

Prefixos de BGP Filtros com Números AS de 4 bytes

Contents

[Introduction](#)

[Informações de Apoio](#)

[Filtro de número AS de 4 bytes](#)

[Configuração alternativa](#)

[Modo Asperante](#)

[Filtrar para o modo Asplain](#)

Introduction

Este artigo descreve como filtrar prefixos de BGP (Border Gateway Protocol) com números de AS (Sistema Autônomo) de 4 bytes no Cisco IOS[®].

Informações de Apoio

Os números AS de 4 bytes foram introduzidos no BGP quando o pool de números AS livres de 2 bytes se tornou baixo. Isso significa que um número AS agora pode ser um número de 2 ou 4 bytes. O RFC 6793 especifica números AS de 4 bytes. Um número AS de 2 bytes é um número entre 1 e 65.535. Um número AS de 4 bytes é um número entre 1 e 4.294.967.295.

O número AS de 4 bytes pode ser representado de forma pontilhada ou como um número simples. Um exemplo de um número AS de 4 bytes pontuado é 7.52359. Este é o **ponto** de notação AS. Um exemplo de um número de 4 bytes simples é 511111. Esta é a notação AS **asplain**. 7.52359 é o mesmo que 511111.

Um roteador Cisco IOS pode usar a notação AS. O padrão nas versões mais recentes do Cisco IOS é simples. Para usar a notação AS pontilhada, o comando **bgp asnotation dot** pode ser usado.

Se, por algum motivo, o administrador de rede não quiser enviar prefixos BGP com um ou mais números AS de 4 bytes no caminho AS em direção a um peer BGP externo (eBGP), esses prefixos podem ser filtrados. Este documento fornece um filtro possível para alcançar isso. Pode haver algumas razões pelas quais você pode não querer que um alto-falante BGP envie prefixos BGP com números AS de 4 bytes no caminho AS. Um exemplo pode ser que há um defeito no alto-falante BGP receptor que resulta em comportamento adverso, o que ocorre somente para prefixos BGP com números AS de 4 bytes no caminho AS.

Filtro de número AS de 4 bytes

Note: O filtro nesta seção só pode ser usado quando o BGP no roteador é executado no modo **dot de notação**.

Esse filtro pode filtrar prefixos com AS PATHS que contêm um ou mais números AS de 4 bytes.

```
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 .*
```

Aqui está um exame deste filtro.

- **[0-9]** significa qualquer dígito.
- O **+** significa uma ou mais ocorrências. Então **[0-9]+** significa qualquer número, porque pode haver vários dígitos presentes.
- **\.** significa **.** (ponto). A barra invertida é necessária para garantir que o roteador não trate o ponto como um caractere especial, mas o trate como um ponto. O **.** é um caractere especial e significa corresponder a qualquer caractere (o curinga).
- O ***** significa que corresponde a zero ou mais do que precede. Então, **.*** significa combinar qualquer coisa.
- O filtro completo **.*[0-9]+\.[0-9]+.*** significa qualquer coisa com um ponto nele.

Aqui está um exemplo:

O roteador 1 (R1) recebe oito prefixos no BGP. O prefixo BGP 10.100.1.1/32 é o único prefixo com um caminho AS que consiste em apenas números AS de 2 bytes. Todos os outros prefixos BGP têm um ou mais números AS de 4 bytes no caminho 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 anuncia somente os prefixos sem qualquer número AS de 4 bytes no caminho AS em direção ao peer 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

Você pode verificar o filtro `.*[0-9]+\.[0-9]+.*` filtros da tabela BGP com o comando `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

3.37515 99 66 3.37515 99 ?

O filtro `.*[0-9]+\.[0-9]+.*` filtra todos os prefixos com um ou mais números AS de 4 bytes no atributo AS PATH.

Aqui está outra maneira de verificar o que o filtro realmente filtra da tabela 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 recebeu somente o prefixo 10.100.1.1/32 de R1 devido ao filtro.

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

```

Configuração alternativa

Uma configuração alternativa é usar um mapa de rota em vez de uma lista de filtros:

```

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

```

Modo Asperante

Quando o comando `bgp asnotation dot` é removido, o que significa que a atribuição é uma simples, a saída não mostra mais os pontos nos números AS de 4 bytes. Essa saída é a mesma mostrada anteriormente, exceto que os números AS pontuados agora são asplanetários.

```

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 ?

```

O comando BGP show com a expressão regular usada no filtro no roteador não filtra os prefixos com um ponto na tabela BGP, porque não há mais prefixos com o formato de número AS de 4 bytes pontuado na tabela BGP. Os números AS de 4 bytes estão agora em formato simples.

```

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

```

O comando show do BGP com lista de filtros também mostra que o filtro não funciona enquanto o roteador está em um modo simples:

```
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 ?
```

O filtro as-path não irá funcionar:

```
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

Filtrar para o modo Asplain

Esse filtro pode ser usado quando o BGP no roteador é executado em modo simples:

```
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, aqui está o mesmo, mas mais compacto:

```
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[0-9][0-9][0-9][0-9]|1[0-9][0-9][0-9][0-9]+)_
ip as-path access-list 4 permit .*
```

Por causa desse filtro, qualquer número AS superior a 65.535 é filtrado. Novamente, isso pode ser aplicado ao vizinho BGP se você usar uma lista de filtros ou um mapa de rota.

Note: A filtragem neste artigo não funciona em uma sessão de BGP interno (iBGP), que inclui em um Refletor de rota.