Implémentation IOS de la caractéristique de l'iBGP PE-CE

Contenu

[Introduction](#)
Informations générales
[IBGP PE-CE de mise en place](#)
[Attribut d'artère de client BGP](#)
[Configurez](#)
[Nouvelle commande](#)
[Détaillé regardez ATTR_SET](#)
[Prochaine manipulation de saut](#)
[RD](#)
[caractéristique de l'iBGP PE-CE avec le local-as](#)
[Règles pour l'échange d'artère entre différents sites de VRF](#)
[Réflexion de Vrf-Lite de Ce-à-CE](#)
[Un Cisco IOS plus ancien sur le routeur PE](#)
[Next-hop-self pour l'eBGP sur le VRF](#)

Introduction

Ce document décrit comment l'Internal Border Gateway Protocol (iBGP) entre le Provider Edge (PE) et la caractéristique de Customer Edge (CE) est mise en application dans le Cisco IOS®.

Informations générales

Jusqu'à la nouvelle caractéristique de l'iBGP PE-CE, l'iBGP entre le PE et le CE (par conséquent sur une interface de Virtual Routing and Forwarding (VRF) sur le routeur PE) n'ont pas été officiellement pris en charge. Une exception est iBGP sur des interfaces de VRF dans un CE de Multi-VRF (Vrf-Lite) installé. La motivation pour déployer cette caractéristique est :

- Le client veut avoir un seul numéro de système autonome (ASN) sur les plusieurs sites du VRF, sans déploiement d'External Border Gateway Protocol (eBGP) avec l'as-override.

- Le client veut fournir la réflexion de route interne vers les Routeurs de la CE, agissant comme si le noyau de fournisseur de services (fournisseur de services) est un réflecteur transparent d'artère (rr).

Avec cette configuration, les sites du VRF peuvent avoir le même ASN que le noyau de fournisseur de services. Cependant, au cas où les ASN des sites de VRF seraient différents que

l'ASN du noyau de fournisseur de services, il peut être fait pour apparaître les mêmes avec l'utilisation du (AS) gens du pays-autonome de système de caractéristique.

IBGP PE-CE de mise en place

Voici les deux parties principales afin de faire fonctionner cette caractéristique :

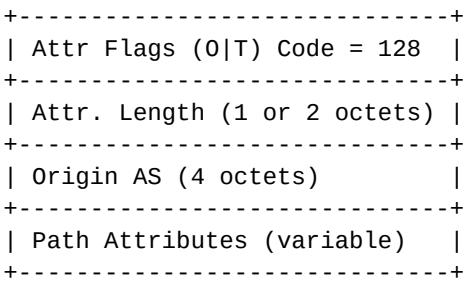
- Un nouvel attribut ATTR_SET a été ajouté au protocole BGP afin de porter les attributs BGP VPN à travers le noyau de fournisseur de services d'une manière transparente.
- Faites au routeur PE un rr pour les sessions d'iBGP vers les Routeurs de la CE dans le VRF et comme rr vers les voisins VPNv4 (les autres Routeurs ou RRs de PE).

Le nouvel attribut ATTR_SET permet au fournisseur de services pour porter tous les attributs BGP du client d'une manière transparente et ne gêne pas les attributs et les stratégies BGP de fournisseur de services. De tels attributs sont la liste de batterie, préférence locale, les communautés, et ainsi de suite.

Attribut d'artère de client BGP

ATTR_SET est le nouvel attribut BGP utilisé afin de porter les attributs BGP VPN du client de fournisseur de services. C'est un attribut transitif facultatif. Dans cet attribut, tous les attributs BGP de client du message de mise à jour BGP, excepté les attributs MP_REACH et MP_UNREACH, peuvent être portés.

L'attribut ATTR_SET a ce format :



Les indicateurs d'attribut sont les indicateurs réguliers d'attribut BGP (référez-vous à RFC 4271). La longueur d'attribut indique si la longueur d'attribut est un ou deux octets. Le but d'origine COMME champ est d'empêcher la fuite d'une artère provenue d'une quant à soit coulé à l'autre COMME sans manipulation appropriée de l'AS_PATH. Le champ de longueur variable d'attributs de chemin porte les attributs BGP VPN qui doivent être portés à travers le noyau de fournisseur de services.

Sur le routeur PE de sortie, les attributs BGP VPN sont poussés dans cet attribut. Sur le routeur PE d'entrée, ces attributs sont sautés de l'attribut, avant que le préfixe BGP soit envoyé au routeur CE. Cet attribut fournit l'isolation des attributs BGP entre le réseau et le client VPN de fournisseur de services et vice versa. Par exemple, l'attribut de liste de batterie de réflexion d'artère de fournisseur de services n'est pas vu et est considéré à l'intérieur du réseau VPN. Mais également, l'attribut de liste de batterie de réflexion d'artère VPN n'est pas vu et est considéré à l'intérieur du réseau de fournisseur de services.

Regardez la figure 1 afin de voir la propagation d'un préfixe BGP de client à travers le réseau de fournisseur de services.

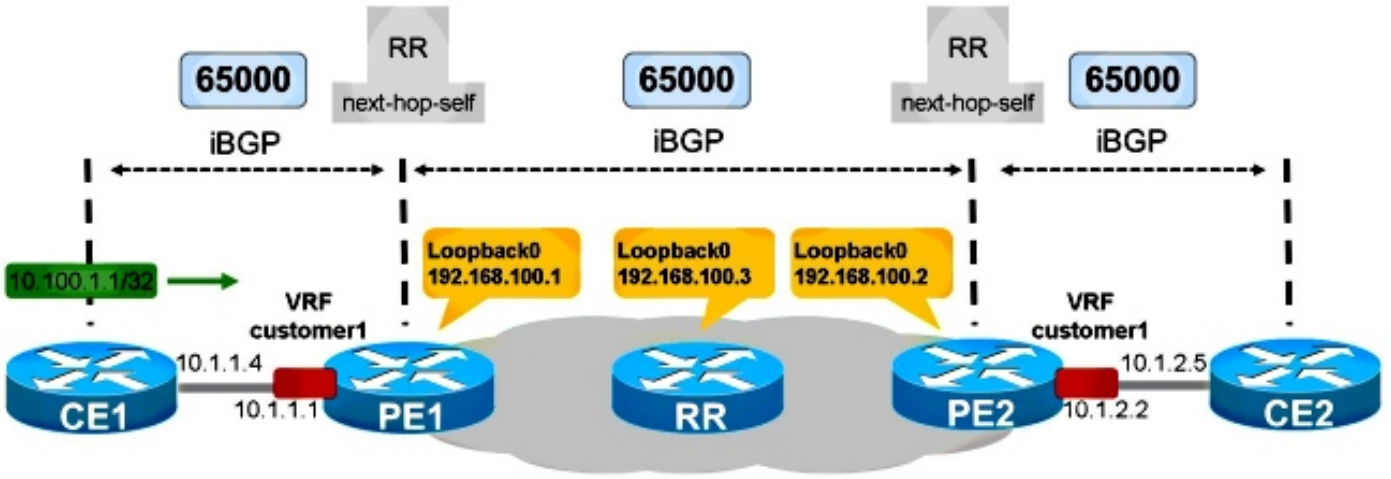


Figure 1

CE1 et CE2 sont dans les mêmes QUE comme réseau de fournisseur de services : 65000. PE1 a l'iBGP configuré vers CE1. PE1 reflète le chemin pour le préfixe 10.100.1.1/32 vers le rr dans le réseau de fournisseur de services. Le rr reflète le chemin d'iBGP vers les Routeurs de PE comme d'habitude. PE2 reflète le chemin vers CE2.

Pour que ceci fonctionne correctement, vous devez :

- Ayez le code sur PE1 et PE2 qui a la prise en charge de fonctionnalité de l'iBGP PE-CE
- Configurez PE1 et PE2 afin d'exécuter la réflexion d'artère sur leur session BGP vers leurs Routeurs respectifs de la CE
- Ayez le next-hop-self sur les Routeurs de PE pour la session BGP vers leurs Routeurs de la CE
- Assurez-vous que chaque site VPN utilise la route différente Distinguishers (RD)

Configurez

Référez-vous à la figure 1.

Voici la configuration nécessaire pour PE1 et PE2 :

```
PE1

vrf definition customer1
rd 65000:1
route-target export 1:1
route-target import 1:1
!
address-family ipv4
exit-address-family
```

```
router bgp 65000
bgp log-neighbor-changes
neighbor 192.168.100.3 remote-as 65000
neighbor 192.168.100.3 update-source Loopback0
!
address-family vpnv4
 neighbor 192.168.100.3 activate
 neighbor 192.168.100.3 send-community extended
exit-address-family
!
address-family ipv4 vrf customer1
 neighbor 10.1.1.4 remote-as 65000
 neighbor 10.1.1.4 activate
 neighbor 10.1.1.4 internal-vpn-client
 neighbor 10.1.1.4 route-reflector-client
 neighbor 10.1.1.4 next-hop-self
exit-address-family PE2

vrf definition customer1
 rd 65000:2
 route-target export 1:1
 route-target import 1:1
 !
 address-family ipv4
 exit-address-family

router bgp 65000
 bgp log-neighbor-changes
 neighbor 192.168.100.3 remote-as 65000
 neighbor 192.168.100.3 update-source Loopback0
 !
 address-family vpnv4
 neighbor 192.168.100.3 activate
 neighbor 192.168.100.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf customer1
 neighbor 10.1.2.5 remote-as 65000
 neighbor 10.1.2.5 activate
 neighbor 10.1.2.5 internal-vpn-client
 neighbor 10.1.2.5 route-reflector-client
 neighbor 10.1.2.5 next-hop-self
 exit-address-family
```

> Remarque: Si le PE n'a pas l'ordre **voisin d'interne-VPN-client de <internal-CE>** pour le voisin de la CE, il ne propage pas les préfixes du CE vers les Routeurs du fournisseur de services RRs/PE.

> Remarque: Si le PE n'est pas le rr dans le VRF, il ne propage pas les préfixes des Routeurs RRs/PE vers le routeur CE.

Nouvelle commande

Il y a une nouvelle commande, interne-VPN-**client voisin de <internal-CE>**, de faire ce travail de feaure. Il doit être configuré sur le routeur PE seulement pour la session d'iBGP vers les Routeurs de la CE.

Remarque: La caractéristique de la CE de Multi-VRF de l'iBGP PE-CE (Vrf-Lite) est encore prise en charge sans ordre **voisin d'interne-VPN-client de <internal-CE>**.

Remarque: Quand l'ordre **voisin d'interne-VPN-client de <internal-CE>** est configuré, les ordres **voisins de next-hop-self de <internal-CE> de route-reflector-client** et de **voisin de <internal-CE>** automatiquement sont aussi bien mis dans la configuration. Quand l'un ou l'autre du **next-hop-self voisin de <internal-CE> de route-reflector-client** et de **voisin de <internal-CE> que des** commandes (ou chacun des deux) sont retirées et une recharge est exécuté, alors elles sont automatiquement remises dans la configuration.

Détaillé regardez ATTR_SET

Référez-vous à la figure 1.

C'est le préfixe annoncé par CE1 :

```
CE1#show bgp ipv4 unicast 10.100.1.1/32
BGP routing table entry for 10.100.1.1/32, version 2
Paths: (1 available, best #1, table default)
 Advertised to update-groups:
    4
 Refresh Epoch 1
 Local
   0.0.0.0 from 0.0.0.0 (10.100.1.1)
     Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best
     rx pathid: 0, tx pathid: 0x0
```

Quand PE1 reçoit le préfixe 10.100.1.1/32 BGP de CE1, il l'enregistre deux fois :

```
PE1#show bgp vpnv4 unicast all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 21
Paths: (2 available, best #1, table customer1)
 Advertised to update-groups:
    5
 Refresh Epoch 1
 Local, (Received from ibgp-pece RR-client)
   10.1.1.4 (via vrf customer1) from 10.1.1.4 (10.100.1.1)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     mpls labels in/out 18/nolabel
     rx pathid: 0, tx pathid: 0x0
 Refresh Epoch 1
 Local, (Received from ibgp-pece RR-client), (ibgp sourced)
   10.1.1.4 (via vrf customer1) from 10.1.1.4 (10.100.1.1)
     Origin IGP, localpref 100, valid, internal
     Extended Community: RT:1:1
     mpls labels in/out 18/nolabel
     rx pathid: 0, tx pathid: 0
```

Le premier chemin est le chemin réel sur PE1, parce qu'il est reçu de CE1.

Le deuxième chemin est le chemin qui est annoncé vers les Routeurs RRs/PE. Il est identifié par l'**ibgp originaire**. Il contient l'attribut ATTR_SET. Notez que ce chemin a un ou plusieurs cibles d'artère (rts) reliées à lui.

PE1 annonce le préfixe comme affiché ici :

```
PE1#show bgp vpnv4 unicast all neighbors 192.168.100.3 advertised-routes
```

```
BGP table version is 7, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

    Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 65000:1 (default for vrf customer1)
 *>i 10.100.1.1/32    10.1.1.4                 0    200      0 i

Total number of prefixes 1
```

C'est comment le rr voit le chemin :

```
RR#show bgp vpnv4 un all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 10
Paths: (1 available, best #1, no table)
 Advertised to update-groups:
    3
 Refresh Epoch 1
 Local, (Received from a RR-client)
   192.168.100.1 (metric 11) (via default) from 192.168.100.1 (192.168.100.1)
     Origin IGP, localpref 100, valid, internal, best
     Extended Community: RT:1:1
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
      ATTR_SET Attribute:
       Originator AS 65000
       Origin IGP
       Aspath
       Med 0
       LocalPref 200
       Cluster list
       192.168.100.1,
       Originator 10.100.1.1
     mpls labels in/out nolabel/18
     rx pathid: 0, tx pathid: 0x0
```

Notez que la préférence locale de ce préfixe de l'unicast VPNv4 est au centre 100. Dans l'ATTR_SET, la préférence locale d'origine de 200 est enregistrée. Cependant, c'est transparent au rr dans le noyau de fournisseur de services.

Sur PE2, vous voyez le préfixe comme affiché ici :

```
PE2#show bgp vpnv4 unicast all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 5
Paths: (1 available, best #1, no table)
 Not advertised to any peer
 Refresh Epoch 2
 Local
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, localpref 100, valid, internal, best
     Extended Community: RT:1:1
     Originator: 10.100.1.1, Cluster list: 192.168.100.3, 192.168.100.1
     ATTR_SET Attribute:
       Originator AS 65000
       Origin IGP
       Aspath
       Med 0
       LocalPref 200
       Cluster list
       192.168.100.1,
       Originator 10.100.1.1
     mpls labels in/out nolabel/18
```

```
     rx pathid: 0, tx pathid: 0x0
BGP routing table entry for 65000:2:10.100.1.1/32, version 6
Paths: (1 available, best #1, table customer1)
 Advertised to update-groups:
    1
 Refresh Epoch 2
 Local, imported path from 65000:1:10.100.1.1/32 (global)
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator AS(ibgp-pece): 65000
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
     mpls labels in/out nolabel/18
     rx pathid:0, tx pathid: 0x0
```

Le premier chemin est celui reçu du rr, avec l'ATTR_SET. Notez que le RD est 65000:1, le RD d'origine. Le deuxième chemin est le chemin importé de la table de VRF avec le RD 65000:1. L'ATTR_SET a été retiré.

C'est le chemin comme vu sur CE2 :

```
CE2#show bgp ipv4 unicast 10.100.1.1/32
BGP routing table entry for 10.100.1.1/32, version 10
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   10.1.2.2 from 10.1.2.2 (192.168.100.2)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator: 10.100.1.1, Cluster list: 192.168.100.2, 192.168.100.1
     rx pathid: 0, tx pathid: 0x0
```

Notez que le prochain-saut est 10.1.2.2, qui est PE2. La liste de batterie contient les Routeurs PE1 et PE2. Ce sont le RRs cette matière à l'intérieur du VPN. Le fournisseur de services rr (10.100.1.3) n'est pas dans la liste de batterie.

La préférence locale de 200 a été préservée à l'intérieur du VPN à travers le réseau de fournisseur de services.

L'unicast BGP vpnv4 de débogage met à jour la commande affiche la mise à jour propagée dans le réseau de fournisseur de services :

```
PE1#
BGP(4): Revise route installing 1 of 1 routes for 10.100.1.1/32 -> 10.1.1.4
(customer1) to customer1 IP table
BGP(4): 192.168.100.3 NEXT_HOP changed SELF for ibgp rr-client pe-ce net
65000:1:10.100.1.1/32,
BGP(4): 192.168.100.3 Net 65000:1:10.100.1.1/32 from ibgp-pece 10.1.1.4 format
ATTR_SET
BGP(4): (base) 192.168.100.3 send UPDATE (format) 65000:1:10.100.1.1/32, next
192.168.100.1, label 16, metric 0, path Local, extended community RT:1:1
BGP: 192.168.100.3 Next hop is our own address 192.168.100.1
BGP: 192.168.100.3 Route Reflector cluster loop; Received cluster-id 192.168.100.1
BGP: 192.168.100.3 RR in same cluster. Reflected update dropped

RR#
BGP(4): 192.168.100.1 rcvd UPDATE w/ attr: nexthop 192.168.100.1, origin i, localpref
100, originator 10.100.1.1, clusterlist 192.168.100.1, extended community RT:1:1,
[ATTR_SET attribute:  originator AS 65000, origin IGP, aspath , med 0, localpref 200,
cluster list 192.168.100.1 , originator 10.100.1.1]
BGP(4): 192.168.100.1 rcvd 65000:1:10.100.1.1/32, label 16
RT address family is not configured. Can't create RTC route
BGP(4): (base) 192.168.100.1 send UPDATE (format) 65000:1:10.100.1.1/32, next
```

```
192.168.100.1, label 16, metric 0, path Local, extended community RT:1:1

PE2#
BGP(4): 192.168.100.3 rcvd UPDATE w/ attr: nexthop 192.168.100.1, origin i, localpref
100, originator 10.100.1.1, clusterlist 192.168.100.3 192.168.100.1, extended community
RT:1:1, [ATTR_SET attribute:  originator AS 65000, origin IGP, aspath , med 0, localpref
200, cluster list 192.168.100.1 , originator 10.100.1.1]
BGP(4): 192.168.100.3 rcvd 65000:1:10.100.1.1/32, label 16
RT address family is not configured. Can't create RTC route
BGP(4): Revise route installing 1 of 1 routes for 10.100.1.1/32 -> 192.168.100.1
(customer1) to customer1 IP table
BGP(4): 10.1.2.5 NEXT_HOP is set to self for net 65000:2:10.100.1.1/32,
```

Remarque: PE1 a reçu sa propre mise à jour du rr et l'a puis relâchée. C'est parce que PE1 et PE2 sont dans le même groupe de mise à jour sur le rr.

Remarque: Si vous voulez vider le message complet de mise à jour dans l'hexadécimal, utilisez le mot clé de **détail** pour la commande de **mises à jour BGP de débogage**.

```
PE2#   debug bgp vpnv4 unicast updates detail
BGP updates debugging is on with detail for address family: VPNv4 Unicast

PE2#
BGP(4): 192.168.100.3 rcvd UPDATE w/ attr: nexthop 192.168.100.1, origin i,
localpref 100, originator 10.100.1.1, clusterlist 192.168.100.3 192.168.100.1,
extended community RT:1:1, [ATTR_SET attribute:  originator AS 65000, origin IGP,
aspath , med 0, localpref 200, cluster list 192.168.100.1 , originator 10.100.1.1]
BGP(4): 192.168.100.3 rcvd 65000:1:10.100.1.1/32, label 17
RT address family is not configured. Can't create RTC route
BGP: 192.168.100.3 rcv update length 125
BGP: 192.168.100.3 rcv update dump: FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
0090 0200 00
PE2#00 7980 0E21 0001 800C 0000 0000 0000 0000 C0A8 6401 0078 0001 1100 00FD E800
0000 010A 6401 0140 0101 0040 0200 4005 0400 0000 64C0 1008 0002 0001 0000 0001 800A
08C0 A864 03C0 A864 0180 0904 0A64 0101 C080 2700 00FD E840 0101 0040 0200 8004 0400
0000 0040 0504 0000 00C8 800A 04C0 A864 0180 0904 0A64 0101
BGP(4): Revise route installing 1 of 1 routes for 10.100.1.1/32 -> 192.168.100.1
(customer1) to customer1 IP table
BGP(4): 10.1.2.5 NEXT_HOP is set to self for net 65000:2:10.100.1.1/32,
```

Prochaine manipulation de saut

Le next-hop-self doit être configuré sur les Routeurs de PE pour cette caractéristique. La raison pour ceci est que normalement le prochain-saut est transporté sans changement avec l'iBGP. Cependant, ici il y a deux réseaux indépendants : le réseau VPN et le réseau de fournisseur de services, qui protocoles distincts d'Interior Gateway de passage (IGP). Par conséquent, la mesure d'IGP ne peut pas être facilement comparée et utilisée pour le meilleur calcul de chemin entre les deux réseaux. L'approche choisie par RFC 6368 est d'avoir le next-hop-self obligatoire pour la session d'iBGP vers le CE, qui évite toute la question ensemble précédemment décrite. Un avantage est que les sites de VRF peuvent exécuter différents IGP avec cette approche.

RD

RFC 6368 mentionne que les différents sites de VRF d'il est recommandé que du même VPN utilisent (le seul) RDS différent. Dans le Cisco IOS, c'est obligatoire pour cette caractéristique.

caractéristique de l'iBGP PE-CE avec le local-as

Référez-vous à la figure 2. Le VPN customer1 a ASN 65001.

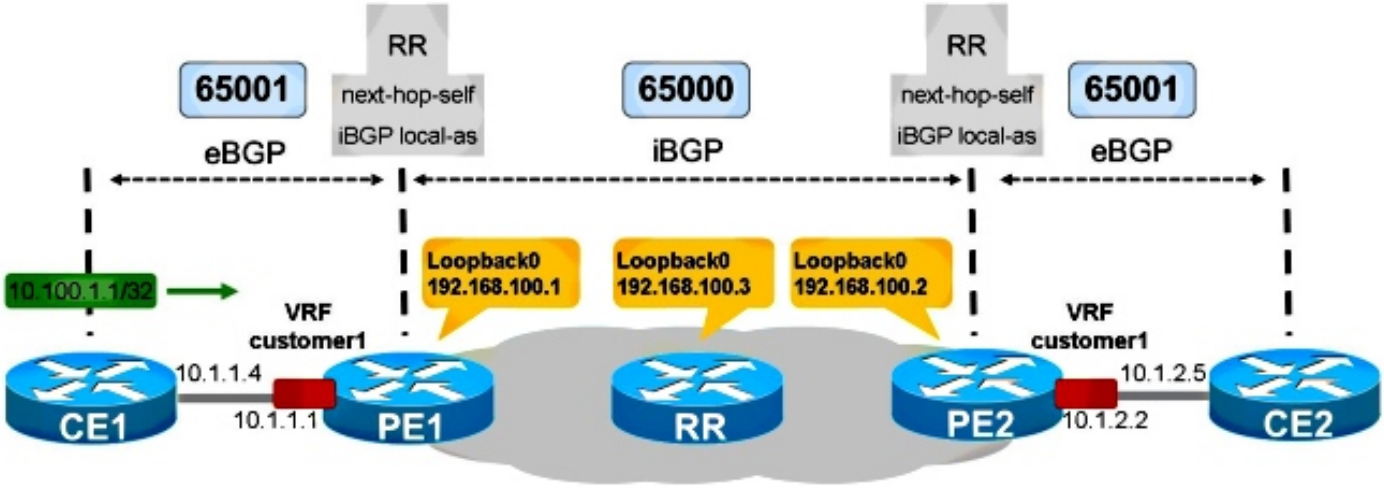


Figure 2

CE1 est dedans EN TANT QUE 65001. Afin de faire ce BGP interne à partir du point de vue de PE1, il a besoin de la caractéristique de local-as d'iBGP.

```
CE1

router bgp 65001
 bgp log-neighbor-changes
 network 10.100.1.1 mask 255.255.255.255
 neighbor 10.1.1.1 remote-as 65001

PE1

router bgp 65000
 bgp log-neighbor-changes
 neighbor 192.168.100.3 remote-as 65000
 neighbor 192.168.100.3 update-source Loopback0
 !
 address-family vpnv4
 neighbor 192.168.100.3 activate
 neighbor 192.168.100.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf customer1
 neighbor 10.1.1.4 remote-as 65001
 neighbor 10.1.1.4 local-as 65001
 neighbor 10.1.1.4 activate
 neighbor 10.1.1.4 internal-vpn-client
 neighbor 10.1.1.4 route-reflector-client
 neighbor 10.1.1.4 next-hop-self
 exit-address-family
```

PE2 et CE2 sont configurés pareillement.

PE1 voit le préfixe BGP comme affiché ici :

```
PE1#show bgp vpnv4 unicast all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 41
Paths: (2 available, best #1, table customer1)
```

```
 Advertised to update-groups:
    5
 Refresh Epoch 1
 Local, (Received from ibgp-pece RR-client)
   10.1.1.4 (via vrf customer1) from 10.1.1.4 (10.100.1.1)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     mpls labels in/out 18/nolabel
     rx pathid: 0, tx pathid: 0x0
 Refresh Epoch 1
 Local, (Received from ibgp-pece RR-client), (ibgp sourced)
   10.1.1.4 (via vrf customer1) from 10.1.1.4 (10.100.1.1)
     Origin IGP, localpref 100, valid, internal
     Extended Community: RT:1:1
     mpls labels in/out 18/nolabel
     rx pathid: 0, tx pathid: 0
```

Le préfixe est un BGP interne.

PE2 voit ceci :

```
PE2#show bgp vpnv4 unicast all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 33
Paths: (1 available, best #1, no table)
 Not advertised to any peer
 Refresh Epoch 5
 Local
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, localpref 100, valid, internal, best
     Extended Community: RT:1:1
     Originator: 10.100.1.1, Cluster list: 192.168.100.3, 192.168.100.1
     ATTR_SET Attribute:
       Originator AS 65001
       Origin IGP
       Aspath
       Med 0
       LocalPref 200
       Cluster list
       192.168.100.1,
       Originator 10.100.1.1
     mpls labels in/out nolabel/18
     rx pathid: 0, tx pathid: 0x0
BGP routing table entry for 65000:2:10.100.1.1/32, version 34
Paths: (1 available, best #1, table customer1)
 Advertised to update-groups:
    5
 Refresh Epoch 2
 Local, imported path from 65000:1:10.100.1.1/32 (global)
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator AS(ibgp-pece): 65001
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
     mpls labels in/out nolabel/18
     rx pathid: 0, tx pathid: 0x0
```

Le créateur DE MÊME QUE 65001, qui est COMME utilisé quand le préfixe est envoyé de PE2 à CE2. Ainsi, DE MÊME QU'est préservée, et ainsi est la préférence locale dans cet exemple.

```
CE2#show bgp ipv4 unicast 10.100.1.1/32
BGP routing table entry for 10.100.1.1/32, version 3
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   10.1.2.2 from 10.1.2.2 (192.168.100.2)
```

```
    Origin IGP, metric 0, localpref 200, valid, internal, best
    Originator: 10.100.1.1, Cluster list: 192.168.100.2, 192.168.100.1
    rx pathid: 0, tx pathid: 0x0
```

Vous voyez des **gens du pays** au lieu de l'COMME chemin. Ceci signifie que c'est une route BGP interne provenue EN TANT QUE 65001, qui est également l'ASN configuré du routeur CE2. Tous les attributs BGP ont été pris de l'attribut ATTR_SET. Ceci adhère aux règles pour l'affaire 1 dans la section suivante.

Règles pour l'échange d'artère entre différents sites de VRF

L'ATTR_SET contient le créateur en date du VRF d'origine. Ceci commençant COMME est vérifié par le PE distant, quand il retire l'ATTR_SET avant qu'il envoie le préfixe au routeur CE.

Cas 1 : Si le commencement EN TANT QU'apparie configuré QUANT au routeur CE, alors les attributs BGP sont pris de l'attribut ATTR_SET quand le PE importe le chemin dans le VRF de destination.

Cas 2 : Si le commencement DE MÊME QUE n'apparie pas configuré QUANT au routeur CE, alors à l'ensemble d'attributs pour le chemin construit sont pris comme affiché ici :

1. Les attributs de chemin sont placés aux attributs contenus dans l'attribut ATTR_SET.

2. Les attributs d'iBGP-particularité sont jetés (LOCAL_PREF, CRÉATEUR, et CLUSTER_LIST).

3. **Le numéro de système autonome d'origine** contenu dans l'attribut ATTR_SET est ajouté à l'AS_PATH et suit au début les règles qui s'appliquent à un BGP externe scrutant entre la source et la destination Ass.

4. Si l'Autonomous System associé avec le VRF est identique que l'Autonomous System de fournisseur VPN et l'attribut as_path de l'artère VPN n'est pas vide, elle sera ajoutée au début à l'attribut as_path de l'artère de VRF.

   Référez-vous au schéma 3. CE1 et PE1 ont EN TANT QUE 65000 et sont configurés avec la configuration de l'iBGP PE-CE. CE2 a ASN 65001. Ceci signifie qu'il y a eBGP entre PE2 et CE2.

Figure 3

PE2 voit l'artère comme suit :

```
PE2#show bgp vpnv4 unicast all 10.100.1.1/32
BGP routing table entry for 65000:1:10.100.1.1/32, version 43
Paths: (1 available, best #1, no table)
 Not advertised to any peer
 Refresh Epoch 6
 Local
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, localpref 100, valid, internal, best
```

```
     Extended Community: RT:1:1
     Originator: 10.100.1.1, Cluster list: 192.168.100.3, 192.168.100.1
     ATTR_SET Attribute:
       Originator AS 65000
       Origin IGP
       Aspath
       Med 0
       LocalPref 200
       Cluster list
       192.168.100.1,
       Originator 10.100.1.1
     mpls labels in/out nolabel/17
     rx pathid: 0, tx pathid: 0x0
BGP routing table entry for 65000:2:10.100.1.1/32, version 44
Paths: (1 available, best #1, table customer1)
 Advertised to update-groups:
    6
 Refresh Epoch 6
 Local, imported path from 65000:1:10.100.1.1/32 (global)
   192.168.100.1 (metric 21) (via default) from 192.168.100.3 (192.168.100.3)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator AS(ibgp-pece): 65000
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
     mpls labels in/out nolabel/17
     rx pathid: 0, tx pathid: 0x0
```

C'est le préfixe comme vu sur CE2 :

```
CE2#show bgp ipv4 unicast 10.100.1.1/32
BGP routing table entry for 10.100.1.1/32, version 5
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 65000
   10.1.2.2 from 10.1.2.2 (192.168.100.2)
     Origin IGP, localpref 100, valid, external, best
     rx pathid: 0, tx pathid: 0x0
```

C'est l'affaire 2. **Le numéro de système autonome d'origine** contenu dans l'attribut ATTR_SET est ajouté à l'AS_PATH par PE2 et suit au début les règles qui s'appliquent à un eBGP scrutant entre la source et la destination AS. Les attributs d'iBGP-particularité sont ignorés par PE2 quand il crée l'artère à annoncer à CE2. Ainsi, la préférence locale est 100 et non 200 (comme vu dans l'attribut ATTR_SET).

Réflexion de Vrf-Lite de Ce-à-CE

Référez-vous à la figure 4.

Figure 4

La figure 4 affiche un routeur CE supplémentaire, CE3, connecté à PE1. CE1 et CE3 chacun des deux sont connectés à PE1 sur le même exemple de VRF : customer1. Ceci signifie que CE1 et CE3 sont des Routeurs de la CE de Multi-VRF (également connus sous le nom de Vrf-Lite) de PE1. PE1 se met comme prochain-saut quand il annonce les préfixes de CE1 à CE3. Dans le cas que ce comportement n'est pas voulu, vous pourriez configurer le **voisin 10.1.3.6 next-hop-unchanged** sur PE1. Afin de configurer ceci, vous devez retirer le **next-hop-self de 10.1.3.6 de voisin** sur PE1. Alors CE3 voit les artères de CE1 avec CE1 pour être le prochain-saut pour ces préfixes BGP. Afin de faire ce travail, vous avez besoin des artères pour ces bgp next-hop dans la table de routage de CE3. Vous avez besoin d'un protocole de routage dynamique (IGP) ou des

artères de charge statique sur CE1, PE1, et CE3 afin de veiller les Routeurs pour avoir une artère pour chaque autres des adresses IP de prochain-saut. Cependant, il y a un problème avec cette configuration.

La configuration sur PE1 est :

```
router bgp 65000
 !
 address-family ipv4 vrf customer1
 neighbor 10.1.1.4 remote-as 65000
 neighbor 10.1.1.4 activate
 neighbor 10.1.1.4 internal-vpn-client
 neighbor 10.1.1.4 route-reflector-client
 neighbor 10.1.1.4 next-hop-self
 neighbor 10.1.3.6 remote-as 65000
 neighbor 10.1.3.6 activate
 neighbor 10.1.3.6 internal-vpn-client
 neighbor 10.1.3.6 route-reflector-client
 neighbor 10.1.3.6 next-hop-unchanged
 exit-address-family
```

Le préfixe de CE1 est vu bien sur CE3 :

```
CE3#show bgp ipv4 unicast 10.100.1.1
BGP routing table entry for 10.100.1.1/32, version 9
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   10.1.1.4 from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
     rx pathid: 0, tx pathid: 0x0
```

Cependant, le préfixe de CE2 est vu sur CE3 comme affiché ici :

```
CE3#show bgp ipv4 unicast 10.100.1.2
BGP routing table entry for 10.100.1.2/32, version 0
Paths: (1 available, no best path)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   192.168.100.2 (inaccessible) from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 100, valid, internal
     Originator: 10.100.1.2, Cluster list: 192.168.100.1, 192.168.100.2
     rx pathid: 0, tx pathid: 0
```

Le bgp next-hop est **192.168.100.2**, l'adresse IP de bouclage de PE2. PE1 n'a pas réécrit le bgp next-hop à lui-même quand il a annoncé le préfixe 10.100.1.2/32 à CE3. Ceci rend ce préfixe inutilisable sur CE3.

Ainsi, dans le cas d'un mélange de la caractéristique de l'iBGP PE-CE à travers MPLS-VPN et iBGP Vrf-Lite, vous devez s'assurer que vous avez toujours le next-hop-self sur les Routeurs de PE.

Vous ne pouvez pas conserver le prochain-saut quand un routeur PE est des rr qui réfléchissent des artères d'iBGP d'un CE à un autre CE à travers des interfaces de VRF localement sur le PE. Quand vous exécutez l'iBGP PE-CE à travers un réseau VPN MPLS, vous devez utiliser l'interne-VPN-**client** pour les sessions d'iBGP vers les Routeurs de la CE. Quand vous avez plus d'un CE local dans un VRF sur un routeur PE, alors vous devez garder le **next-hop-self** pour ces pairs BGP.

Vous pourriez regarder des route-map afin de placer le prochain-saut à l'individu pour des préfixes reçus d'autres Routeurs de PE, mais pas pour des préfixes réfléchis d'autres Routeurs local-connectés de la CE. Cependant, il n'est pas actuellement pris en charge pour placer le prochain-saut à l'individu dans une mise en correspondance de route sortante. Cette configuration est affichée ici :

```
router bgp 65000

 address-family ipv4 vrf customer1
 neighbor 10.1.1.4 remote-as 65000
 neighbor 10.1.1.4 activate
 neighbor 10.1.1.4 internal-vpn-client
 neighbor 10.1.1.4 route-reflector-client
 neighbor 10.1.1.4 next-hop-self
 neighbor 10.1.3.6 remote-as 65000
 neighbor 10.1.3.6 activate
 neighbor 10.1.3.6 internal-vpn-client
 neighbor 10.1.3.6 route-reflector-client
 neighbor 10.1.3.6 route-map NH-setting out
 exit-address-family

ip prefix-list PE-loopbacks seq 10 permit 192.168.100.0/24 ge 32
!

route-map NH-setting permit 10
 description set next-hop to self for prefixes from other PE routers
 match ip route-source prefix-list PE-loopbacks
 set ip next-hop self
!

route-map NH-setting permit 20
 description advertise prefixes with next-hop other than the prefix-list in
route-map entry 10 above
!
```

Cependant, ceci n'est pas pris en charge :

```
PE1(config)#route-map NH-setting permit 10
PE1(config-route-map)# set ip next-hop self
% "NH-setting" used as BGP outbound route-map, set use own IP/IPv6 address for the nexthop not
supported
```

Un Cisco IOS plus ancien sur le routeur PE

Si PE1 exécute un logiciel plus ancien de Cisco IOS qui manque de l'iBGP PE-CE de caractéristique, alors PE1 ne se place jamais comme prochain-saut pour les préfixes reflétés d'iBGP. Ceci signifie que le préfixe reflété BGP (10.100.1.1/32) de CE1 (10.100.1.1) à CE2 - par l'intermédiaire de PE1- aurait CE1 (10.1.1.4) comme prochain-saut.

```
CE3#show bgp ipv4 unicast 10.100.1.1
BGP routing table entry for 10.100.1.1/32, version 32
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   10.1.1.4 from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 200, valid, internal, best
     Originator: 10.100.1.1, Cluster list: 192.168.100.1
     rx pathid: 0, tx pathid: 0x0
```

Le préfixe de CE2 (10.100.1.2/32) est vu avec PE2 comme prochain-saut, parce que PE1 ne fait

pas le next-hop-self pour ce préfixe l'un ou l'autre :

```
CE3#show bgp ipv4 unicast 10.100.1.2
BGP routing table entry for 10.100.1.2/32, version 0
Paths: (1 available, no best path)
 Not advertised to any peer
 Refresh Epoch 1
 Local
   192.168.100.2 (inaccessible) from 10.1.3.1 (192.168.100.1)
     Origin IGP, localpref 100, valid, internal
     Originator: 10.100.1.2, Cluster list: 192.168.100.1, 192.168.100.3, 192.168.100.2
     ATTR_SET Attribute:
       Originator AS 65000
       Origin IGP
       Aspath
       Med 0
       LocalPref 100
       Cluster list
       192.168.100.2,
       Originator 10.100.1.2
     rx pathid: 0, tx pathid: 0
```

Pour que la caractéristique de l'iBGP PE-CE fonctionne correctement, tous les Routeurs de PE pour le VPN où la caractéristique est activée doivent avoir le code pour prendre en charge la caractéristique et pour avoir la fonction activée.

Next-hop-self pour l'eBGP sur le VRF

Référez-vous à la figure 5.

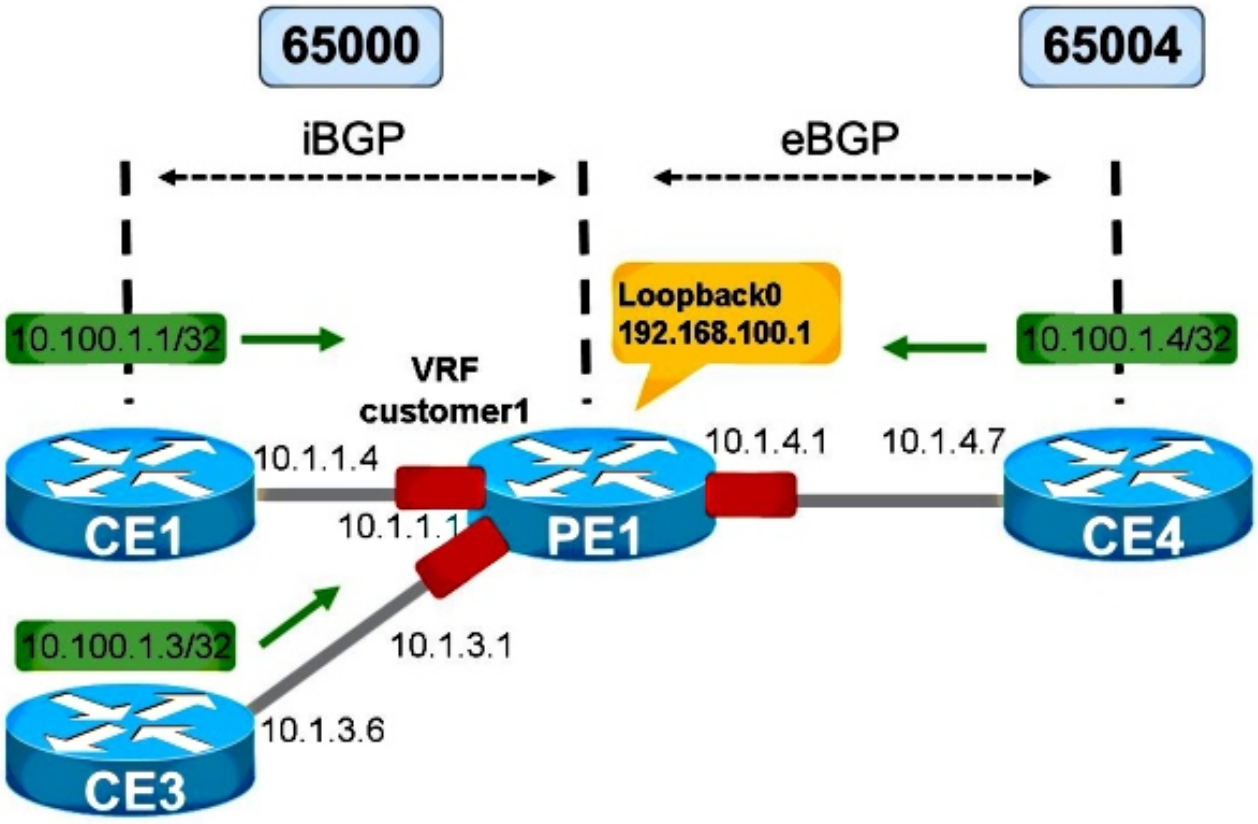


Figure 5

La figure 5 affiche une installation de Vrf-Lite. La session de PE1 vers CE4 est eBGP. La session de PE1 vers CE3 est toujours iBGP.

Pour des préfixes d'eBGP, le prochain-saut est toujours placé à l'individu quand il annonce les préfixes vers un voisin d'iBGP sur le VRF. C'est indépendamment du fait si la session vers le voisin d'iBGP à travers le VRF a le next-hop-self réglé ou pas.

Dans la figure 5, CE3 voit les préfixes de CE4 avec PE1 comme prochain-saut.

```
CE3#show bgp ipv4 unicast 10.100.1.4
BGP routing table entry for 10.100.1.4/32, version 103
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 65004
   10.1.3.1 from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 100, valid, internal, best
     rx pathid: 0, tx pathid: 0x0
```

Ceci se produit avec le next-hop-self sur PE1 vers CE3 ou sans.

Si les interfaces sur PE1 vers CE3 et CE4 sont non dans un VRF, mais dans le contexte global, le next-hop-self vers CE3 fait fait une différence.

Sans next-hop-self sur PE1 vers CE3, vous voyez :

```
PE1#show bgp vrf customer1 vpnv4 unicast neighbors 10.1.3.6
BGP neighbor is 10.1.3.6,  vrf customer1,  remote AS 65000, internal link
...
 For address family: VPNv4 Unicast
 Translates address family IPv4 Unicast for VRF customer1
 Session: 10.1.3.6
 BGP table version 1, neighbor version 1/0
 Output queue size : 0
 Index 12, Advertise bit 0
 Route-Reflector Client
 12 update-group member
 Slow-peer detection is disabled
 Slow-peer split-update-group dynamic is disabled
 Interface associated: (none)
```

Bien que le next-hop-self soit implicitement activé, la sortie n'indique pas ceci.

Avec le next-hop-self sur PE1 vers CE3, vous voyez :

```
PE1#show bgp vrf customer1 vpnv4 unicast neighbors 10.1.3.6
BGP neighbor is 10.1.3.6,  vrf customer1,  remote AS 65000, internal link
..
 For address family: VPNv4 Unicast
...
 NEXT_HOP is always this router for eBGP paths
```

Considérant que, si les interfaces vers CE3 et CE4 sont dans un contexte global, le prochain-saut pour des préfixes de CE4 est CE4 lui-même quand le next-hop-self n'est pas configuré :

```
CE3#show bgp ipv4 unicast 10.100.1.4
BGP routing table entry for 10.100.1.4/32, version 124
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 65004
   10.1.4.7 from 10.1.3.1 (192.168.100.1)
```

```
     Origin IGP, metric 0, localpref 100, valid, internal, best
     rx pathid: 0, tx pathid: 0x0
```

Pour le next-hop-self sur PE1 vers CE3 :

```
CE3#show bgp ipv4 unicast 10.100.1.4
BGP routing table entry for 10.100.1.4/32, version 125
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 1
 65004
   10.1.3.1 from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 100, valid, internal, best
     rx pathid: 0, tx pathid: 0x0
```

Ceci a été fait a basé sur RFC 4364.

Si vous voulez ne pas placer le next-hop-self pour des préfixes d'eBGP vers une session d'iBGP à travers une interface de VRF, vous devez configurer **next-hop-unchanged**. Le soutien de ceci s'est seulement produit avec l'ID de bogue Cisco CSCuj11720.

```
router bgp 65000
...
 address-family ipv4 vrf customer1
 neighbor 10.1.1.4 remote-as 65000
 neighbor 10.1.1.4 activate
 neighbor 10.1.1.4 route-reflector-client
 neighbor 10.1.3.6 remote-as 65000
 neighbor 10.1.3.6 activate
 neighbor 10.1.3.6 route-reflector-client
 neighbor 10.1.3.6 next-hop-unchanged
 neighbor 10.1.4.7 remote-as 65004
 neighbor 10.1.4.7 activate
 exit-address-family
```

Maintenant, CE3 voit CE4 comme prochain-saut pour les préfixes annoncés par CE4 :

```
CE3#show bgp ipv4 unicast 10.100.1.4
BGP routing table entry for 10.100.1.4/32, version 130
Paths: (1 available, best #1, table default)
 Not advertised to any peer
 Refresh Epoch 3
 65004
   10.1.4.7 from 10.1.3.1 (192.168.100.1)
     Origin IGP, metric 0, localpref 100, valid, internal, best
     rx pathid: 0, tx pathid: 0x0
```

Si vous essayez de configurer le mot clé **next-hop-unchanged** pour la session d'iBGP vers CE3 sur code de Cisco IOS avant l'ID de bogue Cisco CSCuj11720, vous rencontrez cette erreur :

```
PE1(config-router-af)#neighbor 10.1.3.6 next-hop-unchanged
%BGP: Can propagate the nexthop only to multi-hop EBGP neighbor
```

Après l'ID de bogue Cisco CSCuj11720, le mot clé **next-hop-unchanged** est valide pour les voisins de multi-alimentation d'eBGP et les voisins de Vrf-Lite d'iBGP.