

# 4 バイトの AS 番号による BGP プレフィックスフィルタ

## 目次

[概要](#)

[背景説明](#)

[4 バイト AS 番号のフィルタ処理](#)

[代替の設定](#)

[Asplain モード](#)

[Asplain モードのフィルタ処理](#)

## 概要

この記事では、Cisco IOS® の 4 バイト自律システム ( AS ) 番号による Border Gateway Protocol ( BGP ) のプレフィックスをフィルタ処理する方法について説明します。

## 背景説明

4 バイト AS 番号は、空いている 2 バイト AS 番号のプールが少なくなってきた時点で BGP に導入されました。そのため、現在の AS 番号は 2 バイトまたは 4 バイトの数値になります。RFC 6793 では、4 バイト AS 番号を規定しています。2 バイト AS 番号は、1 から 65,535 までの数値です。4 バイト AS 番号は、1 から 4,294,967,295 までの数値です。

4 バイト AS 番号は、ドット付きの形式で表すことも、数字のみで表すこともできます。ドット付きの 4 バイト AS 番号の例は、7.52359 です。これは、AS 表記方法 `dot` です。数字のみの 4 バイト番号の例は、511111 です。これは、AS 表記方法 `asplain` です。7.52359 は 511111 と同じです。

Cisco IOS ルータでは、どちらかの表記方法を使用できます。最新の Cisco IOS バージョンのデフォルト設定は、`asplain` です。ドット付きの AS 表記方法を使用するには、`bgp asnotation dot` コマンドを使用します。

ネットワーク管理者が何らかの理由で、AS PATH に 4 バイト AS 番号が 1 つ以上含まれる BGP プレフィックスを外部 BGP ( eBGP ) ピアに向けて送信しないようにする必要がある場合は、それらのプレフィックスをフィルタ処理で除外できます。このドキュメントでは、これを実現するフィルタ処理の 1 つについて説明します。AS PATH に 4 バイト AS 番号が含まれる BGP プレフィックスを BGP スピーカーが送信しないようにする理由は、いくつかあります。1 つの例を挙げると、受信側の BGP スピーカーに異常な動作が発生する問題があり、その問題が AS PATH に 4 バイト AS 番号が含まれる BGP プレフィックスに限って発生する場合です。

## 4 バイト AS 番号のフィルタ処理

注: このセクションで説明するフィルタは、ルータの BGP が `asnotation dot` モードで実行している場合にのみ使用できます。

このフィルタは、1 つ以上の 4 バイト AS 番号を保持する AS PATH のプレフィックスを除外します。

```
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 .*
```

次に、このフィルタの調査結果を示します。

- `[0-9]` は、任意の数字を表します。
- `+` は、1 回以上の出現を表します。したがって、`[0-9]+` は、複数の数字の存在が当てはまるため、任意の数値を表すこととなります。
- `\.` は、`.` (ドット) を表します。バックスラッシュは、ルータがドットを特殊文字として処理しないようにして、単なるドットとして処理させるために必要です。`.` は、特殊文字であり、任意の文字と一致することを表します (ワイルドカード)。
- `*` は、直前のものとの 0 回以上の一致を表します。したがって、`.*` は、すべてに一致することを表します。
- フィルタ全体の `.*[0-9]+\.[0-9]+.*` は、ドットが含まれているすべてのものを表します。

次に例を示します。

ルータ 1 (R1) は、BGP で 8 つのプレフィックスを受信します。BGP プレフィックス `10.100.1.1/32` は、2 バイト AS 番号だけで構成された AS PATH を持つ唯一のプレフィックスです。その他すべての BGP プレフィックスには、AS PATH に 1 つ以上の 4 バイト AS 番号が含まれています。

```
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はAS PATHに4バイトAS番号が含まれていないプレフィックスだけをBGPピア10.1.1.2に向けてアドバタイズします。

```
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
```

フィルタ `.*[0-9]+\.[0-9]+.*` で、BGP テーブルから抽出される内容を確認するには、`show ip bgp regexp`を使用します。

```
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 ?
```

フィルタ `.*[0-9]+\.[0-9]+.*` により、AS PATH 属性に1つ以上の4バイトAS番号を含むプレフィックスはすべて除外されます。

次に、このフィルタにより実際にBGPテーブルから抽出される内容を確認する別の方法を示します。

```
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

フィルタ処理により、R2はR1からプレフィックス10.100.1.1/32のみを受信しています。

```
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

## 代替の設定

代替の設定では、filter-list の代わりに route-map を使用します。

```
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
```

## Asplain モード

bgp asnotation dot コマンドが削除されると、asnotation が asplain になり、4 バイト AS 番号のドットは出力に表示されなくなります。この出力は、前に示したものと同じですが、ドット付きの AS 番号が asplain になっています。

```
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 ?

フィルタに使用した正規表現を指定して、ルータで BGP show コマンドを実行しても、BGP テーブル内のドット付きのプレフィックスはフィルタ処理で除外されません。これは、BGP テーブルにはドット付きの 4 バイト AS 番号形式のプレフィックスが存在しないためです。4 バイト

AS 番号は、asplain 形式になっています。

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

ルータが asplain モードの間は、filter-list を指定した BGP show コマンドでもフィルタが動作しないことが示されます。

```
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 ?
```

次に示すように、as-path フィルタは動作しません。

```
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

## Asplain モードのフィルタ処理

このフィルタは、ルータの BGP が asplain モードで動作しているときに使用できます。

```
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 ?

Total number of prefixes 8

次に示すように、同じものを簡潔にまとめられます。

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 ?

Total number of prefixes 8

このフィルタにより、65,535 よりも大きな AS 番号が除外されます。前述したように、これは BGP ネイバーに filter-list または route-map を使用する場合に適用できます。

**注:** この記事のフィルタリングは、ルート リフレクタに含まれる内部 BGP ( iBGP ) セッションでは動作しません。