Catalyst 6500 S2T CEFエントリについて

内容

概要

このドキュメントでは、Supervisor Sup2Tを搭載したCisco Catalyst 6500が、パケット転送を実現するために使用されるラインカードハードウェアでCisco IOSソフトウェアに設定された(Cisco Express Forwarding)CEFエントリをプログラムする方法について説明します。

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。

- Cisco Express Forwarding（CEF）
- Cisco Catalyst 6500 シリーズ スイッチ
- Cisco Distributed Forwarding Card(DFC)

使用するコンポーネント

このドキュメントの情報は、次のハードウェアとソフトウェアのバージョンに基づいています。

- Cisco Catalyst 6500 WS-X6848-GE-TX（DFC4搭載）ラインカード
- IOSバージョン15.2.1SY5のSupervisor 2T搭載Cisco Catalyst 6500

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。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

レイヤ3スイッチングメカニズムとしてのCEFは、ほとんどのCiscoマルチレイヤスイッチで使用されています。ネットワークの停止、パケット損失、またはパケット遅延のシナリオを日常的にトラブルシューティングするには、CEFの動作を理解する必要があります。

スタンドアロンモードのSup2Tスーパーバイザ、または現在VSSが多くの企業ネットワークによってコアスイッチとして導入されているため、実質的に他

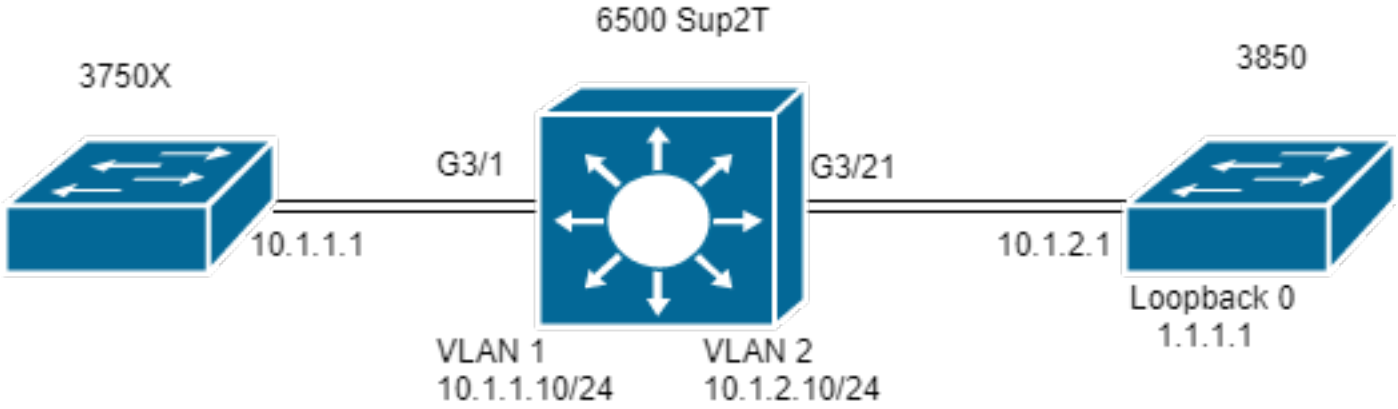
のすべてのルーティングデバイスまたはスイッチングデバイスを集約します。これは、が宛先にパケットを正常に配信するために、ドメイン内およびドメイン間のトラフィックのほとんどを転送することを意味します。これを実現するには、Sup2Tに、静的またはルーティングプロトコルを介して動的に学習された適切なルーティング情報が必要です。

モジュラシャーシでは、スーパーバイザの他に複数のフォワーディングエンジンが存在する場合があります。特定のラインカード（特にC6800-32P10Gなど）には、パケット交換のパフォーマンスを強化するために独自のフォワーディングエンジンが搭載されており、CEFエントリのルックアップがローカルで実行され、異なるラインカードに着信するトラフィックにに最適です。

ソフトウェアの不具合状態、リソースの枯渇、高CPU状態など、すべてのフォワーディングエンジンで共有されるこれらのCEFエントリがHWに割り当てられない場合があり、スイッチにすべてのエントリを更新する十分な時間が存在しないことがあります。

ネットワーク図

ネットワーク
：




```
Switch#show module 3
---------------------- ----------------------------- Mod Ports Card Type Model Serial No. --- --
--- ---------------------------------------- ----------------- ----------- 3 48 CEF720 48 port
10/100/1000mb Ethernet WS-X6848-GE-TX SAL2003X5AH ---- --------------------------- -------------
----- ----------- ------- ------- 3 Distributed Forwarding Card WS-F6K-DFC4-A SAL2003X5AH 1.4 Ok
```

分散フォワーディングエンジンでのCEFエントリの特定

この図では、スタンドアロンの6506スイッチにSupervisor 2Tと、スロット3にDFCを搭載したラインカードWS-6848-GE-TXが取り付けられています。ポートG3/1を介してラインカードに接続されているホスト333377777555550

このために、3750Xには、ネクストホップ10.1.1.10を経由するIPアドレス1.1.1.1へのスタティックルートがあります。これは、Sup2TスイッチのVLAN 1のSVIです。Sup2Tスイッチは、VLAN 2のSup2Tに接続された3850インターフェイスであるネクストホップ10.1.2.1を介して、IP 1.1.1.1/32のスタティックルートエントリに基づいて、このトラフィックを3850スイッチにルーティングする必要があります。

```
MXC.CALO.3750X#show ip route | inc 1.1.1.1
S 1.1.1.1 [1/0] via 10.1.1.10
```

```
MXC.CALO.Sup2T#show ip route | inc 1.1.1.1
S 1.1.1.1 [1/0] via 10.1.2.1
```

```
CALO.MXC.3850#show ip route | inc 1.1.1.1
C 1.1.1.1 is directly connected, Loopback1
```

わかりやすくするために、3750Xと3850スイッチの両方が同じラインカードを介して6500に接続されていることに注意してください。これは、トラフィックがローカルに検索され、ローカルにも転送されることを意味します。

Gi3/1を介してSup2Tスイッチに入力されたパケットは、最終的にフォワーディングエンジンに到達します（これはDFCであるため）。フォワーディングエンジンは、このパケット内の宛先IPアドレスフィールドを解析し、プログラムされたCEFエントリをルックアップしてベストマッチ（最長マスク）を行います。

これはDFCカードであるため、独自のCEFエントリがあることを意味し、それを確認するために、**attach [dec]コマンドまたはattach switch [1-2] mod [dec]コマンドを使用してラインカードに接続する必要があります**。

これで、DFCプロンプトで、コマンド**show platform hardware cefまたはshow platform hardware cef** または**show platform hardware cef vpn 0**が、一般的なルーティングテーブル（VPN 0/VRFなし）用にプログラムされたすべてのCEFエントリを返すようになります。

目的はプレフィックス1.1.1.1/32であるため、コマンド**show platform hardware cef vpn 0 lookup 1.1.1.1**を使用します。このコマンドは、プレフィックス1.1.1.1と、実際にトラフィックを転送するために使用される最適な一致を返します。

```
MXC.CALO.Sup2T#attach 3
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session

MXC.CALO.Sup2T-dfc3#show platform hardware cef vpn 0
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
32 0.0.0.0/32 receive
33 255.255.255.255/32 receive
34 10.1.85.254/32 glean
35 10.1.85.5/32 receive
36 10.1.86.5/32 receive
[snip...]

MXC.CALO.Sup2T-dfc3#show platform hardware cef vpn 0 lookup 1.1.1.1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
262 1.1.1.1/32 Vl2 ,0c11.678b.f6f7
```

CEFエントリが存在します。これは、コマンド**ip route 1.1.1.1 255.255.255.255 10.1.2.1を介してIOSソフトウェアでプログラムされたスタティックエントリの結果としてプログラムされました。**

また、このエントリがヒットし、トラフィックが隣接関係エントリを返す**show platform hardware cef 1.1.1.1 detailコマンドを介して、このエントリで転送されるこ**とを確認できます。

```
MXC.CALO.Sup2T-dfc3#show platform hardware cef 1.1.1.1 detail
Codes: M - mask entry, V - value entry, A - adjacency index, NR- no_route bit
LS - load sharing count, RI - router_ip bit, DF: default bit
CP - copy_to_cpu bit, AS: dest_AS_number, DGTv - dgt_valid bit
DGT: dgt/others value

Format:IPV4 (valid class vpn prefix)
M(262 ): 1 F 2FFF 255.255.255.255
V(262 ): 1 0 0 1.1.1.1
(A:114689, LS:0, NR:0, RI:0, DF:0 CP:0 DGTv:1, DGT:0)
```

最後に、隣接関係エントリは、パケットの書き換え方法と、この隣接関係エントリによってトラフィックが書き換えられるかどうかを示します。

```
MXC.CALO.Sup2T-dfc3#show platform hardware cef adjacencies entry 114689 detail

RIT fields: The entry has a Layer2 Format
_____________________________________________________
|decr_ttl = YES | pipe_ttl = 0 | utos = 0
|_______________|_______________|_____________________
|l2_fwd = 0 | rmac = 0 | ccc = L3_REWRITE
|_______________|_______________|_____________________
|rm_null_lbl = YES| rm_last_lbl = YES| pv = 0
|_______________|_______________|_____________________
|add_shim_hdr= NO | rec_findex = N/A | rec_shim_op = N/A
|_______________|_______________|_____________________
|rec_dti_type = N/A | rec_data = N/A
|_______________________________|_____________________
|modify_smac = YES| modify_dmac = YES| egress_mcast = NO
|_______________________________|_____________________
```

```
|ip_to_mac = NO
|_____________________________________________________
|dest_mac = 0c11.678b.f6f7 | src_mac = d8b1.902c.9680
|_____________________________|_____________________
|
Statistics: Packets = 642
Bytes = 75756                    <<<<
```

dest_macとsrc_macは、主に対象とする値で、このパケットに書き込まれる新しいL2ヘッダーを示します。宛先MACアドレス0c11.678b.f6f7は、3850(1.1.1.1に到達するためのネクストホップ)である10.1.2.1です。

```
MXC.CALO.Sup2T#show ip arp 10.1.2.1
Protocol Address Age (min) Hardware Addr Type Interface
Internet 10.1.2.1 30 0c11.678b.f6f7 ARPA Vlan2
```

また、[Statistics] フィールドには、トラフィックがこの隣接関係エントリに実際にヒットし、それに応じてL2ヘッダーが書き換えられることが示されます。

CEFエントリの削除

CEFエントリの削除は、誤ってプログラムされた可能性のあるエントリ（誤った隣接関係エントリなど）や、トレーニング目的で削除するのに役立ちます。また、ルーティングパスを変更する方法も提供します。

CEFエントリを削除するには、CEFエントリが順番にプログラムされ、ハードウェアインデックスが割り当てられていることを理解する必要があります。次に例を示します。

MXC.CALO.Sup2T-dfc3#show platform hardware cef vpn 0

Codes:decap - decapsulation, + – プッシュラベル

```
MXC.CALO.Sup2T-dfc3#show platform hardware cef vpn 0
...
Index Prefix Adjacency 259 10.1.2.255/32 receive 260 10.1.1.1/32 Vl1 ,a0ec.f930.3f40 261
10.1.2.1/32 Vl2 ,0c11.678b.f6f7 262 1.1.1.1/32 Vl2 ,0c11.678b.f6f7    <<<< Our CEF entry of
interest has a HW index of 262.
...
```

このハードウェアインデックスは、CEFエントリを参照として使用するため、CEFエントリを削除する際に最も重要な要素です。ただし、変更を行うには、ソフトウェアハンドルに変換する必要があります。**test platform hardware cef index-conv hw_to_sw [hw index]**コマンドを使用して、これを実現できます

```
MXC.CALO.Sup2T-dfc3#test platform hardware cef index-conv hw_to_sw 262
hw index: 262 ----> sw handle: 101
```

ソフトウェアハンドルがわかったので、**test platform hardware cef v4-delete [sw handle] mask [mask length] vpn [dec]コマンドを使用して、CEFエントリの削除を続行できます**

```
MXC.CALO.s2TVSS-sw2-dfc3#test platform hardware cef v4-delete 101 mask 32 vpn 0
test_ipv4_delete: done.
```

> **注：** これはホスト固有のルート(1.1.1.1/32)であるため、マスク長の値は32です

ここで、CEFエントリが削除されます。

```
MXC.CALO.Sup2T-dfc3#show platform hard cef vpn 0 1.1.1.1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
```

```
MXC.CALO.Sup2T-dfc3#show platform hard cef vpn 0
```

```
[snip...]
259 10.1.2.255/32 receive
260 10.1.1.1/32 Vl1 ,a0ec.f930.3f40
261 10.1.2.1/32 Vl2 ,0c11.678b.f6f7
288 224.0.0.0/24 receive              <<<<<<< Index 262 no longer exists in the CEF entries.
289 10.1.85.0/24 glean
```

test platform hardware cef vpn 0コマンドがDFCプロンプトの下で実行されていることに注意してください。このように、CEFエントリがスーパーバイザから削除されるのではなく、DFCのCEFテーブルから削除されました。どのフォワーディングエンジンからエントリが削除されるかに注意する必要があります。

トラフィックの変更は、可視性がないリスクがあります（ラボテストの場合）。これは、別のCEFエントリのヒットが原因である可能性があります。常に最も正確なマスク（最長マスク）に一致することを考慮してください。 この実習では、次の項目を確認します。

```
MXC.CALO.Sup2T-dfc3#show plat hard cef vpn 0 lookup 1.1.1.1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
262048 0.0.0.0/0 glean
```

このエントリはパケットに対して実際に何を行うのでしょうか。

```
MXC.CALO.Sup2T-dfc3#show platform hardware cef adjacencies entry 262048
RIT fields: The entry has a Recirc. Format
_____________________________________________________________ |decr_ttl=NO | l2_fwd=NO | ccc = 6 |
add_shim_hdr = YES |_____________|_____________|_________|_____________________ |rc_fidx=0 |
rc_shimop=1 | rc_dti_type=4 | rc_data = 0x10B
|_____________|_____________|_________________|_________________ Statistics: Packets = 2163 Bytes =
255234
```

Taken from a CPU packet capture using Catlayst 6500 NETDR tool. For NETDR capture tool details refer to: [Catalyst 6500 Series Switches Netdr Tool for CPU-Bound Packet Captures]

```
------- dump of incoming inband packet -------
l2idb Po1, l3idb Vl1, routine inband_process_rx_packet, timestamp 01:00:17.841
dbus info: src_vlan 0x1(1), src_indx 0xB40(2880), len 0x82(130)
bpdu 0, index_dir 0, flood 0, dont_lrn 0, dest_indx 0x5FA4(24484), CoS 0
cap1 0, cap2 0
78020800 00018400 0B400100 82000000 1E000464 2E000004 00000010 5FA45BDD
destmac D8.B1.90.2C.96.80, srcmac A0.EC.F9.30.3F.40, shim ethertype CCF0
earl 8 shim header IS present:
version 0, control 64(0x40), lif 1(0x1), mark_enable 1,
feature_index 0, group_id 0(0x0), acos 0(0x0),
ttl 14, dti 4, dti_value 267(0x10B)
10000028 00038080 010B
ethertype 0800
protocol ip: version 0x04, hlen 0x05, tos 0x00, totlen 100, identifier 51573
df 0, mf 0, fo 0, ttl 255, src 10.1.1.1, dst 1.1.1.1
icmp type 8, code 0


------- dump of outgoing inband packet -------
l2idb NULL, l3idb Vl2, routine etsec_tx_pak, timestamp 01:03:56.989
dbus info: src_vlan 0x2(2), src_indx 0x380(896), len 0x82(130)
bpdu 0, index_dir 0, flood 0, dont_lrn 0, dest_indx 0x0(0), CoS 0
cap1 0, cap2 0
00020000 0002A800 03800000 82000000 00000000 00000000 00000000 00000000
destmac 0C.11.67.8B.F6.F7, srcmac D8.B1.90.2C.96.80, shim ethertype CCF0
earl 8 shim header IS present:
version 0, control 0(0x0), lif 16391(0x4007), mark_enable 0,
feature_index 0, group_id 0(0x0), acos 0(0x0),
ttl 15, dti 0, dti_value 540674(0x84002)
000800E0 0003C008 4002
```

```
ethertype 0800
protocol ip: version 0x04, hlen 0x05, tos 0x00, totlen 100, identifier 50407
df 0, mf 0, fo 0, ttl 254, src 10.1.1.1, dst 1.1.1.1
icmp type 8, code 0
```

これで、ラインカード3を介して入力される宛先1.1.1.1のすべてのトラフィックがSHIMヘッダーに再循環され、CPUにパントされます。時には、このCEFエントリの代わりに、ドロップ隣接関係を持つ別の0.0.0.0/0が表示さ**れ、**まったく同じことを行うことがあります。

> **注：**削除されるCEFエントリを評価します。CPU使用率が高くなる原因は、次のとおりです。通常、デフォルトルート0.0.0.0/0が設定され、それに基づいてトラフィックが転送されます（パケット損失が発生します）。

CEFエントリの追加

CEFエントリが追加されると、ほとんどの場合、パケット損失、パケット遅延、または高いCPU使用率を引き起こすプログラミング上の誤った問題が解決されます。CEFエントリをハードウェアにインストールする方法に関する知識は、誤ってプログラムされたエントリを修正するだけでなく、 パケットの再循環、完全に異なるインターフェイスまたはネクストホップへのポイント、ルーテッドパケットの望ましい書き換え、ドロップなどこれらはすべて、ボックスをリロードせずに、設定を削除して設定するか、明確な変更を行います。CEFエントリ 追加は、コンフィギュレーションモードに入らなくても実行できます。（前のセクションで説明したCEFエントリの削除手順でも行ったように）。

基本的に、ネクストホップへの有効なARPエントリがある場合、この例では10.1.2.1であり、何らかの理由でエントリがない場合の2つの状況があります。2番目の状況では、（スタティックARPを使用して）有効なARPエントリを実際に作成するように強制されます。

ステップ1:1.1.1.1のネクストホップである10.1.2.1のスイッチにARPエントリがあります。

```
MXC.CALO.Sup2T#show ip arp 10.1.2.1
Protocol Address Age (min) Hardware Addr Type Interface
Internet 10.1.2.1 2 0c11.678b.f6f7 ARPA Vlan2

MXC.CALO.Sup2T#show ip route | inc 1.1.1.1
S 1.1.1.1 [1/0] via 10.1.2.1
```

ARPエントリは、CEFテーブルのホストルート(/32)としてプログラムされます。

```
MXC.CALO.Sup2T-dfc3#show plat hard cef vpn 0 look 10.1.2.1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
53 10.1.2.1/32 Vl2 ,0c11.678b.f6f7
```

And of course, there is an index for this which again will tell us how a packet should be rewritten to reach 10.1.2.1:

```
MXC.CALO.Sup2T-sw2-dfc3#show plat hard cef vpn 0 10.1.2.1 detail
[snip...]
Format:IPV4 (valid class vpn prefix)
M(53 ): 1 F 2FFF 255.255.255.255
V(53 ): 1 0 0 10.1.2.1
(A:114689, LS:0, NR:0, RI:0, DF:0 CP:0 DGTv:1, DGT:0)
```

Wait, wasn't 114689 adj entry the same used for 1.1.1.1?:

```
MXC.CALO.Sup2T-sw2-dfc3#show plat hard cef 1.1.1.1 de
[snip...]
Format:IPV4 (valid class vpn prefix)
M(54 ): 1 F 2FFF 255.255.255.255
V(54 ): 1 0 0 1.1.1.1
(A:114689, LS:0, NR:0, RI:0, DF:0 CP:0 DGTv:1, DGT:0)
```

同じデータリンクのネクストホップを持つ宛先IPアドレスを持つパケットは、同じインターフェイスを経由して転送し、同じL2ヘッダーで書き換える必要があります。

これは最初はかなり明白に思えますが、実際にはCEFエントリを追加することが最も重要な要素ですが、特定のCEF隣接関係エントリを使用してパケットを書き換える方法を指定する必要があります。

ステップ2：ここで、ARPエントリが自動的に作成されないとすると、スタティックARPエントリを作成する必要があります。

このため、プレフィックス10.1.2.1のネクストホップとして使用されるデバイスのMACアドレスを知る必要があります。そのため、0c11.678b.f6f7に送信されます。すでに**show mac address-table address 0c11.678b.f6f7出力**これは問題ありませんが、スタティックMACエントリを作成する必要があります。

```
MXC.CALO.Sup2T(config)#mac address-table static 0c11.678b.f6f7 vlan 2 int Gi3/21
Displaying entries from DFC switch [2] linecard [3]:

vlan mac address type learn age ports
----+----+---------------+-------+-----+----------+-----------------------------
2 0c11.678b.f6f7 static No - Gi3/21
```

ステップ3：最後に、CEFエントリをプログラムするには、スタティックARPエントリを作成する必要があります。

```
MXC.CALO.Sup2T(config)#arp 10.1.2.1 0c11.678b.f6f7 arpa    <<< Static ARP configuration

MXC.CALO.Sup2T#show ip arp 10.1.2.1
Protocol Address Age (min) Hardware Addr Type Interface
Internet 10.1.2.1 - 0c11.678b.f6f7 ARPA                   <<< Now the static ARP entry is
complete

// Attaching to DFC3...
MXC.CALO.Sup2T-sw2-dfc3#show plat hard cef 10.1.2.1 detail
[snip...]
Format:IPV4 (valid class vpn prefix)
M(53 ): 1 F 2FFF 255.255.255.255
V(53 ): 1 0 0 10.1.2.1
(A:114689, LS:0, NR:0, RI:0, DF:0 CP:0 DGTv:1, DGT:0)

The ARP entry exist in CEF table for DFC3. Same Adjacency Index result as before...
```

これらの隣接関係エントリの動作を理解したので、最後にCEFエントリの追加に進むことができます。最後のセクションでは、プレフィックス1.1.1.1/32のCEFエントリが**test platform hardware cef v4-deleteコマンドで削除されまし**た。次に、**test platform hardware cef v4-insert [prefix]**コマンドを使用して、このコマンドを再度追加します **[マスク長] vpn [vpn番号]隣接関係[隣接関係インデックス]**

これを確認するには、コマンド**test platform hardware cef v4-insert 1.1.1.1 32 vpn 0 adjacency 114689**を使用します。このエントリはDFC CEFテーブルに再度追加されています。

```
MXC.CALO.Sup2T-sw2-dfc3#test platform hardware cef v4-insert 1.1.1.1 32 vpn 0 adjacency 114689
test_ipv4_insert: done: sw_index = 42

MXC.CALO.Sup2T-sw2-dfc3#show plat hard cef vpn 0 1.1.1.1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
54 1.1.1.1/32 Vl2 ,0c11.678b.f6f7

Ping from the 3750X to Loopback 0 is successful and HW forwarded by 6500 DFC.

MXC.CALO.Sup2T-sw2-dfc3#show platform hard cef adj entry 114689

Index: 114689 -- Valid entry (valid = 1) --

RIT fields: The entry has a Layer2 Format
_____________________________________________________
|decr_ttl=YES | l2_fwd=NO | ccc = 4 | add_shim_hdr = NO
|_____________|___________|_________|__________________

Statistics: Packets = 684
Bytes = 80712
```

```
// Logs in 3850
CALO.MXC.385024XU#show logging [snip...] *Jan 23 05:59:56.911: ICMP: echo reply sent, src
1.1.1.1, dst 10.1.1.1, topology BASE, dscp 0 topoid 0 *Jan 23 05:59:57.378: ICMP: echo reply
sent, src 1.1.1.1, dst 10.1.1.1, topology BASE, dscp 0 topoid 0 *Jan 23 05:59:57.390: ICMP: echo
reply sent, src 1.1.1.1, dst 10.1.1.1, topology BASE, dscp 0 topoid 0
```

VRFルーティングテーブルのエントリの追加と削除

上記のすべての手順で行った設定を通じて、**show platform hardware cefコマンド**の**vpn 0**文字列が強制されています。このコマンドはデフォルトで一般ルーティングテーブルまたは**vpn 0**のエントリを返すため、完全に不要と思われる場合でも、この操作は目的に従って行われました。 CEFエントリ1.1.1.1/32を追加および削除したドキュメントでは、必ず特定のルーティングテーブルインスタンス(VRF)にエントリが追加または削除されることに注意してください。ただし、特定のプレフィックスが異なるVRF(ie.10.x.x.x) 誤ったVRFのCEFエントリを削除、追加、または変更すると、悪影響を及ぼす可能性があります。

VRF **TEST**_VRFのプレフィックス1.1.1.1/32を持つCEFエントリ**を削除します**。CEFエントリの追加の詳細については、このドキュメントの「**CEFエントリの追**加」セクションを参照してください。

VRFを追加するには、コマンド**ip vrf forwarding [VRF-NAME]を使用して6500スイッチのSVIを提案されたVRFに変更し、最後にTEST_VRFテーブルに同じスタティックルートを追加**します。

```
MXC.CALO.Sup2T(config)#ip vrf TEST_VRF
MXC.CALO.Sup2T(config-vrf)#int vlan 1
MXC.CALO.Sup2T(config-if)#ip vrf forwarding TEST_VRF
% Interface Vlan1 IPv4 disabled and address(es) removed due to enabling VRF TEST_VRF
MXC.CALO.Sup2T(config-if)#ip add 10.1.1.10 255.255.255.0
MXC.CALO.Sup2T(config-if)#int vlan 2
MXC.CALO.Sup2T(config-if)#ip vrf forwarding TEST_VRF
% Interface Vlan2 IPv4 disabled and address(es) removed due to enabling VRF TEST_VRF
MXC.CALO.Sup2T(config-if)#ip add 10.1.2.10 255.255.255.0
MXC.CALO.Sup2T(config)#ip route vrf TEST_VRF 1.1.1.1 255.255.255.255 10.1.2.1

MXC.CALO.Sup2T#show ip vrf
Name Default RD Interfaces
TEST_VRF <not set> Vl1
Vl2
```

VRFも順番にプログラムされます。これはスイッチ内の最初のVRFで（以前は他のVRFは設定されていません）、このVRFインスタンスのVPN番号は1である必要があります。**show platform hardware cef vpn 1コマンドを実行して、これが正しいことを確認します。**

```
MXC.CALO.Sup2T-sw2-dfc3#show plat hard cef vpn 1
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
34 10.1.1.10/32 receive
35 10.1.1.0/32 receive
36 10.1.1.255/32 receive
38 10.1.2.10/32 receive
43 10.1.2.0/32 receive
44 10.1.2.255/32 receive
53 10.1.2.1/32 Vl2 ,0c11.678b.f6f7
54 1.1.1.1/32 Vl2 ,0c11.678b.f6f7
[snip...]
```

However, usually, switches have hundred or thousands of VRFs and just count them in the 'show ip vrf' command output would be quite difficult. In order to know which VPN number is assigned to a VRF we will run the command "show platform hardware cef vrf [VRF name] [prefix] detail", it will return the actual vpn number for that VRF:

```
Format:IPV4 (valid class vpn prefix)
M(54 ): 1  F 2FFF 255.255.255.255
```

V(54): 1 0 **1**   1.1.1.1   <<<<<<<<<<< The number in red
determines the VPN this prefix belongs to.
(A:114689, LS:0, NR:0, RI:0, DF:0 CP:0 DGTv:1, DGT:0)

このエントリの実際のVPN番号とソフトウェアインデックスを知っておくことが重要です。このエントリの削除や、このVRFインスタンスとの間での追加に進むことができます。

MXC.CALO.Sup2T-sw2-dfc3#**test platform hardware cef index-conv hw_to_sw 54**
hw index: 54 ----> sw handle: 42

MXC.CALO.Sup2T-sw2-dfc3#**test platform hardware cef v4-delete 42 mask 32 vpn 1**
test_ipv4_delete: done.

Result:

MXC.CALO.Sup2T-sw2-dfc3#**show platform hardware cef vpn 1 lookup 1.1.1.1**
Codes: decap - Decapsulation, + - Push Label
Index Prefix Adjacency
262049 0.0.0.0/0 drop

Traffic is now getting punted, and the effects are seen in the 3750X pings to 1.1.1.1:

MXC.CALO.3750X#ping 1.1.1.1 repe 5000000
Sending 5000000, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!
!!!.!!!!!!!!!!!!
!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!.!!!!!
!!
!!!!!!!!!!!!!!!!!!!!!!!!!.!!
!!!.!!!!!!!!!!
!!
!!!!!!!!!!!!!!!!!!!!.!!!
!!!.!!!!!!!!!!!!!!!!!!
!!
!!!!!!!!!!!!!!!!!!!!!!.!!!
!!!.!!!!!!!!!!!!!!!!
[snip...]

// Packet loss
実稼働ネットワークでは、これらのCEFエントリの状態が原因で、パケット損失や音声の途切れやビデオの不良が発生することを考慮してください。したがって、メンテナンス時間帯にこれらのテストを実行することを推奨します。