

# Catalyst 6500シリーズVSS 1440でのパケットフローのトラブルシューティング

## 内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[背景説明](#)

[ネットワーク図](#)

[Catalyst 6500 シリーズ スイッチでの EtherChannel の概要](#)

[ロード バランシング アルゴリズムの決定](#)

[出カインターフェイスの決定：スタンドアロン Catalyst 6500](#)

[出カインターフェイスの決定：VSS](#)

[Catalyst 6500 スイッチでの ECMP の概要](#)

[ロード バランシング アルゴリズムの決定](#)

[出カインターフェイスの決定：スタンドアロン Catalyst 6500](#)

[出カインターフェイスの決定：VSS](#)

[トラブルシューティングのシナリオ](#)

[シナリオ 1：レイヤ 2 MEC を使用する 2 台のアクセス層ホスト間のパケット フロー](#)

[シナリオ 2：レイヤ 2 MEC を使用する 2 台のアクセス層ホスト間のパケット フロー：冗長性の喪失](#)

[シナリオ 3：レイヤ 3 MEC を使用する 2 台のアクセス層ホスト間のパケット フロー](#)

[シナリオ 4：レイヤ 3 MEC を使用する 2 台のアクセス層ホスト間のパケット フロー：冗長性の喪失](#)

[シナリオ 5：ECMP を使用する 2 台のアクセス層ホスト間のパケット フロー](#)

[シナリオ 6：ECMP を使用する 2 台のアクセス層ホスト間のパケット フロー：冗長性の喪失](#)

[関連情報](#)

## 概要

このドキュメントでは、Virtual Switching System ( VSS ) ネットワーク内のパケット フローのトラブルシューティングに関するガイドラインを示します。この例は、VSS を使用しているネットワークのトラブルシューティングを中心にしていますが、ここに示す一般原則は、冗長リンクを含めて設計されているすべてのネットワークで有効です。

## 前提条件

### 要件

次の項目に関する知識があることが推奨されます。

- [Virtual Switching System の概要](#)
- [Virtual Switching System \( VSS \) の Q&A](#)

## [使用するコンポーネント](#)

このドキュメントの情報は、Cisco IOS® ソフトウェア リリース 12.2(33)SXH1 以降が稼働する Supervisor VS-S720-10G-3C/XL が搭載された Cisco Catalyst 6500 シリーズ スイッチに基づくものです。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

## [表記法](#)

ドキュメント表記の詳細は、『[シスコ テクニカル ティップスの表記法](#)』を参照してください。

## [背景説明](#)

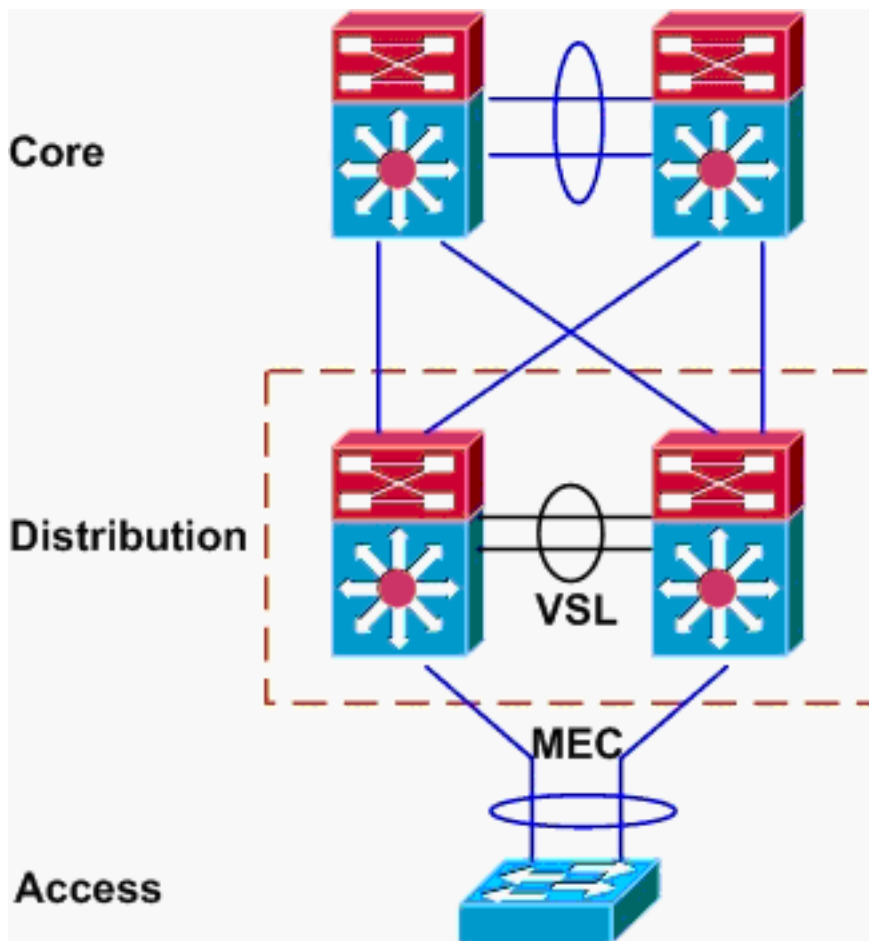
VSS を利用する一般的なネットワーク設計については、『[ネットワーク ダイアグラム](#)』を参照してください。2 台のシスコ スイッチが VSS 用に設定されている場合、これらは、単一の論理スイッチとしてネットワークに表示されます。冗長性を実現するには、仮想スイッチに接続されている各ノードは、物理シャーシごとに 1 つ以上のリンクを含む必要があります。Multi-chassis EtherChannel ( MEC ) を介して冗長リンクを利用することを推奨しますが、等コスト マルチパス ( ECMP ) を使用することもできます。1 台のスイッチが故障した場合にユニキャストおよびマルチキャストのコンバージェンス時間を短縮できるため、接続方法として ECMP よりも MEC を推奨します。

詳細については、『*Cisco Catalyst 6500 Virtual Switching System 配備のベスト プラクティス*』の「[アップストリーム リンクの回復](#)」を参照してください。

VSS は、仮想化を使用しているため、ネットワーク内のパケットのパスをトレースする新しいトラブルシューティング ツールを使用する必要があります。VSS ネットワークでは、ポート チャネル インターフェイスまたは複数のネクストホップ インターフェイスのいずれかが返されるため、MAC アドレス テーブルまたはルーティング テーブルを調べてネクストホップを判別する方法など、一般的なパケット パストラブルシューティング方法は、あまり有用ではありません。このドキュメントは、Catalyst 6500 プラットフォームで使用可能ないずれの Cisco CLI コマンドを使用すると、パケットのパスに関する有用性の高いデータを収集できるのかを示すことを目的としています。

## [ネットワーク図](#)

このドキュメントでは、次のネットワーク セットアップを使用します。



## Catalyst 6500 シリーズ スイッチでの EtherChannel の概要

### ロード バランシング アルゴリズムの決定

いずれの Cisco Catalyst スイッチでも、EtherChannel リンクは、送信元および宛先の MAC、IP、またはレイヤ 4 ポート番号など、パケット ヘッダーにある特定のフィールドのハッシュに基づいて選択されます。この情報は、特定の 1 つのフローに含まれるすべてのパケットで同一であるため、EtherChannel ロードバランシングは、フローベースであるとされることがよくあります。

Catalyst 6500 スイッチでは、**show etherchannel load-balance** コマンドによって、このハッシュに使用されるフィールドを参照できます。

```
PFC-3B#show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
  src-dst-ip
  mpls label-ip
```

```
EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
IPv4: Source XOR Destination IP address
IPv6: Source XOR Destination IP address
MPLS: Label or IP
```

ここでは、IPX、AppleTalk など、IP 以外のトラフィックは、送信元および宛先の MAC アドレスに基づいてハッシュされ、IPv4 および IPv6 のトラフィックは、送信元および宛先の IP アドレスに基づいてハッシュされることが示されています。MPLS パケットのハッシュ処理については、このドキュメントでは取り上げていません。上記の設定は、Catalyst 6500 でのデフォルトです。

IPv6 および IP 以外のパケットについては、他に選択可能なロード バランス設定オプションはありません。一方、IPv4 パケットの場合については、選択可能な他のロード バランス設定を次に示します。

- 宛先 IP
- 宛先 MAC
- 宛先レイヤ 4 ポート
- 宛先 IP とレイヤ 4 ポートの混合 ( PFC-3C のみ )
- 送信元/宛先 IP
- 送信元/宛先 MAC
- 送信元/宛先レイヤ 4 ポート
- 送信元および宛先 IP とレイヤ 4 ポートの混合 ( PFC-3C のみ )
- 送信元 IP
- 送信元 MAC
- 送信元レイヤ 4 ポート
- 送信元 IP とレイヤ 4 ポートの混合 ( PFC-3C のみ )

EtherChannel ロード バランス設定は、**port-channel load-balance** コマンドを使用して変更できません。

```
SW1(config)#port-channel load-balance ?
dst-ip          Dst IP Addr
dst-mac         Dst Mac Addr
dst-mixed-ip-port Dst IP Addr and TCP/UDP Port
dst-port        Dst TCP/UDP Port
mpls            Load Balancing for MPLS packets
src-dst-ip      Src XOR Dst IP Addr
src-dst-mac     Src XOR Dst Mac Addr
src-dst-mixed-ip-port Src XOR Dst IP Addr and TCP/UDP Port
src-dst-port    Src XOR Dst TCP/UDP Port
src-ip          Src IP Addr
src-mac         Src Mac Addr
src-mixed-ip-port Src IP Addr and TCP/UDP Port
src-port        Src TCP/UDP Port
```

PFC-3C ( XL ) の導入に伴って、ロードバランシング アルゴリズムが少し変更されたことへの配慮も重要です。PFC-3C ( XL ) は、Supervisor 720-10GE に搭載されています。PFC-3C 上のハッシュ アルゴリズムでは、IPv4 パケット用および IPv6 パケット用に設定されているフィールドに加えて、VLAN が常に考慮されます。

たとえば、**src-dst-ip enhanced** ( 下記 ) のデフォルト設定の場合、PFC では、送信元と宛先の IP および VLAN を考慮して、ハッシュ値を計算します。入力として使用する VLAN は、パケットの入力 VLAN である必要があります。入力インターフェイスがレイヤ 3 として設定されている場合、このインターフェイスに対する内部 VLAN は、**show vlan internal usage** コマンドで表示される入力である必要があります。

```
PFC-3C#show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
    src-dst-ip enhanced
    mpls label-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
IPv4: Source XOR Destination IP address
IPv6: Source XOR Destination IP address
```

## [出インターフェイスの決定 : スタンドアロン Catalyst 6500](#)

システムのロードバランシング アルゴリズムが決まると、この CLI を使用して、特定の packets に対して選択される EtherChannel 内の物理インターフェイスを判別できます (バージョン 12.2(33)SXH 以降でのみ使用可能)。

```
Router#show etherchannel load-balance hash-result interface port-channel 1 ?
  ip      IP address
  ipv6    IPv6
  l4port  Layer 4 port number
  mac     Mac address
  mixed   Mixed mode: IP address and Layer 4 port number
  mpls    MPLS
```

上記のコマンドでは、データ入力とロードバランシング アルゴリズムで使用するデータが一致していることを確認しないため、このコマンドを使用するときは注意する必要があります。この CLI に入力する情報が多すぎるか少なすぎる場合、プロンプトは物理インターフェイスを 1 つ返します。ただし、返されるインターフェイスは正しくないことがあります。次に、適切に使用されているコマンドの例を示します。

**注 :** スペースの制約により、一部のコマンドが 2 行目に移動します。

src-dst-ip アルゴリズムを使用する PFC-3B システム

```
PFC-3B#show etherchannel load-balance hash-result interface port-channel
1 ip 10.1.1.1 10.2.2.2
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

src-dst-ip enhanced アルゴリズムを使用する PFC-3C システム

```
PFC-3C#show etherchannel load-balance hash-result interface port-channel
1 ip 10.1.1.1 vlan 10 10.2.2.2
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

src-dst-ip enhanced アルゴリズムを使用し、入インターフェイスがレイヤ 3 の PFC-3C システム

```
PFC-3C#show vlan internal usage | include Port-channel 2
1013 Port-channel 2
PFC-3C#
```

```
PFC-3C#show etherchannel load-balance hash-result interface port-channel 1 ip 10.1.1.1 vlan 1013
10.2.2.2
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

src-dst-mixed-ip-port enhanced アルゴリズムを使用する PFC-3CXL システム

```
PFC-3CXL#show etherchannel load-balance hash-result interface port-channel
1 mixed 10.1.1.1 1600 10 10.2.2.2 80
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

## [出インターフェイスの決定 : VSS](#)

スタンドアロン Catalyst 6500 と VSS EtherChannel ハッシュ処理の間には、1つの大きな違いが存在します。その違いとは、VSS の場合、同じスイッチ上に利用可能な EtherChannel リンクがあれば、常にこのリンクにトラフィックが転送される点です。これは、VSL 上の輻輳を最小化することを目的としています。これは、帯域幅をスイッチ間で均等に分割するかどうかという問題です。言い換えると、VSS スイッチの 1 つに 1 つの EtherChannel に含まれる 4 つのアクティブリンクがあり、もう 1 つのスイッチにはリンクが 1 つだけある場合、アクティブリンクが 1 つのスイッチでは、VSL を介していくらかのトラフィックを送信するのではなく、すべてのローカルトラフィックをこの単一のリンクに転送しようとしています。

この違いがあるため、`hash-result` コマンドを使用するときは、VSS スイッチ番号を指定する必要があります。スイッチ ID を `hash-result` CLI に入力しなかった場合、VSS ではスイッチ 1 であると想定します。

src-dst-ip enhanced アルゴリズムを使用する PFC-3C VSS システム

```
VSS-3C#show etherchannel load-balance hash-result interface port-channel
1 switch 1 ip 10.1.1.1 vlan 10 10.2.2.2
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

src-dst-mixed-ip-port enhanced アルゴリズムを使用する PFC-3CXL VSS システム

```
VSS-3CXL#show etherchannel load-balance hash-result interface port-channel
1 switch 2 mixed 10.1.1.1 1600 10 10.2.2.2 80
Computed RBH: 0x1
Would select Gig3/2 of Po1
```

## [Catalyst 6500 スイッチでの ECMP の概要](#)

### [ロード バランシング アルゴリズムの決定](#)

Equal-Cost MultiPath ( ECMP ) は、ルータに、単一のプレフィクスに対する複数の等コストパスがあり、その結果、各パスにトラフィックのロード バランシングが行われる状況を指します。Catalyst 6500 でのロード バランシングは、EtherChannel および MLS CEF での EtherChannel の実装と同様に、フローに基づきます。

Catalyst 6500 では、複数のハッシュ アルゴリズムから選択できます。

- Default : 極性を避けるために各リンクに不均等に重み付けして、送信元と宛先の IP アドレスを使用します。
- Simple : 各リンクに均等に重み付けして、送信元と宛先の IP アドレスを使用します。
- Full : 不均等に重み付けして、送信元と宛先の IP アドレスとレイヤ 4 ポート番号を使用します。
- Full Simple : 各リンクに均等に重み付けして、送信元と宛先の IP アドレスとレイヤ 4 ポート番号を使用します。

```
VSS(config)#mls ip cef load-sharing ?
full      load balancing algorithm to include L4 ports
simple     load balancing algorithm recommended for a single-stage CEF router
```

```
VSS(config)#mls ip cef load-sharing full ?
simple     load balancing algorithm recommended for a single-stage CEF router
```

<cr>

*simple* キーワードおよび CEF の極性については、このドキュメントでは扱いません。詳細については、『[Cisco Express Forwarding によるロード バランシング](#)』を参照してください。

現時点では、使用されているロード シェアリング アルゴリズムを確認する CLI は存在しません。使用中の方法を調べる場合は、`show running-config` コマンドを使用して、実行コンフィギュレーションをチェックする方法が最適です。`mls ip cef load-sharing` で始まる設定がない場合は、送信元と宛先を不均等に重み付けするデフォルトのアルゴリズムが使用されています。

## [出インターフェイスの決定 : スタンドアロン Catalyst 6500](#)

スタンドアロン スイッチでは、このコマンドを使用して、ECMP の出インターフェイスを判別できます。

```
VSS#show mls cef exact-route ?  
  A.B.C.D  src IP address  
  vrf      Show numeric VPN Routing/Forwarding ID
```

次の例では、10.100.4.0/24への等コストルートが存在します。これは、このサブネット内の2つの宛先に対してexact-routeコマンドを使用する例です。

```
SW1#show mls cef exact-route 10.100.3.1 10.100.4.1  
Interface: Gi3/14, Next Hop: 10.100.2.1, Vlan: 1067, Destination Mac: 000b.000b.000b
```

```
SW1#show mls cef exact-route 10.100.3.1 10.100.4.2  
Interface: Gi3/13, Next Hop: 10.100.1.1, Vlan: 1066, Destination Mac: 000c.000c.000c
```

レイヤ 4 ポートをハッシュに含めるフル ロード シェアリング モードにシステムが設定されている場合は、このコマンドを次のように入力します。

```
SW1#show mls cef exact-route 10.100.3.1 10.100.4.1  
% System is configured in full load-sharing mode. Layer 4 ports needed
```

```
SW1#show mls cef exact-route 10.100.3.1 1024 10.100.4.1 80  
Interface: Gi3/14, Next Hop: 10.100.2.1, Vlan: 1067, Destination Mac: 000b.000b.000b
```

```
SW1#show mls cef exact-route 10.100.3.1 1024 10.100.4.1 81  
Interface: Gi3/13, Next Hop: 10.100.1.1, Vlan: 1066, Destination Mac: 000c.000c.000c
```

ここに示すように、`exact-route` コマンドには、無効なインターフェイスを返さないための健全性チェックが組み込まれています。システムがフル モードのときにレイヤ 4 ポートを指定しないなど、入力した情報が少なすぎる場合はエラーが表示されます。デフォルト モードのときにレイヤ 4 ポートを入力するなど、入力した情報が多すぎる場合、関係のない情報は無視されて正しいインターフェイスが返ります。

## [出インターフェイスの決定 : VSS](#)

EtherChannel の場合同様、VSS プログラム自体は、VSL を経由するのではなく、ローカル スイッチ上の ECMP リンクにトラフィックを送信しようとします。VSS では、ローカル スイッチ ECMP の隣接関係だけを使用して各スイッチの MLS CEF テーブルをプログラミングすることにより、これを行います。この事実があるため、有用な出力を取得するためには、exact-route CLI にスイッチ ID を含める必要があります。スイッチ番号を入力しなかった場合、VSS は、アクティブ スイッチに関する情報を提供します。

```
VSS#show mls cef exact-route 10.100.4.1 10.100.3.1 switch 1
Interface: Gi1/1/13, Next Hop: 10.100.1.2, Vlan: 1095, Destination Mac: 0013.5f1d.32c0
```

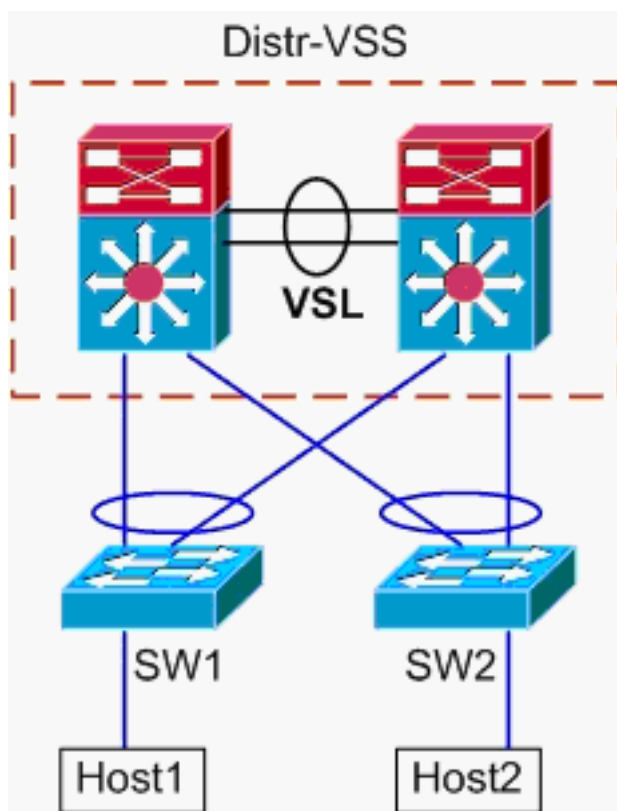
```
VSS#show mls cef exact-route 10.100.4.1 10.100.3.1 switch 2
Interface: Gi2/1/13, Next Hop: 10.100.2.2, Vlan: 1136, Destination Mac: 0013.5f1d.32c0
```

## トラブルシューティングのシナリオ

以下のトラブルシューティングシナリオでは、前述した概念を使用して、Host1 から Host2 へのパケットのフローをトレースする方法を示すことを目的としています。各シナリオは、異なるネットワークトポロジまたはシチュエーションを扱っています。

### シナリオ 1: レイヤ 2 MEC を使用する 2 台のアクセス層ホスト間のパケット フロ

二



#### トポロジ情報

- Host1 IP/MASK : 10.0.1.15/24
- Host1 MAC : 0001.0001.0001
- Host1 のデフォルト ゲートウェイ : 10.0.1.1 : Distr-VSS 上
- Host2 IP : 10.0.2.30
- SW1 と SW2 はいずれも、Distr-VSS に接する EtherChannel トランクを持つ、レイヤ 2 のみで動作している Catalyst 6500 スイッチ

1. Host1 から VSS ディストリビューションへのパスをトレースします。Host2 は、Host1 のサブネット マスクから決まる Host1 の VLAN とは異なる VLAN にあるため、パケットはルーティングのために VSS ディストリビューションに向かう必要があります。Host1 と VSS ディストリビューションの間のパケットのパスを見つけるためには、まず、Host1 のデフォルト ゲートウェイの MAC アドレスを判別する必要があります。大部分のオペレーティング



システムでは、コマンドプロンプトを開いて `arp -a` を発行すると、デフォルト ゲートウェイの IP > MAC のマッピングが表示されます。このコマンドを Host1 に対して発行したとき、10.0.1.1 に対して返された MAC は、000a.000a.000a でした。これで、この MAC を SW1 の MAC アドレス テーブルで検索できます。

```
SW1#show mac-address-table address 000a.000a.000a
```

```
Legend: * - primary entry
         age - seconds since last seen
         n/a - not available
```

```
   vlan   mac address      type    learn    age           ports
-----+-----+-----+-----+-----+-----
```

```
Supervisor:
```

```
*    10  000a.000a.000a   dynamic Yes           0   Po1
```

この出力は、Host1のデフォルトゲートウェイに対応するMACアドレスがPort-channel1を介して学習されたことを示しています。この出力には示されませんが、EtherChannel内のどのリンクが特定の packets に対して選択されているかが示されます。この情報を判別するためには、まず、EtherChannel ロードバランシング アルゴリズムをチェックする必要があります。

```
SW1#show etherchannel load-balance
```

```
EtherChannel Load-Balancing Configuration:
```

```
src-dst-ip
mpls label-ip
```

```
EtherChannel Load-Balancing Addresses Used Per-Protocol:
```

```
Non-IP: Source XOR Destination MAC address
```

```
IPv4: Source XOR Destination IP address
```

```
IPv6: Source XOR Destination IP address
```

```
MPLS: Label or IP
```

次の出力は、IPv4 パケットの場合のアルゴリズムは、src-dst-ipであることを示しています。次に、関連するフロー情報を `hash-result` コマンドに入力します。

```
SW1#show etherchannel load-balance hash-result interface port-channel
```

```
1 ip 10.1.1.1 10.0.2.30
```

```
Computed RBH: 0x1
```

```
Would select Gig3/2 of Po1
```

これで物理的な出力ポイントが判明したため、このポイントのマップ先である VSS 内の物理スイッチを CDP テーブルで表示できます。

```
SW1#show cdp neighbor
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
```

```
Device ID      Local Intrfce    Holdtme    Capability  Platform  Port ID
VSS            Gig 3/2         157        R S I      WS-C6509-EGig 2/1/1
VSS            Gig 3/1         128        R S I      WS-C6509-EGig 1/1/1
```

## 2. VSS ディストリビューションを介するパスをトレースします。まず、ルーティング テーブルを調べて、Host2 がある場所を判別します。

```
VSS#show ip route 10.0.2.30
```

```
Routing entry for 10.0.2.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via Vlan20
```

```
Route metric is 0, traffic share count is 1
```

この前の出力は、Host2がVlan20のVSSに隣接するレイヤ3であることを示しています。

Host2への物理デバイスを見つけるには、ARPテーブルを調べてMACアドレスを見つけます。

。

```
VSS#show ip arp
```

```
Protocol  Address          Age (min)  Hardware Addr  Type   Interface
Internet  10.0.2.1         15        0002.0002.0002  ARPA   Vlan20
```

次に、この出力から得た Host2 の MAC アドレスを使用して、MAC アドレス テーブルから出カインターフェイスを見つけます。

```
VSS#show mac-address-table address 0002.0002.0002
```

```
Legend: * - primary entry
         age - seconds since last seen
         n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	210	Po2

以前のCDP出力から、このフローのパケットがスイッチ2、モジュール1、ポート1に対応するGig2/1/1のVSSに入ったことを思い出してください。ここでも、hash-resultコマンドを使用して、VSSから出る物理的なポイントを判別します。

```
VSS#show etherchannel load-balance
```

```
EtherChannel Load-Balancing Configuration:
  src-dst-mixed-ip-port enhanced
  mpls label-ip
```

```
EtherChannel Load-Balancing Addresses Used Per-Protocol:
```

```
Non-IP: Source XOR Destination MAC address
IPv4: Source XOR Destination IP address
IPv6: Source XOR Destination IP address
MPLS: Label or IP
```

```
VSS#show etherchannel load-balance hash-result interface port-channel
```

```
2 switch 2 ip 10.0.1.15 vlan 10 10.0.2.30
```

```
Computed RBH: 0x6
```

```
Would select Gi2/1/13 of Po2
```

次に、CDP テーブルを使用して、Host2 へのダウンストリーム スイッチに関する情報を検索します。

```
VSS#show cdp nei
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
```

Device ID	Local Infrfce	Holdtme	Capability	Platform	Port ID
SW2	Gig 2/1/13	129	R S I	WS-C6503-	Gig 3/14
SW2	Gig 1/1/13	129	R S I	WS-C6503-	Gig 3/13

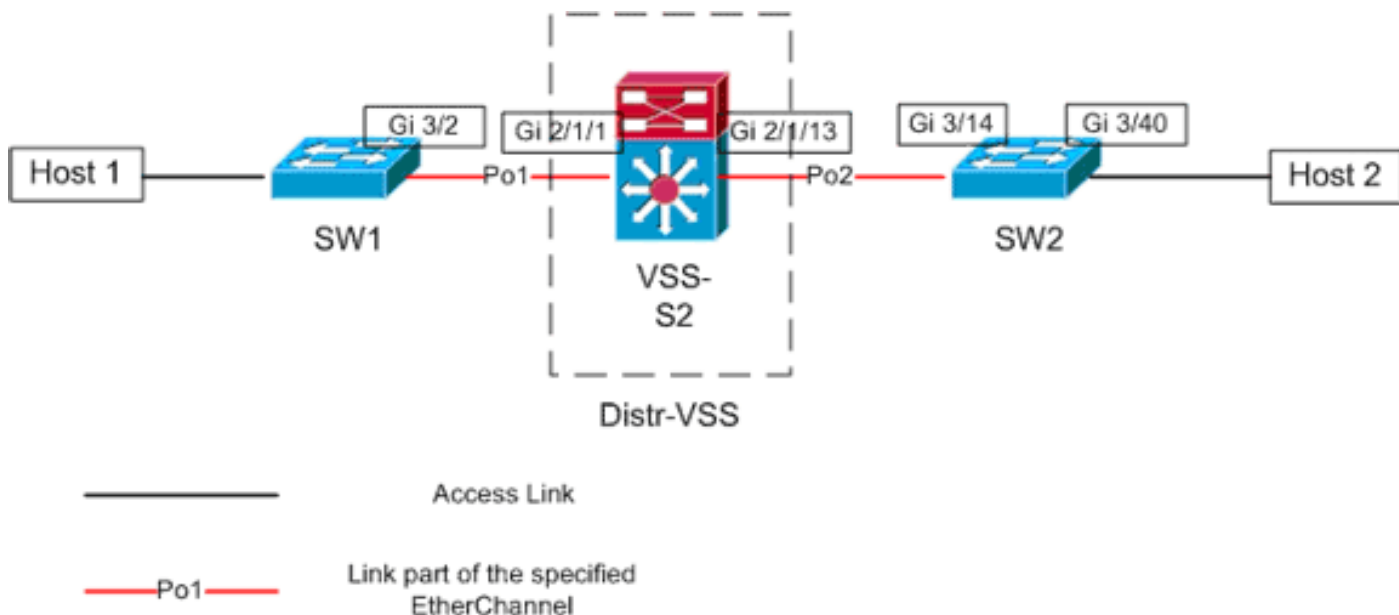
3. Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テーブルを使用して、Host2 が接続されている正確なポートを判別します。

```
SW2#show mac-address-table address 0002.0002.0002
```

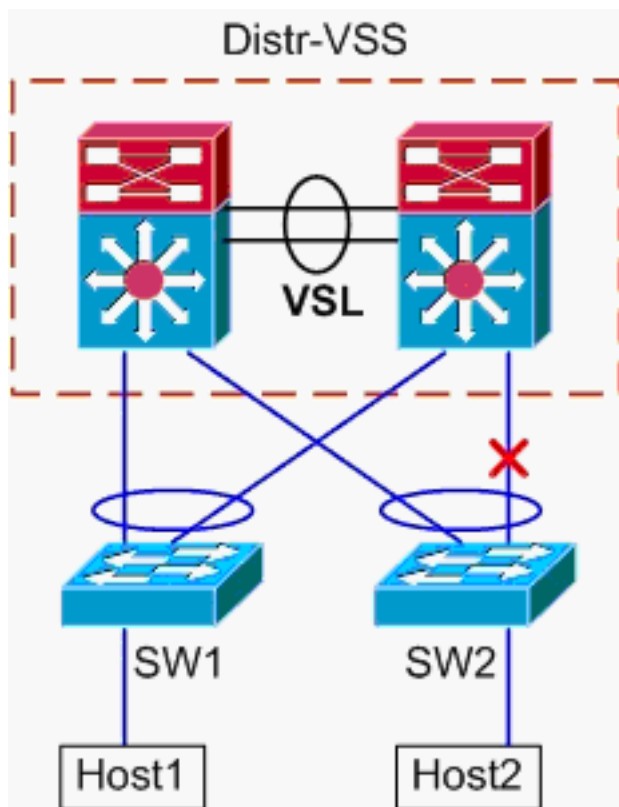
```
Legend: * - primary entry
         age - seconds since last seen
         n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

パケット フロー図



**シナリオ 2：レイヤ 2 MEC を使用する 2 台のアクセス層ホスト間のパケットフロー：冗長性の喪失**



1. Host1 から VSS ディストリビューションへのパスをトレースします。手順は、[シナリオ 1](#) のステップ 1 と同じです。
2. VSS ディストリビューションを介するパスをトレースします。このシナリオは、Distr-VSS スイッチ 2 と SW2 の間のリンクが切断されている点を除き、シナリオ 1 と同じです。このため、スイッチ 2 上に port-channel2 内のアクティブリンクは存在しません。これは、Host1 からのパケットが VSS に入る場所です。したがって、パケットはVSLおよび出力スイッチ1を通過する必要があります。このハッシュ結果の出力は次を示します。

```
VSS#show etherchannel load-balance hash-result interface port-channel 2 switch 2 ip
10.0.1.15 vlan 10 10.0.2.30
Computed RBH: 0x6
Would select Gi1/1/13 of Po2
```

hash-result コマンドは、フレームを送信するために選択される VSL リンクの判別にも使用できます。この場合、Port-channel10 はスイッチ 1 上の VSL、Port-channel20 はスイッチ 2 VSL です。

```
VSS#show etherchannel load-balance hash-result int port-channel 20 switch 2 ip 10.0.1.15
vlan 10 10.0.2.30
Computed RBH: 0x6
Would select Te2/5/4 of Po20
```

次に、CDP テーブルを使用して、Host2 へのダウンストリーム スイッチに関する情報を検索します。

```
VSS#show cdp nei
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW2	Gig 2/1/13	129	R S I	WS-C6503-	Gig 3/14
SW2	Gig 1/1/13	129	R S I	WS-C6503-	Gig 3/13

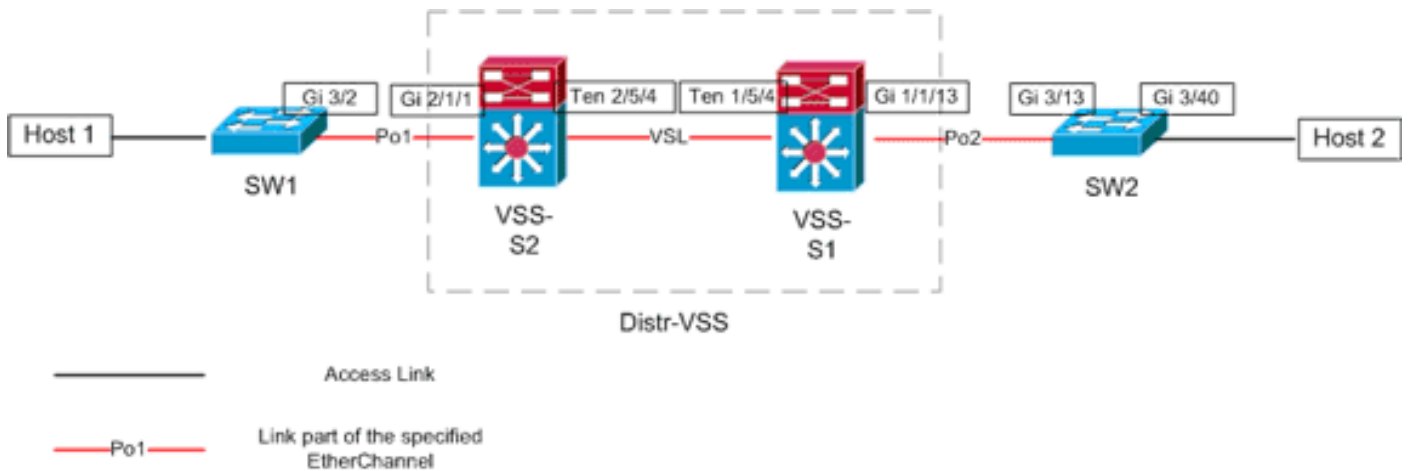
### 3. Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テーブルを使用して、Host2 が接続されている正確なポートを判別します。

```
SW2#show mac-address-table address 0002.0002.0002
```

```
Legend: * - primary entry
        age - seconds since last seen
        n/a - not available
```

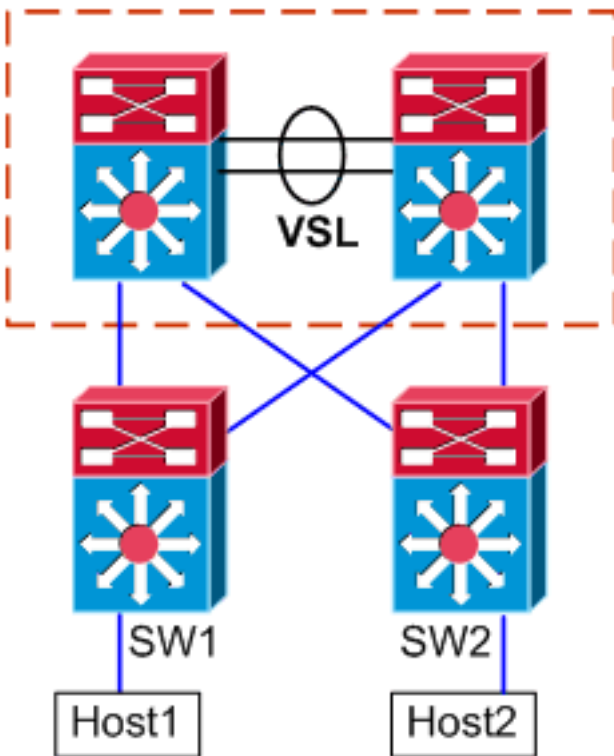
vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

### パケット フロー



### シナリオ 3 : レイヤ 3 MEC を使用する 2 台のアクセス層ホスト間のパケット フロ

## Distr-VSS



### トポロジ情報

- Host1 IP/MASK : 10.0.1.15/24
- Host1 MAC : 0001.0001.0001
- Host1 デフォルト ゲートウェイ : 10.0.1.1 : SW1 上
- Host2 IP : 10.0.2.30
- SW1 と SW2 はいずれも、Distr-VSS に接する、ルーティングされた EtherChannel を持つ、レイヤ 3 で動作している Catalyst 6500 スイッチ

1. Host1 から VSS ディストリビューションへのパスをトレースします。Host1 は、SW1 によってレイヤ 3 で終了しているため、まずは、SW1 のルーティング テーブルを検索して、Host2 のある場所を判別します。

```
SW1#show ip route 10.0.2.30
```

```
Routing entry for 10.0.2.0/24
```

```
Known via "static", distance 1, metric 0
```

```
Routing Descriptor Blocks:
```

```
* 10.100.1.1
```

```
Route metric is 0, traffic share count is 1
```

```
SW1#show ip route 10.100.1.1
```

```
Routing entry for 10.100.1.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via Port-Channel1
```

```
Route metric is 0, traffic share count is 1
```

```
SW1#sh etherchannel 1 summary
```

```
Flags: D - down P - bundled in port-channel
```

```
I - stand-alone s - suspended
```

```
H - Hot-standby (LACP only)
```

```
R - Layer3 S - Layer2
```

```
U - in use N - not in use, no aggregation
```

```
f - failed to allocate aggregator
```

```

M - not in use, no aggregation due to minimum links not met
m - not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
d - default port

w - waiting to be aggregated
Number of channel-groups in use: 4
Number of aggregators:          4

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
1      Po1(RU)         LACP      Gi3/1(P)   Gi3/2(P)
Last applied Hash Distribution Algorithm:  -

```

SW1#show cdp neighbor

```

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID        Local Intrfce   Holdtme    Capability  Platform  Port ID
VSS               Gig 3/2         126        R S I       WS-C6509-EGig 2/1/1
VSS               Gig 3/1         128        R S I       WS-C6509-EGig 1/1/1

```

上記の出力は、10.100.1.1を経由する宛先への単一のルートを示しています。show etherchannelコマンドの出力は、Port-channel1がGig3/1とGig3/2で構成され、CDPテーブルは物理スイッチごとに1つのリンクでVSSに接続されることを示示示します。次に、etherchannel hash-result コマンドを使用して、Host1 から Host2 への出力の正確なポイントを判別する必要があります。

SW1#show etherchannel load-balance

```

EtherChannel Load-Balancing Configuration:
    src-dst-ip
    mpls label-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
IPv4:   Source XOR Destination IP address
IPv6:   Source XOR Destination IP address
MPLS:   Label or IP

```

次の出力は、IPv4 パケットの場合のアルゴリズムは、src-dst-ipであることを示しています。次に、関連するフロー情報を hash-result CLI に入力します。

```

SW1#show etherchannel load-balance hash-result interface port-channel 1 ip 10.1.1.1
10.0.2.30
Computed RBH: 0x1
Would select Gig3/2 of Po1

```

これで、フローは Gi3/2 を介して SW1 から出て、スイッチ 1 上に存在する Gig2/1/1 上の VSS に入ることがはっきりしました。

## 2. VSS デISTRIBUTION を介するパスをトレースします。次に、VSS 上のルーティング テーブル エントリをチェックする必要があります。

```

VSS#show ip route 10.0.2.30
Routing entry for 10.0.2.0/24
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks:
    * 10.200.1.2
      Route metric is 0, traffic share count is 1

VSS#show ip route 10.200.1.2
Routing entry for 10.200.1.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)

```

```
Routing Descriptor Blocks:
* directly connected, via Port-channel2
  Route metric is 0, traffic share count is 1
```

以前のCDP出力から、このフローのパケットがスイッチ2、モジュール1、ポート1に対応するGig2/1/1上のVSSに入ったことを思い出してください。ここでも、hash-resultコマンドを使用してVSSから出る物理的なポイントを確認し、Po1:

```
VSS#show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
  src-dst-mixed-ip-port enhanced
  mpls label-ip
```

```
EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
MPLS: Label or IP
```

```
VSS#show vlan internal usage | include Port-channel 1
1026 Port-channel 1
```

```
VSS#show etherchannel load-balance hash-result interface port-channel 2 switch 2 ip
10.0.1.15 vlan 1026 10.0.2.30
Computed RBH: 0x6
Would select Gi2/1/13 of Po2
```

次に、CDP テーブルを使用して、Host2 へのダウンストリーム スイッチに関する情報を検索します。

```
VSS#show cdp nei
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW2	Gig 2/1/13	129	R S I	WS-C6503-	Gig 3/14
SW2	Gig 1/1/13	129	R S I	WS-C6503-	Gig 3/13

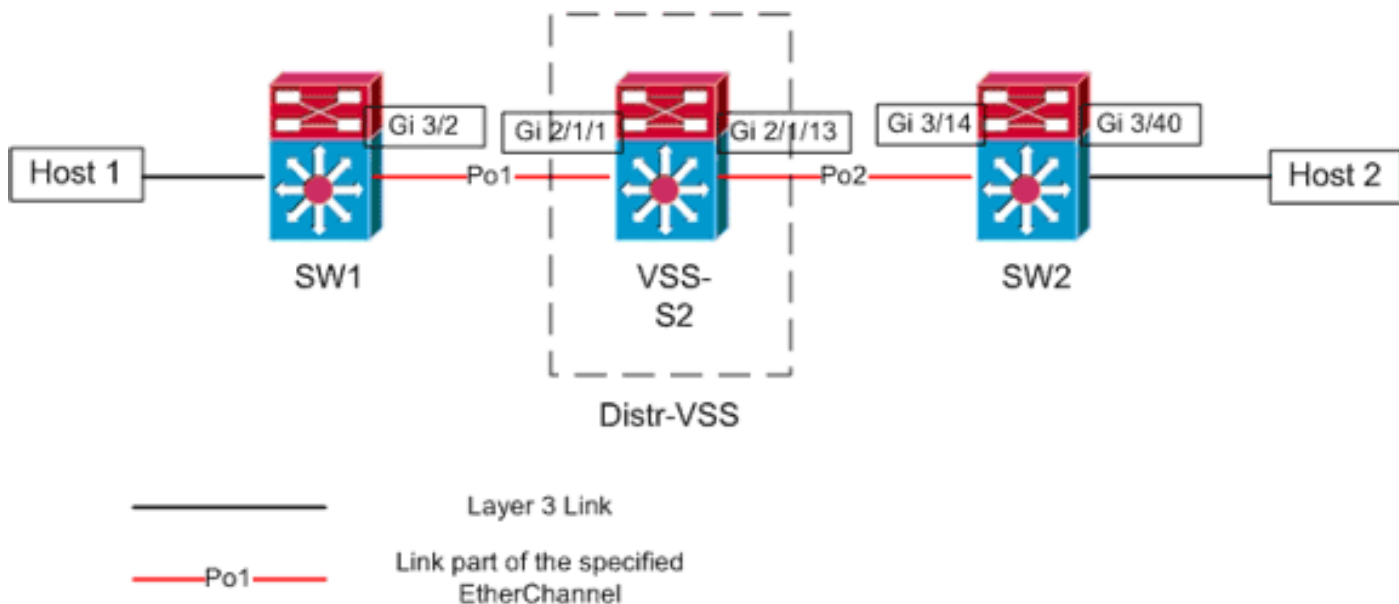
この情報は、前述の CDP 出力によって示されたとおり、パケットが Gig2/1/13 を介して VSS から出て、Gig3/14 上の SW2 に入ることを示しています。

- Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テーブルを使用して、Host2 が接続されている正確なポートを判別します。

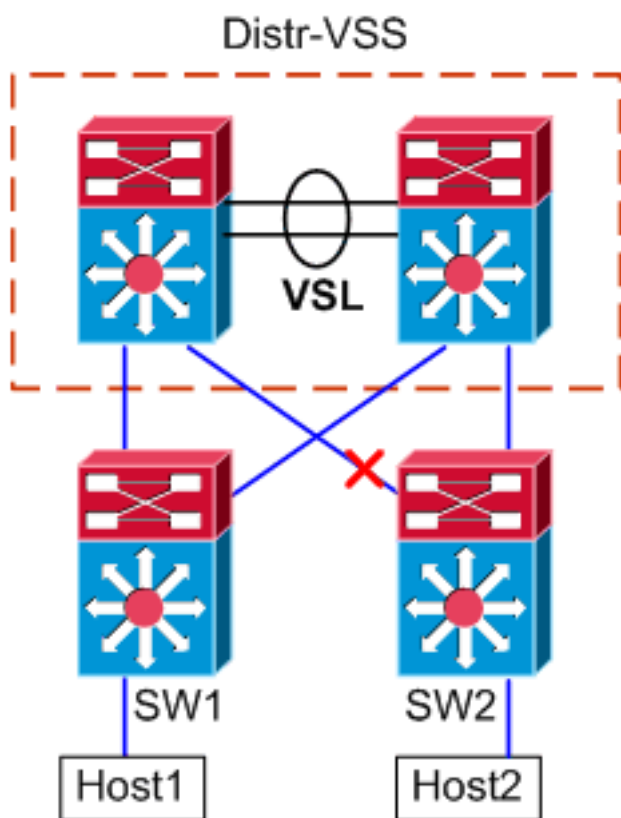
```
SW2#show mac-address-table address 0002.0002.0002
Legend: * - primary entry
        age - seconds since last seen
        n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

パケット フロー図



## シナリオ 4：レイヤ 3 MEC を使用する 2 台のアクセス層ホスト間のパケット フロー：冗長性の喪失



- Host1 から VSS ディストリビューションへのパスをトレースします。手順は、[シナリオ 3](#) のステップ 1 と同じです。
- VSS ディストリビューションを介するパスをトレースします。このシナリオは、Distr-VSS スイッチ 2 と SW2 間のリンクが切断される点を除き、シナリオ 3 と同じです。このため、Host1 からのパケットが VSS に入るため、パケットが VSL および出力スイッチ 1 を通過する必要があるスイッチ 2 には、port-channel 2 内のアクティブなリンクは存在しません。次の hash-result 出力は、これを示します。

```
VSS#show etherchannel load-balance hash-result interface port-channel 2 switch 2 ip
10.0.1.15 vlan 1026 10.0.2.30
Computed RBH: 0x6
Would select Gi1/1/13 of Po2
```



hash-result コマンドは、フレームを送信するために選択される VSL リンクの判別にも使用できます。この場合、Port-channel10 はスイッチ 1 上の VSL、Port-channel20 はスイッチ 2 VSL です。

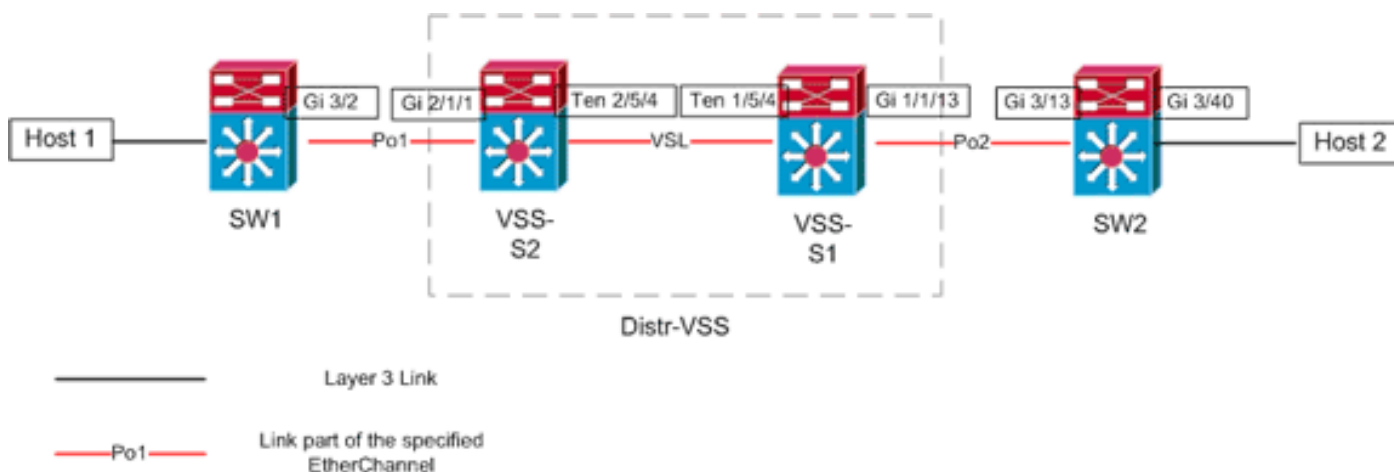
```
VSS#show etherchannel load-balance hash-result int port-channel 20 switch 2 ip 10.0.1.15
vlan 1026 10.0.2.30
Computed RBH: 0x6
Would select Te2/5/4 of Po20
```

3. Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テーブルを使用して、Host2 が接続されている正確なポートを判別します。

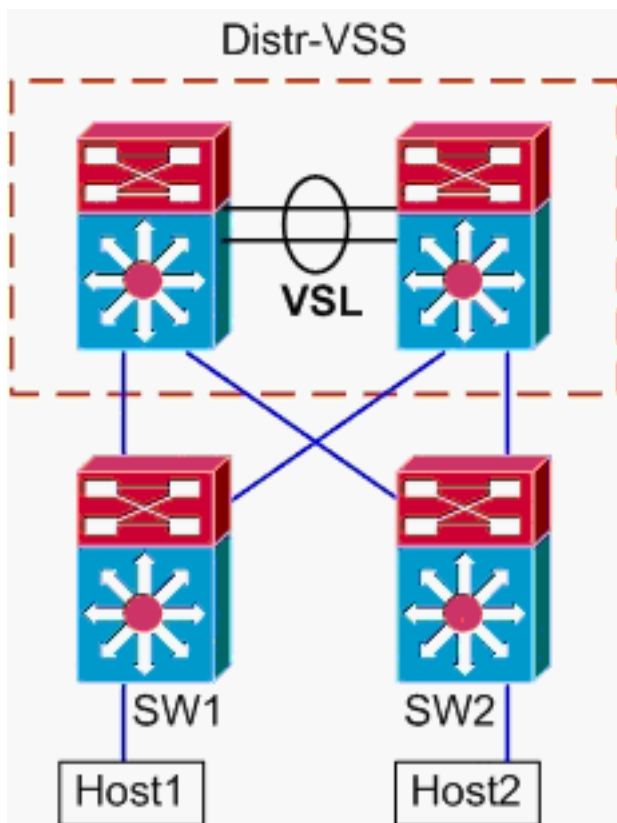
```
SW2#show mac-address-table address 0002.0002.0002
Legend: * - primary entry
age - seconds since last seen
n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

### パケットフロー図



### シナリオ 5 : ECMP を使用する 2 台のアクセス層ホスト間のパケット フロー



## トポロジ情報

- Host1 IP/MASK : 10.0.1.15/24
- Host1 MAC : 0001.0001.0001
- Host1 デフォルト ゲートウェイ : 10.0.1.1 : SW1 上
- Host2 IP : 10.0.2.30
- Catalyst 6500 では、SW1 と SW2 の両方が、Distr-VSS に接するルーティングされたリンクを持ち、接続されたサブネットをレイヤ 3 で終端

1. Host1 から VSS ディストリビューションへのパスをトレースします。Host1 は、SW1 によってレイヤ 3 で終了しているため、まずは、SW1 のルーティング テーブルを検索して、Host2 のある場所を判別します。

```
SW1#show ip route 10.0.2.30
```

```
Routing entry for 10.0.2.0/24
```

```
Known via "static", distance 1, metric 0
```

```
Routing Descriptor Blocks:
```

```
* 10.100.1.1
```

```
Route metric is 0, traffic share count is 1
```

```
10.100.2.1
```

```
Route metric is 0, traffic share count is 1
```

```
SW1#show ip route 10.100.1.1
```

```
Routing entry for 10.100.1.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via GigabitEthernet3/1
```

```
Route metric is 0, traffic share count is 1
```

```
SW1#show ip route 10.100.2.1
```

```
Routing entry for 10.100.2.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via GigabitEthernet3/2
```

```
Route metric is 0, traffic share count is 1
```

```
SW1#show cdp neighbor
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge  
S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
VSS	Gig 3/2	126	R S I	WS-C6509-EGig	2/1/1
VSS	Gig 3/1	128	R S I	WS-C6509-EGig	1/1/1

上記の出力は、10.100.1.1 および 10.100.2.1 を介した等コスト ルートを示します。これらのルートは、それぞれ Gig3/1 および Gig3/2 を介して接続しています。CDP テーブルは、物理スイッチあたり 1 つのリンクで、Gig3/1 と Gig3/2 の両方が VSS に接続していることを示しています。次に、**exact-route** コマンドを使用して、Host1 から Host2 への出力の正確なポイントを判別する必要があります。

```
SW1#show mls cef exact-route 10.0.1.15 10.0.2.30
```

```
Interface: Gi3/1, Next Hop: 10.100.1.1, Vlan: 1030, Destination Mac: 000a.000a.000a
```

これで、フローは Gi3/1 を介して SW1 から出て、スイッチ 1 上に存在する Gig1/1/1 上の VSS に入ることがはっきりしました。

## 2. VSS ディストリビューションを介するパスをトレースします。次に、VSS 上のルーティング テーブル エントリをチェックする必要があります。

```
VSS#show ip route 10.0.2.30
```

```
Routing entry for 10.0.2.0/24
```

```
Known via "static", distance 1, metric 0
```

```
Routing Descriptor Blocks:
```

```
10.200.2.2
```

```
Route metric is 0, traffic share count is 1
```

```
* 10.200.1.2
```

```
Route metric is 0, traffic share count is 1
```

```
VSS#show ip route 10.200.2.2
```

```
Routing entry for 10.200.2.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via GigabitEthernet2/1/13
```

```
Route metric is 0, traffic share count is 1
```

```
VSS#show ip route 10.200.1.2
```

```
Routing entry for 10.200.1.0/24
```

```
Known via "connected", distance 0, metric 0 (connected, via interface)
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via GigabitEthernet1/1/13
```

```
Route metric is 0, traffic share count is 1
```

```
VSS#show cdp nei
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
```

```
S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
```

```
D - Remote, C - CVTA, M - Two-port Mac Relay
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW2	Gig 1/1/13	121	R S I	WS-C6503-	Gig 3/13
SW2	Gig 2/1/13	121	R S I	WS-C6503-	Gig 3/14

ここでは、宛先に対する等コスト パスが再度存在し、スイッチごとに 1 つの出力ポイントです。先に確認したように、パケットはスイッチ 1 上の VSS に入るため、次に、スイッチ 1 を指定して **exact-route** コマンドを発行します。

```
VSS#show mls cef exact-route 10.0.1.15 10.0.2.30 switch 1
```

```
Interface: Gi1/1/13, Next Hop: 10.200.1.2, Vlan: 1095, Destination Mac: 000b.000b.000b
```

この情報は、前述の CDP 出力によって示されたとおり、パケットが Gig1/1/13 を介して VSS から出て、Gig3/13 上の SW2 に入ることを示しています。

## 3. Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テー

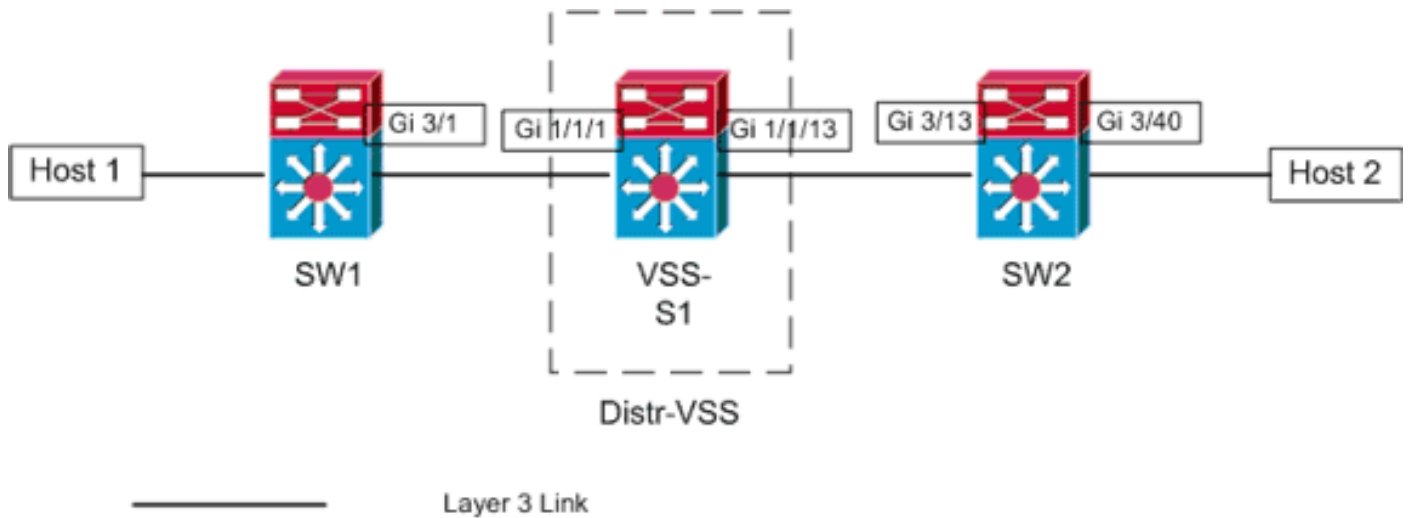
ブルを使用して、Host2 が接続されている正確なポートを判別します。

```
SW2#show mac-address-table address 0002.0002.0002
```

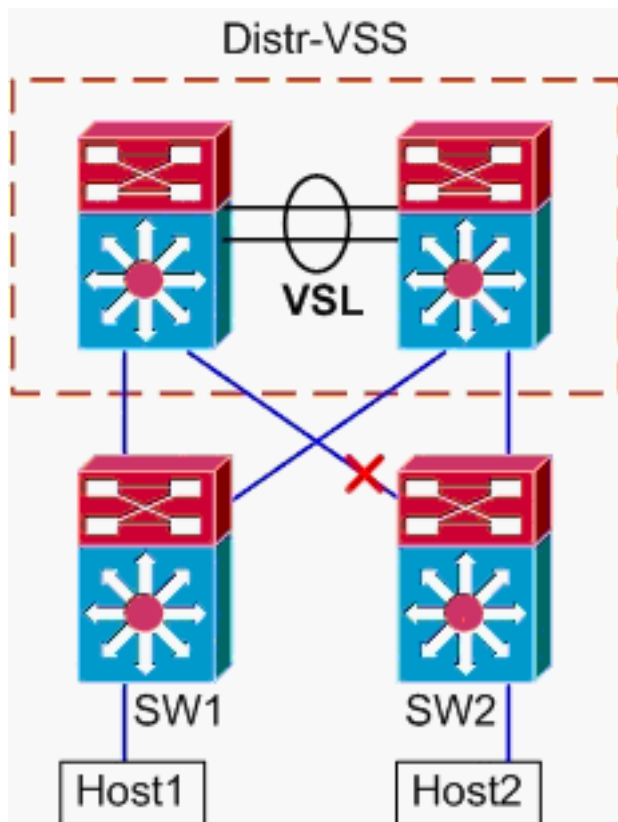
```
Legend: * - primary entry  
age - seconds since last seen  
n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

### パケットフロー図



### シナリオ 6 : ECMP を使用する 2 台のアクセス層ホスト間のパケットフロー : 冗長性の喪失



1. Host1 から VSS ディストリビューションへのパスをトレースします。手順は、シナリオ5のステップ1と同じです。
2. VSS ディストリビューションを介するパスをトレースします。hash-result コマンドは、フ

レームを送信するために選択される VSL リンクの判別にも使用できます。この場合、Port-channel10 はスイッチ 1 上の VSL、Port-channel20 はスイッチ 2 VSL です。入力 VLAN は、Gig1/1/1 の内部 VLAN、つまり入カインターフェイスです。

```
VSS#show vlan internal usage | include 1/1/1
```

```
1026 GigabitEthernet1/1/1
```

```
VSS#show etherchannel load-balance hash-result int port-channel 10 switch
```

```
1 ip 10.0.1.15 vlan 1026 10.0.2.30
```

```
Computed RBH: 0x4
```

```
Would select Te1/5/5 of Po10
```

3. Host2 へのパスをトレースします。最後に、SW2 にログインし、再度 MAC アドレス テーブルを使用して、Host2 が接続されている正確なポートを判別します。

```
SW2#show mac-address-table address 0002.0002.0002
```

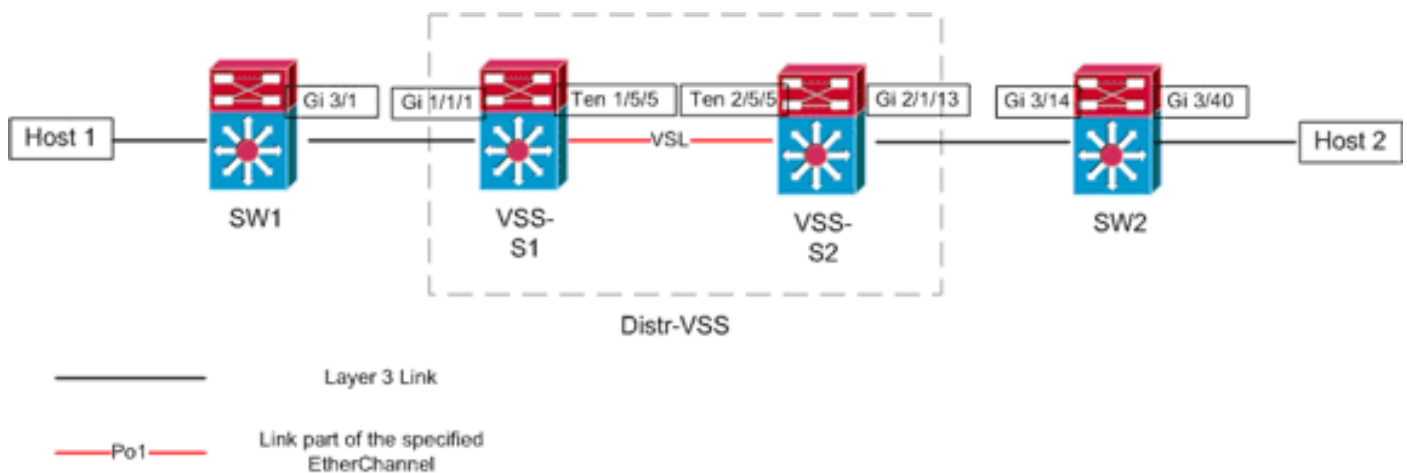
```
Legend: * - primary entry
```

```
age - seconds since last seen
```

```
n/a - not available
```

vlan	mac address	type	learn	age	ports
20	0002.0002.0002	dynamic	Yes	140	Gi3/40

## パケット フロー図



## 関連情報

- [Cisco Catalyst 6500 Virtual Switching System 配備のベスト プラクティス](#)
- [Cisco サービス モジュールの Cisco Catalyst 6500 Virtual Switching System 1440 への統合](#)
- [Cisco Catalyst 6500 Virtual Switching System 1440 製品に関するサポート ページ](#)
- [LAN 製品に関するサポート ページ](#)
- [LAN スイッチング テクノロジーに関するサポート ページ](#)
- [テクニカル サポートとドキュメント - Cisco Systems](#)