VLAN のトラブルシューティング

VXLAN の問題のトラブルシューティング

VXLAN データ パスには、次のパスが含まれます。

- マルチキャスト カプセル化パス：ネイティブ レイヤ 2 パケットは、ネットワーク（レイヤ 2 からレイヤ 3）方向へのアクセスで VXLAN にカプセル化されます。
- マルチキャスト カプセル化解除パス：ネイティブ レイヤ 2 パケットはネットワークの VXLAN でカプセル化解除され、（レイヤ 3 からレイヤ 2 へ）方向にアクセスします。
- ユニキャスト カプセル化パス：ネイティブ レイヤ 2 パケットは、ネットワーク（レイヤ 2 からレイヤ 3）方向へのアクセスで VXLAN にカプセル化されます。
- ユニキャスト カプセル化解除パス：ネイティブのレイヤ 2 パケットがネットワークの VXLAN でカプセル化解除され、（レイヤ 3 からレイヤ 2 へ）方向にアクセスします。

これらのデータ パスを理解すると、VXLAN の問題のトラブルシューティングに役立ちます。

注意 VXLAN の問題をトラブルシューティングするには、Broadcom シェル コマンドを実行する必要があります。Broadcom シェル コマンドは、シスコのサポート担当者の直接監督下または要求された場合のみ注意して使用してください。

（注） Cisco Nexus 9300 シリーズ スイッチは、VXLAN をサポートしています。Cisco Nexus 9500 シリーズ スイッチはサポートしていません。

マルチキャスト カプセル化パスでドロップされたパケット

ネットワークにアクセスする方向にデバイスで ARP 要求またはマルチキャスト パケットがドロップされている場合は、次の手順に従います。

手順の概要

1. Broadcom シェルにアクセスします。
2. **stg show** コマンドの出力を調べて、特定の VLAN のポートが STP 転送状態になっているかどうかを確認します。
3. ポートが VLAN の一部であるかどうかを確認します。
4. **mc show** コマンドの出力を調べて、ローカル VLAN ポートとカプセル化ポートがカプセル化フラッド リストに含まれているかどうかを確認します。
5. **mc show** コマンドの出力が正しくない場合は、Broadcom シェル モードを終了し、**showtech-support pixm**、**show tech-support pixm-all**、**show tech-support pixmc-all** コマンドを実行し、出力を表示します。

手順の詳細

手順

ステップ 1 Broadcom シェルにアクセスします。

例：

```
switch# bcm-shell module 1
Warning: BCM shell access should be used with caution
Entering bcm shell on module 1
Available Unit Numbers: 0
```

ステップ 2 **stg show** コマンドの出力を調べて、特定の VLAN のポートが STP 転送状態になっているかどうかを確認します。

例：

```
bcm-shell.0> stg show
STG 6: contains 1 VLAN (3)
  Disable: xe56-xe95
    Block: xe0-xe22,xe24-xe55
  Forward: xe23,hg
```

この例では、VLAN 3 に eth1/24 があり、アップリンク トンネル ポートが eth2/2 であるため、出力に xe23（1/24）と hg が表示されます。

ステップ 3 ポートが VLAN の一部であるかどうかを確認します。

例：

```
bcm-shell.0> vlan show
vlan 3  ports xe23
(0x00000000000000000000000000000000000000000000000000000100000000), untagged
```

```
none (0x00000000000000000000000000000000
00000000000000000000000000000000) MCAST_FLOOD_UNKNOWN
```

この例では、xe23 は VLAN 3 の一部である必要があります。

ステップ 4 **mc show** コマンドの出力を調べて、ローカル VLAN ポートとカプセル化ポートがカプセル化フラッド リストに含まれているかどうかを確認します。

a) カプセル化フラッド リストを取得します。

例：

```
bcm-shell.0> d chg vfi 3
Private image version: R
VFI.ipipe0[3]:
<VP_1=0xc01,VP_0=0x1803,UUC_INDEX=0x1803,UMC_INDEX=0x1803,RSVD_VP_0=1,BC_INDEX=0x1803>
```

この例では、0x1803 がカプセル化フラッド リストです。

b) カプセル化フラッド リストを **mc show** コマンドに入力します。

例：

```
bcm-shell.0> mc show 0x1803
Group 0xc001803 (VXLAN)
        port hg7, encap id 400053
        port xe23, encap id 400057
```

この例では、hg7 はアップリンク トンネル ポートで、xe23 は VLAN のローカル ポートです。

アップリンクがポート チャネルの場合、ポート チャネルのすべてのメンバーが出力に表示されます。出力に重複エントリが含まれている場合、対応するパケット レプリケーションがあります。

ステップ 5 **mc show** コマンドの出力が正しくない場合は、Broadcom シェル モードを終了し、**showtech-support pixm**、**show tech-support pixm-all**、**show tech-support pixmc-all** コマンドを実行し、出力を表示します。

例：

```
bcm-shell.0> exit
switch# show tech-support pixm
switch# show tech-support pixm-all
switch# show tech-support pixmc-all
```

マルチキャスト カプセル化解除パスでドロップされたパケット

ネットワークがアクセスする方向にデバイスで ARP 要求またはマルチキャスト パケットがドロップされている場合は、次の手順に従います。

手順の概要

1. パケットがスーパーバイザに送信されたかどうか、およびリモート VXLAN トンネル エンドポイント（VTEP）の検出が行われたかどうかを確認します。

2. ハードウェアに mpls_entry が存在する場合は、vlan_xlate テーブルを確認します。
3. vlan_xlate テーブルにマルチキャスト DIP の正しいエントリがある場合は、VLAN フラッディング リストに正しいメンバー（カプセル化トンネル ポートを除く VLAN のメンバー）が表示されているかどうかを確認します。

手順の詳細

手順

ステップ 1 パケットがスーパーバイザに送信されたかどうか、およびリモート VXLAN トンネル エンドポイント（VTEP）の検出が行われたかどうかを確認します。

a) リモート ピアがソフトウェアで学習されたかどうかを確認します。

例：

```
switch# show nve peers
Interface          Peer-IP           VNI        Up Time
-----------------  ---------------   --------   -----------
nve1               100.100.100.5     10000      00:02:23
```

b) mpls_entry テーブルを確認して、リモート ピアがハードウェアで学習されたかどうかを確認します。

例：

```
switch# bcm-shell module 1
bcm-shell.0> d chg mpls_entry | grep SVP
MPLS_ENTRY.ipipe0[12368]:
<VXLAN_SIP:SVP=0x1751,VXLAN_SIP:SIP=0x66666666,VXLAN_SIP:KEY=0x666666668,VXLAN_SIP:HASH_LSB=0x666
VXLAN_SIP:DATA=0
x1751,VALID=1,KEY_TYPE=8,>
```

c) mpls_entry がなく、送信元仮想ポート（SVP）がない場合は、パケットがスーパーバイザに送信されているかどうかを確認し、IPFIB エラーがないかどうかを確認します。

例：

```
bcm-shell.0> show c cpu0
bcm-shell.0> exit
switch# attach module 1
module-1# show system internal ipfib errors
```

ステップ 2 ハードウェアに mpls_entry が存在する場合は、vlan_xlate テーブルを確認します。

例：

```
module-1# exit
switch# bcm-shell module 1
bcm-shell.0> d chg vlan_xlate | grep 0xe1000003
VLAN_XLATE.ipipe0[8464]:
<VXLAN_DIP:KEY=0x7080000192,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=3,VXLAN_DIP:DIP=0xe1000003,
XLAN_DIP
:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

vlan_xlate テーブルには、パケットのマルチキャスト宛先 IP アドレス（DIP）のエントリが 1 つ必要です。この例では、マルチキャストパケットが 225.0.0.3 に送信される場合を示しています。

ステップ 3 vlan_xlate テーブルにマルチキャスト DIP の正しいエントリがある場合は、VLAN フラッディング リストに正しいメンバー（カプセル化トンネル ポートを除く VLAN のメンバー）が表示されているかどうかを確認します。

a) VLAN フラッディング リストを確認します。

例：

```
bcm-shell.0> d chg vfi 3
Private image version: R
VFI.ipipe0[3]:
<VP_1=0xc01,VP_0=0x1803,UUC_INDEX=0x1803,UMC_INDEX=0x1803,RSVD_VP_0=1,BC_INDEX=0x1803>
```

0x1803 のカプセル化フラッド リストの場合、対応するカプセル化解除フラッド リストは 0x1c03 になります。

b) ローカル ポートがカプセル化解除フラッド リストに含まれているかどうかを確認します。

例：

```
bcm-shell.0> mc show
Group 0xc001c03 (VXLAN)
        port xe23, encap id 400057
```

xe23 はカプセル化解除フラッド リストの一部である必要があります。

c) ポートがフォワーディング ステートであり、VLAN の一部であることを確認します。

例：

```
bcm-shell.0> stg show
bcm-shell.0> vlan show
```

ユニキャスト カプセル化パスでドロップされたパケット

単一のネクスト ホップで VTEP に到達でいる場合にドロップユニキャスト パケット

アクセスからネットワーク方向のデバイスでユニキャスト パケットがドロップされ、VTEP が ECMP パスを介して到達可能である場合は、次の手順に従います。

手順の概要

1. リモート ピアがハードウェアで検出されたかどうかを確認します。
2. ネクストホップへの送信元仮想ポート（SVP）のマッピングを取得します。
3. ネクストホップ インデックスからポート番号を取得します。
4. ポート番号からチップ上の物理ポートへのマッピングを取得します。
5. 出力ポートからネクストホップ インデックスへのマッピングを取得します。

6. トンネル パラメータをチェックして、EGR IP トンネルの SIP フィールドに正しいローカル VTEP IP アドレスが表示されていることを確認します。
7. トンネル DIP がプログラムされていることを確認します。

手順の詳細

手順

ステップ 1 リモート ピアがハードウェアで検出されたかどうかを確認します。

例：

```
switch# bcm-shell module 1
bcm-shell.0> d chg mpls_entry | grep SVP
MPLS_ENTRY.ipipe0[12368]:
<VXLAN_SIP:SVP=0x1751,VXLAN_SIP:SIP=0x66666666,VXLAN_SIP:KEY=0x666666668,VXLAN_SIP:HASH_LSB=0x666
VXLAN_SIP:DATA=0
x1751,VALID=1,KEY_TYPE=8,>
```

有効な送信元 IP アドレス（SIP）が存在することを確認します。

この例では、102.102.102.102 がリモート VTEP IP アドレスです。

ステップ 2 ネクストホップへの送信元仮想ポート（SVP）のマッピングを取得します。

例：

```
bcm-shell.0> d chg ing_dvp_table 0x1751
Private image version: R
ING_DVP_TABLE.ipipe0[5969]:
<VP_TYPE=3,NEXT_HOP_INDEX=0x18,NETWORK_PORT=1,ECMP_PTR=0x18,DVP_GROUP_PTR=0x18,>
```

この例では、ネクストホップインデックスは 0x18 です。

ステップ 3 ネクストホップ インデックスからポート番号を取得します。

例：

```
bcm-shell.0> d chg ing_l3_next_hop 0x18
Private image version: R
ING_L3_NEXT_HOP.ipipe0[24]:
<VLAN_ID=0xfff,TGID=0x88,PORT_NUM=8,MTU_SIZE=0x3fff,MODULE_ID=1,L3_OIF=0x1fff,ENTRY_TYPE=2
ENTRY_INFO_UPPER=3,DV
P_RES_INFO=0x7f,>
```

この例では、ポート番号は8です。

ステップ 4 ポート番号からチップ上の物理ポートへのマッピングを取得します。

例：

```
bcm-shell.0> phy info
Phy mapping dump:
      port   id0   id1  addr iaddr          name    timeout
  hg0(  1)  600d  8770   1b1   1b1   TSC-A2/31/4     250000
  hg1(  2)  600d  8770    81    81   TSC-A2/00/4     250000
```

```
  hg2(  3)  600d  8770   1ad   1ad   TSC-A2/30/4     250000
  hg3(  4)  600d  8770    85    85   TSC-A2/01/4     250000
  hg4(  5)  600d  8770   189   189   TSC-A2/23/4     250000
  hg5(  6)  600d  8770    ad    ad   TSC-A2/08/4     250000
  hg6(  7)  600d  8770   185   185   TSC-A2/22/4     250000
  hg7(  8)  600d  8770    b1    b1   TSC-A2/09/4     250000
  xe0(  9)  600d  84f9     0    89      BCM84848     250000
  xe1( 10)  600d  84f9     1    8a      BCM84848     250000
  xe2( 11)  600d  84f9     2    8b      BCM84848     250000
  xe3( 12)  600d  84f9     3    8c      BCM84848     250000
```

この例では、ポート番号 8 は hg7 です。

ステップ 5 出力ポートからネクストホップ インデックスへのマッピングを取得します。

例：

```
bcm-shell.0> g chg egr_port_to_nhi_mapping
EGR_PORT_TO_NHI_MAPPING.hg7[2][0x4001808]=0x18: <NEXT_HOP_INDEX=0x18>
```

この例では、ネクストホップ インデックス 0x18 は hg7 を指しています。

ステップ 6 トンネル パラメータをチェックして、EGR IP トンネルの SIP フィールドに正しいローカル VTEP IP アドレスが表示されていることを確認します。

例：

```
bcm-shell.0> d chg egr_ip_tunnel
Private image version: R
EGR_IP_TUNNEL.epipe0[1]:
<TUNNEL_TYPE=0xb,TTL=0xff,SIP=0x65656565,L4_DEST_PORT=0x2118,ENTRY_TYPE=1,DSCP_SEL=1,>
```

この例では、SIP はローカル VTEP IP アドレス（101.101.101.101）で、L4_DEST_PORT は 0x2118（ポート 8472）で、DSCP_SEL = 1 は内部 DSCP パケットが外部 DSCP パケットにコピーされることを意味します。

ステップ 7 トンネル DIP がプログラムされていることを確認します。

例：

```
bcm-shell.0> d chg egr_dvp_attribute 0x1751
Private image version: R
EGR_DVP_ATTRIBUTE.epipe0[5969]:
<VXLAN:TUNNEL_INDEX=1,VXLAN:DVP_IS_NETWORK_PORT=1,VXLAN:DIP=0x66666666,VP_TYPE=2,>
```

VTEP が ECMP パスを介して到達可能な場合にドロップされるユニキャスト パケット

ネットワーク方向にアクセスするデバイスでユニキャストパケットがドロップされ、VTEP が ECMP パスを介して到達可能である場合は、次の手順に従います。

手順の概要

1. 特定のリモートピア仮想ポート（VP）の ECMP ネクストホップを取得します。
2. ECMP_PTR を 10 進数に変換し、200000 を追加してポート番号を取得します。
3. ECMP ネクストホップ セット内のインターフェイスのリストを取得します。

4. ポート チャネルのメンバーを検索します。
5. 特定のネクストホップ インデックスの物理ネクストホップ インターフェイスを検索します。

手順の詳細

手順

ステップ 1 特定のリモートピア仮想ポート（VP）の ECMP ネクストホップを取得します。

例：

```
bcm-shell.0> d chg ing_dvp_table 0x1751
Private image version: R
ING_DVP_TABLE.ipipe0[5969]:
<VP_TYPE=3,NEXT_HOP_INDEX=0x108,NETWORK_PORT=1,ECMP_PTR=0x108,ECMP=1,DVP_GROUP_PTR=0x108,>
```

この例では、0x1751 は、d chg mpls_entry 出力を使用して取得されたリモート ピア IP アドレスの VP 番号です。

（注）
リモート VTEP が ECMP パスを介して到達可能である場合、出力に ECMP=1 が存在する必要があります。

ステップ 2 ECMP_PTR を 10 進数に変換し、200000 を追加してポート番号を取得します。

例：

```
0x108 (264) + 200000 = 200264
```

この例では、ポート番号は 200264 です。

ステップ 3 ECMP ネクストホップ セット内のインターフェイスのリストを取得します。

例：

```
bcm-shell.0> d chg l3 multipath show 200264
Multipath Egress Object 200264
Interfaces: 100606 100607 100608
Reference count: 2
bcm-shell.0> l3 egress show | grep 100606
100606  00:22:bd:f5:1a:60 4095 4101     1t   0        -1   no   no
bcm-shell.0> l3 egress show | grep 100607
100607  00:22:bd:f5:1a:60 4095 4102     2t   0        -1   no   no
bcm-shell.0> l3 egress show | grep 100608
100608  00:22:bd:f5:1a:60 4095 4103     3t   0        -1   no   no
```

この例では、ネクストホップ インターフェイスはポート チャネルである 1t、2t、および 3t です。

ステップ 4 ポート チャネルのメンバーを検索します。

例：

```
bcm-shell.0> trunk show
Device supports 1072 trunk groups:
  1024 front panel trunks (0..1023), 256 ports/trunk
  48 fabric trunks (1024..1071), 64 ports/trunk
```

```
trunk 0: (front panel, 0 ports)
trunk 1: (front panel, 1 ports)=hg6 dlf=any mc=any ipmc=any psc=portflow (0x9)
trunk 2: (front panel, 1 ports)=hg4 dlf=any mc=any ipmc=any psc=portflow (0x9)
trunk 3: (front panel, 1 ports)=hg7 dlf=any mc=any ipmc=any psc=portflow (0x9)
```

ステップ 5 特定のネクストホップ インデックスの物理ネクストホップ インターフェイスを検索します。

例：

```
bcm-shell.0> g chg egr_port_to_nhi_mapping
EGR_PORT_TO_NHI_MAPPING.hg4[2][0x4001805]=0x5f7: <NEXT_HOP_INDEX=0x5f7>
EGR_PORT_TO_NHI_MAPPING.hg6[2][0x4001807]=0x9b3: <NEXT_HOP_INDEX=0x9b3>
EGR_PORT_TO_NHI_MAPPING.hg7[2][0x4001808]=0x5f8: <NEXT_HOP_INDEX=0x5f8>
```

この例では、ネクストホップ インデックス 0x5f7 は hg4 を指し、0x9b3 は hg6 を指し、0x5f8 は hg7 を指します。

ユニキャスト カプセル化解除パスでドロップされたパケット

方向にアクセスするために、ネットワーク内のデバイスでユニキャストパケットがドロップされる場合は、次の手順に従います。

手順の概要

1. パケットがスーパーバイザに送信されたかどうか、およびリモート VXLAN トンネル エンドポイント（VTEP）の検出が行われたかどうかを確認します。
2. ハードウェアに mpls_entry が存在する場合は、vlan_xlate テーブルを確認します。
3. ユニキャスト DIP エントリが vlan_xlate テーブルに存在するかどうかを確認します。
4. ユニキャスト DIP エントリが vlan_xlate テーブルに存在するかどうかを確認します。
5. 宛先 MAC アドレスがレイヤ 2 MAC アドレス テーブルに表示されていることを確認します。

手順の詳細

手順

ステップ 1 パケットがスーパーバイザに送信されたかどうか、およびリモート VXLAN トンネル エンドポイント（VTEP）の検出が行われたかどうかを確認します。

a) リモート ピアがソフトウェアで学習されたかどうかを確認します。

例：

```
switch# show nve peers
Interface          Peer-IP           VNI        Up Time
-----------------  ----------------  ---------  -----------
nve1               100.100.100.5     10000      00:06:54
```

b) mpls_entry テーブルを確認して、リモート ピアがハードウェアで学習されたかどうかを確認します。

例：

```
switch# bcm-shell module 1
bcm-shell.0> d chg mpls_entry | grep SVP
MPLS_ENTRY.ipipe0[12368]:
<VXLAN_SIP:SVP=0x1751,VXLAN_SIP:SIP=0x66666666,VXLAN_SIP:KEY=0x666666668,VXLAN_SIP:HASH_LSB=0x666
VXLAN_SIP:DATA=0
x1751,VALID=1,KEY_TYPE=8,>
```

c) mpls_entry がなく、送信元仮想ポート（SVP）がない場合は、パケットがスーパーバイザに送信されているかどうかを確認し、IPFIB エラーがないかどうかを確認します。

例：

```
bcm-shell.0> show c cpu0
bcm-shell.0> exit
switch# attach module 1
module-1# show system internal ipfib errors
```

ステップ 2 ハードウェアに mpls_entry が存在する場合は、vlan_xlate テーブルを確認します。

例：

```
module-1# exit
switch# bcm-shell module 1
bcm-shell.0> d chg vlan_xlate | grep 0xe1000003
VLAN_XLATE.ipipe0[8464]:
<VXLAN_DIP:KEY=0x7080000192,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=3,VXLAN_DIP:DIP=0xe1000003,
XLAN_DIP
:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

vlan_xlate テーブルには、パケットのマルチキャスト宛先 IP アドレス（DIP）のエントリが 1 つ必要です。この例では、マルチキャストパケットが 225.0.0.3 に送信される場合を示しています。

ステップ 3 ユニキャスト DIP エントリが vlan_xlate テーブルに存在するかどうかを確認します。

例：

```
bcm-shell.0> d chg vlan_xlate | grep 0x65656565
VLAN_XLATE.ipipe0[14152]:
<VXLAN_DIP:KEY=0x32b2b2b292,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=0x565
VXLAN_DIP:DIP=0x65656565,VXLAN_DIP:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

エントリが存在する場合は、カプセル化が解除されます。

ステップ 4 ユニキャスト DIP エントリが vlan_xlate テーブルに存在するかどうかを確認します。

例：

```
bcm-shell.0> d chg vlan_xlate | grep 0x65656565
VLAN_XLATE.ipipe0[14152]:
<VXLAN_DIP:KEY=0x32b2b2b292,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=0x565
VXLAN_DIP:DIP=0x65656565,VXLAN_DIP:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

エントリが存在する場合は、カプセル化が解除されます。

ステップ 5 宛先 MAC アドレスがレイヤ 2 MAC アドレス テーブルに表示されていることを確認します。

例：

```
bcm-shell.0> l2 show
mac=00:00:bb:01:00:03 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:cc:01:00:0a vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:05 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:aa:01:00:0a vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:07 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:cc:01:00:01 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:08 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:aa:01:00:01 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:cc:01:00:07 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:cc:01:00:02 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:04 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:cc:01:00:04 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:02 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:cc:01:00:09 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:09 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:06 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:cc:01:00:06 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:06 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:09 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:bb:01:00:04 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:bb:01:00:02 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:aa:01:00:08 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:07 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:cc:01:00:08 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:01 vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:cc:01:00:05 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:03 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:bb:01:00:0a vlan=28772 GPORT=0x80000215Unknown GPORT format
mac=00:00:cc:01:00:03 vlan=28772 GPORT=0x80003401Unknown GPORT format
mac=00:00:aa:01:00:05 vlan=28772 GPORT=0x80003401Unknown GPORT format
```

宛先 MAC アドレスが存在する場合、レイヤ 2 転送が発生します。それ以外の場合、パケットはカプセル化解除フラッディングリストを使用してフラッディングされます。

Broadcom シェル テーブルについて

このセクションでは、VXLAN に関する Broadcom シェル テーブルについて説明します。

MPLS エントリ テーブル

MPLS エントリ（mpls_entry）テーブルには、次の情報が含まれます。

- リモート VTEP（SIP）の IP アドレス
- トンネルカプセル化ポート（SVP）
- VLAN と VNID（VFI、VN_ID）間のマッピング

SIPエントリがmpls_entryテーブルにない場合、パケットはVTEP学習のためにスーパーバイザに送信されます。エントリがハードウェアにインストールされると、パケットはスーパーバイザに送信されなくなります。

（注） 一部のパケットは、ソフトウェア転送が VXLAN パケットに対して実行されないため、学習フェーズ中にドロップされます。

（注） スーパーバイザに送信されるパケットは、class-default CPU キューを使用します。現在、VxLAN 専用の COPP クラスはありません。

次の例は、リモート VTEP IP アドレスが 100.100.100.1 で、VLAN 100 が VNID 10000 にマッピングされるテーブルを示しています。

```
bcm-shell.0> d chg mpls_entry
Private image version: R
MPLS_ENTRY.ipipe0[6816]:
<VXLAN_SIP:SVP=8,VXLAN_SIP:SIP=0x64646401,VXLAN_SIP:KEY=0x646464018
VXLAN_SIP:HASH_LSB=0x401,VXLAN_SIP:DATA=8,VALID=1,KEY_TYPE=8,>
MPLS_ENTRY.ipipe0[8680]:
<VXLAN_VN_ID:VN_ID=0x2710,VXLAN_VN_ID:VFI=0x64,VXLAN_VN_ID:KEY=0x27109
VXLAN_VN_ID:HASH_LSB=0x710,VXLAN_VN_ID:DATA=0x64,VALID=1,KEY_TYPE=9,>
```

出力では、VLAN-VNIDマッピングごとに1つのエントリが検索されます。この例では、VN_ID = 0x2710 は 16 進表記の VNID、VFI = 0x64 は 16 進表記のマッピング VLAN、0x64 = 100 は 0x2710 VNID 10000 にマッピングされます。

MAC アドレス ラーニング

VXLAN VLAN で学習された MAC アドレスは、内部変換 VLAN で学習されたものとして表示されます（たとえば、VLAN 100 は VLAN 28772 として表示されます）。

GPORTは、MACアドレスが学習されたポートまたは仮想ポートを参照します。ローカルMACアドレスの場合、GPORT # と前面パネルの port # の間にマッピングがあります。リモート MAC アドレスは、トンネル ポートを指している SVP に対して学習する必要があります。

このテーブルのミスは、VLAN のローカル ポートおよびトンネル ポートにパケットをフラッディングすることを意味します。このテーブルのヒットは、パケットを対応する GPORT に転送することを意味します。GPORT がトンネル ポートの場合は、パケットを VXLAN にカプセル化する必要があります。GPORT がローカル ポートの場合、通常のレイヤ 2 学習 MAC アドレス転送が発生します。

（注） GPORT と前面パネルのポート番号の間のマッピングを取得するには、GPORTと前面パネルのポート番号マッピングの取得 （15 ページ） セクションを参照してください。

入力 DVP テーブル

入力 DVP テーブルは、仮想ポートをネクストホップ インデックスにマッピングします。これはユニキャスト カプセル化パスで使用され、仮想ポートによってインデックスが作成されます。ECMP の場合は、ECMP = 1 フィールドが必要です。

次の例は、VP 0x1751 のネクストホップ インデックスが0x35であることを示しています。

```
bcm-shell.0> d chg ing_dvp_table 0x1751
Private image version: R
ING_DVP_TABLE.ipipe0[5969]:
<VP_TYPE=3,NEXT_HOP_INDEX=0x35,NETWORK_PORT=1,ECMP_PTR=0x35,DVP_GROUP_PTR=0x35,>
```

入力レイヤ 3 ネクスト ホップ

入力レイヤ 3 ネクスト ホップは、特定のネクストホップ インデックスのポート番号を示します。ユニキャスト カプセル化パスで使用されます。phy_info を使用すれば、ポート番号と実際の前面パネルのポート番号の間のマッピングを取得できます。

```
bcm-shell.0> d chg ing_l3_next_hop
ING_L3_NEXT_HOP.ipipe0[16]:
<VLAN_ID=0xfff,TGID=0x9f,PORT_NUM=0x1f,MTU_SIZE=0x3fff,MODULE_ID=1,L3_OIF=0x1fff,ENTRY_TYPE=2
ENTRY_INFO_UPPER=3,DVP_RES_INFO=0x7f,>
```

VLAN 変換テーブル

VLAN 変換テーブルは、VXLAN マルチキャストとユニキャストの両方のカプセル化解除パスで使用されます。次の 3 種類のエントリが含まれます。

- 外部マルチキャスト グループごとに 1 つのエントリ（マルチキャスト DIP）
- ローカル VTEP（ユニキャスト DIP）の 1 つのエントリ
- ポートごとに VLAN ごとに 1 つのエントリ

次の例は、マルチキャスト DIP エントリを示しています。

```
bcm-shell.0> d chg vlan_xlate | grep 0xe1000003
VLAN_XLATE.ipipe0[8464]:
<VXLAN_DIP:KEY=0x7080000192,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=3
VXLAN_DIP:DIP=0xe1000003,VXLAN_DIP:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

次の例は、ユニキャスト DIP エントリを示しています。

```
bcm-shell.0> d chg vlan_xlate | grep 0x65656565
VLAN_XLATE.ipipe0[14152]:
<VXLAN_DIP:KEY=0x32b2b2b292,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=0x565
VXLAN_DIP:DIP=0x65656565,VXLAN_DIP:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

次の例は、VLAN ごと、ポートごとに 1 つのエントリを示しています。

```
bcm-shell.0> d chg vlan_xlate | grep VLAN_ID=3
VLAN_XLATE.ipipe0[3216]:
<XLATE:VLAN_ID=3,XLATE:TGID=0xa0,XLATE:SVP_VALID=1,XLATE:SOURCE_VP=0x201,XLATE:SOURCE_FIELD=0xa0
XLATE:PORT_NUM=0x20,XLATE:OVID=3,XLATE:OTAG=3,XLATE:OLD_VLAN_ID=3,XLATE:MPLS_ACTION=1
```

```
XLATE:MODULE_ID=1,XLATE:KEY=0x1805024,XLATE:ITAG=3,XLATE:INCOMING_VIDS=3,XLATE:HASH_LSB=3
XLATE:GLP=0xa0,XLATE:DISABLE_VLAN_CHECKS=1,XLATE:DATA=0x100a000000000000000001,VLAN_ID=3
VALID=1,TGID=0xa0,SVP_VALID=1,SOURCE_VP=0x201,SOURCE_TYPE=1,SOURCE_FIELD=0xa0,PORT_NUM=0x20,OVID=3
OTAG=3,OLD_VLAN_ID=3,MPLS_ACTION=1,MODULE_ID=1,KEY_TYPE=4,KEY=0x1805024,ITAG=3,INCOMING_VIDS=3
HASH_LSB=3,GLP=0xa0,DISABLE_VLAN_CHECKS=1,DATA=0x100a00000000000000001>
```

EGR ポートから NHI へのマッピング

EGR ポートから NHI へのマッピングは、ネクストホップ インデックスを出力ポートにマッピングします。ユニキャスト カプセル化パスで使用されます。

```
bcm-shell.0> g chg egr_port_to_nhi_mapping
EGR_PORT_TO_NHI_MAPPING.hg7[2][0x4001808]=0x36: <NEXT_HOP_INDEX=0x36>
```

VLAN フラッド インデックス テーブル

VLAN フラッドインデックス（VFI）テーブルには、特定の VLAN または VFI の BC/UUC/UMC インデックスが表示されます。**mcshow** コマンドの出力でフラッディング インデックスを使用して、トンネル カプセル化ポートを含む VLAN のメンバーを検索できます。

次の例は、ポート番号を取得する例を示しています。

```
bcm-shell.0> d chg vfi 3
Private image version: R
VFI.ipipe0[3]:
<VP_1=0xc01,VP_0=0x1803,UUC_INDEX=0x1803,UMC_INDEX=0x1803,RSVD_VP_0=1,BC_INDEX=0x1803>
```

次の例は、このポート番号を phy_info に入力して、前面パネルのポート番号を取得する方法を示しています。

```
bcm-shell.0> d chg ing_l3_next_hop
ING_L3_NEXT_HOP.ipipe0[16]:
<VLAN_ID=0xfff,TGID=0x9f,PORT_NUM=0x1f,MTU_SIZE=0x3fff,MODULE_ID=1,L3_OIF=0x1fff,ENTRY_TYPE=2
ENTRY_INFO_UPPER=3,DVP_RES_INFO=0x7f,

bcm-shell.0> phy info
Phy mapping dump:
      port   id0   id1  addr iaddr                  name    timeout
  hg0(  1)  600d  8770   1b1   1b1            TSC-A0/31/4     250000
  hg1(  2)  600d  8770    81    81            TSC-A0/00/4     250000
  hg2(  3)  600d  8770   1ad   1ad            TSC-A0/30/4     250000
  hg3(  4)  600d  8770    85    85            TSC-A0/01/4     250000
  hg4(  5)  600d  8770   1a9   1a9            TSC-A0/29/4     250000
  hg5(  6)  600d  8770    89    89            TSC-A0/02/4     250000
  hg6(  7)  600d  8770   195   195            TSC-A0/26/4     250000
  hg7(  8)  600d  8770    a1    a1            TSC-A0/05/4     250000
  hg8(  9)  600d  8770   191   191            TSC-A0/25/4     250000
```

次の例は、カプセル化解除ルートを示しています。

```
bcm-shell.0> d chg vlan_xlate
Private image version: R
VLAN_XLATE.ipipe0[768]:
<VXLAN_DIP:NETWORK_RECEIVERS_PRESENT=1,VXLAN_DIP:KEY=0x7080000092,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1
VXLAN_DIP:HASH_LSB=1,VXLAN_DIP:DIP=0xe1000001,VXLAN_DIP:DATA=0x400001,VALID=1,KEY_TYPE=0x12,>
VLAN_XLATE.ipipe0[1472]:
```

```
<VXLAN_DIP:KEY=0x3232320112,VXLAN_DIP:IGNORE_UDP_CHECKSUM=1,VXLAN_DIP:HASH_LSB=0x402
VXLAN_DIP:DIP=0x64646402,VXLAN_DIP:DATA=0x400000,VALID=1,KEY_TYPE=0x12,>
```

（注） NETWORK_RECEIVERS_PRESENT は 0 に設定する必要があります。

GPORTと前面パネルのポート番号マッピングの取得

次の手順に従って、GPORT から前面パネルのポート番号へのマッピングを取得します。

手順の概要

1. GPORT # からローカル ターゲット ロジック（LTL）を取得するには、次の式を使用します：LTL # = 0x10000 - 512 + GPORT #
2. 対象とする LTL の ifindex を取得します。
3. 前面パネル ポートの ifindex を取得します。
4. GPORT から前面パネル ポート番号へのマッピングを表示します。

手順の詳細

手順

ステップ 1 GPORT # からローカルターゲットロジック（LTL）を取得するには、次の式を使用します：LTL # = 0x10000 - 512 + GPORT #

GPORT が 0x201 の場合、LTL は 0x10000 + 0x201 (513) - 0x200 (512) = 0x10001です。

ステップ 2 対象とする LTL の ifindex を取得します。

例：

```
switch# attach module 1
module-1# show system internal pixmc info sdb ltl 0x10001
```

ステップ 3 前面パネル ポートの ifindex を取得します。

例：

```
module-1# exit
switch# show int snmp-ifindex | grep 0x1a002e00
Eth1/24    436219392  (0x1a002e00)
```

ステップ 4 GPORT から前面パネル ポート番号へのマッピングを表示します。

例：

```
switch# bcm-shell module 1
bcm-shell.0> l2 show
```

```
mac=00:00:00:00:00:00 vlan=0 GPORT=0xc000000 Trunk=0^M
mac=00:00:bb:01:00:03 vlan=28772 GPORT=0x80001751Unknown GPORT format ^M
mac=00:00:cc:01:00:0a vlan=28772 GPORT=0x80000201Unknown GPORT format ^M
mac=00:00:bb:01:00:05 vlan=28772 GPORT=0x80001751Unknown GPORT format ^M
mac=00:00:aa:01:00:0a vlan=28772 GPORT=0x80000202Unknown GPORT format ^M
```

この例では、MAC アドレス 00:00:bb:01:00:05 はトンネルを通して学習されるので、GPORT 0x1751 はトンネル SVP に対応します。MAC アドレス 00:00:aa:01:00:0a はローカルに学習されるので、GPORT 0x202 は前面パネル ポートに対応します。

入力ポートのためにどのインターフェイス トラフィックが使用されるかを特定する

次に、特定の出力ポートでトラフィックが使用するインターフェイスを検索する例を示します。

```
switch# show system internal ethpm info interface ethernet 2/3 | grep ns_pid
  IF_STATIC_INFO:
port_name=Ethernet2/3,if_index:0x1a006400,ltl=2543,slot=0,nxos_port=50,dmod=1,dpid=9,unit=0
queue=2064,xbar_unitbmp=0x0
ns_pid=8

- dpid=9 is higig8

switch# bcm-shell module 1
bcm-shell.0> g chg egr_port_to_nhi_mapping
EGR_PORT_TO_NHI_MAPPING.hg7[2][0x4001808]=0x36: <NEXT_HOP_INDEX=0x36>
bcm-shell.0> d chg egr_l3_next_hop 0x36
Private image version: R
EGR_L3_NEXT_HOP.epipe0[54]:
<OVID=0x65,MAC_ADDRESS=0x60735cde6e41,L3MC:VNTAG_P=1,L3MC:VNTAG_FORCE_L=1,L3MC:VNTAG_DST_VIF=0x18
L3MC:RSVD_DVP=1,L3MC:INTF_NUM=0x1065,L3MC:FLEX_CTR_POOL_NUMBER=3,L3MC:FLEX_CTR_OFFSET_MODE=3
L3MC:FLEX_CTR_BASE_COUNTER_IDX=0xe41,L3MC:ETAG_PCP_DE_SOURCE=3,L3MC:ETAG_PCP=1
L3MC:ETAG_DOT1P_MAPPING_PTR=1,L3MC:DVP=0x2b9b,L3:OVID=0x65,L3:MAC_ADDRESS=0x60735cde6e41
L3:IVID=0xc83,L3:INTF_NUM=0x1065,IVID=0xc83,INTF_NUM=0x1065,>
```

VLAN のフラッド リストの検索

次に、特定の VLAN のフラッド リストを検索する例を示します。

```
bcm-shell.0> d chg vfi 3
Private image version: R
VFI.ipipe0[3]:
<VP_1=0xc01,VP_0=0x1803,UUC_INDEX=0x1803,UMC_INDEX=0x1803,RSVD_VP_0=1,BC_INDEX=0x1803>
```

カプセル化ポートがフラッドリストの一部であるかどうかの判別

次に、ネットワーク方向へのアクセスにおいて、カプセル化ポートがフラッドリストの一部であるかどうかを確認する例を示します。

```
bcm-shell.0> mc show 0x1803
Group 0xc001803 (VXLAN)
        port hg7, encap id 400053
        port xe23, encap id 400057
```

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。