Configuración de una Red de Proveedor de Flujo Ascendente con Valores de Comunidad BGP

Contenido

Introducción

Prerequisites

- Requirements
- Componentes Utilizados

Antecedentes

Convenciones

Configuración y control de la política de routing

- Diagrama de la red

Configuraciones

R3

R1

R2

Verificación

Información Relacionada

Introducción

Este documento describe cómo utilizar los valores de la comunidad BGP para controlar la política de ruteo en las redes de proveedores ascendentes.

Prerequisites

Requirements

Este documento requiere una comprensión del protocolo de ruteo BGP (Border Gateway Protocol) y su funcionamiento.

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware. Sin embargo, la información de este documento se basa en esta versión de software:

- ‘Versión de software 12.2 del IOS® de Cisco

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo,

asegúrese de entender el posible impacto de cualquier comando.

Antecedentes

Mientras que las comunidades mismas no modifican el proceso BGP Best Path, las comunidades se pueden utilizar como indicadores para marcar un conjunto de rutas. Los routers de proveedores de servicios ascendentes pueden utilizar estos indicadores para aplicar políticas de ruteo específicas (por ejemplo, la preferencia local) dentro de su red.

Los proveedores se asignan entre los valores de comunidad configurables y los valores de preferencia locales correspondientes dentro de la red del proveedor. Puede tener directivas específicas que requieran la modificación de LOCAL_PREF en el conjunto de redes del proveedor y los valores de comunidad correspondientes en sus actualizaciones de enrutamiento.

Una comunidad es un grupo de prefijos que comparten alguna propiedad común y se puede configurar con el atributo de comunidad BGP. El atributo de la comunidad BGP es un atributo transitivo opcional de longitud variable. El atributo consta de un conjunto de cuatro valores de octeto que especifican una comunidad. Los valores de atributo de comunidad se codifican con un número de sistema autónomo (AS) en los primeros dos octetos, con los otros dos octetos definidos por el AS. Un prefijo puede tener más de un atributo de comunidad. Un altavoz BGP que ve múltiples atributos de comunidad en un prefijo puede actuar basándose en uno, algunos o todos los atributos. Un router tiene la opción de agregar o modificar un atributo de comunidad antes de que el router pase el atributo a otros peers. Para aprender más sobre el atributo de comunidad, consulte Casos Prácticos de BGP.

El atributo de preferencia local es una indicación para el AS de qué trayectoria se prefiere alcanzar una red determinada. Cuando hay varias rutas al mismo destino, se elige la ruta con la preferencia más alta (el valor predeterminado del atributo de preferencia local es 100). Para obtener más información, consulte Casos prácticos.

Convenciones

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

Configuración y control de la política de routing

Nota: Para obtener información adicional sobre los comandos que se utilizan en este documento, use la herramienta Command Lookup (Búsqueda de comandos).

Para simplificar, se supone que el atributo de comunidad y la asignación de atributos de preferencia local se establecen entre el proveedor de servicios ascendente (AS 100) y el dispositivo (AS 30).

Preferencia local	Valores de comunidad

130	100:300
125	100:250

Si los prefijos se anuncian con un atributo de comunidad igual a 100:300, el proveedor de servicio ascendente establece la preferencia local de esas rutas en 130 y 125 si el atributo de comunidad es igual a 100:250.

Esto le da control sobre la política de ruteo dentro de la red del proveedor de servicio si cambia los valores de comunidad de los prefijos anunciados al proveedor de servicio.

En el diagrama de red , el AS 30 quiere utilizar esta política de ruteo con los atributos de la comunidad.

- El tráfico entrante del AS 100 destinado a la red 10.0.10.0/24 viaja a través del link R1-R3. Si el link R1-R3 falla, todo el tráfico viaja a través de R2-R3.
- El tráfico entrante del AS 100 destinado a la red 10.1.0.0/24 viaja a través del link R2-R3. Si el link R2-R3 falla, todo el tráfico viaja a través de R1-R3.

Para lograr esta política de ruteo, R3 anuncia sus prefijos de esta manera:

A R1:

- 10.0.10.0/24 con un atributo comunitario 100:300
- 10.1.0.0/24 con un atributo comunitario 100:250

A R2:

- 10.0.10.0/24 con un atributo comunitario 100:250
- 10.1.0.0/24 con un atributo comunitario 100:300

Una vez que los vecinos BGP R1 y R2 reciben los prefijos de R3, R1 y R2 aplican la política configurada basada en la asignación entre la comunidad y los atributos de preferencia locales (mostrados en la tabla anterior), y así logran la política de ruteo dictada por usted (el AS 30). R1 instala los prefijos en la tabla BGP.

- 10.0.10.0/24 con una preferencia local de 130
- 10.1.0.0/24 con una preferencia local de 125

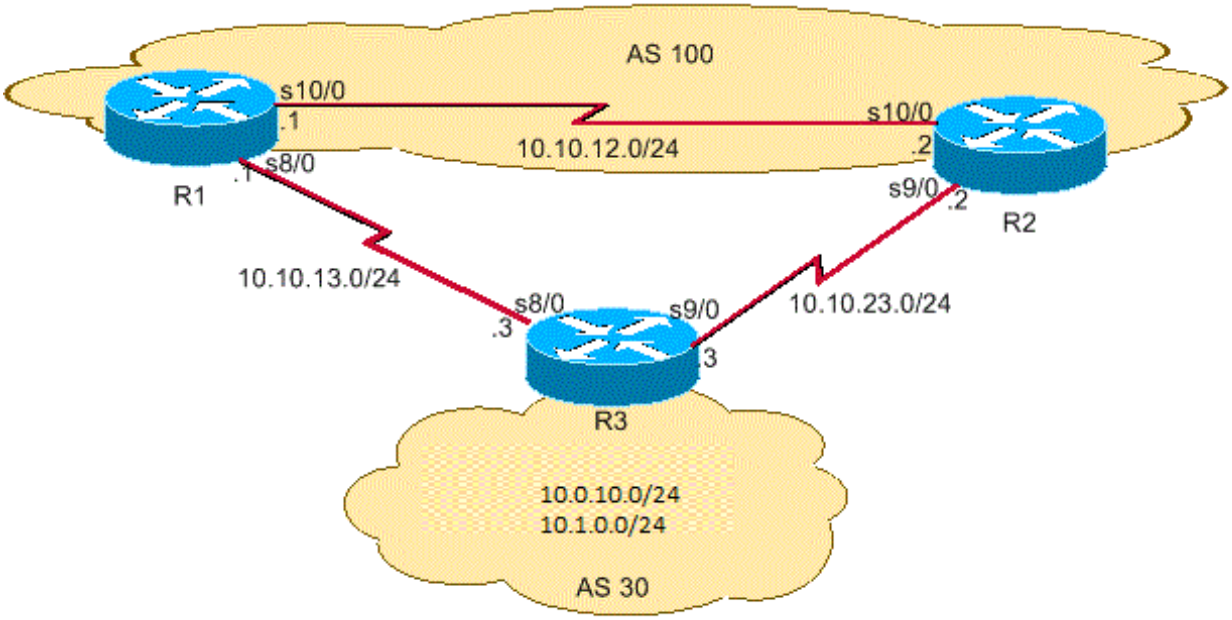
R2 instala el prefijo en su tabla BGP:

- 10.0.10.0/24 con una preferencia local de 125
- 10.1.0.0/24 con una preferencia local de 130

Dado que se prefiere una preferencia local más alta en los criterios de selección de trayectoria

BGP, la trayectoria con una preferencia local de 130 (130 es mayor que 125) se selecciona como la mejor trayectoria dentro del AS 100, y se instala en la tabla de IP Routing de R1 y R2. Para obtener más información sobre los criterios de selección de trayectoria BGP, consulte Algoritmo de Selección de Mejor Trayectoria BGP.

Diagrama de la red



Redes BGP

Configuraciones

En este documento, se utilizan estas configuraciones:

- R3
- R1
- R2

R3

```
hostname R3
!
interface Loopback0
 ip address 10.0.10.0 255.255.255.0
!
interface Ethernet0/0
 ip address 10.1.0.0 255.255.255.1
!
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 10.0.10.0 mask 255.255.255.0
 network 10.1.0.0 mask 255.255.255.1

!--- Network commands announce prefix 10.0.10.0/24 and 10.1.0.0/24.

 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 10.0.10.0 host 255.255.255.0
access-list 102 permit ip host 10.1.0.0 host 255.255.255.1
!
!
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

```
hostname R1
!
interface Loopback0
 ip address 10.200.10.1 255.255.255.0
!
interface Serial8/0
 ip address 10.10.13.1 255.255.255.1

!--- Connected to R3

!
interface Serial10/0
 ip address 10.10.12.1 255.255.255.0

!--- Connected to R2

!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.10.12.2 remote-as 100

!--- Establishes peering with R2

 neighbor 10.10.12.2 next-hop-self
 neighbor 10.10.13.3 remote-as 30

!--- Establishes peering with R3
```

```
 neighbor 10.10.13.3 route-map Peer-R3 in

!--- Configures the inbound policy as defined by route-map "Peer-R3" when peering with R3.

 no auto-summary
!
ip bgp-community new-format

!--- Allows you to configure the BGP community attribute in AA:NN format.

ip community-list 1 permit 100:300
ip community-list 2 permit 100:250

!--- Defines community list 1 and 2.

!
route-map Peer-R3 permit 10
 match community 1
 set local-preference 130

!--- Sets local preference 130 for all routes matching community list 1.

!
route-map Peer-R3 permit 20
 match community 2
 set local-preference 125

!--- Sets local preference 125 for all routes matching community list 2.

!
route-map Peer-R3 permit 30

!--- Without this permit 30 statement, updates that do not match the permit 10 or permit 20 statements are dropped.

!
end
```

R2

```
hostname R2
!
interface Loopback0
ip address 10.0.10.0 255.255.255.0
!
interface Serial9/0
ip address 10.10.23.2 255.255.255.1

!--- Connected to R3

!
interface Serial10/0
ip address 10.10.12.2 255.255.255.0
```

```
!--- Connected to R1

!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.10.12.1 remote-as 100

!--- Establishes iBGP peering with R1

 neighbor 10.10.12.1 next-hop-self
 neighbor 10.10.23.3 remote-as 30

!--- Establishes peering with R3

 neighbor 10.10.23.3 route-map Peer-R3 in

!--- Configures inbound policy as defined by route-map "Peer-R3" when peering with R3.

 no auto-summary
!
ip bgp-community new-format

!--- Allows you to configure the BGP community attribute in AA:NN format.

!
ip community-list 1 permit 100:300
ip community-list 2 permit 100:250

!--- Defines community list 1 and 2.

!
route-map Peer-R3 permit 10
 match community 1
 set local-preference 130

!--- Sets local preference 130 for all routes matching community list 1.

!
route-map Peer-R3 permit 20
 match community 2
 set local-preference 125

!--- Sets local preference 125 for all routes matching community list 2.

!
route-map Peer-R3 permit 30

!--- Without this permit 30 statement, updates that do not match the permit 10 or permit 20 statements are dropped.

!
end
```

Verificación

R1 recibe los prefijos 10.0.10.0/24 y 10.1.0.0/24 con las comunidades 100:300 y 100:250, como se muestra en el siguiente show ip bgp resultado del comando.

Nota: Una vez que estas rutas se instalan en la tabla BGP en función de la política configurada, a los prefijos con la comunidad 100:300 se les asigna la preferencia local 130 y a los prefijos con la comunidad 100:250 se les asigna la preferencia local 125.

<#root>

R1#

show ip bgp 10.0.10.0

```
BGP routing table entry for 10.0.10.0/24, 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 (10.0.10.0)
      Origin IGP, metric 0, localpref 130, valid, external, best
      Community: 100:300

!--- Prefix 10.0.10.0/24 with community 100:300 received from 10.10.13.3 (R3) is assigned local prefere
```

<#root>

R1#

show ip bgp 10.1.0.0

```
BGP routing table entry for 10.1.0.0/24, 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 (10.1.0.0)
      Origin IGP, metric 0, localpref 130, valid, internal, best

!--- Received prefix 10.1.0.0/24 over iBGP from 10.10.12.2 (R2) with local preference 130
```

```
 30
  10.10.13.3 from 10.10.13.3 (198.51.100.1)
    Origin IGP, metric 0, localpref 125, valid, external
    Community: 100:250
```

!--- Prefix 10.1.0.0/24 with community 100:250 received from 10.10.13.3 (R3) is assigned local preference 125.

<#root>

R1#

show ip bgp

```
BGP table version is 4, local router ID is 10.200.10.1
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
*> 10.0.10.0/24    10.10.13.3               0    130      0 30 i
*>i 10.1.0.0/24    10.10.12.2               0    130      0 30 i
*                  10.10.13.3               0    125      0 30 i
```

El show ip bgp comando en R1 confirma que la mejor trayectoria seleccionada en R1 tiene preferencia local (LoclPrf) = 130. De manera similar, R2 recibe los prefijos 10.0.10.0/24 y 10.1.0.0/24 con las comunidades 100:250 y 100:300, como se muestra en negrita en el resultado de este show ip bgp comando:

Nota: Una vez que estas rutas se instalan en la tabla BGP, según la política configurada, a los prefijos con la comunidad 100:300 se les asigna la preferencia local 130 y a los prefijos con la comunidad 100:250 se les asigna la preferencia local 125.

<#root>

R2#

show ip bgp 10.0.10.0

```
BGP routing table entry for 10.0.10.0/24, 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 (10.0.10.0)
      Origin IGP, metric 0, localpref 125, valid, external
      Community: 100:250

!--- Prefix 10.0.10.0/24 with community 100:250 received from 10.10.23.3 (R3) is assigned local prefere
```

30
10.10.12.1 from 10.10.12.1 (10.200.10.1)
Origin IGP, metric 0, localpref 130, valid, internal, best

!--- Received prefix 10.0.10.0/24 over iBGP from 10.10.12.1 (R1) with local preference 130

<#root>

R2#

show ip bgp 10.1.0.0

```
BGP routing table entry for 10.1.0.0/24, 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 (10.1.0.0)
      Origin IGP, metric 0, localpref 130, valid, external, best
      Community: 100:300

!--- Prefix 10.1.0.0/24 with community 100:300 received from 10.10.23.3 (R3) is assigned local preferen
```

<#root>

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
*  10.0.10.0/24      10.10.23.3               0    125      0 30 i
*>i                  10.10.12.1               0    130      0 30 i
*> 10.1.0.0/24       10.10.23.3               0    130      0 30 i
```

Este resultado del show ip bgp comando en R2 confirma que la mejor trayectoria seleccionada en R2 tiene preferencia local (loclPrf) = 130. La ruta IP al prefijo 10.0.10.0/24 prefiere que el link R1-R3 salga de AS 100 hacia AS 30. El show ip route comando en R1 y R2 confirma esta preferencia.

<#root>

R1#

show ip route 10.0.10.0

```
Routing entry for 10.0.10.0/24
  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 10.0.10.0/24 points to next hop 10.10.13.3 which is R3 serial 8/0 in
```

<#root>

R2#

show ip route 10.1.0.0

```
Routing entry for 10.1.0.0/24
  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 10.1.0.0/24 points to next hop R1 (10.10.12.1) on its iBGP link
!--- Thus traffic to network 10.1.0.0/24 from R2 exits through R2-R1 and then R1-R3 link from AS 100 to
```

La ruta IP al prefijo 10.1.0.0/24 prefiere el link R2-R3 para salir de AS 100 hacia AS 30. El show ip route comando en R1 y R2 confirma esta preferencia.

<#root>

R2#

show ip route 10.1.0.0

```
Routing entry for 10.1.0.0/24
  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 10.1.0.0/24 points to next hop 10.10.23.3 which is R3 serial 9/0 interfa
```

<#root>

R1#

show ip route 10.1.0.0

```
Routing entry for 10.1.0.0/24
  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 10.1.0.0/24 points to next hop R2 (10.10.12.2) on its iBGP link.
!--- Thus traffic to network 10.1.0.0/24 from R1 exits through R1-R2 and then R2-R3 link from AS 100 to
```

Si un link falla, por ejemplo el link R1-R3, todo el tráfico debe rastrear el link R2-R3. Puede simular este tráfico si apaga el link entre R1-R3.

<#root>

```
R1#
```

configure terminal

```
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#
```

interface serial8/0

```
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 la tabla de IP Routing para el prefijo 10.0.10.0/24 y 10.1.0.0/24 en R1 y R2. Utilice el link R2-R3 para salir de AS 100.

<#root>

```
R1#
```

show ip route 10.0.10.0

```
Routing entry for 10.0.10.0/24
  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
```

<#root>

```
R1#
```

show ip route 10.1.0.0

```
Routing entry for 10.1.0.0/24
  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 resultado del show comando muestra que la ruta a los prefijos 10.0.10.0/24 y 10.1.0.0/24

apunta al salto siguiente 10.10.12.2, (R2), que se espera. Ahora, observe la tabla de IP Routing en R2 para verificar el siguiente salto del prefijo 10.0.10.0/24 y 10.1.0.0/24. El siguiente salto debe ser R3 para que la política configurada funcione correctamente.

<#root>

R2#

show ip route 10.0.10.0

```
Routing entry for 10.0.10.0/24
  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
```

<#root>

R2#

show ip route 10.1.0.0

```
Routing entry for 10.1.0.0/24
  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
```

El salto siguiente 10.10.23.3 es la interfaz R3 Serial 9/0 en el link R2-R3. Esto confirma que la política configurada funciona como se esperaba.

Información Relacionada

- RFC 1998
- Troubleshooting de BGP
- BGP: Preguntas Frecuentes
- Carga a Compartir con BGP
- Soporte técnico y descargas de Cisco

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).