

Comprendre les entrées CEF Catalyst 6500 S2T

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Diagramme du réseau](#)

[Identifier les entrées CEF sur les moteurs de transfert distribué](#)

[Supprimer les entrées CEF](#)

[Ajouter une entrée CEF](#)

[Ajouter et supprimer des entrées pour les tables de routage VRF](#)

Introduction

Ce document décrit comment Cisco Catalyst 6500 avec Supervisor Sup2T programme les entrées CEF (Cisco Express Forwarding) configurées sur le logiciel Cisco IOS dans le matériel des cartes de ligne utilisé pour effectuer le transfert de paquets.

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 DFC (Distributed Forwarding Card) Cisco

Components Used

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

- Carte de ligne Cisco Catalyst 6500 WS-X6848-GE-TX (avec DFC4).
- Cisco Catalyst 6500 avec Supervisor 2T sur IOS version 15.2.1SY5

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

CEF en tant que mécanisme de commutation de couche 3 est utilisé par la plupart des commutateurs multicouches Cisco. Il est impératif que les ingénieurs réseau comprennent comment CEF fonctionne afin de dépanner des pannes de réseau, des pertes de paquets ou des scénarios de retard de paquets au quotidien.

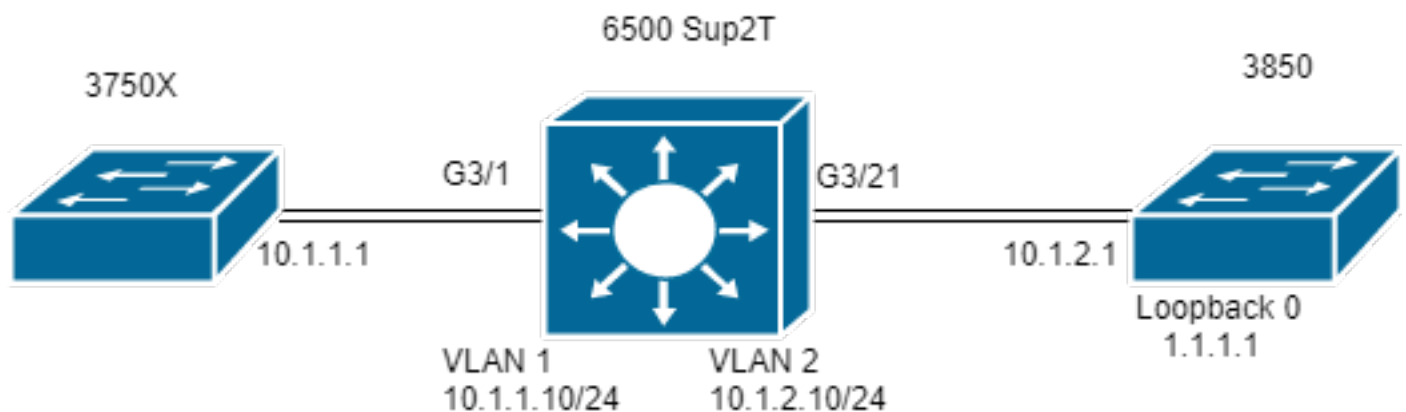
Le superviseur Sup2T en mode autonome ou comme VSS est actuellement déployé par de nombreux réseaux d'entreprise en tant que commutateur principal, agrège pratiquement tous les autres périphériques de routage ou de commutation. Cela signifie également que transfère la plupart du trafic intra et inter-domaine afin de livrer les paquets avec succès à ses destinations. Pour ce faire, les informations de routage appropriées doivent être apprises de manière statique ou dynamique via les protocoles de routage.

Dans un châssis modulaire, plusieurs moteurs de transfert peuvent exister en plus de ceux du superviseur. Certaines cartes de ligne (en particulier les nouvelles générations comme C6800-32P10G) incluent déjà leur propre moteur de transfert afin d'améliorer les performances de commutation de paquets, la recherche d'entrées CEF est exécutée localement et fait que les ressources sont mieux réparties pour le trafic entrant sur différentes cartes de ligne. Ces cartes sont appelées DFC (Distributed Forwarding Cards).

Ces entrées CEF partagées sur tous les moteurs de transfert peuvent ne pas être allouées en matériel pour plusieurs raisons, depuis une condition de défaut logiciel, l'épuisement des ressources jusqu'à des conditions CPU élevées et empêche le commutateur de disposer de suffisamment de temps pour mettre à jour toutes les entrées, cela peut provoquer une série d'événements indésirables.

Diagramme du réseau

Réseau:



```
Switch#show module 3
```

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

Identifier les entrées CEF sur les moteurs de transfert distribué

Dans le schéma, un commutateur 6506 autonome est équipé d'un Supervisor 2T ainsi que d'une carte de ligne WS-6848-GE-TX avec DFC dans le logement 3. L'hôte 3750X, connecté à la carte de ligne via le port G3/1, envoie le trafic vers l'adresse de bouclage 0 1.1.1.1 du 3850.

Pour cela, 3750X dispose d'une route statique vers l'adresse IP 1.1.1.1 à 10.1.1.10 du tronçon suivant, qui est l'interface SVI de VLAN 1 dans le commutateur Sup2T. Le commutateur Sup2T doit acheminer ce trafic vers le commutateur 3850 en fonction d'une entrée de route statique pour IP 1.1.1.1/32 via le tronçon suivant 10.1.2.1, qui est l'interface 3850 connectée 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
```

Par souci de simplicité, les commutateurs 3750X et 3850 sont tous deux connectés au 6500 via la même carte de ligne. Cela signifie que le trafic est recherché localement et transféré localement également.

Un paquet entre dans le commutateur Sup2T via Gi3/1, puis atteint le moteur de transfert (car il s'agit d'une DFC). Le moteur de transfert analyse le champ d'adresse IP de destination dans ce paquet et recherche la meilleure correspondance possible dans les entrées CEF programmées (masque le plus long).

Puisqu'il s'agit d'une carte DFC, cela signifie qu'elle a ses propres entrées CEF et pour les vérifier, il est nécessaire que nous nous attachions à la carte de

ligne avec la commande **attachement [dec] ou attachement commutateur [1-2] mod [dec]** pour VSS.

Maintenant, vous devez être dans l'invite DFC, la commande **show platform hardware cef** ou **show platform hardware cef vpn 0** retourne toutes les entrées CEF programmées pour la table de routage générale (VPN 0/ No VRF).

Puisque l'objectif est le préfixe 1.1.1.1/32, vous utilisez la commande **show platform hardware cef vpn 0 lookup 1.1.1.1**. La commande retourne la meilleure correspondance pour le préfixe 1.1.1.1 et celle qu'elle utilise pour transférer 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, elle a été programmée à la suite de notre entrée statique programmée dans le logiciel IOS via la commande **ip route 1.1.1.1 255.255.255.255 10.1.2.1**.

Vous pouvez également vérifier que cette entrée obtient des résultats et que le trafic est transféré avec cette entrée via les commandes **show platform hardware cef 1.1.1.1 detail** 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)
```

Enfin, l'entrée de contiguïté montre 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 <<<<

```

Les valeurs **dest_mac** et **src_mac** sont les valeurs d'intérêt principal, qui indiquent les nouveaux en-têtes L2 écrits pour ce paquet. L'adresse MAC de destination 0c11.678b.f6f7 est 10.1.2.1, qui est le 3850 (tronçon suivant pour atteindre 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 30 0c11.678b.f6f7 ARPA Vlan2

```

En outre, le champ **Statistiques** indique que le trafic atteint effectivement cette entrée de contiguïté et que les en-têtes L2 sont réécrits en conséquence.

Supprimer les entrées CEF

La suppression des entrées CEF peut nous aider à supprimer toute entrée qui pourrait être programmée à tort (à une entrée de contiguïté erronée par exemple) ou même à des fins de formation. Il permet également de modifier un chemin de routage.

Pour supprimer une entrée CEF, vous devez comprendre que les entrées CEF sont programmées de manière séquentielle et qu'un index matériel est affecté, par exemple :

```

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

```

Codes : décapsulation, + - Étiquette de poussée

```

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 matériel est l'élément le plus important pour supprimer une entrée CEF car elle est utilisée comme référence. Cependant, pour pouvoir effectuer toute modification, il doit être converti en handle de logiciel. Vous pouvez y parvenir avec la commande **test platform hardware cef index-conv hw_to_sw [hw index]**

```

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 handle logiciel, vous pouvez supprimer l'entrée CEF avec la commande test platform hardware cef v4-delete [sw handle] mask [mask length] vpn [dec]

```

```

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

```

Note: La longueur du masque est 32, car il s'agit d'une route spécifique à l'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 **test platform hardware cef vpn 0** a été exécutée sous l'invite DFC. De cette façon, l'entrée CEF a été supprimée de la table CEF de la DFC et NON du superviseur, vous devez être très prudent sur le moteur de transfert dont les entrées sont supprimées.

Une modification du trafic risque de ne pas avoir de visibilité (en cas de test en laboratoire), ceci peut être dû au succès d'une autre entrée CEF. Pensez à toujours faire correspondre le plus exact (masque le plus long). Au cours de ces travaux pratiques, il a atteint :

```

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

```

Que fait donc cette entrée avec le paquet ? :

```

MXC.CALO.Sup2T-dfc3#show platform hardware cef adjacencies entry 262048
RIT fields: The entry has a Recirc. Format
-----|decr_ttl=NO | 12_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 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, 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 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, identifiant 50407
df 0, mf 0, fo 0, ttl 254, src 10.1.1.1, dst 1.1.1.1
icmp type 8, code 0
```

À présent, tout le trafic avec la destination 1.1.1.1 qui passe par la carte de ligne 3 est rédiffusé avec un en-tête de fractionnement et pointé vers le processeur. Parfois, au lieu de cette entrée CEF, une autre 0.0.0.0/0 avec contiguïté de suppression est vue et fait exactement la même chose.

Note: Évaluez les entrées CEF supprimées. Une utilisation élevée du CPU peut être causée par cela. Généralement, une route par défaut 0.0.0.0/0 est configurée et le trafic est transféré en fonction de celle-ci (et entraîne la perte de paquets).

Ajouter une entrée CEF

Lorsqu'une entrée CEF est ajoutée, résout dans la plupart des cas tout problème de mauvaise programmation qui entraîne une perte de paquets, un retard de paquets ou une utilisation élevée du CPU. La connaissance de l'installation des entrées CEF dans le matériel permet non seulement de corriger une entrée mal programmée, mais également de manipuler tout transfert de paquets via la recirculation du paquet, le diriger vers une interface complètement différente ou vers le saut suivant, réécrire un paquet routé comme souhaité et/ou le supprimer, etc. Tout ceci, sans rechargement de la boîte, supprimer et définir la configuration ou toute modification apparente. Entrée CEF l'ajout peut être effectué sans passer en mode de configuration. (Comme vous l'avez également fait avec la procédure de suppression des entrées CEF expliquée dans la section précédente).

En gros, il y a deux situations ici, quand vous avez une entrée ARP valide au saut suivant, dans ce cas 10.1.2.1 et quand vous ne le faites pas (pour une raison quelconque). La deuxième situation vous force à créer une entrée ARP valide (via ARP statique) :

Étape 1. Il y a une entrée ARP dans le commutateur pour 10.1.2.1 qui est le saut suivant 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 ARP est programmée en tant que 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)
```



```
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
```

Ajouter et supprimer des entrées pour les tables de routage VRF

Tout au long de la configuration effectuée à partir de toutes les étapes précédentes, la chaîne **vpn 0** dans les commandes **show platform hardware cef** a été appliquée. Même si cela semble totalement inutile puisque la commande retourne par défaut les entrées pour la table de routage générale ou **vpn 0**, cela a été fait exprès afin de garder toujours à l'esprit que les entrées sont ajoutées ou supprimées d'instances de table de routage spécifiques (VRF), via le document que vous avez ajouté et supprimé l'entrée CEF 1.1.1.1/32. Cependant, certains préfixes sont très susceptibles d'exister dans différents VRF (i. e. 10.x.x.x) et supprimer, ajouter ou modifier une entrée CEF pour un VRF incorrect peut avoir un impact négatif.

Supprimez une entrée CEF avec le préfixe 1.1.1.1/32 pour VRF **TEST_VRF**. Pour obtenir une description détaillée de l'ajout d'entrées CEF, reportez-vous à la section **Ajouter une entrée CEF** de ce document.

Afin d'ajouter le VRF, modifiez les interfaces SVI du commutateur 6500 en VRF proposé avec la commande **ip vrf forwarding [VRF-NAME]** et ajoutez enfin la même route 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
```

Les VRF sont également programmés séquentiellement. Il s'agissait du premier VRF du commutateur (aucun autre VRF n'a été configuré précédemment), le numéro de vpn de cette instance VRF doit donc être 1. Exécutez la commande **show platform hardware cef vpn 1** afin de vérifier que ceci est vrai :

```
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 V12 ,0c11.678b.f6f7
54 1.1.1.1/32 V12 ,0c11.678b.f6f7
[snip...]
```

However, usually, switches have hundred or thousands of VRFs and just count them in the 'show ip

