

Contrainte de cible d'artère

Contenu

[Introduction](#)

[But de contrainte de cible d'artère](#)

[Comportement sans RTC](#)

[Configuration RTC](#)

[Configuration de PE](#)

[Configuration rr](#)

[Comportement de RTC](#)

[PE](#)

[Rr](#)

[L'artère régénèrent la manipulation](#)

[Informations connexes](#)

Introduction

Ce document décrit un mécanisme par lequel l'échange des préfixes VPNv4 et VPNv6 vers des Routeurs de Provider Edge (PE) soit réduit au nécessaire de minimum.

But de contrainte de cible d'artère

Avec le Commutation multiprotocole par étiquette (MPLS) VPN, le réflecteur de pair ou d'artère d'Internal Border Gateway Protocol (iBGP) (rr) envoie tous les préfixes VPN4 et/ou VPN6 aux Routeurs de PE. Le routeur PE relâche les préfixes VPN4/6 pour lesquels il n'y a aucun routage VPN et expédition importants (VRF). C'est un comportement où le rr envoie les préfixes VPN4/6 au routeur PE, du lequel il n'a pas besoin. C'est un gaspillage de capacité de traitement sur le rr et le PE et un gaspillage de bande passante.

Avec la contrainte de cible d'artère (RTC), le rr envoie seulement les préfixes VPN4/6 voulus au PE. « A voulu » signifie que le PE a le VRF important les préfixes spécifiques.

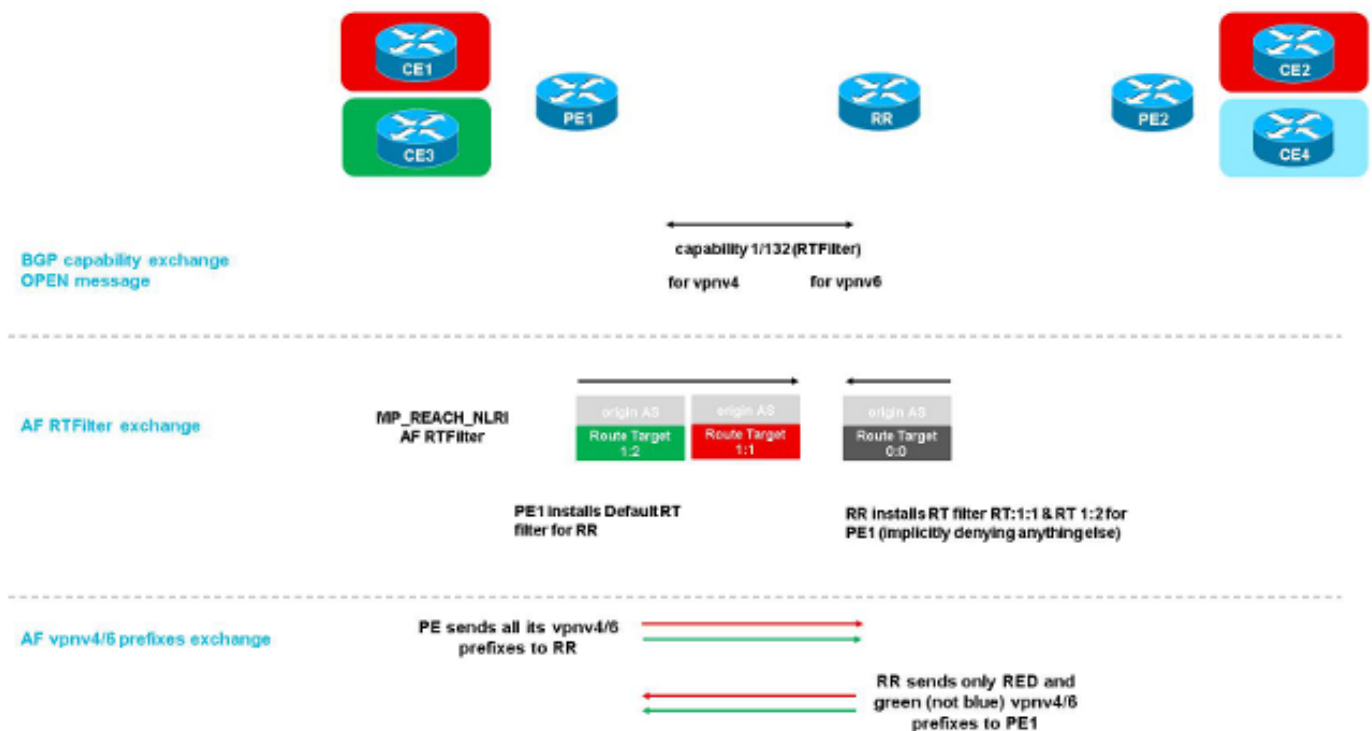
RFC 4684 spécifie le RTC. Le support est par un nouveau rfilter de famille d'adresse pour VPNv4 et VPNv6.

Les informations de filtrage de la cible d'artère (droite) sont obtenues de la liste d'importation VPN droite de tous les vrf sur le routeur PE. Le routeur PE envoie ces informations de filtrage comme mise à jour BGP dans le rfilter de famille d'adresse au rr. Ces informations de filtrage ou l'adhésion droite sont encodées dans les informations d'accessibilité des couches réseau (NLRI) des attributs MP_REACH_NLRI et MP_UNREACH_NLRI.

Le pair de réception BGP traduit ce NLRI en filtre et installe ce filtre sortant sur le pair de envoi. Le pair de réception BGP utilise ce filtre pour décider quels préfixes VPNv4/6 à envoyer ou ne pas envoyer, dépendant sur la présence du rts relié.

Pour que le RTC fonctionne, les deux pairs BGP doivent prendre en charge le RTC. C'est-à-dire, les rr et le PE doivent le prendre en charge. Cependant, le déploiement peut être incrémental, qui signifie que non tous les Routeurs rr et de PE doivent le prendre en charge dans un vont. Le RTC peut fonctionner dans le réseau, avec quelques Routeurs de PE prenant en charge lui et d'autres pas. Sur les Routeurs qui le prennent en charge, le RTC sera déjà en activité. Sur les Routeurs qui ne le prennent en charge pas encore les annonces fonctionneront comme avant, qui est sans RTC (ainsi sans filtrage sortant).

Cette figure affiche le principe du RTC :



Comportement sans RTC

Le rr envoie tous les préfixes VPN4/6 au PE. Le PE relâche ceux pour lesquels il n'y a aucune importation de la droite. Les mises à jour BGP de debug affichent les préfixes abandonnés. En raison REFUSÉ du message « de : la communauté étendue non prise en charge » est donnée.

Un exemple pour l'unicast VPNv4 est comme suit :

```
BGP(4): 10.100.1.3 rcvd UPDATE w/ att: nexthop 10.100.1.1, origin i, localpref 100,
metric 0, originator 10.100.1.1, clusterlist 10.100.1.3, merged path 65003,
AS_PATH , extended community RT:1:2
BGP(4): 10.100.1.3 rcvd 1:2:10.100.1.6/32, label 27 -- DENIED due to: extended
community not supported;
```

Un exemple pour l'unicast VPNv6 est comme suit :

```
BGP(5): 10.100.1.3 rcvd UPDATE w/ attr: nexthop ::FFFF:10.100.1.1, origin i,
localpref 100, metric 0, originator 10.100.1.1, clusterlist 10.100.1.3,
```

```
merged path 65003, AS_PATH , extended community RT:1:2
BGP(5): 10.100.1.3 rcvd [1:2]2001:10:100:1::6/128, label 23 -- DENIED due to:
extended community not supported;
```

Configuration RTC

Configuration de PE

```
vrf definition green
 rd 1:2
  route-target export 1:2
  route-target import 1:2
  !
  address-family ipv4
  exit-address-family
  !
vrf definition red
 rd 1:1
  route-target export 1:1
  route-target import 1:1
  !
  address-family ipv4
  exit-address-family
  !
  address-family ipv6
  exit-address-family

router bgp 1
  bgp log-neighbor-changes
  neighbor 10.100.1.3 remote-as 1
  neighbor 10.100.1.3 update-source Loopback0
  neighbor 10.100.1.4 remote-as 1
  neighbor 10.100.1.4 update-source Loopback0
  !
  address-family vpv4
  neighbor 10.100.1.3 activate
  neighbor 10.100.1.3 send-community both
  neighbor 10.100.1.4 activate
  neighbor 10.100.1.4 send-community both
  exit-address-family
  !
  address-family rtfiler unicast
  neighbor 10.100.1.3 activate
  neighbor 10.100.1.3 send-community extended
  exit-address-family
  !
  address-family ipv4 vrf green
  neighbor 10.1.6.6 remote-as 65003
  neighbor 10.1.6.6 activate
  neighbor 10.1.6.6 send-community both
  exit-address-family
  !
  address-family ipv4 vrf red
  neighbor 10.1.5.5 remote-as 65001
  neighbor 10.1.5.5 activate
  neighbor 10.1.5.5 send-community both
```

exit-address-family

Configuration rr

```
router bgp 1
  bgp log-neighbor-changes
  neighbor 10.100.1.1 remote-as 1
  neighbor 10.100.1.1 update-source Loopback0
  neighbor 10.100.1.2 remote-as 1
  neighbor 10.100.1.2 update-source Loopback0
  !
  address-family vpnv4
  neighbor 10.100.1.1 activate
  neighbor 10.100.1.1 send-community both
  neighbor 10.100.1.1 route-reflector-client
  neighbor 10.100.1.2 activate
  neighbor 10.100.1.2 send-community both
  neighbor 10.100.1.2 route-reflector-client
  exit-address-family
  !
  address-family rtfiler unicast
  neighbor 10.100.1.1 activate
  neighbor 10.100.1.1 send-community both
  neighbor 10.100.1.1 route-reflector-client
  neighbor 10.100.1.1 default-originate
  exit-address-family
```

Comportement de RTC

Quand scruter BGP établit, les pairs permutent la capacité pour le rtfiler, qui est 1/132 (pour VPNV4 et VPNV6).

```
RR1# show bgp rtfiler unicast all neighbors 10.100.1.1
BGP neighbor is 10.100.1.1, remote AS 1, internal link
  BGP version 4, remote router ID 10.100.1.1
  BGP state = Established, up for 00:14:28
  Last read 00:00:01, last write 00:00:56, hold time is 180,
  keepalive interval is 60 seconds
  Neighbor sessions:
    1 active, is not multisession capable (disabled)
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: received
    Address family VPNv4 Unicast: advertised and received
    Address family VPNv6 Unicast: advertised and received
    Address family RT Filter: advertised and received
    Enhanced Refresh Capability: advertised and received
    Multisession Capability:
    Stateful switchover support enabled: NO for session 1
  Message statistics:
    InQ depth is 0
    OutQ depth is 0


```

	Sent	Rcvd
Opens:	1	1
Notifications:	0	0

```

Updates:          6          7
Keepalives:       17         18
Route Refresh:    0          0
Total:            24         30

```

Default minimum time between advertisement runs is 0 seconds

For address family: VPNv4 Unicast

```

Session: 10.100.1.1
BGP table version 65, neighbor version 65/0
Output queue size : 0
Index 19, Advertise bit 1
Route-Reflector Client
19 update-group member
RT Filter activate
Community attribute sent to this neighbor
Slow-peer detection is disabled
Slow-peer split-update-group dynamic is disabled

```

```

                Sent      Rcvd

```

...

For address family: VPNv6 Unicast

```

Session: 10.100.1.1
BGP table version 5, neighbor version 5/0
Output queue size : 0
Index 3, Advertise bit 1
Route-Reflector Client
3 update-group member
RT Filter activate
Community attribute sent to this neighbor
Slow-peer detection is disabled
Slow-peer split-update-group dynamic is disabled

```

...

For address family: RT Filter

```

Session: 10.100.1.1
BGP table version 52, neighbor version 52/0
Output queue size : 0
Index 13, Advertise bit 0
Route-Reflector Client
13 update-group member
NEXT_HOP is always this router for eBGP paths
Community attribute sent to this neighbor
Default information originate, default sent
Slow-peer detection is disabled
Slow-peer split-update-group dynamic is disabled

```

	Sent	Rcvd
Prefix activity:	----	----
Prefixes Current:	1	2 (Consumes 160 bytes)
Prefixes Total:	1	2
Implicit Withdraw:	0	0
Explicit Withdraw:	0	0
Used as bestpath:	n/a	2
Used as multipath:	n/a	0

	Outbound	Inbound
Local Policy Denied Prefixes:	-----	-----
Bestpath from iBGP peer:	2	n/a
Total:	2	0

```

Number of NLRIs in the update sent: max 1, min 0
Last detected as dynamic slow peer: never
Dynamic slow peer recovered: never
Refresh Epoch: 1
Last Sent Refresh Start-of-rib: never

```

```

Last Sent Refresh End-of-rib: never
Last Received Refresh Start-of-rib: never
Last Received Refresh End-of-rib: never

Refresh activity:
Refresh Start-of-RIB      Sent      Rcvd
Refresh End-of-RIB       0         0

```

```

Address tracking is enabled, the RIB does have a route to 10.100.1.1
Connections established 16; dropped 15
Last reset 00:14:28, due to Peer closed the session of session 1
Transport(tcp) path-mtu-discovery is enabled
Graceful-Restart is disabled

```

PE

```
debug bgp all
```

```

BGP: 10.100.1.3 active rcvd OPEN w/ optional parameter type 2 (Capability) len 6
BGP: 10.100.1.3 active OPEN has CAPABILITY code: 1, length 4
BGP: 10.100.1.3 active OPEN has MP_EXT CAP for afi/safi: 1/132
BGP: 10.100.1.3 accept RTC SAFI PE1# show bgp rtfilter unicast rt 1:1
BGP routing table entry for 1:2:1:1, version 3
Paths: (1 available, best #1)
  Advertised to update-groups:
    13
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (10.100.1.1)
      Origin IGP, localpref 100, weight 32768, valid, sourced, local, best
      RT generation: import
      rx pathid: 0, tx pathid: 0x0

```

Le rtfilter AF utilise également des groupes de mise à jour :

```

PE1# show bgp rtfilter unicast all update-group 13
BGP version 4 update-group 13, internal, Address Family: RT Filter
BGP Update version : 12/0, messages 0
Extended-community attribute sent to this neighbor
Topology: global, highest version: 12, tail marker: 12
Format state: Current working (OK, last not in list)
              Refresh blocked (not in list, last not in list)
Update messages formatted 1, replicated 1, current 0, refresh 0, limit 1000
Number of NLRIs in the update sent: max 2, min 0
Minimum time between advertisement runs is 0 seconds
Has 1 member:
  10.100.1.3

```

Vérifiez le RTFilter envoyé par le PE :

```

PE1# show bgp rtfilter unicast all neighbors 10.100.1.3 advertised-routes
BGP table version is 8, local router ID is 10.100.1.1
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
*> 1:2:1:1    0.0.0.0              32768 i

```

```
*> 1:2:1:2          0.0.0.0          32768 i
```

```
Total number of prefixes 2
```

Le codage du préfixe d'adhésion de cible d'artère est de 4 octets pour le numéro de système autonome et de 8 octets pour la cible d'artère, qui est un attribut de la communauté étendue. Dans l'exemple ci-dessus, le préfixe "1:2:1:1" de rfilter est décodé comme suit :

- 1 est le numéro de système autonome
- 2 est le type et le sous-type de l'attribut de la communauté étendue (dans la décimale) (référez-vous à RFC 4360)
- 1:1 est la cible d'artère elle-même

Le rr envoie le filtre par défaut au PE (Rr-client). C'est parce que par conception, le rr veut toutes les artères VPNv4 :

```
BGP(10): (base) 10.100.1.1 send UPDATE (format) 0:0:0:0, next 10.100.1.3,
metric 0, path Local
```

Le PE reçoit et installe un filtre du par défaut droite. Par exemple, il envoie tout au rr :
(mettez au point les mises à jour d'unicast de rfilter BGP)

```
BGP(10): 10.100.1.3 rcvd UPDATE w/ attr: nexthop 10.100.1.3, origin i,
localpref 100, metric 0, community no-export
BGP(10): 10.100.1.3 rcvd 0:0:0:0
BGP(4): Default RT filter installed for 10.100.1.3
```

Le rr reçoit et installe le rfilter de PE1 :
(mettez au point les mises à jour d'unicast de rfilter BGP)

```
BGP(10): 10.100.1.1 rcvd UPDATE w/ attr: nexthop 10.100.1.1, origin i,
localpref 100, metric 0
BGP(10): 10.100.1.1 rcvd 1:2:1:1
BGP(4): 1:2:1:1 RT filter installed for 10.100.1.1
BGP: installing rt filter on 10.100.1.1
BGP: add installed RT filter 1:2:1:1 for 10.100.1.1
BGP(10): 10.100.1.1 rcvd 1:2:1:2
BGP(4): 1:2:1:2 RT filter installed for 10.100.1.1
BGP(4): 1:2:1:2 Initiating an incremental table walk for 10.100.1.1
BGP: installing rt filter on 10.100.1.1
BGP: add installed RT filter 1:2:1:2 for 10.100.1.1
```

Vérifiez les filtres reçus sur le rr :

```
RR1# show bgp vpnv4 unicast all neighbors 10.100.1.1 received rfilters
Address family: VPNv4 Unicast
Extended community filter has: 2 entries with default filtering disabled
Incremental refresh walk mode
Status codes: * valid, S Stale > installed
Route-Target Outbound Filter
*> Extended Community RT:1:2
*> Extended Community RT:1:1
```

Le PE n'installe pas un filtre droite avec le rts spécifique. Le PE a reçu le filtre du par défaut droite du rr, ainsi le PE envoie tous les préfixes VPNv4/v6 :

```
PE1# show bgp vpnv4 unicast all neighbors 10.100.1.3 received rfilters
Address family: VPNv4 Unicast
Extended community filter has: 1 entries with default filtering enabled
Incremental refresh walk mode
```

Afin de créer un filtre du par défaut droite, configurez « le default-originate voisin x.x.x.x » sous le rfilter AF.

Ceci sera créé automatiquement sur le rr pour les peerings de client rr.

Rr

```
router bgp 1

address-family rtfiler unicast
neighbor 10.100.1.1 activate
neighbor 10.100.1.1 send-community both
neighbor 10.100.1.1 route-reflector-client
neighbor 10.100.1.1 default-originate
exit-address-family
```

L'artère régénèrent la manipulation

Quand une nouvelle importation droite est configurée ou quand l'importation droite est retirée, une artère régénèrent est envoyée du PE au rr pour les familles VPNv4/6 d'adresse.

Quand un nouveau VRF est configuré, le PE envoie une artère-régénération au rr.

Dans des les deux cas avec l'active RTC, le rr n'envoie pas tous les préfixes VPNv4/6 au PE. Il envoie seulement le positionnement selon le filtre droite.

Informations connexes

- [Support et documentation techniques - Cisco Systems](#)