

Préfixes BGP avec numéros AS à 4 octets

Contenu

[Introduction](#)

[Informations générales](#)

[Filtre de numéro AS à 4 octets](#)

[Autre configuration](#)

[Mode d'analyse](#)

[Filtre pour le mode Asplain](#)

Introduction

Cet article décrit comment filtrer les préfixes BGP (Border Gateway Protocol) avec des numéros de système autonome (AS) de 4 octets dans Cisco IOS®.

Informations générales

Les numéros AS à 4 octets ont été introduits dans BGP lorsque le pool de numéros AS gratuits à 2 octets est devenu faible. Cela signifie qu'un numéro de système autonome peut maintenant être un numéro à 2 octets ou à 4 octets. Le document RFC 6793 spécifie des numéros de système autonome de 4 octets. Un numéro de système autonome à 2 octets est un nombre compris entre 1 et 65 535. Un numéro AS à 4 octets est un nombre compris entre 1 et 4 294 967 295.

Le numéro de système autonome à 4 octets peut être représenté en pointillés ou en nombres ordinaires. Un exemple de numéro AS à 4 octets en pointillés est 7.52359. C'est le **point de note** AS. 511111 est un exemple de numéro simple à 4 octets. Il s'agit de l'**analyseur** de notation AS. 7.52359 est identique à 511111.

Un routeur Cisco IOS peut utiliser une notation AS. La valeur par défaut dans les versions les plus récentes de Cisco IOS est simple. Afin d'utiliser la notation AS pointillée, la commande **bgp asnotation dot** peut être utilisée.

Si, pour une raison quelconque, l'administrateur réseau ne souhaite pas envoyer de préfixes BGP avec un ou plusieurs numéros AS de 4 octets dans le chemin d'accès AS vers un homologue BGP externe (eBGP), ces préfixes peuvent être filtrés. Ce document fournit un filtre possible pour atteindre cet objectif. Il peut y avoir quelques raisons pour lesquelles vous pouvez ne pas vouloir qu'un haut-parleur BGP envoie des préfixes BGP avec des numéros AS de 4 octets dans le chemin d'accès AS. Un exemple pourrait être qu'il existe un défaut sur le haut-parleur BGP récepteur qui entraîne un comportement défavorable, qui se produit uniquement pour les préfixes BGP avec des numéros AS de 4 octets dans le chemin d'accès AS.

Filtre de numéro AS à 4 octets

Note: Le filtre de cette section ne peut être utilisé que lorsque BGP sur le routeur s'exécute en mode **asnotation dot**.

Ce filtre peut filtrer les préfixes avec les PATHS AS qui contiennent un ou plusieurs numéros AS de 4 octets.

```
router bgp 1
  bgp asnotation dot
  neighbor 10.1.1.2 remote-as 2

  address-family ipv4
  neighbor 10.1.1.2 activate
  neighbor 10.1.1.2 filter-list 1 out

ip as-path access-list 1 deny .*[0-9]+\.[0-9]+.*
ip as-path access-list 1 permit .*
```

Voici un examen de ce filtre.

- **[0-9]** désigne tout chiffre.
- Le **+** signifie une ou plusieurs occurrences. Donc **[0-9]+** signifie n'importe quel nombre, car il peut y avoir plusieurs chiffres présents.
- **\.** signifie **.** (point). La barre oblique inverse est nécessaire pour s'assurer que le routeur ne traite pas le point comme un caractère spécial mais le traite comme un point. **Les.** est un caractère spécial et signifie qu'il correspond à n'importe quel caractère (le caractère générique).
- Le ***** signifie qu'il correspond à zéro ou plus du précédent. Donc, **.*** ça veut dire qu'il y a une correspondance.
- Le filtre complet **.*[0-9]+\.[0-9]+.*** signifie tout point.

Voici un exemple :

Le routeur 1 (R1) reçoit huit préfixes dans BGP. Le préfixe BGP 10.100.1.1/32 est le seul préfixe avec un chemin d'accès AS qui se compose uniquement de numéros AS de 2 octets. Tous les autres préfixes BGP ont un ou plusieurs numéros AS de 4 octets dans le chemin d'accès AS.

```
R1#show bgp ipv4 unicast
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*> 10.100.1.1/32	10.1.3.4	0			0 4 100 200 300 i
*> 10.100.1.2/32	10.1.3.4	0			0 4 1.34464 3.3392 4.37856 i
*> 10.100.1.3/32	10.1.3.4	0			0 4 1.34464 200 4.37856 i
*> 10.100.1.4/32	10.1.3.4	0			0 4 500 4.37856 600 ?
*> 10.100.1.5/32	10.1.3.4	0			0 4 7.41248 3200 400 65000 ?

```
*> 10.100.1.6/32 10.1.3.4 0 0 4 50 3200 400 9.60176 ?
*> 10.100.1.7/32 10.1.3.4 0 0 4 66 1.57464 77 5.17320 99 ?
*> 10.100.1.8/32 10.1.3.4 0 0 4 7.52359 99 66 99 5.18307
3.37515 99 66 3.37515 99 ?
```

R1 annonce uniquement les préfixes sans numéro AS de 4 octets dans le chemin d'accès AS vers l'homologue BGP 10.1.1.2.

```
R1#show bgp ipv4 unicast neighbors 10.1.1.2 advertised-routes
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*> 10.100.1.1/32	10.1.3.4	0		0 4 100 200 300	i

Total number of prefixes 1

Vous pouvez vérifier quels sont les filtres de la table BGP `.*[0-9]+\.[0-9]+.*` à l'aide de la commande `show ip bgp regexp` :

```
R1#show ip bgp regexp .*[0-9]+\.[0-9]+.*
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*> 10.100.1.2/32	10.1.3.4	0		0 4 1.34464 3.3392 4.37856	i
*> 10.100.1.3/32	10.1.3.4	0		0 4 1.34464 200 4.37856	i
*> 10.100.1.4/32	10.1.3.4	0		0 4 500 4.37856 600	?
*> 10.100.1.5/32	10.1.3.4	0		0 4 7.41248 3200 400 65000	?
*> 10.100.1.6/32	10.1.3.4	0		0 4 50 3200 400 9.60176	?
*> 10.100.1.7/32	10.1.3.4	0		0 4 66 1.57464 77 5.17320 99	?
*> 10.100.1.8/32	10.1.3.4	0		0 4 7.52359 99 66 99 5.18307	?

Le filtre `.*[0-9]+\.[0-9]+.*` filtre tous les préfixes avec un ou plusieurs numéros AS de 4 octets dans l'attribut AS PATH.

Voici une autre façon de vérifier ce que le filtre filtre réellement dans la table BGP :

```
R1#show bgp ipv4 unicast filter-list 1
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*> 10.100.1.1/32	10.1.3.4	0		0 4 100 200 300	i

R2 a reçu uniquement le préfixe 10.100.1.1/32 de R1 en raison du filtre.

R2#show bgp ipv4 unicast

```
BGP table version is 6, local router ID is 10.1.1.2
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
*> 10.100.1.1/32	10.1.1.1			0 1 4	100 200 300 i

Autre configuration

Une autre configuration consiste à utiliser une route-map au lieu d'une liste de filtres :

```
router bgp 1
  bgp asnotation dot
  neighbor 10.1.1.2 remote-as 2

  address-family ipv4
  neighbor 10.1.1.2 route-map no-4byte out

ip as-path access-list 1 deny .*[0-9]+\.[0-9]+.*
ip as-path access-list 1 permit .*

route-map no-4byte permit 10
  match as-path 1
```

Mode d'analyse

Lorsque la commande **bgp asnotation dot** est supprimée, ce qui signifie que l'asnotation est simple, le résultat n'affiche plus les points dans les numéros AS à 4 octets. Cette sortie est identique à celle précédemment affichée, à ceci près que les numéros AS en pointillés sont maintenant explicites.

R1#show bgp ipv4 unicast

```
BGP table version is 9, local router ID is 10.100.1.100
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
*> 10.100.1.1/32	10.1.3.4	0		0 4 100 200 300	i
*> 10.100.1.2/32	10.1.3.4	0		0 4 100000 200000 300000	i
*> 10.100.1.3/32	10.1.3.4	0		0 4 100000 200 300000	i
*> 10.100.1.4/32	10.1.3.4	0		0 4 500 300000 600	?
*> 10.100.1.5/32	10.1.3.4	0		0 4 500000 3200 400 65000	?
*> 10.100.1.6/32	10.1.3.4	0		0 4 50 3200 400 650000	?
*> 10.100.1.7/32	10.1.3.4	0		0 4 66 123000 77 345000 99	?
*> 10.100.1.8/32	10.1.3.4	0		0 4 511111 99 66 99 345987	
234123 99 66 234123 99	?				

La commande BGP show avec l'expression régulière utilisée dans le filtre sur le routeur ne filtre pas les préfixes avec un point dans la table BGP, car il n'y a plus de préfixes avec le format de numéro AS à 4 octets en pointillés dans la table BGP. Les numéros AS à 4 octets sont maintenant

au format simple.

```
R1#show ip bgp regexp .*[0-9]+\.[0-9]+.*
R1#
```

La commande BGP show avec filter-list indique également que le filtre ne fonctionne pas tant que le routeur est en mode clair :

```
R1#show bgp ipv4 unicast filter-list 1
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*>  10.100.1.1/32    10.1.3.4         0           0 4 100 200 300 i
*>  10.100.1.2/32    10.1.3.4         0           0 4 100000 200000 300000 i
*>  10.100.1.3/32    10.1.3.4         0           0 4 100000 200 300000 i
*>  10.100.1.4/32    10.1.3.4         0           0 4 500 300000 600 ?
*>  10.100.1.5/32    10.1.3.4         0           0 4 500000 3200 400 65000 ?
*>  10.100.1.6/32    10.1.3.4         0           0 4 50 3200 400 650000 ?
*>  10.100.1.7/32    10.1.3.4         0           0 4 66 123000 77 345000 99 ?
*>  10.100.1.8/32    10.1.3.4         0           0 4 511111 99 66 99 65509 56
100000 ?
```

Le filtre as-path ne fonctionne pas :

```
R1#show bgp ipv4 unicast neighbors 10.1.1.2 advertised-routes
```

```
BGP table version is 9, local router ID is 10.100.1.100
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
*>  10.100.1.1/32    10.1.3.4         0           0 4 100 200 300 i
*>  10.100.1.2/32    10.1.3.4         0           0 4 100000 200000 300000 i
*>  10.100.1.3/32    10.1.3.4         0           0 4 100000 200 300000 i
*>  10.100.1.4/32    10.1.3.4         0           0 4 500 300000 600 ?
*>  10.100.1.5/32    10.1.3.4         0           0 4 500000 3200 400 65000 ?
*>  10.100.1.6/32    10.1.3.4         0           0 4 50 3200 400 650000 ?
*>  10.100.1.7/32    10.1.3.4         0           0 4 66 123000 77 345000 99 ?
*>  10.100.1.8/32    10.1.3.4         0           0 4 511111 99 66 99 65509 56
100000 ?
```

```
   Network          Next Hop        Metric LocPrf Weight Path
```

Total number of prefixes 8

Filtre pour le mode Asplain

Ce filtre peut être utilisé lorsque BGP sur le routeur s'exécute en mode d'analyse :

```
ip as-path access-list 4 deny _6553[6-9]_
ip as-path access-list 4 deny _655[4-9][0-9]_
ip as-path access-list 4 deny _65[6-9][0-9][0-9]_
```

```
ip as-path access-list 4 deny _[6][6-9][0-9][0-9][0-9]_  
ip as-path access-list 4 deny _[7-9][0-9][0-9][0-9][0-9]_  
ip as-path access-list 4 deny _[1-9][0-9][0-9][0-9][0-9][0-9]+_  
ip as-path access-list 4 permit .*
```

Ou, ici, c'est la même chose, mais plus compacte :

```
ip as-path access-list 4 deny _(6553[6-9]|655[4-9][0-9]|65[6-9][0-9][0-9]|6[6-9]  
[0-9][0-9][0-9]|[7-9][0-9][0-9][0-9][0-9]|[1-9][0-9][0-9][0-9][0-9]+)_  
ip as-path access-list 4 permit .*
```

En raison de ce filtre, tout numéro de système autonome supérieur à 65 535 est filtré. Encore une fois, ceci peut être appliqué au voisin BGP si vous utilisez une liste de filtres ou une route-map.

Note: Le filtrage de cet article ne fonctionne pas sur une session BGP interne (iBGP), qui inclut sur un réflecteur de route.