

Configurar a agregação de rotas BGP em IOS® XE

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações sobre agregação de rota BGP](#)

[Configurar](#)

[Diagrama de Rede](#)

[Configurações](#)

[Exemplo 1](#)

[Exemplo 2](#)

[Exemplo 3](#)

[Exemplo 4](#)

[Exemplo 5](#)

[Verificar](#)

[Troubleshooting](#)

[Cenário 1](#)

[Cenário 2](#)

Introdução

Este documento descreve como configurar a agregação de rota BGP com argumentos opcionais.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento dos tópicos listados:

- Operações básicas do BGP (Border Gateway Protocol)
- Listas de prefixos
- Mapas de rotas

Componentes Utilizados

As informações neste documento são baseadas no software Cisco IOS XE versão 17.x.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações sobre agregação de rota BGP

A agregação de rotas BGP permite combinar várias rotas específicas em uma única rota sumarizada (rota agregada) para reduzir o tamanho da tabela de roteamento e a sobrecarga de anúncio.

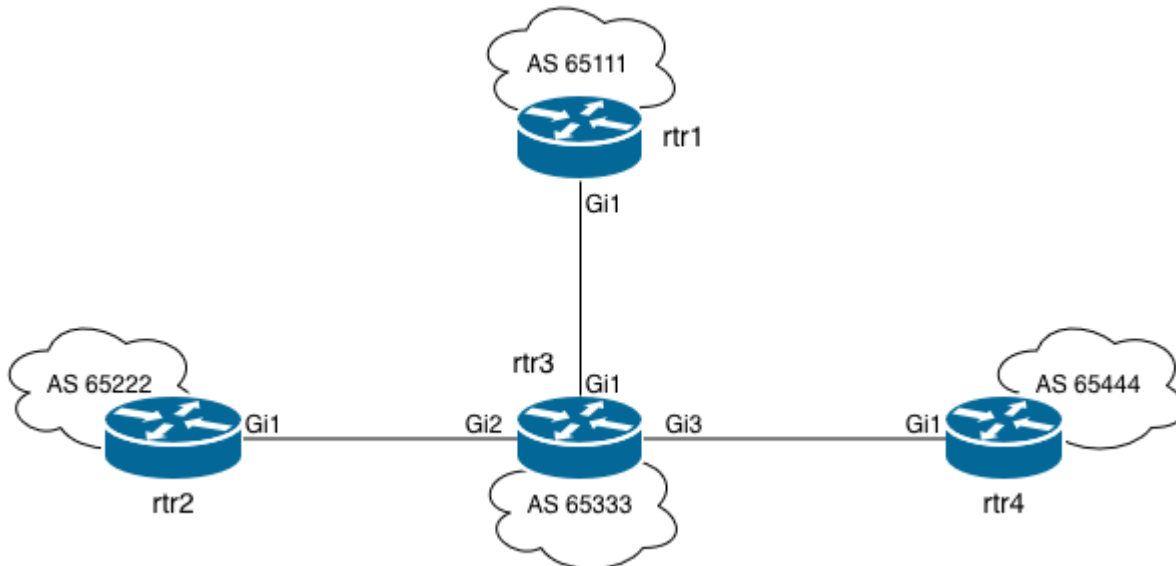
As palavras-chave opcionais incluem:

- `as-set`: Gera um atributo de caminho do conjunto AS contendo todos os números AS das rotas agregadas.
- `somente resumo`: Anuncia somente a rota agregada e suprime rotas mais específicas.
- `suppress-map <map-name>`: Suprime rotas mais específicas selecionadas com base em um mapa de rotas.
- `advertise-map <map-name>`: Anuncia a rota agregada condicionalmente com base em um mapa de rotas.
- `attribute-map <map-name>`: Define atributos na rota agregada usando um mapa de rotas.

Por padrão, o comando `aggregate-address` gera apenas um endereço de sumarização se houver pelo menos mais uma rota específica presente, sem herdar o caminho de AS.

Configurar

Diagrama de Rede



Configurações

Esta é a configuração inicial de rtr3.

```
<#root>
```

```
rtr3#
```

```
show running-config | sec router bgp
```

```
router bgp 65333
bgp log-neighbor-changes
neighbor 10.13.13.1 remote-as 65111
neighbor 10.23.23.2 remote-as 65222
neighbor 10.34.34.4 remote-as 65444
```

Esta é a tabela BGP em rtr3.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 9, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
```

RPKI validation codes: V valid, I invalid, N Not found

```
Network Next Hop Metric LocPrf Weight Path
*> 10.1.1.0/32 10.13.13.1 0 0 65111 i
*> 10.1.1.1/32 10.13.13.1 0 0 65111 i
*> 10.1.1.2/32 10.13.13.1 0 0 65111 i
*> 10.1.1.3/32 10.13.13.1 0 0 65111 i
*> 10.2.2.0/32 10.23.23.2 0 0 65222 i
*> 10.2.2.1/32 10.23.23.2 0 0 65222 i
*> 10.2.2.2/32 10.23.23.2 0 0 65222 i
*> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Observe como rtr3 tem todos os prefixos específicos vindos de rtr1 (AS 65111) e rtr2 (AS 65222). rtr3 anuncia esses prefixos para rtr4 e adiciona o 65333 AS ao atributo AS PATH.

Isso é o que o rtr4 recebe:

```
<#root>
```

```
rtr4#
```

```
show ip bgp
```

```
BGP table version is 9, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.1.1.0/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.1/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.2/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.3/32 10.34.34.3 0 65333 65111 i
*> 10.2.2.0/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.1/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.2/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.3/32 10.34.34.3 0 65333 65222 i
```

Exemplo 1

Configure o BGP para anunciar somente o endereço agregado.

Para o primeiro exemplo prático, você deseja que rtr4 receba apenas o prefixo 10.0.0.0/8.

<#root>

```
rtr3(config)#router bgp 65333
rtr3(config-router)#
```

```
aggregate-address 10.0.0.0 255.0.0.0 summary-only
```

```
rtr3(config-router)#exit
rtr3(config)#
```

Tabela BGP em rtr4 após criar o agregado:

<#root>

rtr4#

```
show ip bgp
```

```
BGP table version is 18, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.0.0.0 10.34.34.3 0 0 65333 i
```

Observe como o caminho AS é 65333 (o roteador que originou o resumo).

Exemplo 2

Para o segundo exemplo prático, você verá como controlar o AS-PATH original.

Esta é a configuração rtr3.

<#root>

```
rtr3#configure terminal
rtr3(config)#router bgp 65333
rtr3(config-router)#
```

```
aggregate-address 10.0.0.0 255.0.0.0 as-set summary-only
```

```
rtr3(config-router)#exit
```

Esta é a tabela BGP em rtr4.

```
<#root>
```

```
rtr4#
```

```
show ip bgp
```

```
BGP table version is 36, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

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

```
*> 10.0.0.0 10.34.34.3 0 0 65333 {65111,65222} i
```

Observe como você tem todos os caminhos AS que rtr3 está agregando.

Exemplo 3

suprimir prefixos BGP usando mapas de rotas.

Para o terceiro exemplo prático, você vai configurar um mapa de rota para filtrar prefixos vindos de rtr1.

```
<#root>
```

```
rtr3(config)#
```

```
ip prefix-list suppress_rtr1 permit 10.1.1.0/24 le 32
```

```
rtr3(config)#route-map
```

```
SUPPRESS-RTR1
```

```
permit 10
rtr3(config-route-map)#match ip address prefix-list
```

```
suppress_rtr1
```

```
rtr3(config)#  
rtr3(config)#router bgp 65333  
rtr3(config-router)#aggregate-address 10.0.0.0 255.0.0.0
```

```
suppress-map SUPPRESS-RTR1
```

```
rtr3(config-router)#end  
rtr3#
```

Esta é a tabela BGP em rtr3.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 114, local router ID is 10.34.34.3  
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,  
t secondary path, L long-lived-stale,  
Origin codes: i - IGP, e - EGP, ? - incomplete  
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path  
*> 10.0.0.0 0.0.0.0 32768 i  
s> 10.1.1.0/32 10.13.13.1 0 0 65111 i  
s> 10.1.1.1/32 10.13.13.1 0 0 65111 i  
s> 10.1.1.2/32 10.13.13.1 0 0 65111 i  
s> 10.1.1.3/32 10.13.13.1 0 0 65111 i  
*> 10.2.2.0/32 10.23.23.2 0 0 65222 i  
*> 10.2.2.1/32 10.23.23.2 0 0 65222 i  
*> 10.2.2.2/32 10.23.23.2 0 0 65222 i  
*> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Observe como somente os prefixos provenientes de rtr1 estão sendo suprimidos.

Esta é a tabela BGP em rtr4.

```
<#root>
```

```
rtr4#
```

```
show ip bgp
```

```
BGP table version is 114, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.0.0.0 10.34.34.3 0 0 65333 i
*> 10.2.2.0/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.1/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.2/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.3/32 10.34.34.3 0 65333 65222 i
rtr4#
```

Exemplo 4

Anuncie uma rota sumarizada somente se um prefixo dentro de um intervalo existir na tabela BGP.

No quarto exemplo, você usará o mesmo mapa de rota configurado anteriormente, que suprime todos os prefixos vindos de rtr1.

```
rtr3(config)#router bgp 65333
rtr3(config-router)#aggregate-address 10.0.0.0 255.0.0.0 advertise-map SUPPRESS-RTR1 summary-only
```

O advertise-map define uma condição, a agregação summary-only é gerada apenas se qualquer prefixo dentro do intervalo 10.1.1.0/24 existir na tabela BGP.

Esta é a tabela BGP em rtr3.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 148, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
```

RPKI validation codes: V valid, I invalid, N Not found

```
Network Next Hop Metric LocPrf Weight Path
*> 10.0.0.0 0.0.0.0 32768 i
s> 10.1.1.0/32 10.13.13.1 0 0 65111 i
s> 10.1.1.1/32 10.13.13.1 0 0 65111 i
s> 10.1.1.2/32 10.13.13.1 0 0 65111 i
s> 10.1.1.3/32 10.13.13.1 0 0 65111 i
s> 10.2.2.0/32 10.23.23.2 0 0 65222 i
s> 10.2.2.1/32 10.23.23.2 0 0 65222 i
s> 10.2.2.2/32 10.23.23.2 0 0 65222 i
s> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Esta é a saída quando não há prefixos que correspondam ao mapa de rota:

```
<#root>
```

```
rtr3#
```

```
show run | section router bgp
```

```
router bgp 65333
 aggregate-address 10.0.0.0 255.0.0.0 summary-only advertise-map SUPPRESS-RTR1
 neighbor 10.13.13.1 remote-as 65111
 neighbor 10.23.23.2 remote-as 65222
 neighbor 10.34.34.4 remote-as 65444
!
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 31, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.2.2.0/32 10.23.23.2 0 0 65222 i
*> 10.2.2.1/32 10.23.23.2 0 0 65222 i
*> 10.2.2.2/32 10.23.23.2 0 0 65222 i
*> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Observe que os prefixos provenientes de rtr2 não estão sendo suprimidos, nem a rota agregada está sendo gerada.

Exemplo 5

Configure atributos BGP com mapas de rotas.

```
<#root>
rtr3(config)#route-map

BGP-ATTR

  permit 10
rtr3(config-route-map)#set community

no-export

rtr3(config-route-map)#exit
rtr3(config)#router bgp 65333
rtr3(config-router)#aggregate-address 10.0.0.0 255.0.0.0

attribute-map BGP-ATTR

  summary-only
```

Se você verificar para ver o endereço agregado gerado, você perceberá que ele diz não anunciado a nenhum peer. Isso é esperado neste cenário específico porque todos os vizinhos são externos (eBGP) e você está definindo a comunidade conhecida de não exportação. Como consequência do uso do argumento `summary-only`, o rtr4 não recebe nenhuma rota.

Esta é a tabela BGP em rtr3

```
<#root>
rtr3#

show ip bgp 10.0.0.0

BGP routing table entry for 10.0.0.0/8, version 20
Paths: (1 available, best #1, table default, not advertised to EBGp peer)
Not advertised to any peer
Refresh Epoch 1
Local, (aggregated by 65333 10.34.34.3)
0.0.0.0 from 0.0.0.0 (10.34.34.3)
Origin IGP, localpref 100, weight 32768, valid, aggregated, local, atomic-aggregate, best
Community: no-export
rx pathid: 0, tx pathid: 0x0
Updated on Jun 12 2026 23:14:53 UTC
```

Verificar

Para verificar se a agregação de rota BGP está funcionando corretamente, você pode verificar no roteador receptor (rtr4 neste exemplo) se você está recebendo apenas o que queremos ver. Por exemplo, somente o resumo, o resumo com todos os prefixos específicos, o resumo e somente alguns prefixos específicos e assim por diante. Você pode usar principalmente os comandos abaixo:

- show ip bgp
- show ip bgp route-map <map-name>
- show running-config | section router bgp
- debug ip bgp update

Consulte os cenários de Troubleshooting para obter mais detalhes.

Troubleshooting

Cenário 1

A rota sumarizada não é recebida e prefixos específicos ainda são vistos.

```
<#root>
```

```
rtr4#
```

```
show ip bgp
```

```
BGP table version is 9, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.1.1.0/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.1/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.2/32 10.34.34.3 0 65333 65111 i
*> 10.1.1.3/32 10.34.34.3 0 65333 65111 i
*> 10.2.2.0/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.1/32 10.34.34.3 0 65333 65222 i
```

```
*> 10.2.2.2/32 10.34.34.3 0 65333 65222 i
*> 10.2.2.3/32 10.34.34.3 0 65333 65222 i
```

Verifique se `summary-only` está configurado como um argumento para o `aggregate-address` e se a máscara de sub-rede está correta.

Esta é a configuração `rtr3`.

```
<#root>
```

```
rtr3(config)#router bgp 65333
rtr3(config-router)#
```

```
aggregate-address 10.0.0.0 255.255.255.0 summary-only
```

```
rtr3(config-router)#exit
rtr3(config)#
```

No exemplo, `summary-address` está configurado, mas a máscara de sub-rede não está correta. Somente prefixos na rede `10.0.0.0/24` caem no agregado, o que quebra a regra para agregação de rota BGP. Todos os prefixos vindos de `rtr1` e `rtr2` caem fora do intervalo de resumo e se você verificar a tabela BGP em `rtr3`, a supressão não está acontecendo.

`rtr3` - configuração corrigida.

```
<#root>
```

```
rtr3(config)#router bgp 65333
rtr3(config-router)#
```

```
aggregate-address 10.0.0.0 255.0.0.0 summary-only
```

```
rtr3(config-router)#exit
rtr3(config)#
```

Observe que os prefixos específicos agora estão sinalizados como suprimidos.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 18, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.0.0.0 0.0.0.0 32768 i
s> 10.1.1.0/32 10.13.13.1 0 0 65111 i
s> 10.1.1.1/32 10.13.13.1 0 0 65111 i
s> 10.1.1.2/32 10.13.13.1 0 0 65111 i
s> 10.1.1.3/32 10.13.13.1 0 0 65111 i
s> 10.2.2.0/32 10.23.23.2 0 0 65222 i
s> 10.2.2.1/32 10.23.23.2 0 0 65222 i
s> 10.2.2.2/32 10.23.23.2 0 0 65222 i
s> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Cenário 2

A opção Suprimir Mapa está configurada para suprimir um intervalo, mas nada está sendo suprimido.

Considere a tabela BGP em rtr3. Você deseja suprimir todos os prefixos 10.2.2.0/24, mas depois de aplicar as configurações, ela não está funcionando.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 37, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.1.1.0/32 10.13.13.1 0 0 65111 i
*> 10.1.1.1/32 10.13.13.1 0 0 65111 i
*> 10.1.1.2/32 10.13.13.1 0 0 65111 i
*> 10.1.1.3/32 10.13.13.1 0 0 65111 i
*> 10.2.2.0/32 10.23.23.2 0 0 65222 i
*> 10.2.2.1/32 10.23.23.2 0 0 65222 i
*> 10.2.2.2/32 10.23.23.2 0 0 65222 i
*> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Verifique a configuração do BGP em rtr3.

```
<#root>
```

```
rtr3#
```

```
show run | section router bgp
```

```
router bgp 65333
 aggregate-address 10.0.0.0 255.0.0.0 suppress-map SUPPRESS-RTR2
 neighbor 10.13.13.1 remote-as 65111
 neighbor 10.23.23.2 remote-as 65222
 neighbor 10.34.34.4 remote-as 65444
rtr3#
```

Verifique o mapa de rotas configurado.

```
<#root>
```

```
rtr3#
```

```
show route-map SUPPRESS-RTR2
```

```
route-map SUPPRESS-RTR2, permit, sequence 10
Match clauses:
ip address prefix-lists:
```

```
suppress-rtr2
```

```
Set clauses:
Policy routing matches: 0 packets, 0 bytes
```

Verifique a lista de prefixos configurados.

```
<#root>
```

```
rtr3#
```

```
show ip prefix-list suppress-rtr2
```

```
ip prefix-list suppress-rtr2: 1 entries
```

```
seq 5 permit 10.2.2.0/24
```

```
rtr3#
```

A lista de prefixos no exemplo corresponde exatamente ao prefixo 10.2.2.0, razão pela qual os prefixos mais específicos não estão sendo suprimidos. Você deve usar o operador menor ou igual a para corresponder prefixos mais específicos.

Corrija a configuração da lista de prefixos.

```
<#root>
```

```
rtr3#configure terminal
rtr3(config)#no ip prefix-list suppress-rtr2
rtr3(config)#

ip prefix-list suppress-rtr2 permit 10.2.2.0/24 le 32

rtr3(config)#end
rtr3#
```

Esta é a tabela BGP rtr3 após a configuração correta da lista de prefixos suppress-rtr2.

```
<#root>
```

```
rtr3#
```

```
show ip bgp
```

```
BGP table version is 14, local router ID is 10.34.34.3
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,
t secondary path, L long-lived-stale,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
Network Next Hop Metric LocPrf Weight Path
*> 10.0.0.0 0.0.0.0 32768 i
*> 10.1.1.0/32 10.13.13.1 0 0 65111 i
*> 10.1.1.1/32 10.13.13.1 0 0 65111 i
*> 10.1.1.2/32 10.13.13.1 0 0 65111 i
*> 10.1.1.3/32 10.13.13.1 0 0 65111 i
s> 10.2.2.0/32 10.23.23.2 0 0 65222 i
s> 10.2.2.1/32 10.23.23.2 0 0 65222 i
s> 10.2.2.2/32 10.23.23.2 0 0 65222 i
s> 10.2.2.3/32 10.23.23.2 0 0 65222 i
```

Você aprendeu como usar argumentos opcionais de BGP para configurar rotas agregadas. Você também recebeu dois exemplos típicos que podem causar falha na agregação de rotas. Esses são os erros mais comuns que podem ser encontrados nas configurações. Se você estiver tendo problemas para gerar uma rota agregada e não puder detectar a configuração problemática, use o comando `debug ip bgp update` para coletar detalhes mais específicos.

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.