

Comprenez les entrées CEF de Catalyst 6500 S2T

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Informations générales](#)

[Diagramme du réseau](#)

[Identifiez les entrées CEF au-dessus des engines de transfert distribué](#)

[Entrées CEF d'effacement](#)

[Ajoutez une entrée CEF](#)

[Ajoutez et supprimez les entrées pour des Tableaux de routage de VRF](#)

Introduction

Ce document décrit comment Cisco Catalyst 6500 avec le superviseur Sup2T programme les entrées CEF (de Cisco Express Forwarding) configurées sur le logiciel de Cisco IOS dans le matériel de linecards utilisé pour réaliser le transfert de paquet.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- [Cisco Express Forwarding \(CEF\)](#)
- [Commutateurs de la gamme Cisco Catalyst 6500](#)
- Carte de transfert distribué Cisco (DFC)

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Linecard de Cisco Catalyst 6500 WS-X6848-GE-TX (avec DFC4).
- Cisco Catalyst 6500 avec le superviseur 2T sur la version IOS 15.2.1SY5

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est vivant, assurez-vous que vous comprenez l'impact potentiel de n'importe quelle commande.

Informations générales

Le CEF comme mécanisme de commutation de couche 3 est utilisé par la plupart des commutateurs multicouches de Cisco. Il est impératif que les ingénieurs réseau comprennent comment des travaux de CEF afin de dépanner des scénarios de retard de pannes de réseau, de perte de paquets ou de paquet quotidiennement.

Superviseur Sup2T en mode autonome ou pendant que le VSS est actuellement déployé par beaucoup de réseaux d'entreprise pendant qu'un principal commutateur, agrège pratiquement tous autres périphériques de routage ou de commutation. Ceci signifie également qu'en avant le plus intra et le trafic inter de domaine afin de livrer avec succès les paquets à ses destinations. Pour que ceci soit réalisé, Sup2T doit avoir les informations de routage appropriées apprises statiquement ou dynamiquement par l'intermédiaire des protocoles de routage.

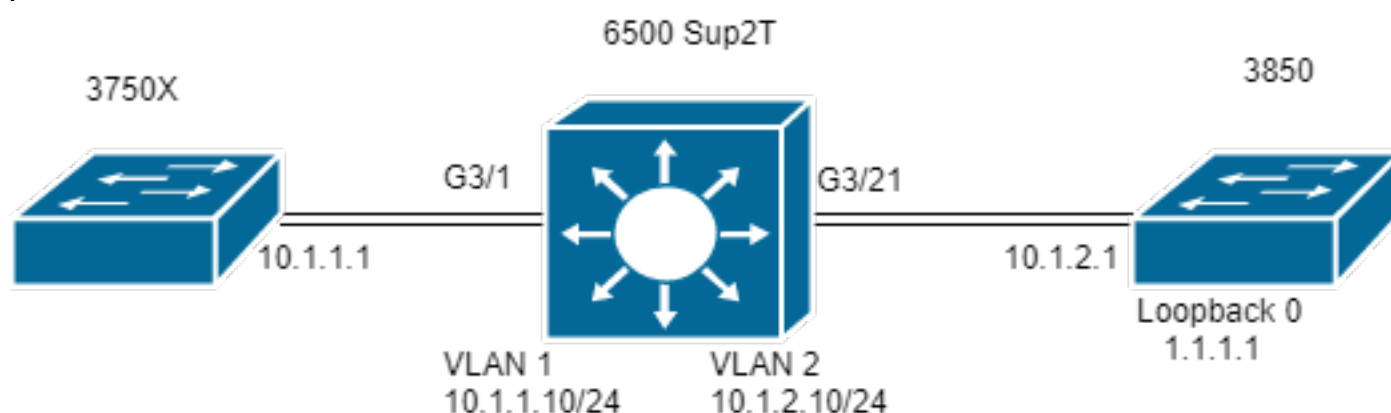
Dans un châssis modulaire, les plusieurs engines d'expédition pourraient exister sans compter que le superviseur. Certains linecards (particulièrement la nouvelle génération ceux telle que C6800-32P10G) incluent déjà leur propre engine d'expédition afin d'améliorer des performances de commutation par paquets, la consultation des entrées CEF exectued localement et cause des ressources d'être distribuées mieux pour le trafic ce des d'entrée au-dessus de différents linecards. Ceux-ci sont connus en tant que cartes d'expédition de Distributed (DFC).

Ces entrées CEF partagées à travers toutes les engines d'expédition pourraient pour être allouées dans HW pour de plusieurs raisons, d'un état d'erreur de logiciel, des ressources que l'épuisement à la CPU élevée conditionne et empêche le commutateur pour avoir assez de temps de mettre à jour toutes les entrées, ceci la boîte entraîne une gamme d'événements indésirables.

Diagramme du réseau

Réseau

:



```
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
```

Identifiez les entrées CEF au-dessus des engines de transfert distribué

Dans le diagramme, un commutateur 6506 autonome a un superviseur 2T installé aussi bien qu'un linecard WS-6848-GE-TX avec un DFC dans l'hôte 3750X de l'emplacement 3. qui est connecté au linecard par le port G3/1 envoie au trafic au bouclage 3850's 0 adresses 1.1.1.1.

Pour ceci, 3750X a une artère statique à l'adresse IP 1.1.1.1 par le prochain saut 10.1.1.10 qui est le SVI du VLAN 1 dans le commutateur Sup2T. Le commutateur Sup2T doit conduire ce trafic au commutateur 3850 basé dans une entrée de route statique pour IP 1.1.1.1/32 par l'intermédiaire du prochain saut 10.1.2.1 qui est l'interface connectée 3850 au Sup2T dans le VLAN 2.

```
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
```

Rendez-vous compte que dans l'intérêt de la simplicité, les deux 3750X et 3850 Commutateurs sont connectés aux 6500 par le même linecard. Ceci signifie que le trafic est recherché localement et aussi localement expédié.

Un commutateur des d'entrée Sup2T de paquet par l'intermédiaire de Gi3/1, atteint par la suite l'engine d'expédition (puisque c'est un DFC). L'engine d'expédition analyse le champ d'adresse IP de destination en ce paquet et une consultation au-dessus des entrées CEF programmées pour la meilleure correspondance (le plus long masque).

Puisque c'est une carte DFC, il signifie qu'il a ses propres entrées CEF et pour les vérifier, il est que nous se relier au linecard avec le **[dec] d'attache de commande ou relie le [dec] modèle du commutateur [1-2]** pour le VSS.

Maintenant, vous devriez être dans la demande DFC, **cef de matériel de show platform de** commande ou retour du **vpn 0 de cef de matériel de show platform que** toutes les entrées CEF ont programmé pour la table de routage générale (VPN 0 aucun VRF).

Puisque l'objectif est le préfixe 1.1.1.1/32 vous utilisez la **consultation 1.1.1.1 du vpn 0 de cef de matériel de show platform de** commande. La commande renvoie la meilleure correspondance pour le préfixe 1.1.1.1 et celui qu'il l'utilise pour expédier réellement le trafic :

```
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 V12 ,0c11.678b.f6f7
```

L'entrée CEF est là, il a obtenu programmé suite à notre entrée statique programmée en logiciel IOS par l'intermédiaire de l'**artère 1.1.1.1 255.255.255.255 10.1.2.1 d'IP de** commande.

Vous pouvez également vérifier que cette entrée obtient des hit et le trafic est expédié avec cette entrée par l'intermédiaire du **détail de 1.1.1.1 de cef de matériel de show platform de** commandes qui renvoie une entrée de contiguïté :

```
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)
```

En conclusion, l'entrée de contiguïté affiche comment le paquet est réécrit et si le trafic est réécrit par cette entrée de contiguïté :

```
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 <<<<

```

Le dest_mac et le src_mac sont les valeurs d'intérêt principal, qui indiquent les nouvelles en-têtes L2 qui sont écrites pour ce paquet. L'adresse MAC 0c11.678b.f6f7 de destination est 10.1.2.1 qui est les 3850 (prochain saut pour atteindre 1.1.1.1) :

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 <<<<

```

En outre, le domaine de **statistiques** prouve que le trafic est frappe réellement cette entrée de contiguïté et les en-têtes L2 sont réécrits en conséquence.

Entrées CEF d'effacement

Les entrées CEF d'effacement peuvent nous aider à supprimer n'importe quelle entrée qui pourrait être incorrectement programmée (à une entrée fausse de contiguïté par exemple) ou même pour l'exercer. Il fournit également une manière de modifier un chemin de routage.

Afin de supprimer une entrée CEF, vous devez comprendre que des entrées CEF sont programmées séquentiellement et faire assigner un index de matériel, par exemple :

Vpn 0 de cef de matériel de la plate-forme MXC.CALO.Sup2T-dfc3#show

Codes : decap - Décapsulage, + - Poussez l'étiquette

```
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 V11 ,a0ec.f930.3f40 261
10.1.2.1/32 V12 ,0c11.678b.f6f7 262 1.1.1.1/32 V12 ,0c11.678b.f6f7 <<<< Our CEF entry of
interest has a HW index of 262.
...

```

Cet index de matériel est l'élément le plus important pour supprimer une entrée CEF puisqu'il l'a utilisé comme référence. Cependant, afin de faire n'importe

quelle modification là-dessus, il doit être converti en traitement de logiciel. Vous pouvez réaliser ceci avec le **hw_to_sw d'index-conv de cef de matériel de la plate-forme de test de** commande [**l'index de hw**]

```
MXC.CALO.Sup2T-dfc3#test platform hardware cef index-conv hw_to_sw 262
```

```
hw index: 262 ----> sw handle: 101
```

Maintenant que vous connaissez le traitement de logiciel, vous pouvez poursuivre la suppression d'entrée CEF avec le **[dec] de vpn de masque du cef v4-delete [traitement de matériel de la plate-forme de test de** commande **commutateur] [longueur de masque]**

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

Note: La valeur de longueur de masque est 32 puisque c'est une artère spécifique d'hôte (1.1.1.1/32)

Maintenant, notre entrée CEF est supprimée :

```
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 V11 ,a0ec.f930.3f40
```

```
261 10.1.2.1/32 V12 ,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
```

Notez que la commande du **vpn 0 de cef de matériel de la plate-forme de test** a été exécutée sous la demande DFC. De cette façon, l'entrée CEF a été retirée de la table CEF du DFC et PAS du superviseur, vous devez faire attention vraiment sur de quelle engine d'expédition les entrées sont retirées.

Un changement du trafic a le risque sans visibilité (en cas d'essai en laboratoire), ceci peut être dû au hit d'une autre entrée CEF. Considérez apparier toujours le plus précis (le plus long masque). Dans ce laboratoire, il frappe :

```
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
```

Ainsi ce qui fait cette entrée faites réellement avec le paquet ? :

```
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
```

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 V11, 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, identifieur 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 V12, 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, identifieur 50407
df 0, mf 0, fo 0, ttl 254, src 10.1.1.1, dst 1.1.1.1
icmp type 8, code 0
```

Maintenant, tous trafiquent avec la destination de 1.1.1.1 que des d'entrée par le linecard 3 est recyclé avec l'en-tête de cale et donnée un coup de volée à la CPU. Parfois, au lieu de cette entrée CEF, encore 0.0.0.0/0 avec la **baisse** adjacency est vu et fait le précis la même chose.

Note: Évaluez quelles entrées CEF sont retirées. Une utilisation du CPU élevé peut être provoqué par en raison de ceci. Généralement un default route 0.0.0.0/0 est configuré et le trafic est expédié basé sur lui (et entraîne la perte de paquets).

Ajoutez une entrée CEF

Quand une entrée CEF est ajoutée, résout dans la plupart des cas n'importe quel problème d'erreur de programmation qui entraîne la perte de paquets, le retard de paquet ou l'utilisation du CPU élevé. La connaissance de la façon d'installer les entrées CEF dans le matériel, fournit non seulement la capacité de corriger une entrée misprogrammée, mais également de manipuler n'importe quel transfert de paquet par le recyclage du paquet, de l'indiquer une interface complètement différente ou un prochain saut, de réécrire un paquet routé comme désiré et/ou de le relâcher, etc. Toute la ceci, sans recharge de la case, retire et place la configuration ou n'importe quelle modification apparente. L'ajout d'entrée CEF peut être fait sans obtenir dans le mode de configuration aussi. (Comme vous avez également fait avec la section passée de méthode de dépose d'entrée CEF dedans expliquée).

Fondamentalement, il y a deux situations ici, quand vous avez une entrée valide d'ARP au prochain saut, dans ce cas 10.1.2.1 et quand vous ne faites pas (pour une raison quelconque). La deuxième situation vous force pour créer réellement une entrée valide d'ARP (par l'intermédiaire de l'ARP statique) :

Étape 1. Il y a une entrée d'ARP dans le commutateur pour 10.1.2.1 qui est le prochain saut pour 1.1.1.1.

```
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
Une entrée d'ARP est programmée comme route hôte (/32) dans la table CEF :
```

```
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 V12 ,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)

N'importe quel paquet avec n'importe quelle adresse IP de destination qui a le même prochain saut de liaison de données devrait être expédié par la même interface et être réécrit avec les mêmes en-têtes L2.

Quoique ceci pourrait sembler assez évident au début, c'est réellement l'élément le plus important pour ajouter une entrée CEF, vous doit lui dire qu'un paquet devrait être réécrit avec une entrée spécifique de contiguïté CEF.

Étape 2. Maintenant, supposez qu'il n'y a aucune entrée d'ARP automatiquement créée pour ceci, ainsi vous devez créer une entrée statique d'ARP.

Afin de faire ceci, vous devez connaître l'adresse MAC du périphérique qui est utilisé comme prochain-saut pour le préfixe 10.1.2.1, ainsi il est envoyé à 0c11.678b.f6f7. S'il y a déjà une entrée d'adresse MAC dans la sortie de commande du **show mac address-table address 0c11.678b.f6f7** qui est bien, sinon alors vous devez créer une entrée de MAC statique :

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
```

Étape 3. En conclusion, une entrée statique d'ARP doit être créée pour qu'une entrée CEF soit programmée :

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...

Maintenant que vous comprenez ce que ces entrées de contiguïté font, vous pouvez finalement poursuivre pour ajouter une entrée CEF. Dans la dernière section, l'entrée CEF pour le préfixe 1.1.1.1/32 a été supprimée par la commande du **cef v4-delete de matériel de la plate-forme de test**. Maintenant, ajoutez-le de retour par le **cef v4-insert [préfixe] [longueur de matériel de la plate-forme de test de commande de masque] contiguïté de vpn [nombre de vpn] [l'index de contiguïté]**

Afin de vérifier ceci, utilisez la **config** 114689 du vpn 0 du cef v4-insert 1.1.1.1 32 de matériel de la plate-forme de test de commande. L'entrée a été ajoutée de retour dans la table CEF DFC :

```
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 V12 ,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
```

Ajoutez et supprimez les entrées pour des Tableaux de routage de VRF

Dans toute la configuration faite à partir de toutes les étapes précédentes, la chaîne du **vpn 0** dans les commandes de **cef de matériel de show platform** a été imposée. Même si il semble complètement inutile puisque la commande par défaut renvoie les entrées pour la table de routage générale ou le **vpn 0**, ceci a été faite sur le but afin d'avoir toujours à l'esprit que des entrées sont ajoutées ou supprimées des exemples spécifiques de table de routage (vrf), par le document que vous avez ajouté et avez supprimé l'entrée CEF 1.1.1.1/32. Cependant, certains préfixes sont très pour exister dans différents vrf (c.-à-d. 10.x.x.x) et l'effacement, ajouter ou modifier une entrée CEF pour un VRF faux peut entraîner une incidence négative.

Supprimez une entrée CEF avec le préfixe 1.1.1.1/32 pour le VRF **TEST_VRF**. Pour une description détaillée de l'ajout des entrées CEF, référez-vous à l'**ajouter par** section d'**entrée CEF de** ce document.

Afin d'ajouter le VRF, le changement SVI du commutateur 6500 au VRF proposé avec l'**ip vrf forwarding de** commande **[VRF-NAME]** et ajouter finalement la même artère statique dans notre table **TEST_VRF** :

```
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> V11
```

```
V12
```



```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! .!!!!!!!!!!!!!!!!!!!!!!  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! .!!!!!!!!!!!!!!!!!!!!!!  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! .!!!!!!!!!!!!!!!!!!!!!!  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! .!!!!!!!!!!!!!!!!!!!!!!  
[snip...]
```

// Packet loss

Considérez cela dans un réseau de production, perte de paquets et l'audio ou le vidéo variable du mauvais est dû expérimenté à l'état de ces entrées CEF. Par conséquent, il est recommandé pour réaliser ces essais dans une fenêtre de maintenance.