップがプライマリの役割を果たします。

（注） 優先順位の低いプライマリデバイスのプリエンプションは、オプションの遅延時間を指定して有効にします。

VRRP のアドバタイズメント

プライマリ仮想デバイスは、同じグループ内の他の VRRP デバイスに 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では、プライマリ仮想デバイスの優先順位と状態が伝達されます。VRRP アドバタイズメントは、（VRRP グループ設定に基づいて）IPv4 または IPv6 パケットにカプセル化され、VRRP グループに割り当てられた適切なマルチキャストアドレスに送信されます。IPv4 では、マルチキャスト アドレスは 224.0.0.18 です。IPv6 では、マルチキャスト アドレスは FF02:0:0:0:0:0:0:12 です。アドバタイズメントは、デフォルトでは 1 秒に 1 回送信されますが、この間隔は設定可能です。

シスコデバイスでは、VRRPv2 からの変更点であるミリ秒タイマーを設定できます。ミリ秒タイマー値は、プライマリ デバイスとバックアップ デバイスの両方に手動で設定する必要があります。バックアップデバイス上の **show vrrp** コマンド出力に表示されるプライマリ アドバタイズメント値は、常に 1 秒です。これはバックアップデバイス上のパケットでミリ秒値が受け入れられないためです。

ミリ秒タイマーは、絶対に必要な場合以外は使用しないようにし、使用する場合は慎重な検討とテストが必要です。ミリ秒の値は望ましい状況でのみ動作します。ミリ秒のタイマー値の使用は、VRRPv3 も含めてサポートしている限り、サードパーティ ベンダーと互換性があります。タイマー値は 100 ～ 40000 ミリ秒の範囲で指定できます。

VRRPv3 プロトコル サポートの設定方法

ここでは、VRRPv3 プロトコルサポートに関する設定情報について説明します。

VRRP グループの作成とカスタマイズ

VRRP グループを作成するには、次の手順を実行します。ステップ 6 ～ 14 はそのグループのカスタマイズ オプションで、これらは省略可能です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**fhrp version vrrp v3** 例： `Device(config)# fhrp version vrrp v3`	VRRPv3 および VRRS を設定する機能をイネーブルにします。
ステップ 4	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 0/0/0`	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**vrrp** *group-id* **address-family** {**ipv4** | **ipv6**} 例： `Device(config-if)# vrrp 3 address-family ipv4`	VRRP グループを作成し、VRRP 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	**address** *ip-address* [**primary** | **secondary**] 例： `Device(config-if-vrrp)# address 100.0.1.10 primary`	VRRP グループのプライマリ アドレスまたはセカンダリ アドレスを指定します。

	コマンドまたはアクション	目的
		（注） IPv6 の VRRPv3 では、グループを動作可能にするため、プライマリ仮想リンクローカル IPv6 アドレスが設定されている必要があります。プライマリ リンクローカル IPv6 アドレスがグループに確立されると、セカンダリ グローバル アドレスを追加できます。
ステップ 7	**description** *group-description* 例： `Device(config-if-vrrp)# description group 3`	（任意）VRRP グループの説明を指定します。
ステップ 8	**match-address** 例： `Device(config-if-vrrp)# match-address`	（任意）アドバタイズメントパケットのセカンダリアドレスを設定したアドレスと照合します。 （注） セカンダリアドレスの照合は、デフォルトで有効になっています。
ステップ 9	**preempt delay minimum** *seconds* 例： `Device(config-if-vrrp)# preempt delay minimum 30`	（任意）優先順位の低いプライマリデバイスのプリエンプションは、オプションの遅延時間を指定して有効にします。 （注） プリエンプションはデフォルトでイネーブルです。
ステップ 10	**priority** *priority-level* 例： `Device(config-if-vrrp)# priority 3`	（任意）VRRP グループのプライオリティを指定します。 VRRP グループの優先度はデフォルトで 100 です。
ステップ 11	**timers advertise** 間隔 例： `Device(config-if-vrrp)# timers advertise 1000`	（任意）アドバタイズメントタイマーをミリ秒で設定します。 アドバタイズメントタイマーはデフォルトで 1000 ミリ秒に設定されています。

	コマンドまたはアクション	目的
ステップ 12	**vrrpv2** 例： `Device(config-if-vrrp)# vrrpv2`	（任意）互換モードで VRRPv2 設定デバイスのサポートを有効にします。
ステップ 13	**vrrs leader** *vrrs-leader-name* 例： `Device(config-if-vrrp)# vrrs leader leader-1`	（任意）VRRSに登録され、フォロワーに使用されるリーダーの名前を指定します。 （注） 登録済みの VRRS 名はデフォルトで使用不可になっています。
ステップ 14	**shutdown** 例： `Device(config-if-vrrp)# shutdown`	（任意）VRRP グループの VRRP 設定をディセーブルにします。 （注） VRRP の設定は、VRRP グループに対してはデフォルトでイネーブルになっています。
ステップ 15	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

FHRP クライアントの初期化前の遅延時間の設定

インターフェイス上のすべてのFHRPクライアントの初期化の前に遅延期間を設定するには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**fhrp version vrrp v3** 例： `Device(config)# fhrp version vrrp v3`	VRRPv3 および VRRS を設定する機能をイネーブルにします。
ステップ 4	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 0/0/0`	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**fhrp delay** {[**minimum**] [**reload**] *seconds*} 例： `Device(config-if)# fhrp delay minimum 5`	インターフェイスの起動後に、FHRP クライアントの初期化の遅延期間を指定します。 範囲は 0 ～ 3600 秒です。
ステップ 6	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

VRRPv3 プロトコル サポートの設定例

ここでは、VRRPv3 プロトコルサポートの設定例を示します。

例：デバイス上の VRRPv3 のイネーブル化

次の例は、デバイスで VRRPv3 をイネーブルにする方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config-if-vrrp)# end
```

例：VRRP グループの作成とカスタマイズ

次に、VRRP グループを作成およびカスタマイズする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# vrrp 3 address-family ipv4
```

```
Device(config-if-vrrp)# address 100.0.1.10 primary
Device(config-if-vrrp)# description group 3
Device(config-if-vrrp)# match-address
Device(config-if-vrrp)# preempt delay minimum 30
Device(config-if-vrrp)# end
```

（注） 上の例では、グローバル コンフィギュレーション モードで **fhrp version vrrp v3** コマンドが使用されています。

例：FHRP クライアントの初期化前の遅延時間の設定

次の例は、FHRP クライアントの初期化前の遅延時間の設定方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# fhrp delay minimum 5
Device(config-if-vrrp)# end
```

（注） 上記の例では、インターフェイスが表示されてから FHRP クライアントの初期化に 5 秒間の遅延時間が指定されています。遅延時間は 0 ～ 3600 秒の範囲で指定できます。

例：VRRP ステータス、設定、および統計情報の詳細

以下は、VRRP グループのステータス、設定、および統計情報の詳細の出力例です。

```
Device> enable
Device# show vrrp detail

 GigabitEthernet1/0/1 - Group 3 - Address-Family IPv4
  Description is "group 3"
  State is MASTER
  State duration 53.901 secs
  Virtual IP address is 100.0.1.10
  Virtual MAC address is 0000.5E00.0103
  Advertisement interval is 1000 msec
  Preemption enabled, delay min 30 secs (0 msec remaining)
  Priority is 100
  Master Router is 10.21.0.1 (local), priority is 100
  Master Advertisement interval is 1000 msec (expires in 832 msec)
  Master Down interval is unknown
  VRRPv3 Advertisements: sent 61 (errors 0) - rcvd 0
  VRRPv2 Advertisements: sent 0 (errors 0) - rcvd 0
  Group Discarded Packets: 0
    VRRPv2 incompatibility: 0
    IP Address Owner conflicts: 0
```

```
    Invalid address count: 0
    IP address configuration mismatch : 0
    Invalid Advert Interval: 0
    Adverts received in Init state: 0
    Invalid group other reason: 0
  Group State transition:
    Init to master: 0
    Init to backup: 1 (Last change Sun Mar 13 19:52:56.874)
    Backup to master: 1 (Last change Sun Mar 13 19:53:00.484)
    Master to backup: 0
    Master to init: 0
    Backup to init: 0

Device# exit
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
FHRP コマンド	『First Hop Redundancy Protocols Command Reference』
VRRPv2 の設定	『*Configuring VRRP*』
VRRPv3 コマンド	この章で使用するコマンドの完全な構文および使用方法の詳細。

標準および RFC

標準/RFC	タイトル
RFC5798	『*Virtual Router Redundancy Protocol*』

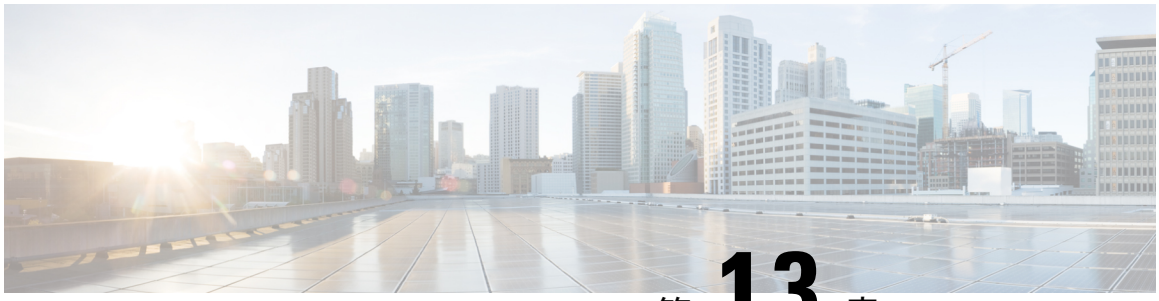
VRRPv3 プロトコルサポートの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	VRRPv3 プロトコルのサポート	VRRP は、デバイスのグループを使用して単一の仮想デバイスを形成し、冗長性を実現します。これにより、仮想デバイスをデフォルト ゲートウェイとして使用するように、LAN クライアントを設定できます。デバイスのグループを表す仮想デバイスは、「VRRP グループ」とも呼ばれます。 VRRPv3 プロトコルのサポート機能は、IPv4 と IPv6 アドレスをサポートするための機能を提供します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 13 章

WCCP の設定

このセクションでは、WCCP の設定について説明します。

はじめに

Web Cache Communication Protocol（WCCP）はシスコが開発したコンテンツルーティング テクノロジーです。IP パケットを代行受信し、IP パケットに指定されている宛先とは別の宛先にそのパケットをリダイレクトします。パケットは、インターネット上にある宛先の Web サーバーから、クライアントのローカルのコンテンツエンジンにリダイレクトされるのが一般的です。WCCP の展開シナリオによっては、Web サーバーからクライアント方向でもトラフィックをリダイレクトする必要があります。WCCP を使用すると、コンテンツエンジンをネットワーク インフラストラクチャに統合できます。

このマニュアルの作業では、ネットワークにコンテンツエンジンが設定済みであることを前提にしています。

WCCP の前提条件

- WCCP を使用するには、インターネットに接続されたインターフェイス上で IP を設定する必要があります。また、別のインターフェイスをコンテンツエンジンに接続する必要があります。
- コンテンツエンジンに接続するインターフェイスは、ファストイーサネットインターフェイスまたはギガビット イーサネット インターフェイスにする必要があります。

WCCP に関する制約事項

General

Web キャッシュ通信プロトコルバージョン 2（WCCPv2）には、次の制限が適用されます。

- WCCP は、IPv4 ネットワークだけで動作します。
- シスコエクスプレス フォワーディングをイネーブルにすると、WCCPによってネットワーク アドレス変換（NAT）がバイパスされます。
- WCCP には、ネットワークで同時に設定された NAT およびゾーンベース ファイアウォールとの相互運用性がありません。
- サービスグループは、最大 32 のコンテンツエンジンおよび 32 のスイッチで構成できます。
- マルチキャストクラスタにサービスを提供するスイッチの場合、存続可能時間（TTL）の値を 15 以下に設定する必要があります。
- クラスタのすべてのコンテンツエンジンは、クラスタにサービスを提供するすべてのデバイスと通信できるように設定する必要があります。
- マルチキャスト アドレスは、224.0.0.0 ～ 239.255.255.255 の範囲にする必要があります。
- 同じクライアント インターフェイスで同時に最大 8 個のサービス グループがサポートされます。
- レイヤ 2 のリライト転送メソッドはサポートされますが、Generic Routing Encapsulation（GRE）はサポートされません。
- レイヤ 2 モードが導入されている場合、コンテンツエンジンにレイヤ 2 を直接接続する必要があります。1 ホップ以上離れたレイヤ 3 接続はサポートされていません。
- Ternary CAM（TCAM）フレンドリ マスクベースの割り当てはサポートされますが、ハッシュ バケットベースの方式はサポートされません。
- TCAM スペースがなくなると、トラフィックはリダイレクトされず、通常どおりに転送されます。
- WCCP バージョン 2 規格では、最大 256 個のマスクをサポートします。ただし、Cisco Catalyst 9000 シリーズ スイッチは、単一のマスクへのマスク割り当てテーブルのみをサポートします。
- マスク割り当てに設定されているコンテンツエンジンが、割り当て方式としてハッシュが選択されているファームに参加しようとする場合、キャッシュ エンジンの割り当て方式が既存のファームの方式と一致しない限り、ファームに参加できません。
- WCCP リダイレクションは、マルチプロトコル ラベル スイッチング（MPLS）およびポートチャネル インターフェイスではサポートされていません。

- WCCP 高可用性は、モジュラ、スタッキング、および StackWise Virtual（SVL）モードではサポートされていません。
- コマンド **show ip wccp <*service_group*> detail** の出力に含まれる **packets redirected** カウンタは、CPU スイッチングでパケットがリダイレクトされた場合にのみ値が増加します。Cisco Express Forwarding（CEF）が着信 WCCP リダイレクション、L2 転送方式、およびマスク割り当てで使用されている場合は常に、すべての WCCP トラフィックがハードウェア経由でリダイレクトされます。トラフィックがハードウェア経由でリダイレクトされた場合、カウンタの値は増加しません。

Catalyst 9000 シリーズ スイッチのアクセス制御リスト

WCCP がマスク割り当てを使用している場合、リダイレクトリストはアプライアンスのマスク情報にマージされ、その結果としてマージされた ACL は Catalyst 9000 シリーズ スイッチ ハードウェアに渡されます。リダイレクト リストのプロトコルが IP であるか、サービス グループ プロトコルと完全に一致する場合、その許可 ACL または拒否 ACL のエントリだけが、アプライアンスのマスク情報にマージされます。

次の制約事項がリダイレクト リスト ACL に適用されます。

- ACL は、IPv4 拡張 ACL にする必要があります。
- 個々の発信元または宛先のポート番号だけを指定できます。ポート範囲は指定できません。
- 個々の発信元または宛先のポート番号以外の有効な一致基準は **dscp** と **tos** のみです。
- **fragments**、**time-range**、**options** キーワードや、TCP フラグは使用できません。
- リダイレクト ACL がこれらの制約事項を満たさない場合、次のエラー メッセージがログに記録されます。

```
WCCP-3-BADACE: Service <service group>, invalid access-list entry (seq:<sequence>,
reason:<reason>)
```

WCCP に関する情報

WCCP の概要

WCCP は、Cisco Content Engine（または WCCP を実行する他のコンテンツエンジン）を使用して、ネットワークのトラフィックパターンをローカライズし、ローカルでコンテンツ要求を実行できるようにします。トラフィックのローカライズによって伝送コストを引き下げ、ダウンロード時間を短縮できます。

WCCP によって、Cisco IOS XE プラットフォームはコンテンツ要求を透過的にリダイレクトできます。透過的リダイレクションを使用すると、ユーザーは、Web プロキシを使用するようにブラウザを設定せずに、コンテンツ要求をローカルで実行できます。ユーザーはターゲット

URL を使用してコンテンツを要求できます。また、ユーザーの要求はコンテンツ エンジンに自動的にリダイレクトされます。この場合の「透過的」とは、エンドユーザーが要求したファイル（Webページなど）が、元々指定していたサーバーからではなく、コンテンツエンジンから送信されることをそのユーザーが意識しないという意味です。

要求を受信したコンテンツエンジンは、独自のローカルキャッシュからサービスを提供しようとします。要求した情報が存在しない場合、コンテンツ エンジンから独自の要求が元のターゲット サーバーに発行され、必要な情報が取得されます。コンテンツエンジンは、要求された情報を取得すると、要求元のクライアントに転送し、以降の要求に対応するためにキャッシュします。その結果、ダウンロードのパフォーマンスが最大になり、送信コストが大幅に削減されます。

WCCPにより、一連のコンテンツエンジン（コンテンツエンジンクラスタと呼ばれる）が1つまたは複数のデバイスにコンテンツを提供できるようになります。ネットワーク管理者は、このようなクラスタ処理機能によって容易にコンテンツエンジンを拡張し、高いトラフィック負荷を管理できます。シスコ クラスタ処理テクノロジーを使用すると、各クラスタ メンバを同時に実行できるため、リニア スケーラビリティが実現します。クラスタ処理コンテンツ エンジンによって、キャッシュ ソリューションのスケーラビリティ、冗長性、および可用性が大幅に改善されます。最大 32 個のコンテンツ エンジンをクラスタ処理し、目的の容量まで拡張できます。

WCCP マスク割り当て

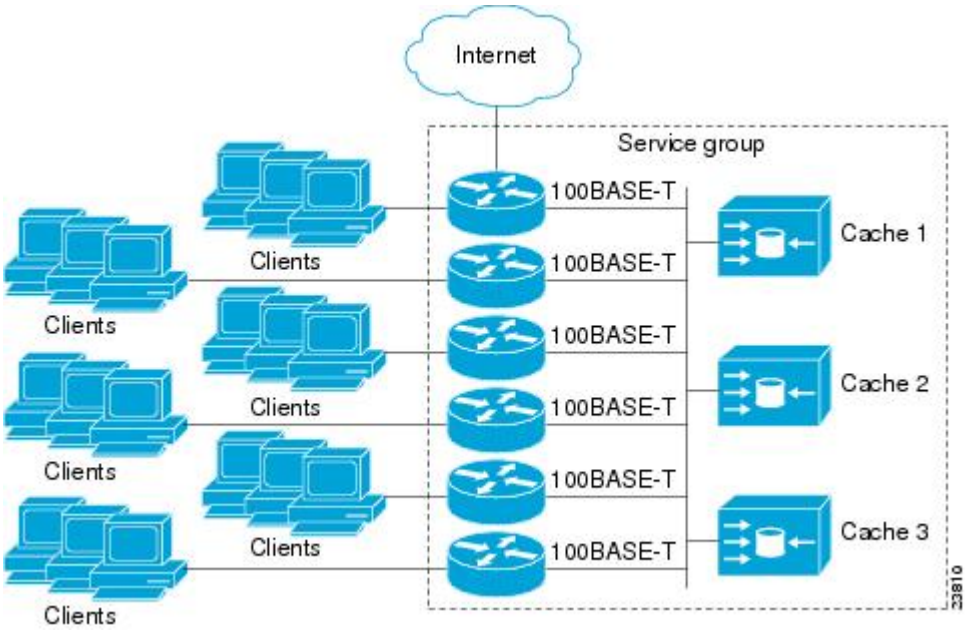
WCCPマスク割り当て機能によって、（デフォルトのハッシュ割り当て方式ではなく）WCCP サービスのロード バランシング方式としてマスク割り当てを使用できます。

Application and Content Networking System（ACNS）ソフトウェアを実行するコンテンツエンジンの場合、**mask-assign** キーワードを指定した **wccp custom-web-cache** コマンドを使用して、マスク割り当てを設定します。Cisco Wide Area Application Services（WAAS）ソフトウェアを実行するコンテンツエンジンの場合、**mask-assign** キーワードを指定した **wccp tcp-promiscuous** コマンドを使用して、マスク割り当てを設定します。

WCCPv2 の設定

複数のデバイスが WCCPv2 を使用して 1 つのコンテンツエンジンクラスタにサービスを提供できます。次の図に、複数のデバイスを使用した設定例を示します。

図 *19 : WCCPv2* を使用した *Cisco* コンテンツ エンジン ネットワーク構成



クラスタ、および同じサービスを実行しているクラスタに接続するデバイス内のコンテンツエンジンのサブセットは、サービスグループと呼ばれます。利用可能なサービスには、TCP および UDP リダイレクションが含まれます。

WCCPv2 の場合、各コンテンツエンジンがサービスグループ内のすべてのデバイスを認識している必要があります。サービスグループ内のすべてのデバイスのアドレスを指定するには、次のいずれかのメソッドを選択する必要があります。

- ユニキャスト：グループ内の各デバイスの IP アドレスリストを、各コンテンツエンジンで設定します。この場合、グループ内の各デバイスのアドレスは、設定の際、コンテンツエンジンごとに明示的に指定する必要があります。
- マルチキャスト：単一のマルチキャストアドレスを各コンテンツエンジンで設定します。マルチキャスト アドレス メソッドの場合、コンテンツエンジンは、サービスグループのすべてのスイッチに提供されるシングルアドレス通知を送信します。たとえば、コンテンツエンジンは、パケットを常にマルチキャストアドレス 224.0.0.100 に送信するように指示できます。その場合、マルチキャストパケットは、WCCP を使用してリッスンしているグループ用に設定されたサービスグループ内のすべてのデバイスに送信されます（詳細については、**ip wccp group-listen** インターフェイス コンフィギュレーション コマンドを参照してください）。

マルチキャスト オプションの場合に必要な操作は、各コンテンツ エンジンで単一のアドレスを指定することだけなので、設定が容易です。このオプションを使用して、サービス グループからルータを動的に追加および削除できます。毎回、異なるアドレス リストを使用してコンテンツ エンジンを再設定する必要はありません。

WCCPv2 での設定は次の順序で行います。

1. 各コンテンツエンジンは、ルータリストを使用して設定されます。
2. 各コンテンツエンジンは、各自の存在と、通信の確立に使用されたすべてのデバイスのリストについて通知します。ルータは、グループ内のコンテンツ エンジンのビュー（リスト）で応答します。
3. そのビューがクラスタ内のすべてのコンテンツエンジンで一貫している場合、1 つのコンテンツエンジンがリードとして指定され、デバイスがパケットのリダイレクト時に展開する必要のあるポリシーが設定されます。

HTTP 以外のサービスの WCCPv2 サポート

WCCPv2 では、さまざまな UDP および TCP トラフィックを含め、HTTP（TCP ポート 80 トラフィック）以外のトラフィックのリダイレクションが可能です。WCCPv2 では他のポート宛てのパケットをリダイレクトできます。たとえば、プロキシ Web キャッシュ処理、ファイル転送プロトコル（FTP）キャッシング、FTP プロキシの処理、80 以外のポートの Web キャッシング、Real Audio、ビデオアプリケーション、およびテレフォニーアプリケーションに使用されるポートなどです。

各種の利用可能なサービスに対応するため、WCCPv2 は複数のサービスグループという概念を導入しました。サービス情報は、ダイナミックサービス識別番号（98 など）または事前定義したサービスキーワード（**web-cache** など）を使用して、WCCP コンフィギュレーション コマンドで指定します。この情報は、サービス グループ メンバーが同じサービスを使用または提供していることを確認するために使用されます。

サービス グループのコンテンツ エンジンは、プロトコル（TCP または UDP）によってリダイレクトされるトラフィックと、最大 8 個の発信元ポートまたは宛先ポートを指定します。各サービス グループにはプライオリティ ステータスが割り当てられます。ダイナミック サービスのプライオリティは、コンテンツエンジンによって割り当てられます。プライオリティ値の範囲は、0 ～ 255 です（0 が最も低いプライオリティ）。事前定義した Web キャッシュ サービスには、240 のプライオリティが割り当てられています。

複数デバイスでの WCCPv2 サポート

WCCPv2 では、複数のデバイスをキャッシュエンジンのクラスタに追加できます。サービスグループで複数のデバイスを使用すると、冗長構成、インターフェイスの集約、およびリダイレクトの負荷分散が可能になります。WCCPv2 は、サービスグループごとに最大 32 のデバイスをサポートします。各サービス グループの確立および保守は独立して行われます。

WCCPv2 での MD5 セキュリティ

WCCPv2 には、パスワードとハッシュメッセージ認証コード-メッセージダイジェスト（HMAC MD5）規格を使用して、サービスグループの一部になるスイッチとコンテンツエンジンを制御できる、オプションの認証機能があります。共有秘密キー MD5 ワンタイム認証（**ip wccp password** *password* グローバル コンフィギュレーション コマンドを使用して設定）では、メッセージを代行受信、検査、およびリプレイから保護します。

WCCPv2 での Web キャッシュ パケットのリターン

エラーまたは過負荷のために、コンテンツエンジンが、キャッシュした要求オブジェクトを提供できない場合、コンテンツエンジンは、元々指定されていた宛先サーバーに転送するように、要求をデバイスに返します。WCCPv2 には、機能していないコンテンツ エンジンから返送された要求を判断できるパケットのチェック機能があります。デバイスは、この情報を使用して（要求をコンテンツエンジンクラスタに再送信しようとするのではなく）要求を元の宛先サーバーに転送できます。このプロセスのエラー処理はクライアントに意識されません。

コンテンツエンジンがパケットを拒否し、パケット返送機能を開始する場合、一般的に次のような理由があります。

- コンテンツ エンジンが過負荷になり、パケットを処理する余裕がなくなった場合
- コンテンツエンジンが、パケットのキャッシング機能が低下する特定の条件についてフィルタリングしている場合（たとえば、IP 認証が有効になった場合）

WCCPv2 での負荷分散

WCCPv2 を使用すると、個々のコンテンツエンジンに割り当てる負荷を調整して、空きリソースを効率的に使用できるようになります。さらに、クライアントに対して高い Quality Of Service（QoS）を確保できます。WCCPv2 を使用すると、指定したコンテンツ エンジンが特定のコンテンツ エンジン上の負荷を調整し、クラスタ内のコンテンツ エンジン全体で負荷を分散できます。WCCPv2 では負荷分散を実行するために、次の 3 つの方法を使用します。

- ホットスポット処理：個々のハッシュバケットをすべてのコンテンツエンジンに分散できます。WCCPv2 の登場までは、1 つのハッシュバケットの情報を転送できるのは、1 つのコンテンツエンジンに対してのみでした。
- ロードバランシング：過負荷のコンテンツエンジンから、空き容量がある他のメンバに負荷を移行するように、コンテンツエンジンに割り当てるハッシュバケットセットを調整できます。
- 負荷制限：コンテンツエンジンの容量を超えないように、スイッチが負荷を選択してリダイレクトできるようにします。

これらのハッシュ処理パラメータを使用すると、コンテンツエンジンの過負荷を防ぎ、障害が発生する可能性を軽減します。

WCCP バイパス パケット

WCCP は IP パケットを代行受信し、IP ヘッダーに指定されている宛先以外の宛先に、そのパケットをリダイレクトします。パケットは、インターネット上にある Web サーバーから、宛先のローカルの Web キャッシュにリダイレクトされるのが一般的です。

場合によっては、Web キャッシュでリダイレクトされたパケットを適切に管理できず、パケットを変更せずに元のデバイスに返送することがあります。このようなパケットはバイパスパケットと呼ばれ、カプセル化なしのレイヤ 2 転送（L2）を使用して、発信元のデバイスに返送

されます。デバイスはカプセル化を解除し、通常どおりにパケットを転送します。入力インターフェイスと関連付けられている VRF（関連付けられている VRF がない場合はグローバルテーブル）は、パケットを宛先にルーティングするときに使用されます。

WCCP クローズド サービスおよびオープン サービス

パケットを代行受信し、Cisco スイッチまたはルータによって外部 WCCP クライアントデバイスにリダイレクトするアプリケーションの場合、WCCP クライアントデバイスを使用できないと、状況によってはアプリケーションのパケットをブロックする必要があります。このブロックを実行するには、WCCP クローズド サービスを設定します。WCCP サービスがクローズドに設定されている場合、サービスを提供するもののアクティブなクライアントデバイスを持たないパケットは破棄されます。

デフォルトでは、WCCP はオープン サービスとして動作します。この場合、中間デバイスがなくても、クライアントとサーバー間の通信は正常に進行します。

ip wccp service-list コマンドは、クローズドモードとオープンモード両方のサービスに使用できます。アプリケーション プロトコル タイプまたはポート番号を登録するには、**service-list** キーワードと *service-access-list* 引数を使用します。オープンサービスまたはクローズドサービスを選択するには、**mode** キーワードを使用します。

WCCP 発信 ACL チェック

入力インターフェイスで WCCP のリダイレクションが有効になっている場合、パケットは WCCP によってリダイレクトされ、代わりに IP ヘッダーで指定された宛先以外のインターフェイスで出力されます。パケットは、引き続き入力インターフェイスで設定された ACL の影響下にあります。ただし、リダイレクションによって、パケットが元の出力インターフェイスで設定された ACL をバイパスする可能性があります。元の出力インターフェイスで ACL が設定されているためにドロップされたパケットは、リダイレクト出力インターフェイスに送信される場合があります。その結果、セキュリティ上の問題が発生する可能性があります。WCCP アウトバウンド ACL チェック機能を有効にすると、リダイレクトされたパケットは、元の出力インターフェイスで設定された ACL 条件の対象になります。

WCCP サービス グループ

WCCP は、Cisco IOS XE ソフトウェアのコンポーネントで、定義済みの特性を持つトラフィックを元の宛先から代替の宛先へとリダイレクトします。一般的な WCCP アプリケーションには、リモート Web サーバー宛ての発信トラフィックをローカル Web キャッシュにリダイレクトして、応答時間を改善し、ネットワーク リソースの使用状況を最適化する機能があります。

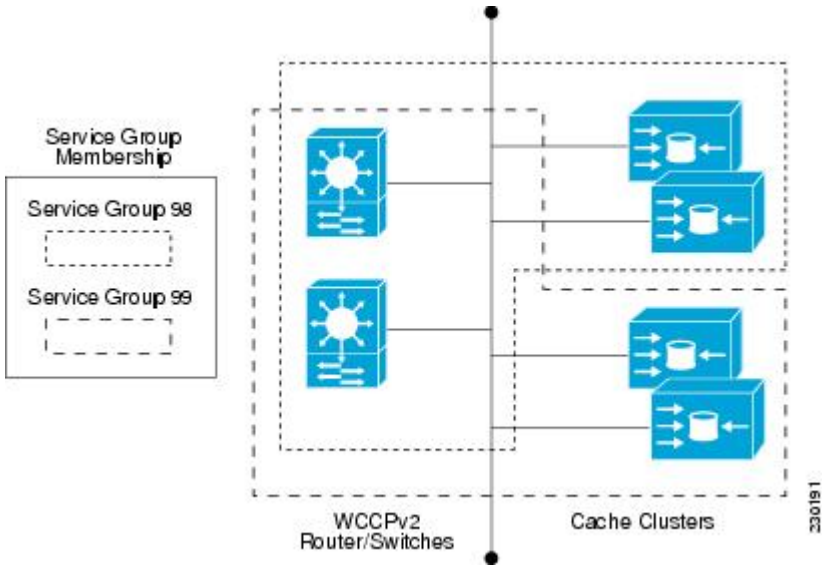
リダイレクトに選択されるトラフィックの性質は、コンテンツエンジンで指定されるサービス グループ（下の図を参照）によって定義され、WCCP を使用してスイッチやルータに伝達されます。

WCCPv2 は、サービスグループごとに最大 32 のスイッチをサポートします。各サービス グループの確立および保守は独立して行われます。

WCCPv2 では、トラフィックの代行受信およびリダイレクションを行うために使用されている論理リダイレクション サービスを基にサービス グループを使用します。標準のサービスは Web キャッシュです。Web キャッシュは TCP ポート 80（HTTP）トラフィックを代行受信し、そのトラフィックをコンテンツ エンジンにリダイレクトします。Web キャッシュ サービスの特徴はスイッチとコンテンツエンジンの両方から認識されているため、このサービスは既知のサービスと呼ばれます。サービスの識別よりも詳細な既知のサービスの説明は必要ありません。標準の Web キャッシュ サービスを指定するには、**ip wccp** コマンドと **web-cache** キーワードを使用します。

（注） スイッチでは同時に複数のサービスが実行できます。また、スイッチとコンテンツエンジンは、同時に複数のサービスグループの一部になることができます。

図 *20 : WCCP* サービス グループ



ダイナミック サービスは、コンテンツエンジンによって定義されます。コンテンツエンジンは、代行受信するプロトコルまたはポート、およびトラフィックの配信方法をスイッチに指示します。ダイナミック サービス グループのトラフィックの特性に関する情報は、スイッチ自体にはありません。この情報は、グループに参加する最初のコンテンツエンジンから提供されるためです。ダイナミック サービスでは、1 つのプロトコルに最大 8 ポートを指定できます。

たとえば、Cisco Content Engine ではダイナミック サービス 99 を使用して、リバース プロキシ サービスを指定します。ただし、他のコンテンツ エンジン デバイスでは、その他のサービスにこのサービス番号を使用する可能性があります。

WCCP：すべてのサービスを確認

インターフェイスは、WCCP サービスを複数使用して設定できます。1 つのインターフェイスに複数の WCCP サービスを設定する場合、サービスの優先順位は、他の設定済みサービスの

プライオリティと比較した、そのサービスの相対的なプライオリティによって変わります。各 WCCP サービスには、定義の一部にプライオリティ値があります。複数の WCCP サービスを使用してインターフェイスを設定する場合、パケットの優先順位は、プライオリティ順でサービス グループに対して対応付けられます。

（注） WCCP サービスグループの優先順位は、Cisco IOS XE ソフトウェアで設定できません。

ip wccp check services all コマンドを使用すると、すべての設定済みサービスを一致についてチェックし、必要に応じてそのサービスに関するリダイレクションを実行するように WCCP を設定できます。パケットのリダイレクト先キャッシュは、リダイレクト ACL およびサービスの優先順位で制御できます。複数の WCCP サービスをサポートするには、**ip wccp check services all** コマンドをグローバルレベルで設定する必要があります。

WCCP サービスをリダイレクト ACL を使用して設定する場合、IP パケットに一致するサービスが見つかるまで、プライオリティ順にサービスがチェックされます。パケットに一致するサービスがない場合、パケットはリダイレクトされません。サービスがパケットに一致し、サービスにリダイレクト ACL が設定されている場合、IP パケットは ACL に対してチェックされます。ACL によってパケットが拒否される場合、**ip wccp check services all** コマンドを設定しない限り、低い優先順位のサービスにパケットは渡されません。**ip wccp check services all** コマンドを設定すると、インターフェイスで設定されている残りの低い優先順位のサービスに対して、引き続きパケットのマッチングが試行されます。

WCCP のトラブルシューティングのヒント

WCCP をイネーブルにすると、CPU の使用率が非常に高くなる場合があります。WCCP カウンタを使用すると、直接スイッチでバイパストラフィックを確認できます。また、その原因が WCCP の有効化による CPU の使用率の高さにあるかどうかを示すことができます。場合によっては 10 % のバイパス トラフィックが標準で、他の状況では 10 % が高いこともあります。ただし、25% を超える数値の場合、Web キャッシュの状況をより詳しく調査する必要があります。

バイパス トラフィックのレベルが高いことをカウンタが示している場合、次の手順は、コンテンツ エンジンのバイパス カウンタを確認し、コンテンツ エンジンがトラフィックのバイパスを選択した理由を判定します。さらに詳細に調査するには、コンテンツ エンジン コンソールにログインし、CLI を使用します。カウンタを使用すると、バイパスするトラフィックの割合を決定できます。

特定のサービスに関してデバイスで保持している WCCP 統計情報（カウント）を削除するには、**clear wccp** コマンドを使用します。

すべての WCCP グローバル統計情報（カウント）を表示するには、**show wccp** コマンドを使用します。

WCCP の設定方法

次の設定作業では、ネットワークで使用するコンテンツエンジンのインストールと設定が完了していることを前提としています。クラスタでコンテンツエンジンを設定してから、ルータまたはスイッチの WCCP 機能を設定する必要があります。コンテンツエンジンの設定とセットアップ作業については、『Cisco Cache Engine User Guide』を参照してください。

WCCP の設定

WCCP を設定するには、次の作業を実行します。

ip wccp {**web-cache** | *service-number*} グローバル コンフィギュレーション コマンドを使用して WCCP サービスを設定しない限り、WCCP はデバイスに対して無効です。特定の形式の **ip wccp** コマンドを最初に使用したときに、WCCP が有効になります。

サービスグループのデバイスとコンテンツエンジンのパスワードを設定するには、**ip wccp web-cache password** コマンドを使用します。MD5 パスワードセキュリティの場合、サービスグループのパスワードを使用して、サービスグループに参加させる各デバイスおよびコンテンツエンジンを設定する必要があります。パスワードの長さは、8 文字以下である必要があります。サービスグループの各コンテンツエンジンまたはデバイスは、WCCP メッセージヘッダーの検証後すぐに、受信した WCCP パケットのセキュリティコンポーネントを認証します。認証に失敗したパケットは廃棄されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** **例：** `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **例：** `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password* [**0** | **7**]] **例：** `Device(config)# ip wccp web-cache password pwd`	デバイスで有効にする Web キャッシュまたはダイナミックサービスを指定します。サービスグループで使用する IP マルチキャストアドレスを指定します。使用するアクセスリストを指定します。MD5 認証を使用するかどうかを指定します。WCCP サービスを有効にします。

	コマンドまたはアクション	目的
		• （注） パスワードの長さは、8文字以内にする必要があります。
ステップ 4	**interface** *type number* 例： `Device(config)# interface Gigabitethernet 0/0`	Web キャッシュ サービスを実行するインターフェイス番号を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip wccp** {**web-cache** | *service-number*} **redirect** {**in** | **out**} 例： `Device(config-if)# ip wccp web-cache redirect in`	WCCP を使用して、発信インターフェイスまたは受信インターフェイスでパケットのリダイレクションをイネーブルにします。 • **out** および **in** キーワードオプションに示されているとおり、発信インターフェイスまたは受信インターフェイスのリダイレクションを指定できます。
ステップ 6	**exit** 例： `Device(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 7	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 0/2/0`	リダイレクトからトラフィックを除外するインターフェイス番号を対象として、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 wccp redirect exclude in** 例： `Device(config-if)# ip wccp redirect exclude in`	（任意）指定したインターフェイスのトラフィックをリダイレクションから除外します。

クローズド サービスの設定

WCCP 用のサービス グループの数を指定し、クローズド サービスまたはオープン サービスとしてサービス グループを設定し、オプションで全サーバーのチェックを指定するには、この作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

ステップ 3	次のいずれかのコマンドを入力します。
	• **ip wccp** *service-number* [**service-list** *service-access-list* **mode** {**open** | **closed**}]
	• または
	• **ip wccp web-cache mode** {**open** | **closed**}
	例：
	`Device(config)# ip wccp 90 service-list 120 mode closed`
	または
	`Device(config)# ip wccp web-cache mode closed`
		ダイナミック WCCP サービスをクローズドまたはオープンとして設定します。
		または
		Web キャッシュ サービスをクローズドまたはオープンとして設定します。
		（注） Web キャッシュ サービスをクローズド サービスとして設定する場合、サービス アクセス リストを指定できません。
		（注） ダイナミック WCCP サービスをクローズド サービスとして設定する場合、サービス アクセス リストを指定する必要があります。

ステップ 4	**ip wccp check services all** 例： `Device(config)# ip wccp check services all`	（任意）WCCP サービスのチェックをイネーブルにします。 • このコマンドを使用すると、一致について他の設定済みサービスをチェックし、必要に応じてそのサービスについてリダイレクションを実行するように WCCP を設定できます。パケットのリダイレクト先キャッシュは、サービス記述だけでなく、リダイレクト ACL によって制御できます。

	コマンドまたはアクション	目的
		（注） **ip wccp check services all** コマンドは、すべてのサービスに適用され、単一のサービスには関連付けられないグローバル WCCP コマンドです。
ステップ 5	**ip wccp** {**web-cache** | *service-number*} 例： `Device(config)# ip wccp 201`	WCCP サービス ID を指定します。 • 標準の Web キャッシュ サービスまたはダイナミック サービス番号（0 ～ 255）を指定できます。 • 指定できるサービスの最大数は256です。
ステップ 6	**exit** 例： `Device(config)# exit`	特権 EXEC モードに戻ります。

マルチキャストアドレスへのデバイスの登録

サービスグループにマルチキャストアドレスオプションを使用する場合、デバイスがインターフェイスでマルチキャスト ブロードキャストを待ち受けるように設定する必要があります。

リダイレクトされたトラフィックが仲介デバイスを経由する必要のあるネットワーク設定の場合、経由対象のデバイスは、IPマルチキャストルーティングを実行するように設定する必要があります。仲介デバイスの経由を有効にするには、次の2つのコンポーネントを設定してください。

- **ip multicast-routing** グローバル コンフィギュレーション コマンドを使用して、IP マルチキャストルーティングを有効にします。
- **ip wccp group-listen** インターフェイス コンフィギュレーション コマンドを使用して、キャッシュエンジンの接続先のインターフェイスが、マルチキャストの送信を受信できるようにします。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip multicast-routing** [**vrf** *vrf-name*] [**distributed**] 例： `Device(config)# ip multicast-routing`	IP マルチキャスト ルーティングを有効にします。
ステップ 4	**ip wccp** {**web-cache** | *service-number*} **group-address** *multicast-address* 例： `Device(config)# ip wccp 99 group-address 239.1.1.1`	サービス グループのマルチキャスト アドレスを指定します。
ステップ 5	**interface** *type number* 例： `Device(config)# interface ethernet 0/0`	コンテンツ エンジンの接続先インターフェイスが、Web キャッシュ サービスが実行するマルチキャスト送信を受信できるようにし、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**ip pim** {**sparse-mode** | **sparse-dense-mode** | **dense-mode** [**proxy-register** { **list** *access-list* | **route-map** *map-name*}]} 例： `Device(config-if)# ip pim dense-mode`	（任意）インターフェイスで Protocol Independent Multicast（PIM）をイネーブルにします。 （注） Catalyst 9000 シリーズ スイッチで **ip wccp group-listen** コマンドを適切に動作させるには、**ip wccp group-listen** コマンドに加えて、**ip pim** コマンドを入力する必要があります。
ステップ 7	**ip wccp** {**web-cache** | *service-number*} **group-listen** 例： `Device(config-if)# ip wccp 99 group-listen`	インターフェイスを設定して、WCCP の IP マルチキャスト パケットの受信をイネーブルまたはディセーブルにします。

WCCP サービス グループのアクセス リストの使用

どのトラフィックをどのコンテンツエンジンに送信するかを決定するためにアクセスリストを使用するようにデバイスを設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* **remark** *remark* 例： `Device(config)# access-list 1 remark Give access to user1`	（任意）アクセス リスト エントリに関してユーザーにわかりやすいコメントを追加します。 • 最大 100 文字の注釈をアクセス リスト エントリの前または後に指定できます。
ステップ 4	**access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**] 例： `Device(config)# access-list 1 permit 172.16.5.22 0.0.0.0`	キャッシュエンジンへのトラフィックリダイレクトを有効または無効にし、送信元アドレスとワイルドカードマスクに基づいて指定された送信元を許可するアクセスリストを作成します。 • すべてのアクセスリストには、1つ以上の許可文が必要です。許可文は、最初のエントリである必要はありません。 • 標準 IP アクセス リストには、1 ～ 99 または 1300 ～ 1999 の番号を付けます。 • *source-wildcard* を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。 • 必要に応じて、*source source-wildcard* の代わりに、キー

	コマンドまたはアクション	目的
		ワード **any** を使用して、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。 • この例では、ホスト 172.16.5.22 がアクセス リストに合格できます。
ステップ 5	**access-list** *access-list-number* **remark** *remark* 例： `Device(config)# access-list 1 remark Give access to user1`	（任意）アクセス リスト エントリに関してユーザーにわかりやすいコメントを追加します。 • 最大 100 文字の注釈をアクセス リスト エントリの前または後に指定できます。
ステップ 6	**access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} | [**log**] 例： `Device(config)# access-list 1 deny 172.16.7.34 0.0.0.0`	送信元アドレスおよびワイルドカード マスクに基づいて、指定した送信元を拒否します。 • *source-wildcard* を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。 • 必要に応じて、*source source-wildcard* の代わりに省略形 any を使用すると、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。 • この例では、ホスト 172.16.7.34 はアクセス リストへの合格が拒否されます。
ステップ 7	アクセス リストの基礎とする送信元の指定が完了するまで、ステップ 3 ～ 6 の手順を繰り返します。	明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。
ステップ 8	**ip wccp web-cache group-list** *access-list* 例： `Device(config) ip wccp web-cache group-list 1`	パケットを受け入れるコンテンツエンジンの IP アドレスをデバイスに示します。

	コマンドまたはアクション	目的
ステップ 9	**ip wccp web-cache redirect-list** *access-list* 例： `Device(config)# ip wccp web-cache redirect-list 1`	（任意）特定のクライアントのキャッシングをディセーブルにします。

WCCP 発信 ACL チェックのイネーブル化

（注） ハードウェアですべてのリダイレクションを実行する場合、発信 ACL チェック処理をイネーブルにすると、リダイレクションのモードは変わります。ショートカットをインストールする前に、追加の ACL チェックがソフトウェアで実行できるように、最初のパケットは切り替えられます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password*] 例： `Device(config)# ip wccp web-cache`	Cisco Content Engine のサービス グループまたはコンテンツ エンジンのサービス グループのサポートをイネーブルにし、リダイレクト ACL リストまたはグループ ACL を設定します。 （注） **web-cache** キーワードは WCCP バージョン 1 とバージョン 2 に使用することができ、*service-number* 引数は WCCP バージョン 2 のみに使用できます。
ステップ 4	**ip wccp check acl outbound** 例：	WCCP によってリダイレクトされたパケットの出力インターフェイスのアクセ

	コマンドまたはアクション	目的
	`Device(config)# ip wccp check acl outbound`	スコントロールリスト（ACL）をチェックします。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーションを終了します。

WCCP 設定の確認およびモニタリング

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show ip wccp** [**web-cache** |*service-number*] [**detail view**] 例： `Device# show ip wccp 24 detail`	WCCP に関連するグローバル情報を表示します。たとえば、実行されているプロトコル バージョン、ルータ サービス グループのコンテンツ エンジンの数、ルータに接続できるコンテンツ エンジン グループ、使用するアクセス リストなどです。 • *service-number*：（任意）コンテンツエンジンで制御される Web キャッシュサービスグループのダイナミック番号。有効な範囲は 0 ～ 99 です。Cisco Content Engine を使用する Web キャッシュの場合、逆プロキシ サービスは 99 の値で示されます。 • **web-cache**：（任意）Web キャッシュサービスの統計情報。 • **detail**：（任意）検出済み、または検出されていない特定のサービスグループまたは Web キャッシュの他のメンバ。

	コマンドまたはアクション	目的
		• **view**：（任意）ルータまたはすべての Web キャッシュに関する情報。
ステップ 3	**show ip interface** 例： `Device# show ip interface`	「Web Cache Redirect is enabled / disabled」など、いずれかの **ip wccp redirection** コマンドがインターフェイスで設定されているかどうかに関するステータスを表示します。
ステップ 4	**more system:running-config** 例： `Device# more system:running-config`	（任意）実行されている構成ファイルのコンテンツを表示します（**show running-config** コマンドと同じです）。

WCCP の設定例

例：一般的な WCCPv2 セッションの設定

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100 password password
Device(config)# ip wccp source-interface GigabitEthernet 0/1/0
Device(config)# ip wccp check services all
! Configures a check of all WCCP services.
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device(config)# interface GigabitEthernet 0/2/0
Device(config-if)# ip wccp redirect exclude in
Device(config-if)# exit
```

例：デバイスとコンテンツエンジンのパスワードの設定

```
Device# configure terminal
Device(config)# ip wccp web-cache password password1
```

例：Web キャッシュ サービスの設定

```
Device# configure terminal
Device(config)# ip wccp web-cache
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# copy running-config startup-config
```

次に、ギガビットインターフェイス 0/1/0 に到達する HTTP トラフィックのリダイレクションを有効にするセッションの設定例を示します。

```
Device# configure terminal
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# show ip interface GigabitEthernet 0/1/0
.
.
.
WCCP Redirect inbound is enabled
WCCP Redirect exclude is disabled
.
.
.
```

例：逆プロキシ サービスの実行

次の例では、Cisco Cache Engine を使用してサービス グループを設定し、ダイナミック サービス 99 を使用して逆プロキシ サービスを実行しているという前提です。

```
Device# configure terminal
Device(config)# ip wccp 99
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

例：マルチキャストアドレスへのデバイスの登録

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache group-listen
```

次に、マルチキャストアドレス 224.1.1.1 を使用してリバースプロキシサービスを実行するようにデバイスを設定する例を示します。リダイレクションは、ギガビットイーサネットインターフェイス 0/1/0 経由で送信されるパケットに適用されます。

```
Device# configure terminal
Device(config)# ip wccp 99 group-address 224.1.1.1
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

例：アクセス リストの使用

セキュリティを改善するには、標準のアクセスリストを使用して、現在のデバイスに登録するコンテンツエンジンで有効なアドレスがどの IP アドレスかをデバイスに通知します。次に、サンプル ホストのアクセス リスト番号が 10 である標準的なアクセス リストの設定セッション例を示します。

```
Device(config)# access-list 10 permit host 10.1.1.1
```

```
Device(config)# access-list 10 permit host 10.1.1.2
Device(config)# access-list 10 permit host 10.1.1.3
Device(config)# ip wccp web-cache group-list 10
```

特定のクライアント、サーバー、またはクライアント/サーバー ペアに対してキャッシングをディセーブルにするには、WCCP アクセス リストを使用します。次に、10.1.1.1 から 10.3.1.1 に送信される要求がキャッシュをバイパスし、その他すべての要求は通常どおりに処理される例を示します。

```
Device(config)# ip wccp web-cache redirect-list 120
Device(config)# access-list 120 deny tcp host 10.1.1.1 any
Device(config)# access-list 120 deny tcp any host 10.3.1.1
Device(config)# access-list 120 permit ip any any
```

次の例では、ギガビットイーサネット 0/1/0 を介して受信した Web 関連のパケットを、209.165.200.224 以外の任意のホストにリダイレクトするようにデバイスを設定します。

```
Device(config)# access-list 100 deny ip any host 209.165.200.224
Device(config)# access-list 100 permit ip any any
Device(config)# ip wccp web-cache redirect-list 100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
```

例：WCCP 発信 ACL チェックの設定

次に、ネットワーク 10.0.0.0 からのトラフィックがギガビットイーサネットインターフェイス 0/1/0 を離れないようにアクセスリストを設定する例を示します。発信 ACL チェックはイネーブルなので、WCCP はそのトラフィックをリダイレクトしません。WCCP は、パケットのリダイレクト前に、ACL に対してパケットをチェックします。

```
Device(config)# ip wccp web-cache
Device(config)# ip wccp check acl outbound
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip access-group 10 out
Device(config-if)# exit
Device(config)# ip wccp web-cache redirect-list redirect-out
Device(config)# access-list 10 deny 10.0.0.0 0.255.255.255
Device(config)# access-list 10 permit any
```

発信 ACL チェックをディセーブルにする場合、ネットワーク 10.0.0.0 からの HTTP パケットを Web キャッシュにリダイレクトします。そのネットワーク アドレスを使用するユーザーは、ネットワーク管理者が回避しようとしても、Web ページを取得できます。

例：WCCP 設定の確認

次に、特権 EXEC モードで **more system:running-config** コマンドを使用して設定の変更を検証する例を示します。次に、Web キャッシュサービスおよびダイナミックサービス 99 の両方をデバイスで有効にする例を示します。

```
Device# more system:running-config

 Building configuration...
```

```
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
hostname router4
!
enable secret 5 $1$nSVy$faliJsVQXVPW.KuCxZNTh1
enable password password1
!
ip subnet-zero
ip wccp web-cache
ip wccp 99
ip domain-name cisco.com
ip name-server 10.1.1.1
ip name-server 10.1.1.2
ip name-server 10.1.1.3
!
!
!
interface GigabitEthernet0/1/1
ip address 10.3.1.2 255.255.255.0
no ip directed-broadcast
ip wccp web-cache redirect in
ip wccp 99 redirect in
no ip route-cache
no ip mroute-cache
!
interface GigabitEthernet0/1/0
ip address 10.4.1.1 255.255.255.0
no ip directed-broadcast
ip wccp 99 redirect in
no ip route-cache
no ip mroute-cache
!
interface Serial0
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
!
interface Serial1
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
!
ip default-gateway 10.3.1.1
ip classless
ip route 0.0.0.0 0.0.0.0 10.3.1.1
no ip http server
!
!
!
line con 0
transport input none
line aux 0
transport input all
```

```
line vty 0 4
password password1
login
!
end
```

次に、WCCP に関連したグローバル統計情報を表示する方法の例を示します。

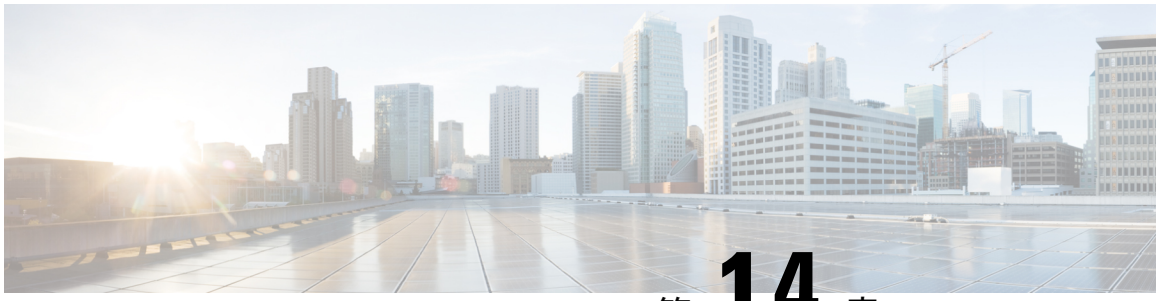
```
Device# show ip wccp web-cache detail

WCCP Client information:
WCCP Client ID:      10.1.1.2
Protocol Version:    2.0
State:               Usable
Redirection:         L2
Packet Return:       L2
Packets Redirected:  0
Connect Time:        00:20:34
Assignment:          MASK
Mask  SrcAddr    DstAddr    SrcPort DstPort
----  -------    -------    ------- -------
0000: 0x00000000 0x00001741 0x0000  0x0000
Value SrcAddr    DstAddr   SrcPort DstPort CE-IP
----- -------    -------   ------- ------- -----
0000: 0x00000000 0x00000000 0x0000 0x0000 0x3C010102 (10.1.1.2)
0001: 0x00000000 0x00000001 0x0000 0x0000 0x3C010102 (10.1.1.2)
0002: 0x00000000 0x00000040 0x0000 0x0000 0x3C010102 (10.1.1.2)
0003: 0x00000000 0x00000041 0x0000 0x0000 0x3C010102 (10.1.1.2)
0004: 0x00000000 0x00000100 0x0000 0x0000 0x3C010102 (10.1.1.2)
0005: 0x00000000 0x00000101 0x0000 0x0000 0x3C010102 (10.1.1.2)
0006: 0x00000000 0x00000140 0x0000 0x0000 0x3C010102 (10.1.1.2)
```

show ip wccp web-cache コマンドの詳細については、『*Cisco IOS IP Application Services Command Reference*』を参照してください。

WCCP の機能情報

表 *6 : WCCP* の機能情報

機能名	リリース	機能情報
Cisco Catalyst 9300 シリーズ スイッチでの WCCP サポート	Cisco IOS XE Everest 16.6.1	Web Cache Communication Protocol（WCCP）はシスコが開発したコンテンツルーティングテクノロジーです。IP パケットを代行受信し、IP パケットに指定されている宛先とは別の宛先にそのパケットをリダイレクトします。 WCCP を使用すると、コンテンツエンジンをネットワーク インフラストラクチャに統合できます。 .



第 14 章

拡張オブジェクト トラッキングの設定

拡張オブジェクトトラッキングの制約事項

すべてのトラッキング設定は、デバイスのリロード後にすべてのレイヤ3サブインターフェイスで再設定する必要があります。トラッキング設定を再設定する前に、すべてのレイヤ3サブインターフェイスがアクティブになるように、デバイスでのすべてのリロード操作を完了しておく必要があります。

拡張オブジェクト トラッキングに関する情報

ここでは、拡張オブジェクトトラッキングに関する情報について説明します。

拡張オブジェクト トラッキングの概要

拡張オブジェクト トラッキング機能が導入される前は、ホットスタンバイ ルータ プロトコル（HSRP）に単純なトラッキング メカニズムが内蔵されていますた。このメカニズムでは、インターフェイスのラインプロトコルのステートしか追跡することができませんでした。インターフェイスのラインプロトコル ステートがダウンになった場合、ルータのHSRP優先度は削減され、より高い優先度のもう 1 つの HSRP ルータがアクティブになることができます。

拡張オブジェクトトラッキング機能は、HSRPからトラッキング メカニズムを分離させて、独立したトラッキング プロセスを別途生成します。これにより、HSRP以外のプロセスがこのトラッキング プロセスを使用できます。この機能を使用すると、インターフェイスのラインプロトコルのステートに加えて他のオブジェクトも追跡できます。

HSRP、仮想ルータ冗長プロトコル（VRRP）、Gateway Load Balancing Protoco（GLBP）などのクライアント プロセスで、トラッキング オブジェクトに対する興味を登録し、追跡対象オブジェクトの状態が変化したときに通知を受け取るようにすることができます。

各追跡対象オブジェクトには、トラッキング コマンドライン インターフェイス（CLI）で指定される一意の番号があります。クライアント プロセスは、この番号を使用して特定のオブジェクトを追跡します。トラッキング プロセスは、追跡対象オブジェクトに値の変化がないかどうかを定期的にポーリングし、（アップまたはダウン値など）変化があれば登録されているクライアント プロセスに通知します。ただちに通知する場合と、指定された時間遅延後に通知する場合があります。同じオブジェクトを複数のクライアントが追跡して、オブジェクトのステートが変化した場合に、それぞれが異なるアクションを実行できます。

複数のオブジェクトを組み合わせて 1 つのリストにして追跡することもできます。このリストの状態判定には、重みしきい値またはパーセンテージを使用します。オブジェクトの組み合わせには、ブール ロジックを使用できます。「AND」ブール関数を使用する追跡リストの場合、リスト内の各オブジェクトがアップ ステートでないと追跡対象オブジェクトはアップになりません。「OR」ブール関数を使用する追跡リストの場合、リスト内の 1 つのオブジェクトだけがアップ ステートであれば追跡対象オブジェクトはアップになります。

インターフェイス ラインプロトコルまたは IP ルーティング ステートのトラッキング

インターフェイス ラインプロトコル ステートまたはインターフェイス IP ルーティング ステートのいずれかを追跡できます。IP ルーティング ステートを追跡する場合、オブジェクトをアップするには次の 3 つの条件が必要です。

- インターフェイス上で IP ルーティングがイネーブル、かつアクティブになっている。
- インターフェイス ラインプロトコル ステートが使用可能な状態（アップ）にある。
- 既知のインターフェイス IP アドレスを使用している。

この 3 つの条件がすべて合致しないと、IP ルーティング ステートはダウンになります。

追跡リスト

オブジェクトの追跡リストは、ブール式、重みしきい値、またはパーセントしきい値を使用して設定できます。トラッキング対象リストには 1 つまたは複数のオブジェクトが含まれます。オブジェクトは存在していないと追跡リストに追加できません。

- 設定にブール式による演算を指定する場合は、「AND」または「OR」演算子を使用します。
- 追跡リストのステートを重みしきい値で判定する場合は、追跡リスト内の各オブジェクトに重み番号を割り当てます。追跡リストのステートは、このしきい値に合致したかどうかで判定されます。各オブジェクトのステートは、すべてのオブジェクトの重みの合計と各オブジェクトのしきい値の重みを比較して判定されます。

- 追跡リストをパーセントしきい値で判定する場合は、追跡リスト内のすべてのオブジェクトにパーセントしきい値を割り当てます。各オブジェクトのステートは、各オブジェクトに割り当てたパーセンテージとリストを比較して判定されます。

他の特性のトラッキング

拡張オブジェクト トラッキングを使用して他の特性を追跡することもできます。

- **track ip route reachability** グローバル コンフィギュレーション コマンドを使用すると、IP ルートの到達可能性を追跡できます。
- **track ip route metric threshold** グローバル コンフィギュレーション コマンドを使用すると、ルートがしきい値を超えているか下回っているかを確認できます。
- **track resolution** グローバル コンフィギュレーション コマンドを使用すると、ルーティング プロトコルのメトリック解決のデフォルト値を変更できます。
- **track timer tracking** コンフィギュレーション コマンドを使用すると、トラッキング対象オブジェクトを定期的にポーリングするようにトラッキングプロセスを設定できます。

拡張オブジェクトトラッキング設定を確認する場合は、 **show track** 特権 EXEC コマンドを使用してください。

IP SLA オブジェクト トラッキング

Cisco IOS IP サービス レベル契約（SLA）は、ネットワーク パフォーマンスの測定と診断を行うツールです。ネットワーク パフォーマンスを測定するためのトラフィック生成には、アクティブ モニタリングが使用されます。Cisco IP SLA 動作は、ネットワークのトラブルシューティングや設計、分析に使用できるリアルタイム メトリックを収集します。

IP SLA 動作のオブジェクト トラッキングを活用すると、クライアントは IP SLA オブジェクトの出力を追跡して、その情報をアクションのトリガーに使用できます。各 IP SLA 動作は、OK または OverThreshold のような簡易ネットワーク管理プロトコル（SNMP）動作の戻りコード値を保持しているため、トラッキング プロセス側で解釈できます。ステートと到達可能性という IP SLA 動作の 2 つの側面をトラッキングできます。ステートの場合、戻りコードが OK のとき、トラック ステートがアップします。リターン コードが OK ではないとき、トラック ステートはダウンします。到達可能性の場合、戻りコードが OK または OverThreshold のとき、到達可能性がアップします。リターン コードが OK ではないとき、到達可能性はダウンします。

スタティック ルート オブジェクト トラッキング

拡張オブジェクトトラッキングを使用したスタティック ルーティング サポートにより、デバイスで ICMP ping を使用して、設定済みのスタティックルートまたは DHCP ルートがダウンしていることを認識できます。トラッキングを有効にしている場合、システムはルート ステートを追跡し、ステートの変化をクライアントに通知できます。スタティック ルート オブジェク

ト トラッキングは、プライマリ ゲートウェイへの接続状態をモニターするために、Cisco IP SLA を使用して ICMP ping を生成します。

拡張オブジェクト トラッキングの設定方法

ここでは、拡張オブジェクトトラッキングに関する設定情報について説明します。

インターフェイスでのライン ステート プロトコルまたは IP ルーティング ステートのトラッキングの設定

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 ルーティング ステートを追跡するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number* **interface** *interface-id* **line-protocol** 例： `Device(config)# track 33 interface gigabitethernet 1/0/1 line-protocol`	（任意）インターフェイスのラインプロトコル ステートを追跡するための追跡リストを作成し、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-number：追跡対象オブジェクトの番号です。指定できる範囲は1 ～ 500 です。 • **interface** interface-id は、追跡されるインターフェイスです。
ステップ 4	**delay** { *object-number* **up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。

	コマンドまたはアクション	目的
ステップ 5	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**track** *object-number* **interface** *interface-id* **ip routing** 例： `Device(config)# track 33 interface gigabitethernet 1/0/1 ip routing`	（任意）インターフェイスのIPルーティング ステートを追跡するための追跡リストを作成し、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。IP ルート追跡では、ルーティング テーブル内の IP ルートおよびインターフェイスの IP パケット ルーティング機能を追跡します。 • object-number：追跡対象オブジェクトの番号です。指定できる範囲は1 ～ 500 です。 • **interface** interface-id は、追跡されるインターフェイスです。
ステップ 7	**delay** { *object-number* **up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 8	**end**	特権 EXEC モードに戻ります。
ステップ 9	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。

追跡リストの設定

ここでは、追跡リストに関する設定情報について説明します。

重みしきい値による追跡リストの設定

重みしきい値による追跡を行うには、複数オブジェクトを含んだ追跡リストを作成し、重みをしきい値として使用することを指定したあと、各オブジェクトに重み値を設定します。各オブジェクトのステートは、アップであるすべてのオブジェクトの重み合計と各オブジェクトのしきい値の重みを比較して判定されます。

重みしきい値のリストには、「NOT」ブール演算子を使用できません。

重みしきい値を使用してオブジェクトの追跡リストを作成し、各オブジェクトに重み値を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *track-number* **list threshold** {**weight**} 例： `Device(config)# track 4 list threshold weight`	トラッキング対象リスト オブジェクトを設定し、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。指定できる track-number の範囲は 1 ～ 500 です。 • **threshold—**追跡リストのステートがしきい値に基づくことを指定します。 • **weight—** しきい値が重みに基づくことを指定します。
ステップ 4	**object** *object-number* [**weight** *weight-number*] 例： `Device(config)#object 2 weight 15`	追跡対象のオブジェクトを指定します。指定できる範囲は 1 ～ 500 です。任意の **weight***weight-number* には、オブジェクトのしきい値の重みを指定します。範囲は 1 ～ 255 です。 （注） オブジェクトは存在していないと追跡リストに追加できません。
ステップ 5	**threshold weight** {**up** *number* | [**down** *number*]} 例： `Device(config-track)# threshold weight up 30 down 10`	（任意）重みしきい値を指定します。 • **up***number*：範囲は 1 ～ 255 です。 • **down***number*：（任意）範囲は **up***number* で選択した数値によって異なります。**up***number* を 25 に設定すると、down number の範囲は 0 ～ 24 になります。

	コマンドまたはアクション	目的
ステップ 6	**delay** { **up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 7	**end**	特権 EXEC モードに戻ります。
ステップ 8	**show track** *object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 9	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

パーセントしきい値による追跡リストの設定

パーセントしきい値による追跡を行うには、複数オブジェクトを含んだ追跡リストを作成し、パーセンテージをしきい値として使用することを指定したあと、リスト内のすべてのオブジェクトにパーセンテージを指定します。リストのステートは、各オブジェクトに割り当てたパーセンテージとリストを比較して判定されます。

パーセントしきい値のリストには、「NOT」ブール演算子を使用できません。

パーセントしきい値を使用してオブジェクトの追跡リストを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**track** *track-number* **list threshold** {**percentage**} 例： `Device(config)# track 4 list threshold percentage`	トラッキング対象リスト オブジェクトを設定し、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。指定できる track-number の範囲は 1 ～ 500 です。 • **threshold—**追跡リストのステートがしきい値に基づくことを指定します。 • **percentage—** しきい値がパーセンテージに基づくことを指定します。
ステップ 4	**object** *object-number* 例： `Device(config)# object 1`	追跡対象のオブジェクトを指定します。指定できる範囲は 1 ～ 500 です。 （注） オブジェクトは存在していないと追跡リストに追加できません。
ステップ 5	**threshold percentage** {**up** *number* | [**down***number*]} 例： `Device(config)# threshold percentage up 51 down 10`	（任意）パーセントしきい値を指定します。 • **up***number*：範囲は 1 ～ 100 です。 • **down***number*：（任意）範囲は **up***number* で選択した数値によって異なります。**up***number* を 25 に設定すると、down number の範囲は 0 ～ 24 になります。
ステップ 6	**delay** { **up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 7	**end**	特権 EXEC モードに戻ります。
ステップ 8	**show track** *object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 9	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

HSRP オブジェクト トラッキングの設定

特定のオブジェクトを追跡し、そのオブジェクトのステートに基づいてHSRPプライオリティを変更できるようにスタンバイ HSRP グループを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number* {**interface** *interface-id* {**line-protocol** | **ip routing**} | **ip route** *ip address/prefix-length* {**metric threshold** | **reachability**} **list** {**boolean** {**and** | **or**}} | {**threshold** {**weight** | **percentage**}}}	（任意）設定されたステートを追跡するための追跡リストを作成し、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-number：追跡対象オブジェクトの番号です。指定できる範囲は 1 ～ 500 です。 • 追跡するインターフェイスを指定するには、**interface** *interface-id* を入力します。 • インターフェイス ラインプロトコルの状態を追跡するには **line-protocol** を入力します。また、インターフェイス IP ルーティングの状態を追跡するには、**ip routing** を入力します。 • IP ルートの状態を追跡するには、**ip route** *ip-address/prefix-length* を入力します。 • しきい値メトリックを追跡する場合は **metric threshold**、ルートが到達可能かどうかを追跡するには **reachability** を入力します。

	コマンドまたはアクション	目的
		デフォルトの up しきい値は 254、デフォルトの down しきい値は 255 です。 • リスト内の一連のオブジェクトを追跡するには、**list** を入力します。 （注） 追跡するインターフェイスごとにこの手順を繰り返してください。
ステップ 4	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**interface** { *interface-id*	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	**standby** [*group-number*] **ip** [*ip-address* **secondary**]]	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成（またはイネーブルに）します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を入力します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。このキーワードが省略された場合、設定されたアドレ

	コマンドまたはアクション	目的
		スはプライマリ IP アドレスになります。
ステップ 7	**standby** [*group-number*] **track** [*object-number* [**decrement** *priority-decrement*]]	特定のオブジェクトを追跡し、そのオブジェクト ステートに基づいてホットスタンバイ プライオリティを変更できるように HSRP を設定します。 • （任意）*group-number*：追跡が適用されるグループ番号を入力します。 • *object-number*：追跡対象のオブジェクト番号を入力します。指定できる範囲は 1 ～ 500 で、デフォルトは 1 です。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。このキーワードが省略された場合、設定されたアドレスはプライマリ IP アドレスになります。 • （任意）**decrement***priority-decrement*：追跡対象のオブジェクトがダウンになった場合（またはアップに戻った場合）に、ルータのホットスタンバイの優先順位を減少（または増加）させる幅を指定します。指定できる範囲は 1 ～ 255 で、デフォルトは 10 です。
ステップ 8	**end**	特権 EXEC モードに戻ります。
ステップ 9	**show standby**	スタンバイ ルータの IP アドレスおよび追跡ステートを確認します。
ステップ 10	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IP SLA オブジェクト トラッキングの設定

IP SLA 動作のステートまたは IP SLA IP ホストの到達可能性を追跡するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number* **ip sla** *operation-number* {**state** | **reachability**} 例： `Device(config)# track 2 ip sla 123 state`	トラッキング コンフィギュレーション モードを開始し、IP SLA 動作のステートを追跡します。 • *object-number* の範囲は 1 ～ 500 です。 • *operation-number* の範囲は 1 ～ 2147483647 です。
ステップ 4	**delay** { **up***seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 5	**end**	特権 EXEC モードに戻ります。
ステップ 6	**show track** *object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スタティック ルート オブジェクト トラッキングの設定

ここでは、スタティック ルート オブジェクト トラッキングに関する設定情報について説明します。

スタティック ルーティング用のプライマリ インターフェイスの設定

スタティック ルーティングのプライマリ インターフェイスを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	プライマリまたはセカンダリ インターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**description** *string*	インターフェイスに説明を追加します。
ステップ 5	**ip address** *ip-address mask* [**secondary**]	インターフェイスのプライマリまたはセカンダリ IP アドレスを設定します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。

DHCP のプライマリ インターフェイスの設定

DHCP のプライマリ インターフェイスを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	プライマリまたはセカンダリ インターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**description** *string*	インターフェイスに説明を追加します。
ステップ 5	**ip dhcp client route track** *number*	DHCP クライアントを設定し、追加されたルートを指定の追跡番号に関連付けます。有効な数値は 1 ～ 500 です。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。

IP SLA モニタリング エージェントの設定

プライマリ インターフェイスおよびエージェント状態をモニターするトラック オブジェクトを使用して、IP アドレスの ping を実行するように IP SLA エージェントを設定することができます。

Cisco IP SLA でネットワーク モニタリングを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla** *operation number*	Cisco IP SLA 動作の設定を開始し、IP SLA 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	**icmp-echo** { *destination ip-address* | *destination hostname* [**source - ipaddr** {*ip-address* | *hostname* **source-interface** *interface-id*]	Cisco IP SLA エンドツーエンド ICMP エコー応答時間動作を設定し、IP SLA ICMP エコー 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	**timeout** *milliseconds*	要求パケットの応答に対する動作の待機時間を設定します。
ステップ 6	**frequency** *seconds*	動作がネットワークに送信される頻度を設定します。
ステップ 7	**threshold** *milliseconds*	反応イベントを生成し、その動作の履歴情報を保存するしきい値（ヒステリシス）の上限を設定します。
ステップ 8	**exit**	IP SLA ICMP エコー 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	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] **start-time** *time* | **pending** | **now** | **after** *time*] **ageout** *seconds*] [**recurring**] 例： `Device(config)# track 2 200 state`	単一の IP SLA 動作のスケジューリング パラメータを設定します。 • *object-number* の範囲は 1 ～ 500 です。 • *operation-number* の範囲は 1 ～ 2147483647 です。
ステップ 10	**track** *object-number* **rtr** *operation-number* **state reachability**	Cisco IOS IP SLA 動作の状態を追跡し、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。
ステップ 11	**end**	特権 EXEC モードに戻ります。
ステップ 12	**show track** *object-number*	指定したオブジェクトが追跡されているかどうかを確認します。

	コマンドまたはアクション	目的
ステップ 13	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ルーティング ポリシーおよびデフォルト ルートの設定

オブジェクト トラッキングを使用してバックアップ スタティック ルーティングのルーティング ポリシーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number*	拡張 IP アクセス リストを定義します。オプションの文字を設定します。
ステップ 4	**route-map** *map tag* [**permit** | **deny**] [*sequence-number*]	ルートマップ コンフィギュレーション モードを開始し、特定のルーティングから別のルーティングへの再配信ルートの条件を定義します。
ステップ 5	**match ip address** {*access-list number* [**permit**|**deny**] [*sequence-number*]	標準または拡張アクセス リストに許可された宛先ネットワーク番号アドレスを持つルートを配信し、パケットのポリシー ルーティングを実行します。複数の番号または名前を入力できます。
ステップ 6	**set ip next-hop dynamic dhcp**	DHCP ネットワーク専用。DHCP クライアントが学んだ最新のゲートウェイへのネクスト ホップを設定します。

	コマンドまたはアクション	目的
ステップ 7	**set interface** *interface-id*	スタティック ルーティング ネットワーク専用。ポリシー ルーティングのルート マップ一致条件をパスした出力パケットの送信場所を指定します。
ステップ 8	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 9	**ip local policy route-map** *map tag*	ルートマップを特定し、ローカル ポリシー ルーティングに使用します。
ステップ 10	**ip route** *prefix mask* {*ip address* | *interface-id* [*ip address*]} [*distance*] [*name*] [**permanent** | **track** *track-number*] [*tag tag*]	スタティック ルーティング ネットワーク専用。スタティック ルートを確立します。**track***track-number* を入力し、設定したトラックオブジェクトがアップの場合に限り、静的ルートがインストールされるように指定します。
ステップ 11	**end**	特権 EXEC モードに戻ります。
ステップ 12	**show ip route track table**	IP ルート トラック テーブルの情報を表示します。
ステップ 13	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

拡張オブジェクト トラッキングのモニタリング

下の表に示す特権EXECコマンドまたはユーザーEXECコマンドを使用して、拡張オブジェクトの追跡情報を表示します。

表 *7:* 追跡情報を表示するコマンド

コマンド	目的
show ip route track table	IP ルート トラック テーブルの
show track [*object-number*]	すべての追跡リストまたは指定
show track brief	すべてのインターフェイスまたテータスおよび設定を表示しま

コマンド	目的
show track interface [brief]	追跡対象のインターフェイス オブ
show track ip [*object-number*] [**brief**] **route**	追跡対象 IP ルート オブジェクトの
show track resolution	追跡対象パラメータの解像度を表
show track timer	追跡対象のポーリング インターバ

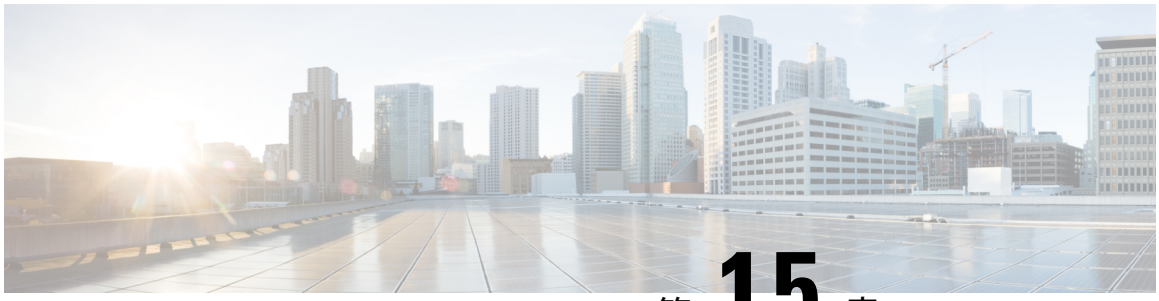
拡張オブジェクトトラッキングの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	拡張オブジェクト トラッキング	拡張オブジェクトトラッキングでは、インターフェイスのラインプロトコル ステート トラッキングのみを許可する HSRP と比較して、高度なトラッキングが可能です。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 15 章

TCP MSS 調整の設定

TCP MSS 調整の制約事項

- サブインターフェイスは TCP MSS 調整をサポートしません。
- TCP MSS 調整は、レイヤ 3 GRE トンネルでの TCP ストリームの入力パケットキャプチャでのみ機能し、出力パケットキャプチャでは機能しません。

TCP MSS 調整に関する情報

トランスミッション コントロール プロトコル（TCP）最大セグメントサイズ（MSS）調整機能では、ルータを通過する一時的なパケット（特に SYN ビットが設定された TCP セグメント）の最大セグメントサイズを設定することができるようになります。切り捨てを回避するために、SYN パケットの中間ルータで MSS 値を指定するには、インターフェイス コンフィギュレーション モードで ip tcp adjust-mss コマンドを使用します。

ホスト（通常は PC）がサーバーと TCP セッションを開始するときは、TCP SYN パケットの MSS オプション フィールドを使って IP セグメント サイズをネゴシエートします。MSS フィールドの値は、ホスト上の MTU 設定によって決まります。PC のデフォルト MSS 値は 1500 バイトです。

PPP over Ethernet（PPPoE）標準は、1,492 バイトのみの MTU をサポートします。ホストと PPPoE での MTU サイズの不一致は、ホストとサーバーの間にあるルータで 1500 バイトのパケットが損失し、PPPoE を介した TCP セッションが終了する原因となる場合があります。ホストでパス MTU（パス全体で正しい MTU を検出）が有効になっていても、システム管理者が

パス MTU を機能させるためにホストからリレーする必要がある ICMP エラーメッセージを無効にすることがあるため、セッションがドロップされることがあります。

ip tcp adjust-mss コマンドで TCP SYN パケットの MSS 値を調整すると、TCP セッション損失防止の役に立ちます。

ip tcp adjust-mss コマンドは、ルータを通過する TCP 接続に対してのみ有効です。

ほとんどの場合、ip tcp adjust-mss コマンドの max-segment-size 引数の最適値は 1,452 バイトです。この値に、20 バイトの IP ヘッダー、20 バイトの TCP ヘッダー、および 8 バイトの PPPoE ヘッダーが追加されて、イーサネット リンクの MTU サイズと同じ 1500 バイトのパケットになります。

サポートされるインターフェイス

TCP MSS 調整は、次のインターフェイスでのみサポートされます。

- 物理層 3 インターフェイス
- SVI
- レイヤ 3 ポートチャネル
- レイヤ 3 GRE トンネル

TCP MSS 調整の設定方法

ここでは、TCP MSS 調整の設定情報について説明します。

一時的な TCP SYN パケットの MSS 値の設定

始める前に

ルータを通過する一時的なパケット（特に SYN ビットが設定された TCP セグメント）の MSS を設定するには、この作業を実行します。

ip tcp adjust-mss 1452 コマンドを使用することを推奨します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device>enable`	特権 EXEC モードをイネーブルにします。 プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device#config terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)#interface GigabitEthernet 1/0/0`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip tcp adjust-mss** *max-segment-size* 例： `Device(config-if)#ip tcp adjust-mss 1452`	ルータを通過する TCP SYN パケットの MSS 値を調整します。 max-segment-size 引数には、MSS をバイト単位で指定します。範囲は 500 ～ 1460 です。
ステップ 5	**end** 例： `Device(config-if)#end`	グローバル コンフィギュレーション モードに戻ります。

IPv6 トラフィックの MSS 値の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device>enable`	特権 EXEC モードをイネーブルにします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Device#config terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)#interface GigabitEthernet 1/0/0`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 tcp adjust-mss** *max-segment-size* 例：	デバイスを通過する TCP DF パケットの MSS 値を調整します。

	コマンドまたはアクション	目的
	`Device(config-if)#ipv6 tcp adjust-mss 1440`	max-segment-size 引数には、MSS をバイト単位で指定します。指定できる範囲は 40 ～ 1440 です。
ステップ 5	**end** 例： `Device(config-if)#end`	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、特権 EXEC モードに戻ります。

TCP MSS 調整の設定例

ここでは、TCP MSS 調整の設定例を示します。

例：TCP MSS 調整の設定

```
Device(config)#vpdn enable
Device(config)#no vpdn logging
Device(config)#vpdn-group 1
Device(config-vpdn)#request-dialin
Device(config-vpdn-req-in)#protocol pppoe
Device(config-vpdn-req-in)#exit
Device(config-vpdn)#exit
Device(config)#interface GigabitEthernet 0/0/0
Device(config-if)#ip address 192.168.100.1.255.255.255.0
Device(config-if)#ip tcp adjust-mss 1452
Device(config-if)#ip nat inside
Device(config-if)#exit
```

例：IPv6 トラフィックの TCP MSS 調整の設定

```
Device>enable
Device#configure terminal
Device(config)#interface GigabitEthernet 0/0/0
Device(config)#ipv6 tcp adjust-mss 1440
Device(config)#end
```

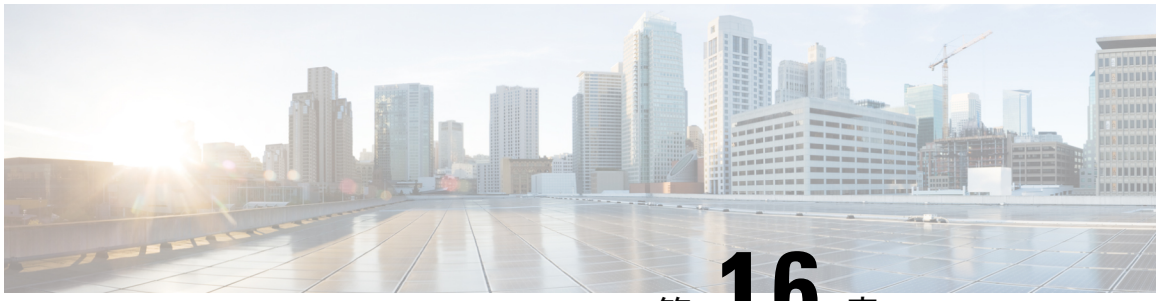
TCP MSS 調整の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	トランスミッション コントロール プロトコル（TCP）最大セグメントサイズ（MSS）調整	TCP MSS 調整機能では、ルータを通過する一時的なパケット（特に SYN ビットが設定された TCP セグメント）の最大セグメントサイズを設定することができるようになります。この機能は、TCP SYN パケットの MSS 値を調整することで TCP セッション損失防止の役に立ちます。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。



第 16 章

IPv6 の拡張ネイバー探索キャッシュ管理

IPv6 の拡張ネイバー探索キャッシュ管理

ネイバー探索プロトコルは、障害のあるノードまたはデバイス、およびリンク層アドレスの変更を検出できるネイバー到達不能検出を実行します。ネイバー到達不能検出プロセスは、ホストからホスト、ホストからデバイス、デバイスからホストへの通信など、ホストとネイバーノード間の全パスの到達可能性情報を保持します。

ネイバーキャッシュは、リンクレイヤアドレスへの IPv6 リンクローカルアドレスまたはグローバルアドレスに関するマッピング情報を保持します。ネイバーキャッシュは、ネイバー到達不能検出プロセスを使用して、ネイバーの到達可能性の状態に関する情報も保持します。ネイバーは、次の 5 つのうちいずれかの状態になります。

- DELAY：ネイバーの解決は保留になっており、トラフィックがこのネイバーに流れる可能性があります。
- INCOMPLETE：アドレスの解決中であり、リンク層アドレスはまだ不明です。
- PROBE：ネイバーの解決中であり、トラフィックがこのネイバーに流れる可能性があります。
- REACHABLE：最後の到達可能時間間隔内でネイバーに到達可能であることがわかっています。
- STALE：ネイバーは解決を必要としており、トラフィックがこのネイバーに流れる可能性があります。

非送信要求ネイバーアドバタイズメントからエントリを収集するネイバー探索プロトコルを設定するには、**ipv6 nd na glean** コマンドを使用します。

ネットワークの中断時にネイバーのネイバー探索キャッシュエントリを保持するようにネイバー探索プロトコルを設定するには、**ipv6 nd nud retry** コマンドを使用します。

ネイバーへのトラフィックフローがない場合でも、ネイバー探索キャッシュエントリを保持するようにネイバー探索プロトコルを設定するには、**ipv6 nd cache expire refresh** コマンドを使用します。

IPv6 ネイバー探索のパラメータのカスタマイズ

IPv6 ネイバー探索のパラメータをカスタマイズするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface gigabitethernet 1/1/4`	インターフェイスタイプと ID を指定します。インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 nd nud retry** *base interval max-attempts* [*final-wait-time*] 例： `Device(config-if)# ipv6 nd nud retry 1 1000 3`	ネイバー到達不能検出でネイバー送信要求を再送信する回数を設定します。
ステップ 5	**ipv6 nd cache expire** *expire-time-in-seconds* [**refresh**] 例： `Device(config-if)# ipv6 nd cache expire 7200`	IPv6 ネイバー探索キャッシュエントリの期限が切れるまでの時間を設定します。
ステップ 6	**ipv6 nd na glean** 例： `Device(config-if)# ipv6 nd na glean`	IPv6 ネイバー探索キャッシュエントリの期限が切れるまでの時間を設定します。

	コマンドまたはアクション	目的
ステップ 7	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。
ステップ 8	**show ipv6 interface** 例： `Device# show ipv6 interface`	（任意）ネイバー探索キャッシュ管理と IPv6 用に設定されたインターフェイスのユーザビリティのステータスを表示します。

例：IPv6 ネイバー探索のパラメータのカスタマイズ

次の例では、IPv6 ネイバーアドバタイズメントの収集が有効になっており、IPv6 ネイバー探索キャッシュの有効期限は 7200 秒（2 時間）に設定されています。

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel 189
Device(config-if)# no ip address
Device(config-if)# ipv6 address 2001:BD8::/64
Device(config-if)# ipv6 nd reachable-time 2700000
Device(config-if)# ipv6 nd na glean
Device(config-if)# ipv6 nd cache expire 7200
Device(config-if)# no ipv6 redirects
Device(config-if)# end
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	「*IP* アドレッシングサービス」のセクションを参照 *Command Reference (Catalyst 9300 Series Switches)*
IPv6 ネイバー探索インスペクションの詳細	「セキュリティ」のセクションを参照*Software Configuration Guide (Catalyst 9300 Switches)*

IPv6 ネイバー探索の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IPv6 の拡張ネイバー探索キャッシュ管理	ネイバー探索プロトコルは、障害のあるノードまたはルータ、およびリンク層アドレスの変更を検出できるネイバー到達不能検出を実行します。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn [英語] からアクセスします。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。