

Usando valores da comunidade do BGP para controlar a política de roteamento na rede de provedor de upstream

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Material de Suporte](#)

[Convenções](#)

[Configurar](#)

[Controlando a Política de Roteamento](#)

[Diagrama de Rede](#)

[Configurações](#)

[Verificar](#)

[Informações Relacionadas](#)

Introdução

Este documento demonstra como o atributo de comunidade Border Gateway Protocol (BGP) pode ser usado para controlar as políticas de roteamento em sua rede de provedor de serviços upstream.

Pré-requisitos

Requisitos

Este documento exige uma compreensão do protocolo de roteamento BGP e de sua operação. Para mais informação, refira [Casos Práticos do BGP](#).

Componentes Utilizados

Este documento não se restringe a versões de software e hardware específicas. Contudo, a informação neste documento é baseada nestes versão de software e hardware:

- Software Release 12.2(27) de Cisco IOS®
- Cisco 2500 Series Routers

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 você estiver trabalhando em uma rede ativa, certifique-se de que entende o impacto potencial de qualquer comando antes de utilizá-lo.

[Material de Suporte](#)

Quando as comunidades elas mesmas não alterarem o [processo de tomada de decisão BGP](#), as comunidades podem ser usadas como bandeiras a fim marcar um grupo de rotas. O serviço upstream que os roteadores de provedor podem então usar estas bandeiras para aplicar o roteamento específico policia (por exemplo, preferência local) dentro de sua rede.

Os fornecedores estabelecem um mapeamento entre valores de comunidade configuráveis do cliente e os valores correspondentes da preferência local dentro da rede de provedor. A ideia é que os clientes com políticas específicas que exigem a alteração do LOCAL_PREF na rede de provedor ajustam os valores de comunidade correspondentes em suas atualizações de roteamento.

Uma comunidade é um grupo de prefixos que compartilham de alguma propriedade comum e podem ser configurados com o atributo de comunidade do BGP. [O atributo de comunidade do BGP é um atributo transitivo opcional de comprimento variável](#). O atributo consiste em um grupo de quatro valores do octeto que especificam uma comunidade. Os valores de atributo de comunidade são codificados com um número de sistema autônomo (AS) nos primeiros dois octetos, com os dois octetos permanecendo definidos pelo COMO. Um prefixo que pode ter mais de um atributo de comunidade. Um interlocutor do BGP que vê atributos de comunidades múltiplas em um prefixo pode agir com base em um, alguns ou todos os atributos. Um roteador tem a opção para adicionar ou alterar um atributo de comunidade antes que o roteador passe o atributo sobre a outros pares. A fim aprender mais sobre o atributo de comunidade, refira [Casos Práticos do BGP](#).

O atributo de preferência local é uma indicação ao COMO que o trajeto é preferido a fim alcançar uma determinada rede. Quando existem vários caminhos para o mesmo destino, o caminho com a preferência mais elevada é preferido (o valor padrão do atributo de preferência local é 100). Para mais informação, refira o [atributo de preferência local](#).

[Convenções](#)

Para obter mais informações sobre convenções de documento, consulte as [Convenções de dicas técnicas Cisco](#).

[Configurar](#)

[Controlando a Política de Roteamento](#)

Nesta seção, você encontrará informações para configurar os recursos descritos neste documento.

Note: Para localizar informações adicionais sobre os comandos usados neste documento, utilize a Ferramenta Command Lookup (somente clientes [registrados](#)).

Para a simplificação, este mapeamento do atributo de comunidade e o atributo de preferência local são supostos para ser estabelecidos entre o fornecedor de serviço upstream (AS100) e o cliente (AS30).

Preferência local	Valores de comunidade
130	100:300
125	100:250

Se o cliente anuncia os prefixos com um atributo de comunidade igual a 100:300, o provedor de serviços de upstream define a preferência local desses roteadores como 130 e 125, se o atributo da comunidade é igual a 100:250.

Isto dá-lhe o potencial controlar a política de roteamento dentro da rede de provedor de serviços se você muda os valores de comunidade dos prefixos anunciados ao provedor de serviços.

[No diagrama da rede](#), o cliente AS30 deseja conseguir esta política de roteamento com os atributos de comunidade.

- O tráfego de entrada do AS100 destinado à rede 6.6.6.0/24 vem através do link R1-R3. Caso que o link R1-R3 falha, todo o tráfego entra com o R2-R3.
- O tráfego de entrada do AS100 destinado à rede 7.7.7.0/24 vem através do link R2-R3. Caso que o link R2-R3 falha, todo o tráfego entra com o R1-R3.

A fim conseguir esta política de roteamento, o R3 anuncia seus prefixos como segue:

Ao r1:

- 6.6.6.0/24 com um atributo de comunidade 100:300
- 7.7.7.0/24 com atributo de comunidade de 100:250

Para R2:

- 6.6.6.0/24 com um atributo de comunidade 100:250
- 7.7.7.0/24 com um atributo de comunidade 100:300

Uma vez que o r1 dos vizinhos de BGP e o R2 recebem os prefixos do R3, o r1 e o R2 aplicam a política preconfigurada baseada no mapeamento entre a comunidade e os atributos de preferência local (mostrados [nesta tabela](#)), e conseguem assim a política de roteamento ditado pelo cliente (AS30). O r1 instala os prefixos na tabela de BGP.

- 6.6.6.0/24 com uma preferência local de 130
- 7.7.7.0/24 com uma preferência local de 125

O R2 instala o prefixo em sua tabela de BGP:

- 6.6.6.0/24 com uma preferência local de 125
- 7.7.7.0/24 com uma preferência local de 130

Como uma preferência local mais alta tem prioridade nos critérios de seleção de caminho do BGP, o caminho com a preferência local de 130 (130 é maior do que 125) é selecionado como o melhor caminho no AS 100 e é instalado na tabela de roteamento de IP de R1 e R2. Para obter mais informações sobre do critério de seleção de caminho do BGP, refira o [algoritmo de seleção de caminho do melhores BGP](#).

[Diagrama de Rede](#)

Este documento utiliza a configuração de rede mostrada neste diagrama:

Configurações

Este documento utiliza as seguintes configurações:

- [R3](#)
- [R1](#)
- [R2](#)

R3

```
Current configuration : 2037 bytes
!
version 12.2
!
hostname R3
!
interface Loopback0
 ip address 6.6.6.1 255.255.255.0
!
interface Ethernet0/0
 ip address 7.7.7.1 255.255.255.0
!
interface Serial8/0
 ip address 10.10.13.3 255.255.255.0
 !--- Interface connected to R1. ! interface Serial9/0
 ip address 10.10.23.3 255.255.255.0 !--- Interface
connected to R2. ! router bgp 30 network 6.6.6.0 mask
255.255.255.0 network 7.7.7.0 mask 255.255.255.0 !---
Network commands announce prefix 6.6.6.0/24 !--- and
7.7.7.0/24. neighbor 10.10.13.1 remote-as 100 !---
Establishes peering with R1. neighbor 10.10.13.1 send-
community - !--- Without this command, the community
attributes !--- are not sent to the neighbor. neighbor
10.10.13.1 route-map Peer-R1 out !--- Configures
outbound policy as defined by !--- route-map "Peer-R1"
when peering with R1. neighbor 10.10.23.2 remote-as 100
!--- Establishes peering with R2. neighbor 10.10.23.2
send-community !--- Configures to send community
attribute to R2. neighbor 10.10.23.2 route-map Peer-R2
out !--- Configures outbound policy as defined by !---
route-map "Peer-R2" when peering with R2. no auto-
summary ! ip classless ip bgp-community new-format !---
Allows you to configure the BGP community !--- attribute
in AA:NN format. ! access-list 101 permit ip host
6.6.6.0 host 255.255.255.0
access-list 102 permit ip host 7.7.7.0 host
255.255.255.0
!
!
route-map Peer-R1 permit 10
 match ip address 101
 set community 100:300
 !--- Sets community 100:300 for routes matching access-
list 101.
!
route-map Peer-R1 permit 20
 match ip address 102
 set community 100:250
 !--- Sets community 100:250 for routes matching access-
```

```

list 102.

!
route-map Peer-R2 permit 10
  match ip address 101
  set community 100:250
  !--- Sets community 100:250 for routes matching access-
list 101.

!
route-map Peer-R2 permit 20
  match ip address 102
  set community 100:300
  !--- Sets community 100:300 for routes matching access-
list 102.

!
end

```

R1

```

Current configuration : 2037 bytes
!
version 12.2
!
hostname R3
!
interface Loopback0
  ip address 6.6.6.1 255.255.255.0
!
interface Ethernet0/0
  ip address 7.7.7.1 255.255.255.0
!
interface Serial8/0
  ip address 10.10.13.3 255.255.255.0
  !--- Interface connected to R1. ! interface Serial9/0
  ip address 10.10.23.3 255.255.255.0 !--- Interface
connected to R2. ! router bgp 30 network 6.6.6.0 mask
255.255.255.0 network 7.7.7.0 mask 255.255.255.0 !---
Network commands announce prefix 6.6.6.0/24 !--- and
7.7.7.0/24. neighbor 10.10.13.1 remote-as 100 !---
Establishes peering with R1. neighbor 10.10.13.1 send-
community - !--- Without this command, the community
attributes !--- are not sent to the neighbor. neighbor
10.10.13.1 route-map Peer-R1 out !--- Configures
outbound policy as defined by !--- route-map "Peer-R1"
when peering with R1. neighbor 10.10.23.2 remote-as 100
!--- Establishes peering with R2. neighbor 10.10.23.2
send-community !--- Configures to send community
attribute to R2. neighbor 10.10.23.2 route-map Peer-R2
out !--- Configures outbound policy as defined by !---
route-map "Peer-R2" when peering with R2. no auto-
summary ! ip classless ip bgp-community new-format !---
Allows you to configure the BGP community !--- attribute
in AA:NN format. ! access-list 101 permit ip host
6.6.6.0 host 255.255.255.0
access-list 102 permit ip host 7.7.7.0 host
255.255.255.0
!
!
route-map Peer-R1 permit 10
  match ip address 101
  set community 100:300

```

```

!--- Sets community 100:300 for routes matching access-
list 101.

!
route-map Peer-R1 permit 20
  match ip address 102
  set community 100:250
  !--- Sets community 100:250 for routes matching access-
list 102.

!
route-map Peer-R2 permit 10
  match ip address 101
  set community 100:250
  !--- Sets community 100:250 for routes matching access-
list 101.

!
route-map Peer-R2 permit 20
  match ip address 102
  set community 100:300
  !--- Sets community 100:300 for routes matching access-
list 102.

!
end

```

R2

```

Current configuration : 2037 bytes
!
version 12.2
!
hostname R3
!
interface Loopback0
  ip address 6.6.6.1 255.255.255.0
!
interface Ethernet0/0
  ip address 7.7.7.1 255.255.255.0
!
interface Serial8/0
  ip address 10.10.13.3 255.255.255.0
  !--- Interface connected to R1. ! interface Serial9/0
  ip address 10.10.23.3 255.255.255.0 !--- Interface
connected to R2. ! router bgp 30 network 6.6.6.0 mask
255.255.255.0 network 7.7.7.0 mask 255.255.255.0 !---
Network commands announce prefix 6.6.6.0/24 !--- and
7.7.7.0/24. neighbor 10.10.13.1 remote-as 100 !---
Establishes peering with R1. neighbor 10.10.13.1 send-
community - !--- Without this command, the community
attributes !--- are not sent to the neighbor. neighbor
10.10.13.1 route-map Peer-R1 out !--- Configures
outbound policy as defined by !--- route-map "Peer-R1"
when peering with R1. neighbor 10.10.23.2 remote-as 100
!--- Establishes peering with R2. neighbor 10.10.23.2
send-community !--- Configures to send community
attribute to R2. neighbor 10.10.23.2 route-map Peer-R2
out !--- Configures outbound policy as defined by !---
route-map "Peer-R2" when peering with R2. no auto-
summary ! ip classless ip bgp-community new-format !---
Allows you to configure the BGP community !--- attribute
in AA:NN format. ! access-list 101 permit ip host

```

```

6.6.6.0 host 255.255.255.0
access-list 102 permit ip host 7.7.7.0 host
255.255.255.0
!
!
route-map Peer-R1 permit 10
  match ip address 101
  set community 100:300
  !--- Sets community 100:300 for routes matching access-
list 101.
!
route-map Peer-R1 permit 20
  match ip address 102
  set community 100:250
  !--- Sets community 100:250 for routes matching access-
list 102.
!
route-map Peer-R2 permit 10
  match ip address 101
  set community 100:250
  !--- Sets community 100:250 for routes matching access-
list 101.
!
route-map Peer-R2 permit 20
  match ip address 102
  set community 100:300
  !--- Sets community 100:300 for routes matching access-
list 102.
!
end

```

Verificar

O r1 recebe os prefixos 6.6.6.0/24 e 7.7.7.0/24 com as comunidades 100:300 e 100:250, segundo as indicações de corajoso nas **saídas BGP IP da mostra** desta seção.

Note: Uma vez que estas rotas são instaladas na tabela de BGP baseada na política configurada, os prefixos com a comunidade 100:300 estão atribuídos a preferência local 130 e os prefixos com a comunidade 100:250 são atribuídos a preferência local 125.

```

R1# show ip bgp 6.6.6.0
BGP routing table entry for 6.6.6.0/24, version 2
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  10.10.12.2
  30
  10.10.13.3 from 10.10.13.3 (6.6.6.1)
    Origin IGP, metric 0, localpref 130, valid, external, best
    Community: 100:300
!--- Prefix 6.6.6.0/24 with community 100:300 received from !--- 10.10.13.3 (R3) is assigned
local preference 130.

```

```

R1# show ip bgp 7.7.7.0
BGP routing table entry for 7.7.7.0/24, version 4
Paths: (2 available, best #1, table Default-IP-Routing-Table)

```

```

Advertised to non peer-group peers:
10.10.13.3
30
  10.10.12.2 from 10.10.12.2 (192.168.50.1)
    Origin IGP, metric 0, localpref 130, valid, internal, best
!--- Received prefix 7.7.7.0/24 over iBGP from 10.10.12.2 !--- (R2) with local preference 130.

30
10.10.13.3 from 10.10.13.3 (6.6.6.1)
  Origin IGP, metric 0, localpref 125, valid, external
  Community: 100:250
!--- Prefix 7.7.7.0/24 with community 100:250 received from !--- 10.10.13.3 (R3) is assigned
local preference 125.

```

```

R1# show ip bgp
BGP table version is 4, local router ID is 200.200.200.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 6.6.6.0/24	10.10.13.3	0	130	0	30 i
*>i7.7.7.0/24	10.10.12.2	0	130	0	30 i
*	10.10.13.3	0	125	0	30 i

O comando **show ip bgp** no r1 confirma que o melhor caminho selecionado no r1 é com preference(LocPrf) local = 130.

A similaridade, R2 recebe os prefixos 6.6.6.0/24 e 7.7.7.0/24 com as comunidades 100:250 e 100:300, segundo as indicações de corajoso na saída do comando **show ip bgp** desta seção.

Note: Uma vez que estas rotas são instaladas na tabela de BGP, com base na política configurada, prefixos com a comunidade que 100:300 estão atribuídos a preferência local 130 e os prefixos com a comunidade 100:250 são atribuídos a preferência local 125.

```

R2# show ip bgp 6.6.6.0
BGP routing table entry for 6.6.6.0/24, version 2
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  10.10.23.3
  30
    10.10.23.3 from 10.10.23.3 (6.6.6.1)
      Origin IGP, metric 0, localpref 125, valid, external
      Community: 100:250
!--- Prefix 6.6.6.0/24 with community 100:250 received from !--- 10.10.23.3 (R3) is assigned
local preference 125.

```

```

R2# show ip bgp 6.6.6.0
BGP routing table entry for 6.6.6.0/24, version 2
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  10.10.23.3
  30
    10.10.23.3 from 10.10.23.3 (6.6.6.1)
      Origin IGP, metric 0, localpref 125, valid, external
      Community: 100:250
!--- Prefix 6.6.6.0/24 with community 100:250 received from !--- 10.10.23.3 (R3) is assigned
local preference 125.

```

```

R2# show ip bgp 7.7.7.0
BGP routing table entry for 7.7.7.0/24, version 3
Paths: (1 available, best #1, table Default-IP-Routing-Table)

```



```
Advertised to non peer-group peers:
10.10.12.1
30
  10.10.23.3 from 10.10.23.3 (6.6.6.1)
    Origin IGP, metric 0, localpref 130, valid, external, best
    Community: 100:300
!--- Prefix 7.7.7.0/24 with community 100:300 received from !--- 10.10.23.3 (R3) is assigned
local preference 130.
```

```
R2# show ip bgp
BGP table version is 3, local router ID is 192.168.50.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
* 6.6.6.0/24	10.10.23.3	0	125	0	30 i
*>i	10.10.12.1	0	130	0	30 i
*> 7.7.7.0/24	10.10.23.3	0	130	0	30 i

Esta saída do comando **show ip bgp** no R2 confirma o melhor caminho selecionado no R2 é com preference(localPrf) local = 130.

A rota IP para prefixar 6.6.6.0/24 prefere o link R1-R3 que retira fora do AS100 para o AS30. O comando **show ip route** no r1 e no R2 confirma aquele.

```
R1# show ip route 6.6.6.0
Routing entry for 6.6.6.0/24
  Known via "bgp 100", distance 20, metric 0
  Tag 30, type external
  Last update from 10.10.13.3 3d21h ago
  Routing Descriptor Blocks:
  * 10.10.13.3, from 10.10.13.3, 3d21h ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
!--- On R1, the IP route to prefix 6.6.6.0/24 points !--- to next hop 10.10.13.3 which is R3
serial 8/0 !--- interface on the R1-R3 link.
```

```
R2# show ip route 6.6.6.0
Routing entry for 6.6.6.0/24
  Known via "bgp 100", distance 200, metric 0
  Tag 30, type internal
  Last update from 10.10.12.1 3d21h ago
  Routing Descriptor Blocks:
  * 10.10.12.1, from 10.10.12.1, 3d21h ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
!--- On R2, IP route to prefix 6.6.6.0/24 points !--- to next hop R1 (10.10.12.1) on its iBGP
link. !--- Thus traffic to network 6.6.6.0/24 from R2 !--- exits through R2-R1 and then R1-R3
link from !--- AS 100 towards AS 30.
```

A rota IP para prefixar 7.7.7.0/24 prefere o link R2-R3 que retira fora do AS100 para o AS30. O comando **show ip route** no r1 e no R2 confirma aquele.

```
R2# show ip route 7.7.7.0
Routing entry for 7.7.7.0/24
  Known via "bgp 100", distance 20, metric 0
  Tag 30, type external
  Last update from 10.10.23.3 3d22h ago
  Routing Descriptor Blocks:
  * 10.10.23.3, from 10.10.23.3, 3d22h ago
    Route metric is 0, traffic share count is 1
```

AS Hops 1

!--- On R2, IP route to prefix 7.7.7.0/24 points !--- to next hop 10.10.23.3 which is R3 serial 9/0 !--- interface on R2-R3 link.

```
R1# show ip route 7.7.7.0
```

```
Routing entry for 7.7.7.0/24
  Known via "bgp 100", distance 200, metric 0
  Tag 30, type internal
  Last update from 10.10.12.2 3d22h ago
  Routing Descriptor Blocks:
  * 10.10.12.2, from 10.10.12.2, 3d22h ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
```

!--- On R1, IP route to prefix 7.7.7.0/24 points !--- to next hop R2 (10.10.12.2) on its iBGP link. !--- Thus traffic to network 7.7.7.0/24 from R1 !--- exits through R1-R2 and then R2-R3 link !--- from AS 100 towards AS 30.

No caso de uma falha de um link, por exemplo o link R1-R3, todo o tráfego deve seguir o link R2-R3. Você pode simular aquele se você fecha o link entre o R1-R3.

```
R1# conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
R1(config)#int s8/0
```

```
R1(config-if)#shut
```

```
R1(config-if)#
```

```
3d22h: %BGP-5-ADJCHANGE: neighbor 10.10.13.3 Down Interface flap
```

```
3d22h: %LINK-5-CHANGED: Interface Serial8/0, changed state to
  administratively down
```

```
3d22h: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial8/0,
  changed state to down
```

Observe a tabela de IP Routing para os prefixos 6.6.6.0/24 e 7.7.7.0/24 no R1 e R2. Use o link R2-R3 a fim retirar fora do AS100.

```
R1# show ip route 6.6.6.0
```

```
Routing entry for 6.6.6.0/24
  Known via "bgp 100", distance 200, metric 0
  Tag 30, type internal
  Last update from 10.10.12.2 00:01:47 ago
  Routing Descriptor Blocks:
  * 10.10.12.2, from 10.10.12.2, 00:01:47 ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
```

```
R1# show ip route 7.7.7.0
```

```
Routing entry for 7.7.7.0/24
  Known via "bgp 100", distance 200, metric 0
  Tag 30, type internal
  Last update from 10.10.12.2 3d22h ago
  Routing Descriptor Blocks:
  * 10.10.12.2, from 10.10.12.2, 3d22h ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
```

Este **show command output (resultado do comando show)** mostra que a rota aos pontos dos prefixos 6.6.6.0/24 e 7.7.7.0/24 ao salto seguinte 10.10.12.2, (R2), que é esperado. Olhe agora a tabela de IP Routing no salto seguinte da verificação R2to do prefixo 6.6.6.0/24 e 7.7.7.0/24. O salto seguinte deve ser R3 para a política configurada a fim trabalhar com sucesso.

```
R2# show ip route 6.6.6.0
Routing entry for 6.6.6.0/24
  Known via "bgp 100", distance 20, metric 0
  Tag 30, type external
  Last update from 10.10.23.3 00:04:10 ago
  Routing Descriptor Blocks:
  * 10.10.23.3, from 10.10.23.3, 00:04:10 ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
```

```
R2# show ip route 7.7.7.0
Routing entry for 7.7.7.0/24
  Known via "bgp 100", distance 20, metric 0
  Tag 30, type external
  Last update from 10.10.23.3 3d22h ago
  Routing Descriptor Blocks:
  * 10.10.23.3, from 10.10.23.3, 3d22h ago
    Route metric is 0, traffic share count is 1
    AS Hops 1
```

O salto seguinte 10.10.23.3 é R3 relação da série 9/0 no link R2-R3. Isto confirma os trabalhos configurados da política como esperado.

[Informações Relacionadas](#)

- [RFC 1998](#)
- [Troubleshooting de BGP](#)
- [BGP: Perguntas mais freqüentes](#)
- [Configurações de exemplo para compartilhamento de carga com BGP em ambientes únicos e multihomed](#)
- [Página de suporte de BGP](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)