

Dépannage de la convergence BGP lente en raison de politiques de routage non optimales sur IOS-XR

Table des matières

[Introduction](#)

[Fond](#)

[Problème](#)

[Solution](#)

[Vérification](#)

[Conclusion](#)

Introduction

Ce document décrit comment diagnostiquer un problème de convergence BGP lente sur les routeurs Cisco IOS® XR qui se produit en raison de politiques de route non optimales.

Fond

Le temps de convergence BGP se compose d'un certain nombre de facteurs. L'une d'elles est le temps de traitement des mises à jour BGP entrantes ou sortantes par les politiques de routage configurées. Il existe plusieurs façons d'écrire une stratégie de routage pour effectuer une tâche spécifique. Un moyen optimal permet d'améliorer la convergence BGP, de minimiser les pertes de trafic potentielles et d'éviter les boucles de routage temporaires. Cisco IOS® XR inclut un outil de profilage qui mesure le temps passé par une politique de routage spécifique afin d'estimer son temps de traitement.

Problème

La convergence BGP lente est parfois le résultat de politiques de route écrites d'une manière non optimale.

Solution

Il existe un outil de profilage de politiques pour les politiques de route qui peut être utilisé sans impact sur les performances afin de mesurer le temps passé dans chaque instruction d'une politique de route à un point d'attache spécifique. Vous pouvez vérifier la durée d'exécution de la stratégie de routage à ce point d'association spécifique. Par défaut, le profilage est activé uniquement pour les statistiques de stratégie de routage agrégé.

```
router bgp 65000
neighbor 10.0.54.6
remote-as 65000
update-source Loopback0
address-family ipv4 unicast
route-policy INGRESS-ROUTE-POLICY in
!
neighbor 10.0.54.11
remote-as 65001
ebgp-multihop 255
update-source Loopback0
address-family ipv4 unicast
route-policy EGRESS-ROUTE-POLICY out
```

<#root>

RP/0/RSP1/CPU0:XR1#

show pc1 protocol bgp speaker-0 neighbor-in-dflt default-IPv4-Uni-10.0.54.6 policy profile

```
Policy profiling data
Policy : INGRESS-ROUTE-POLICY
Pass : 1440233
Drop : 0
# of executions : 1440233
Total execution time : 57095msec      <=====
```

RP/0/RSP1/CPU0:XR1#

show bgp ipv4 unicast neighbors 10.0.54.11 | i Update group

```
Update group: 0.3 Filter-group: 0.5 No Refresh request being processed
RP/0/RSP1/CPU0:XR1#
```

RP/0/RSP1/CPU0:XR1#

show pc1 protocol bgp speaker-0 neighbor-out-dflt default-IPv4-Uni-UpdGrp-0.3-Out policy profile

```
Policy profiling data
Policy : EGRESS-ROUTE-POLICY
Pass : 726751
Drop : 0
# of executions : 726751
Total execution time : 108099msec <=====
RP/0/RSP1/CPU0:XR1#
```

Vous voyez le temps cumulé passé à traiter INGRESS-ROUTE-POLICY et EGRESS-ROUTE-POLICY.

Le profilage peut être appliqué à la politique de routage d'entrée ou de sortie pour n'importe quel point d'attache.

<#root>

RP/0/RSP1/CPU0:XR1#

show pc1 protocol bgp speaker-0 ?

debug-policy	Attachpoint name
permet	Attachpoint name
import	Attachpoint name
export	Attachpoint name
interafi-import	Attachpoint name
source-rt	Attachpoint name
interafi-export	Attachpoint name
retain-rt	Attachpoint name
addpath	Attachpoint name
neighbor-in-dflt	Attachpoint name
neighbor-in-vrf	Attachpoint name
neighbor-out-dflt	Attachpoint name
neighbor-out-vrf	Attachpoint name
orf-dflt	Attachpoint name
orf-vrf	Attachpoint name
dampening-dflt	Attachpoint name
dampening-vrf	Attachpoint name
default-originate-dflt	Attachpoint name
default-originate-vrf	Attachpoint name
clear-policy	Attachpoint name
show-policy-node0_RSP1_CPU0	Attachpoint name
aggregation-dflt	Attachpoint name
aggregation-vrf	Attachpoint name
nexthop	Attachpoint name
allocate-label	Attachpoint name
label-mode	Attachpoint name
l2vpn-import	Attachpoint name
l2vpn-export	Attachpoint name
redistribution-dflt	Attachpoint name
redistribution-vrf	Attachpoint name
rib-install-dflt	Attachpoint name
rib-install-vrf	Attachpoint name
network-dflt	Attachpoint name
network-vrf	Attachpoint name
redistribution-dflt	Attachpoint name
redistribution-vrf	Attachpoint name
rib-install-dflt	Attachpoint name
rib-install-vrf	Attachpoint name
network-dflt	Attachpoint name
network-vrf	Attachpoint name
l2vpn-export-mp2mp	Attachpoint name
l2vpn-export-vfi	Attachpoint name
l2vpn-export-evi	Attachpoint name
l2vpn-export-mspw	Attachpoint name
l2vpn-export-instance	Attachpoint name
WORD	Attachpoint name

RP/0/RSP1/CPU0:XR1#

Vous pouvez effacer les statistiques si nécessaire.

<#root>

RP/0/RSP1/CPU0:XR1#

```
clear pcl protocol bgp speaker-0 neighbor-in-dflt default-IPv4-Uni-10.0.54.6 policy profile
```

```
RP/0/RSP1/CPU0:XR1#
```

```
clear pcl protocol bgp speaker-0 neighbor-out-dflt default-IPv4-Uni-UpdGrp-0.3-Out policy profile
```

Si vous activez debug pcl profile detail, alors vous obtenez des statistiques détaillées par entrée de stratégie de route.

```
<#root>
```

```
RP/0/RSP1/CPU0:XR1#
```

```
debug pcl profile detail
```

Ces résultats ont été collectés après la réception et la propagation de l'échelle complète de la table BGP Internet.

```
<#root>
```

```
RP/0/RSP1/CPU0:XR1#
```

```
show pcl protocol bgp speaker-0 neighbor-in-dflt default-IPv4-Uni-10.0.54.6 policy profile
```

```
Policy profiling data
```

```
Policy : INGRESS-ROUTE-POLICY
```

```
Pass : 720100
```

```
Drop : 0
```

```
# of executions : 720100
```

```
Total execution time : 222788msec <===== about 3.7 minutes to process ingress updates
```

Node Id	Num visited	Exec time	Policy engine operation
PXL_0_1	720100	221796msec	if as-path aspath-match ... then <=====
			<truePath>
PXL_0_3	3525	3msec	set local-preference 150
	3525	0msec	<end-policy/>
			</truePath>
			<falsePath>
PXL_0_2	716575	225msec	set local-preference 50
	716575	82msec	<end-policy/>
			</falsePath>

```
RP/0/RSP1/CPU0:XR1#
```

```
show pcl protocol bgp speaker-0 neighbor-out-dflt default-IPv4-Uni-UpdGrp-0.3-Out policy profile
```

```
Policy profiling data
```

```
Policy : EGRESS-ROUTE-POLICY
```

```
Pass : 720105
```

```
Drop : 0
```

```
# of executions : 720105
```

```
Total execution time : 221975msec <===== about 3.7 minutes to process egress updates
```

Node Id	Num visited	Exec time	Policy engine operation
PXL_0_1	720105	3005msec	if as-path aspath-match ... then <truePath>
PXL_0_5	0	0msec	set med 70
	0	0msec	<end-policy/>
			</truePath>
			<falsePath>
PXL_0_2	720105	218008msec	if as-path aspath-match ... then <=====
			<truePath>
PXL_0_3	25	0msec	set med 80
	25	0msec	<end-policy/>
			</truePath>
			<falsePath>
PXL_0_4	720080	145msec	set med 90
	720080	76msec	<end-policy/>
			</falsePath>
			</falsePath>

RP/0/RSP1/CPU0:XR1#

Comme vous pouvez le voir, les lignes PXL_0_1 pour INGRESS-ROUTE-POLICY et PXL_0_2 pour EGRESS-ROUTE-POLICY sont particulièrement longues et ralentissent la convergence.

Si vous les corrélerez avec les politiques de routage, vous pouvez voir que AS-PATH-SET-11 dans INGRESS-ROUTE-POLICY et AS-PATH-SET-22 dans EGRESS-ROUTE-POLICY sont à l'origine du problème. Chaque ensemble de chemins AS est constitué de 100 lignes d'expression régulière, ce qui est une façon assez inefficace d'écrire une politique puisqu'elle n'exploite aucune optimisation possible ou la puissance de regex.

```
route-policy INGRESS-ROUTE-POLICY
  if as-path in AS-PATH-SET-11 then
    set local-preference 150
  else
    set local-preference 50
  endif
end-policy
```

```
route-policy EGRESS-ROUTE-POLICY
  if as-path in AS-PATH-SET-21 then
    set med 70
  elseif as-path in AS-PATH-SET-22 then
    set med 80
  else
    set med 90
  endif
end-policy
```

```
as-path-set AS-PATH-SET-11
  ios-regex '^65101 65201_',
  ios-regex '^65102 65202_',
  ios-regex '^65103 65203_',
  ios-regex '^65104 65204_',
  ios-regex '^65105 65205_',
```

```

--- removed 90 similar lines ---
ios-regex '^65195 65295_',
ios-regex '^65196 65296_',
ios-regex '^65197 65297_',
ios-regex '^65198 65298_',
ios-regex '^65199 65299_'
end-set

as-path-set AS-PATH-SET-21
ios-regex '^$'
end-set

as-path-set AS-PATH-SET-22
ios-regex '^65169(_65169)*$',
ios-regex '^65392(_65392)*$',
ios-regex '^65133(_65133)*$',
ios-regex '^65231(_65231)*$',
ios-regex '^65161(_65161)*$',
--- removed 90 similar lines ---
ios-regex '^65281(_65281)*$',
ios-regex '^65336(_65336)*$',
ios-regex '^65238(_65238)*$',
ios-regex '^65381(_65381)*$',
ios-regex '^65103(_65103)*$'
end-set

```

Afin d'améliorer les performances de la politique, vous pouvez évaluer la configuration des ensembles de chemins AS avec l'opération de correspondance as-path native au lieu d'une expression régulière. Vous pouvez également utiliser des expressions régulières à l'intérieur de stratégies de routage de manière réduite, ce qui permet de réduire le nombre de lignes d'expression régulière utilisées.

Ce tableau répertorie les critères de correspondance de chemin AS proposés par le langage RPL (Route Policy Language). Les fonctions de correspondance natives utilisent un algorithme de correspondance binaire qui offre de meilleures performances par rapport au moteur de correspondance d'expression régulière. La plupart des scénarios de correspondance ios-regex courants peuvent être écrits avec eux (ou avec leurs combinaisons).

Commande	Description
is-local	Détermine si le routeur (ou un autre routeur au sein de ce système autonome ou de cette confédération) est à l'origine de la route
longueur	Effectue une vérification conditionnelle en fonction de la longueur du chemin AS
neighbor-is	Teste le ou les numéros du système autonome en tête du chemin AS par rapport à une séquence d'une ou de plusieurs valeurs ou paramètres

	intégraux.
provient de	Teste un chemin AS par rapport à la séquence AS depuis le début avec le numéro de système autonome qui a émis une route.
intercommunication	Teste si l'entier ou le paramètre spécifié apparaît n'importe où dans le chemin AS ou si la séquence d'entiers et de paramètres apparaît.
longueur unique	Effectue des vérifications spécifiques en fonction de la longueur du chemin AS en ignorant les doublons

Vérification

Il s'agit de politiques de routage réorganisées, écrites à l'aide de critères de correspondance natifs. Ceci permet de réduire sensiblement le temps de traitement.

```
route-policy INGRESS-ROUTE-POLICY
  if as-path in AS-PATH-SET-11 then
    set local-preference 150
  else
    set local-preference 50
  endif
end-policy
```

```
route-policy EGRESS-ROUTE-POLICY
  if as-path is-local then
    set med 70
  elseif as-path in AS-PATH-SET-22 and as-path unique-length is 1 then
    set med 80
  else
    set med 90
  endif
end-policy
```

```
as-path-set AS-PATH-SET-11
  neighbor-is '65101 65201',
  neighbor-is '65102 65202',
  neighbor-is '65103 65203',
  neighbor-is '65104 65204',
  neighbor-is '65105 65205',
  --- removed 90 similar lines ---
  neighbor-is '65195 65295',
  neighbor-is '65196 65296',
  neighbor-is '65197 65297',
  neighbor-is '65198 65298',
  neighbor-is '65199 65299'
end-set
```

```
as-path-set AS-PATH-SET-22
```

```

originates-from '65169',
originates-from '65392',
originates-from '65133',
originates-from '65231',
originates-from '65161',
--- removed 90 similar lines ---
originates-from '65281',
originates-from '65336',
originates-from '65238',
originates-from '65381',
originates-from '65103'
end-set

```

Ces sorties sont collectées après la réception et la propagation de l'échelle complète de la table BGP Internet.

<#root>

RP/0/RSP1/CPU0:XR1#

show pci protocol bgp speaker-0 neighbor-in-dflt default-IPv4-Uni-10.0.54.6 policy profile

```

Policy profiling data
Policy : INGRESS-ROUTE-POLICY
Pass : 720100
Drop : 0
# of executions : 720100
Total execution time : 9612msec <===== about 10 seconds to process ingress updates

```

Node Id	Num visited	Exec time	Policy engine operation
PXL_0_1	720100	8540msec	if as-path aspath-match ... then <truePath> set local-preference 150 <end-policy/> </truePath> <falsePath>
PXL_0_3	7128	2msec	
	7128	1msec	
PXL_0_2	712972	276msec	set local-preference 50 <end-policy/> </falsePath>
	712972	80msec	

RP/0/RSP1/CPU0:XR1#

show pci protocol bgp speaker-0 neighbor-out-dflt default-IPv4-Uni-UpdGrp-0.3-Out policy profile

```

Policy profiling data
Policy : EGRESS-ROUTE-POLICY
Pass : 720126
Drop : 0
# of executions : 720126
Total execution time : 12399msec <===== about 12 seconds to process egress updates

```

Node Id	Num visited	Exec time	Policy engine operation
PXL_0_1	720126	190msec	if as-path is-local then <truePath> set med 70 <end-policy/>
PXL_0_7	0	0msec	
	0	0msec	

			</truePath>
			<falsePath>
PXL_0_2	720126	11190msec	if as-path aspath-match ... then
			<truePath>
PXL_0_4	262734	65msec	if as-path unique-length is 1 then
			<truePath>
PXL_0_5	25	0msec	set med 80
	25	0msec	<end-policy/>
			</truePath>
			<falsePath>
PXL_0_6	720101	164msec	set med 90
	720101	57msec	<end-policy/>
			</falsePath>
			</truePath>
			<falsePath>
			<reference>
GOTO :			PXL_0_6
			</reference>
			</falsePath>
			</falsePath>

RP/0/RSP1/CPU0:XR1#

Les lignes ios-regex peuvent également être réduites. Cela permet également d'améliorer les performances.

```

route-policy INGRESS-ROUTE-POLICY
  if as-path in (ios-regex '^ (65101_65201|65102_65202|65103_65203|65104_65204|65105_65205|65106_65206|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65111_65211|65112_65212|65113_65213|65114_65214|65115_65215|65116_65216|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65121_65221|65122_65222|65123_65223|65124_65224|65125_65225|65126_65226|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65131_65231|65132_65232|65133_65233|65134_65234|65135_65235|65136_65236|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65141_65241|65142_65242|65143_65243|65144_65244|65145_65245|65146_65246|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65151_65251|65152_65252|65153_65253|65154_65254|65155_65255|65156_65256|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65161_65261|65162_65262|65163_65263|65164_65264|65165_65265|65166_65266|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65171_65271|65172_65272|65173_65273|65174_65274|65175_65275|65176_65276|6
    set local-preference 150
  endif
  if as-path in (ios-regex '^ (65181_65281|65182_65282|65183_65283|65184_65284|65185_65285|65186_65286|6
    set local-preference 150
  else
    set local-preference 50
  endif

```



```
PXL_0_6                                set med 80
--- removed lines ---
GOTO :                                PXL_0_3
                                     </reference>
                                     </falsePath>
```

RP/0/RSP1/CPU0:XR1#

Conclusion

Les performances d'une stratégie de route peuvent être améliorées avec une correspondance as-path native ou des modèles d'expression régulière réduits.

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.