

Configurar o recurso de alocação no BGP

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Conventions](#)

[Exemplo](#)

[Informações de Apoio](#)

[Configurar](#)

[Diagrama de Rede](#)

[Configurações](#)

[Verificar](#)

[Troubleshooting](#)

[Mensagem de erro](#)

[Informações Relacionadas](#)

Introdução

Este documento descreve uma situação em que dois roteadores de filial se conectam através de um ISP e executam o BGP entre eles.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Provedor de Internet (ISP)
- Protocolo de gateway de borda (BGP)

Componentes Utilizados

Este documento não se restringe a versões de software e hardware específicas.

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.

Conventions

Para configurar os roteadores de borda do provedor (PE) para permitir o anúncio de todos os prefixos que contêm números de sistema autônomo (ASNs) duplicados, use o comando `neighbor allowas-in` no modo de configuração do roteador. Para desativar o anúncio de leitura do ASN do roteador PE, use a forma enodesse comando.

`neighbor-address allowas-in [number]`

no neighbor lowas-in [número]

ip-address	Endereço IP do roteador vizinho.
número	(Opcional) Especifica o número de vezes para permitir o anúncio do ASN de um roteador PE. O intervalo é 1 a 10. Se nenhum número for fornecido, o valor padrão de 3 vezes será usado.

Em uma configuração de hub e spoke, um roteador PE faz um novo anúncio de todos os prefixos que contêm números de sistema autônomo duplicados. Use o comando `neighbor allowas-in` para configurar dois VRFs em cada roteador PE para receber e anunciar novamente os prefixos da seguinte forma:

- Uma instância de Virtual Private Network Routing and Forwarding (VRF) recebe prefixos com ASNs de todos os roteadores PE e os anuncia aos roteadores PE vizinhos.
- O outro VRF recebe prefixos com ASNs do roteador de borda do cliente (CE) e os anuncia novamente a todos os roteadores PE na configuração de hub e spoke.

Você controla o número de vezes que um ASN é anunciado especificando um número de 1 a 10.

Exemplo

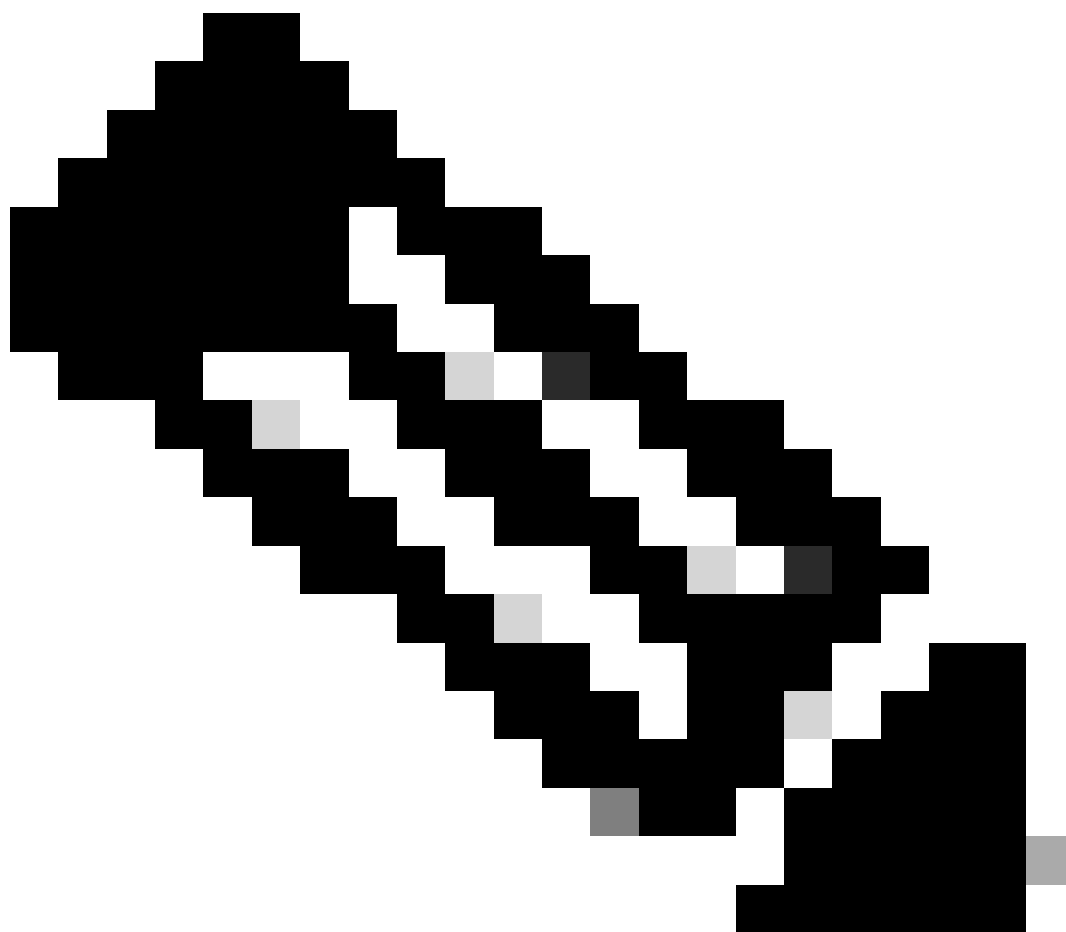
Este exemplo mostra como configurar o roteador PE com ASN 100 para permitir prefixos da família de endereços VRF Rede Virtual Privada (VPN - Virtual Private Network) IPv4 vrf1. O roteador PE vizinho com o endereço IP 192.168.255.255 está definido para ser anunciado novamente para outros roteadores PE com o mesmo ASN seis vezes.

```
Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf vrf1
Router(config-router)# neighbor 192.168.255.255 allowas-in 6
```

Informações de Apoio

Este documento descreve um cenário onde dois roteadores de filial são conectados através de

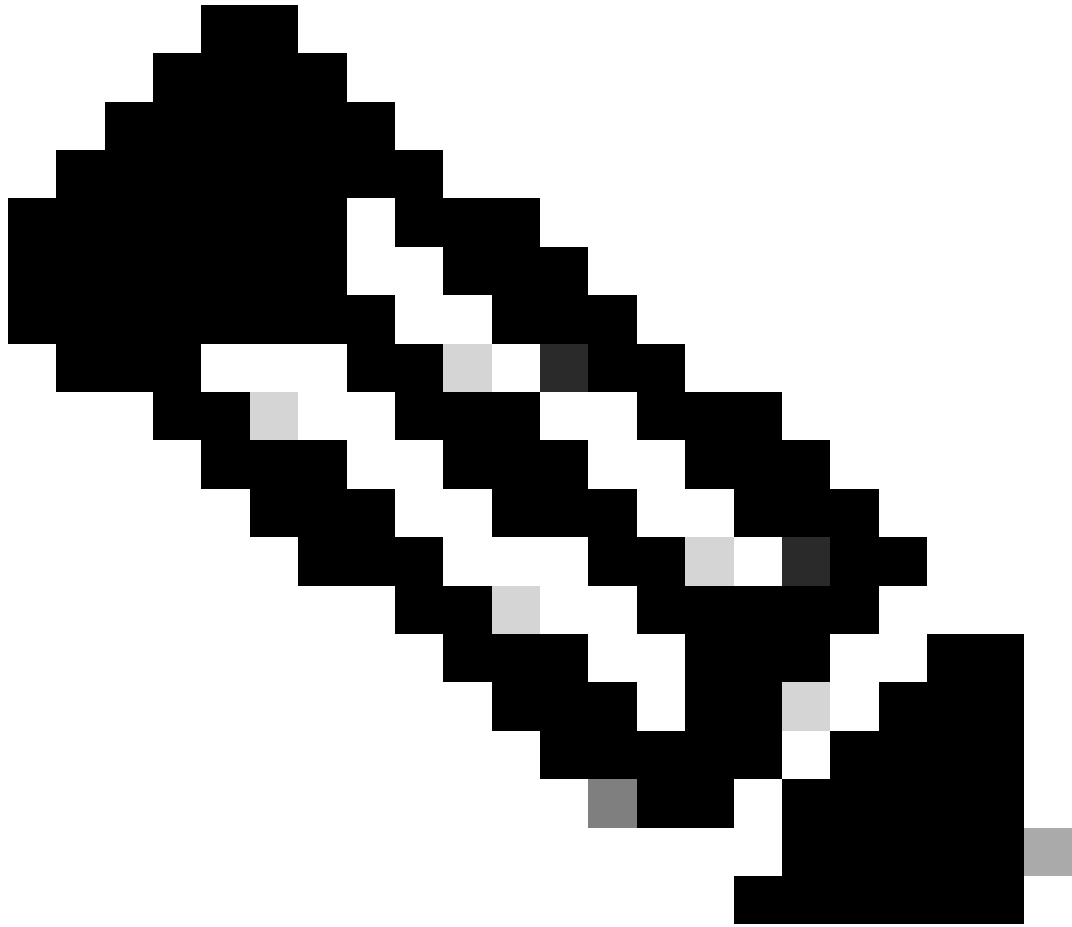
um provedor de Internet (ISP) e executam o protocolo BGP entre eles. Os dois roteadores de filial (R1 e R2), embora em locais diferentes, compartilham o mesmo número AS. Depois que as rotas chegam de uma filial (R1, neste caso) para a rede do provedor de serviços (SP), elas podem ser marcadas com o AS do usuário. Quando o SP passa para o outro roteador de filial (R2), por padrão, as rotas podem ser descartadas se a outra filial também executar o BGP com o SP e usar o mesmo número AS. Neste cenário, o comando `neighbor allowas-in` é emitido para permitir que o BGP do outro lado injete atualizações. Este documento fornece uma configuração de exemplo que ajuda você a entender o recurso `Allowas-in` no BGP.



Observação: esse recurso só pode ser usado para peers eBGP verdadeiros. Você não pode usar esse recurso para dois peers que são membros de sub-ASs de confederação diferentes.

Configurar

Esta seção apresenta as informações para configurar os recursos descritos neste documento.



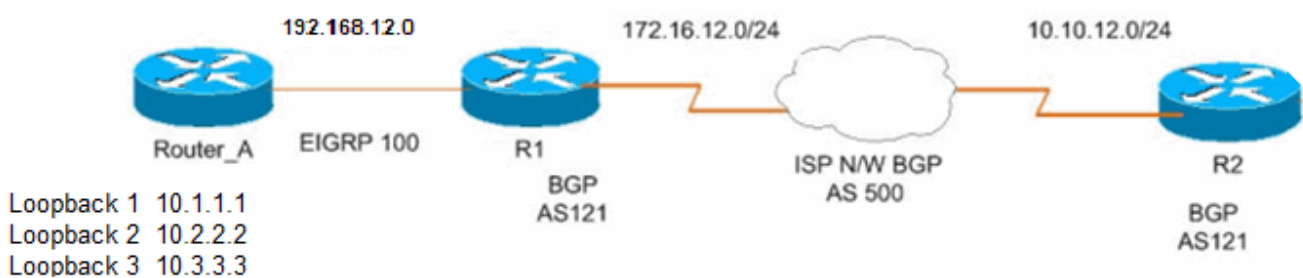
Observação: use a Command Lookup Tool para obter mais informações sobre os comandos usados neste documento.



Observação: somente usuários registrados da Cisco podem acessar ferramentas e informações internas da Cisco.

Diagrama de Rede

Este documento utiliza a seguinte configuração de rede:



Configurações

Este documento utiliza as seguintes configurações:

- [roteador A](#)
- [Roteador R1](#)
- [Roteador R2](#)

Configuração no Router_A

```
<#root>
Router_A#
interface Loopback1
 ip address 10.1.1.1 255.255.255.255
!
interface Loopback2
 ip address 10.2.2.2 255.255.255.255
!
interface Loopback3
 ip address 10.3.3.3 255.255.255.255
!
interface GigabitEthernet0/1
 no switchport
 ip address 192.168.12.2 255.255.255.0
!
router eigrp 100
 network 10.1.1.1 0.0.0.0
 network 10.2.2.2 0.0.0.0
 network 10.3.3.3 0.0.0.0
 network 192.168.12.0
 auto-summary
!
```

Configuração no roteador R1

```
<#root>
R1#
interface Loopback22
 ip address 10.22.22.22 255.255.255.255
!
interface FastEthernet0/0
 ip address 192.168.12.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 172.16.12.1 255.255.255.0
!
```

```

!
router eigrp 100
 network 192.168.12.0
 no auto-summary
!
router bgp 121
 no synchronization
 bgp router-id 10.22.22.22
 bgp log-neighbor-changes
 network 10.22.22.22 mask 255.255.255.255

!--- This is the advertising loopback address.

 redistribute eigrp 100

!--- This shows the redistributing internal routes in BGP.

 neighbor 172.16.12.2 remote-as 500

!--- This shows the EBGP connection with ISP.

 neighbor 172.16.12.2 ebgp-multihop 5
 no auto-summary
!

```

Este exemplo mostra que o EIGRP é executado entre o Router_A e o R1:

```
<#root>
```

```
r1#
```

```
show ip eigrp neighbors
```

```
IP-EIGRP neighbors for process 100
```

H	Address	Interface	Hold Uptime (sec)	SRTT (ms)	RT0	Q	Seq Cnt Num
0	192.168.12.2	Fa0/0	14 01:17:12	828	4968	0	7

Este exemplo mostra como o Roteador R1 aprende rotas do Roteador_A através do EIGRP:

```
<#root>
```

```
r1#
```

```
show ip route eigrp 100
```

```

D    10.0.0.1/8 [90/156160] via 192.168.12.2, 00:02:24, FastEthernet0/0
D    10.0.0.2/8 [90/156160] via 192.168.12.2, 00:02:24, FastEthernet0/0
D    10.0.0.3/8 [90/156160] via 192.168.12.2, 00:02:24, FastEthernet0/0

```

Este exemplo mostra como o Roteador R1 estabelece uma conexão BGP com um ISP executando o BGP AS500:

```
<#root>
```

```
r1#
```

```
show ip bgp summary
```

```
BGP router identifier 10.22.22.22, local AS number 121
BGP table version is 19, main routing table version 19
7 network entries using 924 bytes of memory
7 path entries using 364 bytes of memory
5/4 BGP path/bestpath attribute entries using 840 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 1 (at peak 2) using 32 bytes of memory
BGP using 2184 total bytes of memory
BGP activity 40/33 prefixes, 42/35 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
172.16.12.2	4	500	86	76	19	0	0	00:25:13	2

Este exemplo mostra como R1 anuncia as rotas BGP aprendidas:

```
<#root>
```

```
r1#
```

```
show ip bgp
```

```
BGP table version is 19, local router ID is 10.22.22.22
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 10.0.0.1	192.168.12.2	156160		32768	?
*> 10.0.0.2	192.168.12.2	156160		32768	?
*> 10.0.0.3	192.168.12.2	156160		32768	?
*> 10.10.12.0/24	172.16.12.2	0		0 500	i
*> 10.22.22.22/32	0.0.0.0	0		32768	i
r> 172.16.12.0/24	172.16.12.2	0		0 500	i
*> 192.168.12.0	0.0.0.0	0		32768	?

```
<#root>
```

```
r1#
```

```
ping 10.10.12.2
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.12.2, timeout is 2 seconds:
!!!!
!--- This is the connectivity with Router 2 across the Internet cloud.
```


Configuração no roteador R2

```
<#root>
R2#
interface Loopback33
 ip address 10.33.33.33 255.255.255.255
!
interface Serial1/0
 ip address 10.10.12.1 255.255.255.0

router bgp 121
 no synchronization
 bgp router-id 10.33.33.33
 bgp log-neighbor-changes
 network 10.33.33.33 mask 255.255.255.255

!--- This is the advertising loopback address.

 neighbor 10.10.12.2 remote-as 500

!--- This is the EBGP connection with ISP.

 neighbor 10.10.12.2 ebgp-multihop 5
 no auto-summary
```

O roteador R2 não aprende nenhuma rota do roteador R1.

Esse é um comportamento natural porque o BGP tenta evitar loops de roteamento. Por exemplo, o anúncio de leitura de todos os prefixos que contêm números de sistemas autônomos (ASNs) duplicados é desabilitado por padrão.

As rotas EIGRP redistribuídas (10.0.0.1, 10.0.0.2, 10.0.0.3) e a rota interna BGP 10.22.22.22 de R1 não são recebidas por R2 porque se originam do mesmo ASN na Internet. Como R2 vê seu próprio número AS (121) no AS-PATH, R2 não toma essas rotas.

```
<#root>
r2#
show ip bgp

BGP table version is 20, local router ID is 10.33.33.33
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric LocPrf Weight Path
r> 10.10.12.0/24    10.10.12.2             0           0 500 i
*> 10.33.33.33/32  0.0.0.0              0          32768 i
*> 172.16.12.0/24  10.10.12.2             0           0 500 i
```

Para permitir o anúncio de todos os prefixos que contêm ASNs duplicados, use o comando `neighbor allowas-in` no modo de configuração do roteador R2.

```
<#root>
```

```
r2(config-router)#
```

```
neighbor 10.10.12.2 allowas-in
```

```
r2#
```

```
clear ip bgp*
```

```
r2#
```

```
show ip bgp
```

```
BGP table version is 10, local router ID is 10.33.33.33
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,  
r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 10.0.0.1	10.10.12.2			0 500	121 ?
*> 10.0.0.2	10.10.12.2			0 500	121 ?
*> 10.0.0.3	10.10.12.2			0 500	121 ?
r> 10.10.12.0/24	10.10.12.2	0		0 500	i
*> 10.22.22.22/32	10.10.12.2			0 500	121 i
* 10.33.33.33/32	10.10.12.2			0 500	121 i
*>	0.0.0.0	0		32768	i
*> 172.16.12.0/24	10.10.12.2	0		0 500	i
*> 192.168.12.0	10.10.12.2			0 500	121 ?

Agora tente fazer ping de R1 para R2:

```
<#root>
```

```
r2#
```

```
ping 10.22.22.22
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.22.22.22, timeout is 2 seconds:
```

```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/57/60 ms
```

Verificar

No momento, não há procedimento de verificação disponível para esta configuração.

Troubleshooting

Mensagem de erro

A mensagem de erro **%BGP% Neighbor A.B.C.D rcv bogus route : AS loop** é recebida.

Essa notificação significa que a rota BGP recebida pelo roteador CE tem seu próprio número AS no caminho AS e é considerada um loop de roteador para o roteador CE. Como solução alternativa, configure o roteador CE com o recurso `allocou-in`, conforme ilustrado no exemplo anterior.

Informações Relacionadas

- [Protocolo de gateway de borda \(BGP\)](#)
- [Suporte técnico e downloads da Cisco](#)

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.