

Comment les routeurs BGP utilisent le discriminateur de sorties multiples pour la meilleure sélection de chemin

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[L'attribut MED](#)

[Exemple](#)

[La commande `bgp deterministic-med`](#)

[Exemples](#)

[Informations connexes](#)

[Introduction](#)

Ce document explique l'utilisation de la commande **bgp deterministic-med** et explique comment elle peut effectuer la sélection de chemin basée sur le discriminateur de sorties multiples (MED).

[Conditions préalables](#)

[Conditions requises](#)

Aucune spécification déterminée n'est requise pour ce document.

[Composants utilisés](#)

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Les informations présentées dans ce document ont été créées à partir de périphériques dans un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si vous travaillez dans un réseau opérationnel, assurez-vous de bien comprendre l'impact potentiel de toute commande avant de l'utiliser.

[Conventions](#)

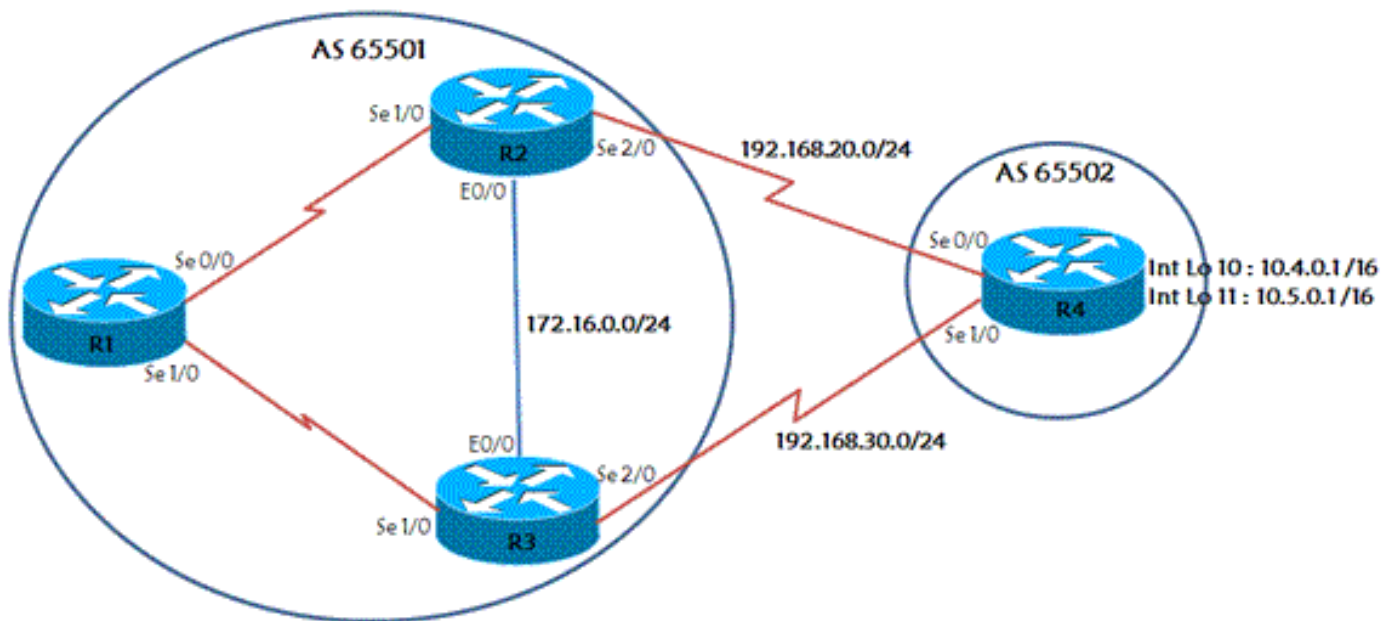
Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous aux

L'attribut MED

MED est un attribut non transitif facultatif. MED est un conseil aux voisins externes au sujet du chemin préféré dans un Autonomous System (AS) qui a des points d'entrée multiples. Le MED est également connu comme le métrique externe d'une route. L'attribut MED ayant la valeur la plus basse est préférée à l'attribut MED ayant la valeur la plus élevée.

Cette section décrit un exemple d'utilisation du MED pour influencer la décision de routage prise par un AS voisin.

Topologie :



Exemple

Dans ce scénario, AS 65502 est un client de l'ISP qui a AS 65501. R4 est connecté à deux routeurs différents du côté de l'ISP pour des raisons de redondance et annonce deux réseaux à l'ISP : 10.4.0.0/16 et 10.5.0.0/16. Une partie de la configuration appropriée est montrée dans cette section.

R4

```
!  
version 12.3  
!  
hostname r4  
!  
ip cef  
!  
!  
interface Loopback10  
 ip address 10.4.0.1 255.255.0.0  
!  
interface Loopback11  
 ip address 10.5.0.1 255.255.0.0  
!
```

```
interface Serial0/0
  ip address 192.168.20.4 255.255.255.0
!
interface Serial1/0
  ip address 192.168.30.4 255.255.255.0
!
router bgp 65502
  no synchronization
  bgp log-neighbor-changes
  network 10.4.0.0 mask 255.255.0.0
  network 10.5.0.0 mask 255.255.0.0
  neighbor 192.168.20.2 remote-as 65501
  neighbor 192.168.30.3 remote-as 65501
  no auto-summary
!
ip classless
!
!
line con 0
  exec-timeout 0 0
line aux 0
line vty 0 4
  exec-timeout 0 0
  login
!
!
end
```

R2

```
!
version 12.3
!
hostname r2
!
ip cef
!
!
interface Loopback0
  ip address 2.2.2.2 255.255.255.255
!
interface Ethernet0/0
  ip address 172.16.0.2 255.255.255.0
!
interface Serial1/0
  ip address 192.168.1.2 255.255.255.0
  serial restart-delay 0
!
interface Serial2/0
  ip address 192.168.20.2 255.255.255.0
  serial restart-delay 0
!
router ospf 1
  log-adjacency-changes
  redistribute connected
  passive-interface Serial2/0
  network 2.2.2.2 0.0.0.0 area 0
  network 172.16.0.2 0.0.0.0 area 0
  network 192.168.1.2 0.0.0.0 area 0
  network 192.168.20.2 0.0.0.0 area 0
!
router bgp 65501
  no synchronization
```

```

bgp log-neighbor-changes
neighbor 1.1.1.1 remote-as 65501
neighbor 1.1.1.1 update-source Loopback0
neighbor 3.3.3.3 remote-as 65501
neighbor 3.3.3.3 update-source Loopback0
neighbor 192.168.20.4 remote-as 65502
no auto-summary
!
ip classless
!
!
line con 0
  exec-timeout 0 0
  transport preferred all
  transport output all
line aux 0
  transport preferred all
  transport output all
line vty 0 4
  exec-timeout 0 0
  login
  transport preferred all
  transport input all
  transport output all
!
end

```

Les configurations de R1 et de R3 sont semblables à R2. R3 a un eBGP homologue à R4 et un iBGP homologue à R1.

R1 a un iBGP homologue à R2 et un iBGP homologue à R3. Regardons ce que les tables BGP R1, R2 et R3 affichent pour les deux réseaux annoncés par R4 :

```

r2# show ip bgp 10.4.0.1
BGP routing table entry for 10.4.0.0/16, version 7
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
    1.1.1.1 3.3.3.3
    65502
      192.168.20.4 from 192.168.20.4 (4.4.4.4)
        Origin IGP, metric 0, localpref 100, valid, external, best
    65502
      192.168.30.4 (metric 74) from 3.3.3.3 (3.3.3.3)
        Origin IGP, metric 0, localpref 100, valid, internal

```

```

r2# show ip bgp 10.5.0.1
BGP routing table entry for 10.5.0.0/16, version 6
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
    1.1.1.1 3.3.3.3
    65502
      192.168.30.4 (metric 74) from 3.3.3.3 (3.3.3.3)
        Origin IGP, metric 0, localpref 100, valid, internal
    65502
      192.168.20.4 from 192.168.20.4 (4.4.4.4)
        Origin IGP, metric 0, localpref 100, valid, external, best

```

```

r3# show ip bgp 10.4.0.1
BGP routing table entry for 10.4.0.0/16, version 8
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:

```

```
1.1.1.1 2.2.2.2
65502
  192.168.20.4 (metric 74) from 2.2.2.2 (2.2.2.2)
    Origin IGP, metric 0, localpref 100, valid, internal
65502
  192.168.30.4 from 192.168.30.4 (4.4.4.4)
    Origin IGP, metric 0, localpref 100, valid, external, best
```

r3# **show ip bgp 10.5.0.1**

```
BGP routing table entry for 10.5.0.0/16, version 10
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Advertised to non-peer-group peers:
    1.1.1.1 2.2.2.2
      65502
        192.168.30.4 from 192.168.30.4 (4.4.4.4)
          Origin IGP, metric 0, localpref 100, valid, external, best
      65502
        192.168.20.4 (metric 74) from 2.2.2.2 (2.2.2.2)
          Origin IGP, metric 0, localpref 100, valid, internal
```

r1# **show ip bgp 10.4.0.1**

```
BGP routing table entry for 10.4.0.0/16, version 11
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
    65502
      192.168.20.4 (metric 128) from 2.2.2.2 (2.2.2.2)
        Origin IGP, metric 0, localpref 100, valid, internal, best
    65502
      192.168.30.4 (metric 128) from 3.3.3.3 (3.3.3.3)
        Origin IGP, metric 0, localpref 100, valid, internal
```

r1# **show ip bgp 10.5.0.1**

```
BGP routing table entry for 10.5.0.0/16, version 10
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Not advertised to any peer
    65502
      192.168.30.4 (metric 128) from 3.3.3.3 (3.3.3.3)
        Origin IGP, metric 0, localpref 100, valid, internal
    65502
      192.168.20.4 (metric 128) from 2.2.2.2 (2.2.2.2)
        Origin IGP, metric 0, localpref 100, valid, internal, best
```

Comme nous pouvons le voir, R2 et R3 choisissent tous deux comme meilleur chemin la route externe depuis R4 prévue d'après l'algorithme de sélection du meilleur chemin BGP. Reportez-vous à l'[Algorithme de sélection du meilleur chemin BGP](#) pour plus d'informations.

De même, R1 choisit R2 pour accéder aux 2 réseaux, ce qui est conforme à la règle de meilleur chemin BGP : sélectionner le chemin d'ID de routeur le plus bas, quand tous les autres paramètres sont équivalents. Puisque l'ID du routeur R2 est 2.2.2.2 et l'ID du routeur R3 est 3.3.3.3, R2 est choisi. Dans cette configuration de base, tout le trafic vers les deux réseaux dans AS 65502 passe de R1 par R2, puis à R4 par défaut. Supposez maintenant que R4 souhaite équilibrer la charge du trafic qu'il reçoit d'AS 65501. Pour ce faire sans demander à l'ISP R4 d'apporter des modifications, vous pouvez configurer R4 pour qu'il utilise le MED pour obliger le trafic d'un réseau à traverser un chemin et le trafic de l'autre réseau à traverser l'autre chemin.

Voici la configuration de R4 après avoir appliqué la configuration nécessaire :

R4
!
version 12.3

```

!
hostname r4
!
ip cef
!
!
!
interface Loopback10
 ip address 10.4.0.1 255.255.0.0
!
interface Loopback11
 ip address 10.5.0.1 255.255.0.0
!
interface Serial10/0
 ip address 192.168.20.4 255.255.255.0
!
interface Serial11/0
 ip address 192.168.30.4 255.255.255.0
!
router bgp 65502
 no synchronization
 bgp log-neighbor-changes
 network 10.4.0.0 mask 255.255.0.0
 network 10.5.0.0 mask 255.255.0.0
 neighbor 192.168.20.2 remote-as 65501
 neighbor 192.168.20.2 route-map setMED-R2 out
 neighbor 192.168.30.3 remote-as 65501
 neighbor 192.168.30.3 route-map setMED-R3 out
 no auto-summary
!
ip classless
no ip http server
!
!
access-list 1 permit 10.4.0.0 0.0.255.255
access-list 2 permit 10.5.0.0 0.0.255.255
!
route-map setMED-R3 permit 10
  match ip address 1
  set metric 200
!
route-map setMED-R3 permit 20
  match ip address 2
  set metric 100
!--- The route-map MED-R3 is applying a MED of 200 to the 10.4.0.0/16 !--- network and a MED of 100 to the 10.5.0.0/16 network. !--- The route-map is being applied outbound towards R3. !
route-map setMED-R2 permit 10
  match ip address 1
  set metric 100
!
route-map setMED-R2 permit 20
  match ip address 2
  set metric 200
!--- The route-map MED-R2 is applying a MED of 100 to the 10.4.0.0/16 !--- network and a MED of 200 to the 10.5.0.0/16 network. !--- The route-map is being applied outbound towards R2. !!! line con 0 exec-timeout 0 0 line aux 0 line vty 0 4 exec-timeout 0 0 login !! end

```

Note: Vous devez effacer la session BGP avec la commande **clear ip bgp * soft out** , par exemple, pour que ces configurations agissent.

R1 considère maintenant la route passant par R2 comme le meilleur chemin pour le réseau 10.4.0.0/16 parce que la mise à jour reçue de R2 a un MED de 100, contre un MED de 200 pour R3. De même, R1 utilise R3 et la liaison R3 - R4 pour accéder à 10.5.0.0/16 :

```
r1# show ip bgp 10.4.0.1
BGP routing table entry for 10.4.0.0/16, version 14
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Flag: 0x800
    Not advertised to any peer
    65502
        192.168.20.4 (metric 128) from 2.2.2.2 (2.2.2.2)
            Origin IGP, metric 100, localpref 100, valid, internal, best
r1#sh ip bgp 10.5.0.1
BGP routing table entry for 10.5.0.0/16, version 13
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Flag: 0x800
    Not advertised to any peer
    65502
        192.168.30.4 (metric 128) from 3.3.3.3 (3.3.3.3)
            Origin IGP, metric 100, localpref 100, valid, internal, best
```

Regardons ce qu'affiche R2 :

```
r2# show ip bgp 10.4.0.1
BGP routing table entry for 10.4.0.0/16, version 10
Paths: (1 available, best #1, table Default-IP-Routing-Table)
    Advertised to non peer-group peers:
    1.1.1.1 3.3.3.3
    65502
        192.168.20.4 from 192.168.20.4 (4.4.4.4)
            Origin IGP, metric 100, localpref 100, valid, external, best

r2# show ip bgp 10.5.0.1
BGP routing table entry for 10.5.0.0/16, version 11
Paths: (2 available, best #1, table Default-IP-Routing-Table)
    Advertised to non peer-group peers:
    192.168.20.4
    65502
        192.168.30.4 (metric 74) from 3.3.3.3 (3.3.3.3)
            Origin IGP, metric 100, localpref 100, valid, internal, best
    65502
        192.168.20.4 from 192.168.20.4 (4.4.4.4)
            Origin IGP, metric 200, localpref 100, valid, external
```

La raison pour laquelle R2 montre seulement un chemin 10.4.0.0/16 est que R3 retire (envoie une mise à jour avec une métrique inaccessible) la mise à jour pour 10.4.0.0/16 une fois qu'il note que R3 utilise R2 pour accéder à 10.4.0.0/16 (après avoir exécuté le meilleur chemin BGP sur tous les chemins disponibles) :

```
r3# show ip bgp 10.4.0.0
BGP routing table entry for 10.4.0.0/16, version 20
Paths: (2 available, best #1, table Default-IP-Routing-Table)
    Advertised to non peer-group peers:
    192.168.30.4
    65502
        192.168.20.4 (metric 74) from 2.2.2.2 (2.2.2.2)
            Origin IGP, metric 100, localpref 100, valid, internal, best
    65502
        192.168.30.4 from 192.168.30.4 (4.4.4.4)
```

```
Origin IGP, metric 200, localpref 100, valid, external
```

Ceci permet à R2 d'économiser de la mémoire puisqu'il ne doit pas stocker ces informations inutiles. Au cas où la session BGP entre R2 et R4 échouerait, R2 enverrait une mise à jour inaccessible à R3 pour 10.4.0.0/16. Cette mise à jour conduirait R3 à envoyer une mise à jour avec la route R3 pour 10.4.0.0/16 par l'intermédiaire de R4 à R2. R2 a pu commencer le routage par l'intermédiaire de R3.

[La commande bgp deterministic-med](#)

L'activation de la commande de **bgp deterministic-med** retire n'importe quelle dépendance temporelle de meilleures décisions basées sur med de chemin. Il s'assure qu'une comparaison précise de MED est faite à travers toutes les routes reçues du même Autonomous System (QUE).

Si vous désactivez **bgp deterministic-med**, l'ordre dans lequel les routes sont reçues peut affecter les décisions de meilleur chemin basé sur MED. Ceci peut se produire quand la même route est reçue depuis plusieurs AS ou depuis des sous-AS de confédération, avec exactement la même longueur de chemin mais différents MED.

[Exemples](#)

Par exemple, considérez les routes suivantes :

```
r3# show ip bgp 10.4.0.0
BGP routing table entry for 10.4.0.0/16, version 20
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  192.168.30.4
  65502
    192.168.20.4 (metric 74) from 2.2.2.2 (2.2.2.2)
      Origin IGP, metric 100, localpref 100, valid, internal, best
  65502
    192.168.30.4 from 192.168.30.4 (4.4.4.4)
      Origin IGP, metric 200, localpref 100, valid, external
```

L'ordre de réception des routes BGP est entry3, entry2 et entry1 (entry3 est l'entrée la plus ancienne dans la table BGP et entry1 est la plus récente).

[Un routeur BGP avec la commande bgp deterministic-med désactivée](#)

Un routeur BGP avec la commande **bgp deterministic-med** désactivée choisit entry2 et non entry1, en raison d'une métrique d'IGP inférieure pour atteindre le NEXT_HOP (le MED n'a pas été utilisé dans cette décision parce qu'entry1 et entry2 proviennent de deux AS différents). Il préfère alors entry3 à entry2 parce qu'il est externe. Cependant, entry3 a un MED plus élevé qu'entry1. Pour plus d'informations sur le critère de sélection de chemin BGP, référez-vous à l'[algorithme de sélection du meilleur chemin BGP](#).

[Un routeur BGP avec la commande bgp deterministic-med activée](#)

Dans ce cas, les routes provenant du même AS sont groupées ensemble et les meilleures entrées de chaque groupe sont comparées. Dans l'exemple donné, il y a deux AS, AS 1 et AS 2.

Group 1: **entry1: ASPATH 1, MED 100, internal, IGP metric to NEXT_HOP 10**
entry3: ASPATH 1, MED 200, external

Group 2: **entry2: ASPATH 2, MED 150, internal, IGP metric to NEXT_HOP 5**

Dans le groupe 1, le meilleur chemin est entry1 en raison du MED inférieur (le MED est utilisé dans cette décision puisque les chemins viennent des mêmes AS). Dans le groupe 2, il y a seulement une entrée (entry2). Le meilleur chemin alors est déterminé en comparant les gagnants de chaque groupe (par défaut, le MED n'est pas utilisé dans cette comparaison parce que les gagnants de chaque groupe proviennent d'AS différents - activer **bgp always-compare-med** modifie ce comportement par défaut). Désormais, lorsqu'entry1 (le gagnant du groupe 1) est comparé avec entry2 (le gagnant du groupe 2), entry2 est le gagnant puisqu'il a la meilleure métrique d'IGP vers le relais suivant.

Si **bgp always-compare-med** est également activé, lorsqu'entry1 (le gagnant du groupe 1) est comparé avec entry2 (le gagnant du groupe 2), entry1 est le gagnant en raison du Med inférieur.

Cisco recommande d'activer **bgp deterministic-med** dans tous les nouveaux déploiements du réseau. En outre, si la commande **bgp always-compare-med** est activée, les décisions MED de BGP sont toujours déterministes.

Pour plus d'informations sur les commandes **bgp deterministic-med** et **bgp always-compare-med**, consultez [En quoi la commande bgp deterministic-med diffère de la commande bgp always-compare-med](#).

[Informations connexes](#)

- [Exemple de configuration pour BGP avec deux fournisseurs de services différents \(multihébergement\)](#)
- [Page de support BGP](#)
- [Commandes BGP](#)
- [Études de cas BGP](#)
- [Support et documentation techniques - Cisco Systems](#)