Fallstudien zum Border Gateway Protocol (BGP)

Inhalt

Einleitung

Dieses Dokument enthält eine Beschreibung von fünf Kundenreferenzen zum Border Gateway Protocol (BGP).

Voraussetzungen

Anforderungen

Es gibt keine spezifischen Anforderungen für dieses Dokument.

Verwendete Komponenten

Dieses Dokument ist nicht auf bestimmte Software- und Hardware-Versionen beschränkt.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie unter Cisco Technical Tips Conventions (Technische Tipps von Cisco zu Konventionen).

BGP-Anwenderberichte 1

Das BGP, das in RFC 1771 definiert ist, ermöglicht das schleifenfreie Interdomain-Routing zwischen autonomen Systemen (ASs). Ein AS ist ein Router-Set, das einer einzelnen technischen Administration unterliegt. Router in einem AS können mehrere Interior Gateway Protocols (IGPs) verwenden, um Routing-Informationen innerhalb des AS auszutauschen. Die Router können ein externes Gateway-Protokoll verwenden, um Pakete außerhalb des AS weiterzuleiten.

Wie funktioniert BGP?

BGP verwendet TCP als Transportprotokoll auf Port 179. Zwei BGP-Router bilden eine TCP-Verbindung zwischen ihnen. Bei diesen Routern handelt es sich um Peer-Router. Die Peer-Router tauschen Nachrichten aus, um die Verbindungsparameter zu öffnen und zu bestätigen.

BGP-Router tauschen Informationen zur Netzwerkerreichbarkeit aus. Diese Informationen sind in erster Linie ein Hinweis auf die vollständigen Pfade, die eine Route nehmen muss, um das Zielnetzwerk zu erreichen. Die Pfade sind BGP-AS-Nummern. Diese Informationen helfen bei der Erstellung eines Diagramms von ASs, die schleifenfrei sind. Das Diagramm zeigt auch, wo Routingrichtlinien angewendet werden müssen, um einige Einschränkungen für das Routingverhalten durchzusetzen.

Bei zwei beliebigen Routern, die eine TCP-Verbindung zum Austausch von BGP-Routing-Informationen bilden, handelt es sich um Peers oder Nachbarn. BGP-Peers tauschen zunächst die vollständigen BGP-Routing-Tabellen aus. Nach diesem Austausch senden die Peers inkrementelle Updates, wenn sich die Routing-Tabelle ändert. BGP speichert eine Versionsnummer der BGP-Tabelle. Die Versionsnummer ist für alle BGP-Peers identisch. Die Versionsnummer ändert sich, wenn BGP die Tabelle mit geänderten Routing-Informationen aktualisiert. Das Senden von Keepalive-Paketen stellt sicher, dass die Verbindung zwischen den BGP-Peers aktiv ist. Benachrichtigungspakete werden aufgrund von Fehlern oder besonderen Bedingungen versendet.

eBGP und iBGP

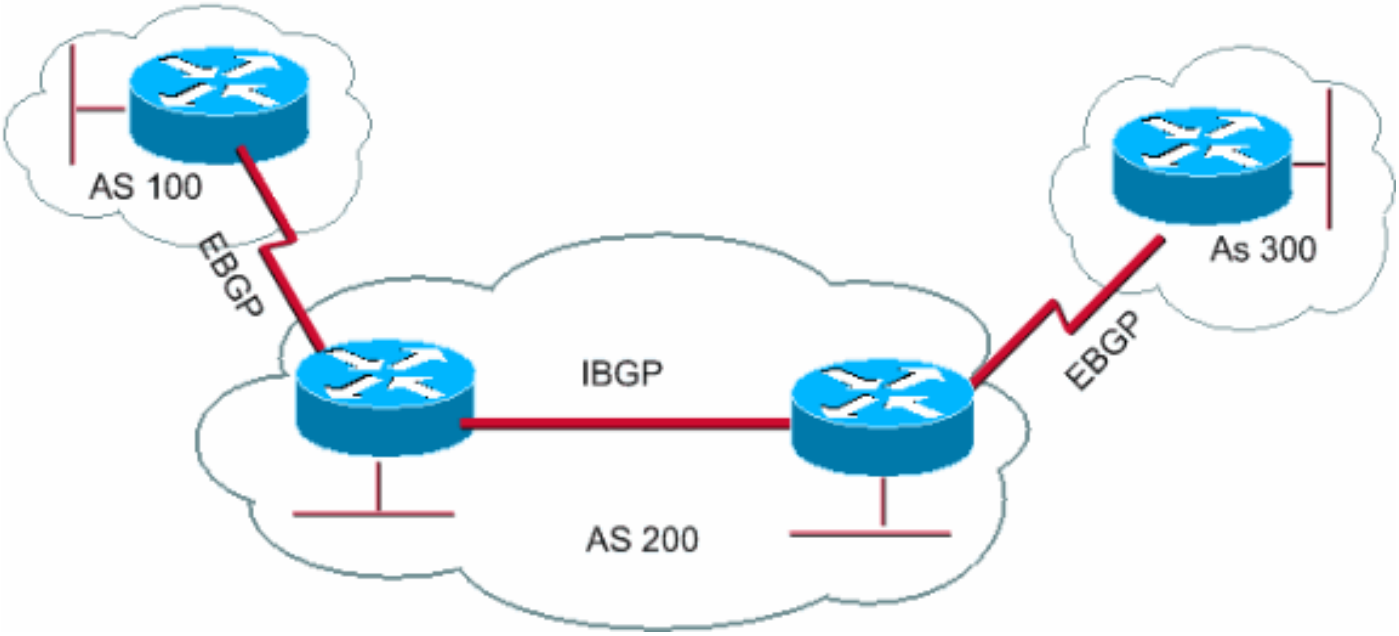
Wenn ein AS über mehrere BGP-Router verfügt, kann das AS als Transit-Service für andere ASs dienen. Wie das nächste Diagramm in diesem Abschnitt zeigt, ist AS200 ein Übertragungs-AS für AS100 und AS300.

Um die Informationen an externe AS zu senden, muss die Erreichbarkeit der Netzwerke gewährleistet sein. Um die Netzwerkerreichbarkeit sicherzustellen, werden folgende Prozesse durchgeführt:

- Internes BGP (iBGP)-Peering zwischen Routern innerhalb eines AS
- Neuverteilung von BGP-Informationen an IGPs, die im AS ausgeführt werden

Wenn BGP zwischen Routern ausgeführt wird, die zu zwei unterschiedlichen AS gehören, wird dies als externes BGP (eBGP) bezeichnet. Wenn BGP zwischen Routern im selben AS ausgeführt

wird, wird dies als iBGP bezeichnet.



BGP wird zwischen Routern im selben AS ausgeführt

BGP-Routing aktivieren

Führen Sie diese Schritte aus, um BGP zu aktivieren und zu konfigurieren.

Angenommen, Sie möchten zwei Router, RTA und RTB, nutzen, die über BGP kommunizieren. Im ersten Beispiel befinden sich RTA und RTB in unterschiedlichen AS. Im zweiten Beispiel gehören beide Router demselben AS.

1. Definieren Sie den Router-Prozess und die AS-Nummer, zu der die Router gehören.

   Geben Sie diesen Befehl ein, um BGP auf einem Router zu aktivieren:

   <#root>

   router bgp <autonomous-system>

   ```
   RTA#
   router bgp 100

   RTB#
   router bgp 200
   ```

   Diese Anweisungen weisen darauf hin, dass RTA BGP ausführt und zu AS100 gehört. RTB läuft mit BGP und gehört zu AS200.

2. Definieren Sie BGP-Nachbarn.

   Die BGP-Nachbarformation gibt die Router an, die versuchen, über das BGP zu kommunizieren. Im nächsten Abschnitt wird dieser Prozess erläutert.

Formular BGP-Nachbarn

Zwei BGP-Router werden zu Nachbarn, nachdem die Router eine TCP-Verbindung untereinander hergestellt haben. Die TCP-Verbindung ist wichtig, damit die beiden Peer-Router den Austausch von Routing-Updates starten können.

Nach dem Herstellen der TCP-Verbindung senden die Router offene Nachrichten, um Werte auszutauschen. Die von den Routern ausgetauschten Werte umfassen die AS-Nummer, die von den Routern ausgeführte BGP-Version, die BGP-Router-ID und die Haltezeit. Nach der Bestätigung und Annahme dieser Werte erfolgt der Aufbau der Nachbarverbindung. Jeder andere Status als "`Established`" weist darauf hin, dass die beiden Router keine Nachbarn geworden sind und dass die Router keine BGP-Updates austauschen können.

Führen Sie diesen neighbor Befehl aus, um eine TCP-Verbindung herzustellen:

```
<#root>
```

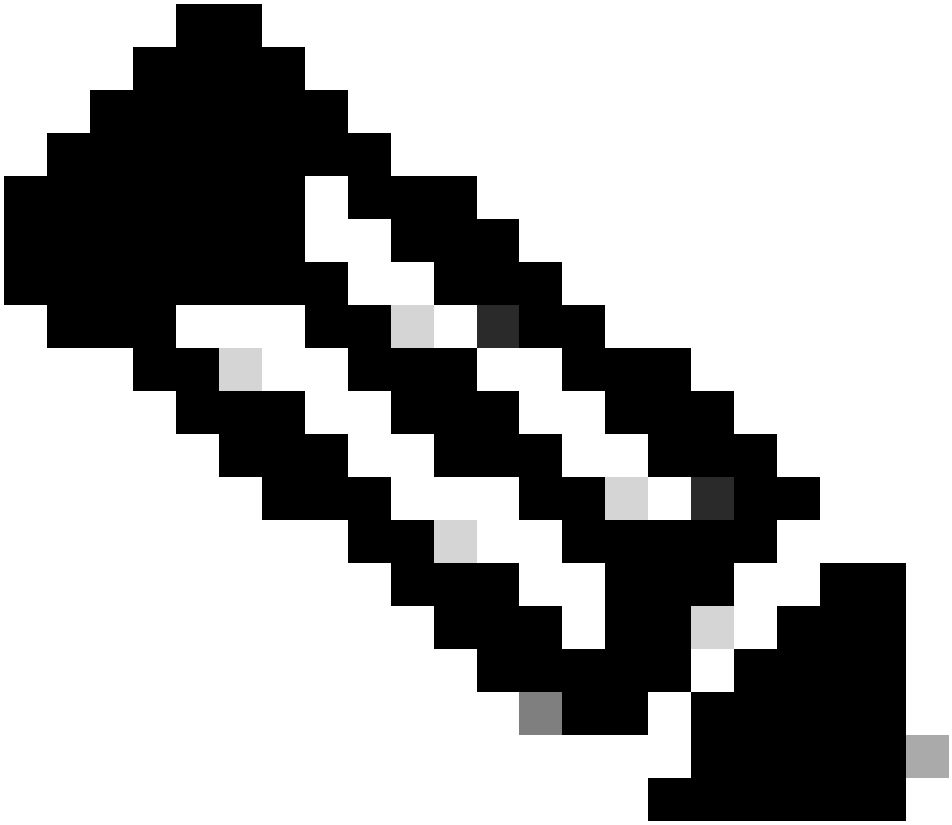
neighbor <ip-address> remote-as <number>

Die ***Nummer*** im Befehl ist die AS-Nummer des Routers, mit dem Sie eine Verbindung zum BGP herstellen möchten. Die ***IP-Adresse*** ist die Next-Hop-Adresse mit direkter Verbindung für das eBGP. Bei iBGP ist ***ip-address*** jede IP-Adresse des anderen Routers.

Die beiden IP-Adressen, die Sie für den neighbor Befehl des Peer-Routers verwenden, *müssen* sich gegenseitig erreichen können. Eine Möglichkeit, die Erreichbarkeit zu überprüfen, ist ein erweiterter Ping zwischen den beiden IP-Adressen. Der erweiterte Ping-Befehl erzwingt, dass der Ping-Router die im Befehl angegebene IP-Adresse als Quelle verwendet neighbor . Der Router muss diese Adresse anstelle der IP-Adresse der Schnittstelle verwenden, von der das Paket ausgeht.

Wenn BGP-Konfigurationsänderungen vorgenommen werden, *müssen* Sie die Nachbarverbindung zurücksetzen, damit die neuen Parameter wirksam werden. .

- clear ip bgp address



Hinweis: Die Adresse ist die Adresse des Nachbarn.

•

clear ip bgp *

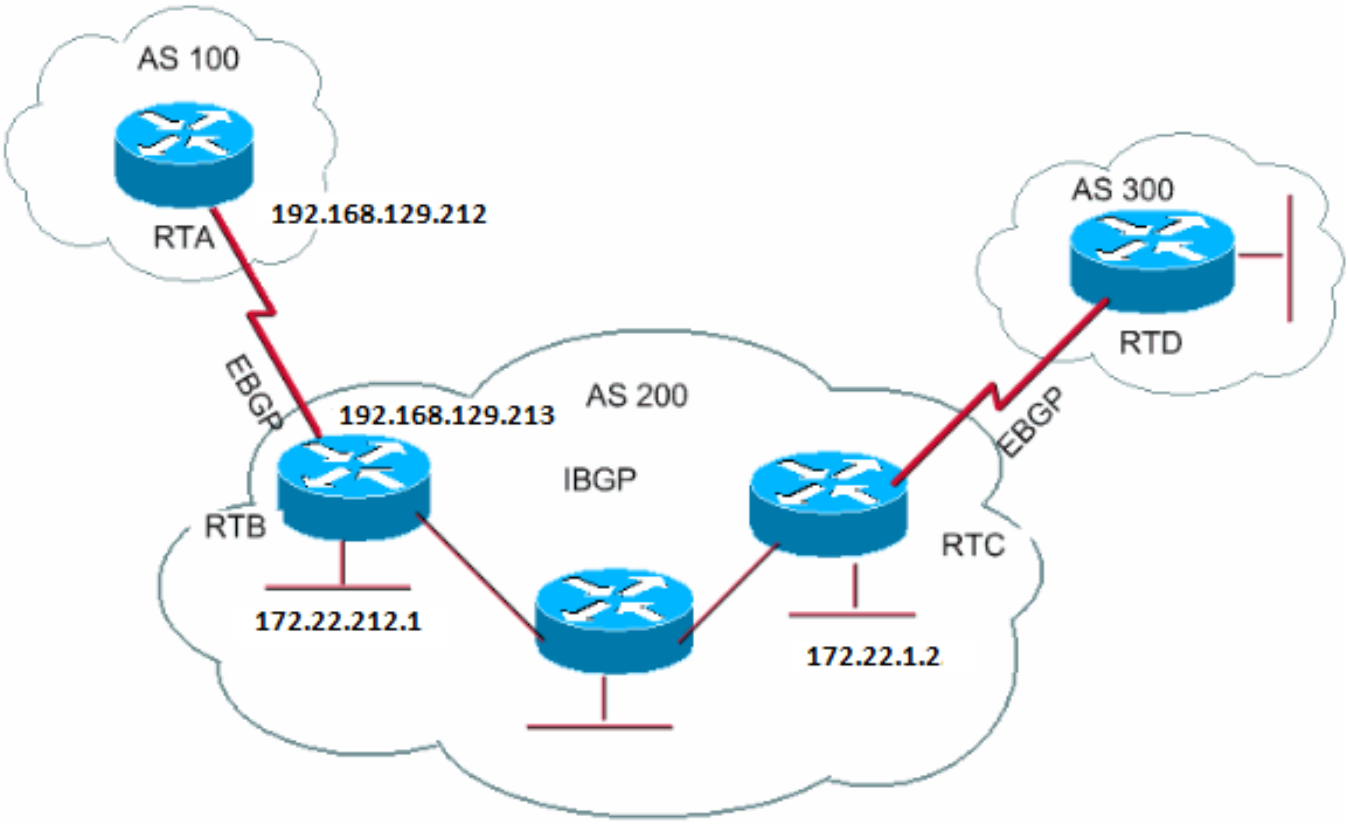
Mit diesem Befehl werden alle Nachbarverbindungen gelöscht.

BGP-Sitzungen beginnen standardmäßig mit der BGP-Version 4 und handeln bei Bedarf nach unten bis zu früheren Versionen aus. Sie können Aushandlungen verhindern und die von den Routern verwendete BGP-Version zwingen, mit einem Nachbarn zu kommunizieren. Geben Sie diesen Befehl im Router-Konfigurationsmodus ein:

<#root>

neighbor {ip address | peer-group-name} version <value>

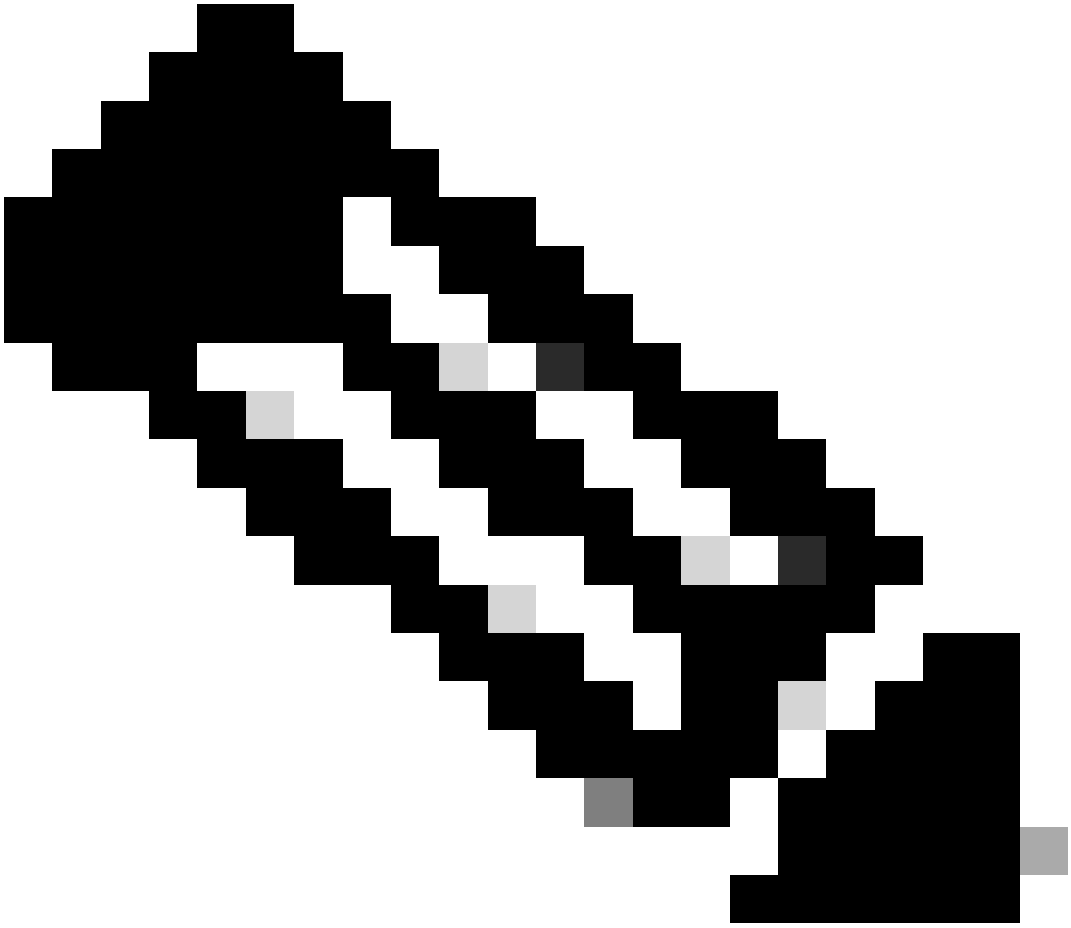
Nachfolgend finden Sie ein Beispiel für die neighbor Befehlskonfiguration:



```
RTA#
router bgp 100
 neighbor 192.168.129.213 remote-as 200

RTB#
router bgp 200
 neighbor 192.168.129.212 remote-as 100
 neighbor 172.22.1.2 remote-as 200

RTC#
router bgp 200
 neighbor 172.22.212.1 remote-as 200
```

In diesem Beispiel wird eBGP auf RTA und RTB ausgeführt. RTB und RTC führen iBGP aus. Die Remote-AS-Nummer verweist auf ein externes oder ein internes AS und gibt damit entweder eBGP oder iBGP an. Die eBGP-Peers haben eine direkte Verbindung, die iBGP-Peers haben jedoch keine direkte Verbindung. iBGP-Router benötigen keine direkte Verbindung. Es muss jedoch ein IGP vorhanden sein, das ausgeführt wird und es den beiden Nachbarn ermöglicht, einander zu erreichen.

Dieser Abschnitt enthält ein Beispiel für die Informationen, die mit dem Befehl **show ip bgp neighbors** angezeigt werden.



Hinweis: Achten Sie besonders auf den BGP-Status. Jeder andere Zustand als "Etabliert" weist darauf hin, dass die Peers nicht verfügbar sind. Beachten Sie auch die folgenden Punkte:

- Die `BGP-Version` mit 4
- Die `ID des Remote-Routers`

Diese Nummer ist die höchste IP-Adresse des Routers oder, falls vorhanden, die höchste Loopback-Schnittstelle.

•

Die `Tabellenversion`

Die `Tabellenversion` gibt den Zustand der Tabelle an. Bei jedem Eintreffen neuer Informationen wird die Version in der Tabelle erhöht. Eine Version, die weiterhin inkrementiert, gibt an, dass es eine Route-Flap gibt, die die kontinuierliche Aktualisierung von Routen verursacht.

<#root>

```
Router#
```

show ip bgp neighbors

```
  BGP neighbor is 192.168.129.213, remote AS 200, external link
  BGP version 4, remote router ID 172.22.12.1
```

BGP state = Established

```
, table version = 3, up for 0:10:59
  Last read 0:00:29, hold time is 180, keepalive interval is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Received 2828 messages, 0 notifications, 0 in queue
  Sent 2826 messages, 0 notifications, 0 in queue
  Connections established 11; dropped 10
```

BGP- und Loopback-Schnittstellen

Die Verwendung einer Loopback-Schnittstelle zur Definition von Nachbarn ist beim iBGP üblich, beim eBGP jedoch nicht. Normalerweise

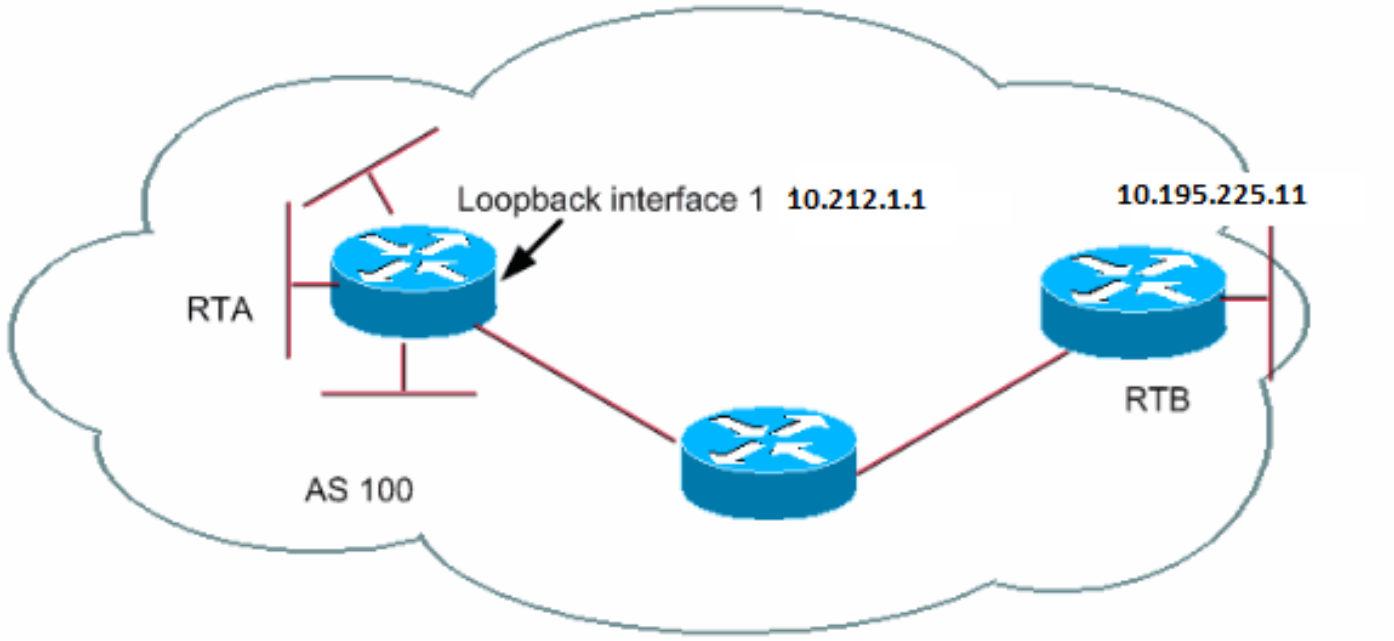
verwenden Sie die Loopback-Schnittstelle, um sicherzustellen, dass die IP-Adresse des Nachbarn erhalten bleibt und von der ordnungsgemäß funktionierenden Hardware unabhängig ist. Im Fall von eBGP haben Peer-Router häufig eine direkte Verbindung, und Loopback ist nicht anwendbar.

Wenn Sie die IP-Adresse einer Loopback-Schnittstelle im neighbor Befehl verwenden, benötigen Sie eine zusätzliche Konfiguration auf dem benachbarten Router. Der Nachbar-Router muss das BGP über die Verwendung einer Loopback-Schnittstelle anstelle einer physischen Schnittstelle informieren, um die BGP-Nachbar-TCP-Verbindung zu initiieren. Führen Sie den folgenden Befehl aus, um eine Loopback-Schnittstelle anzugeben:

```
<#root>
```

neighbor <ip-address> update-source <interface>

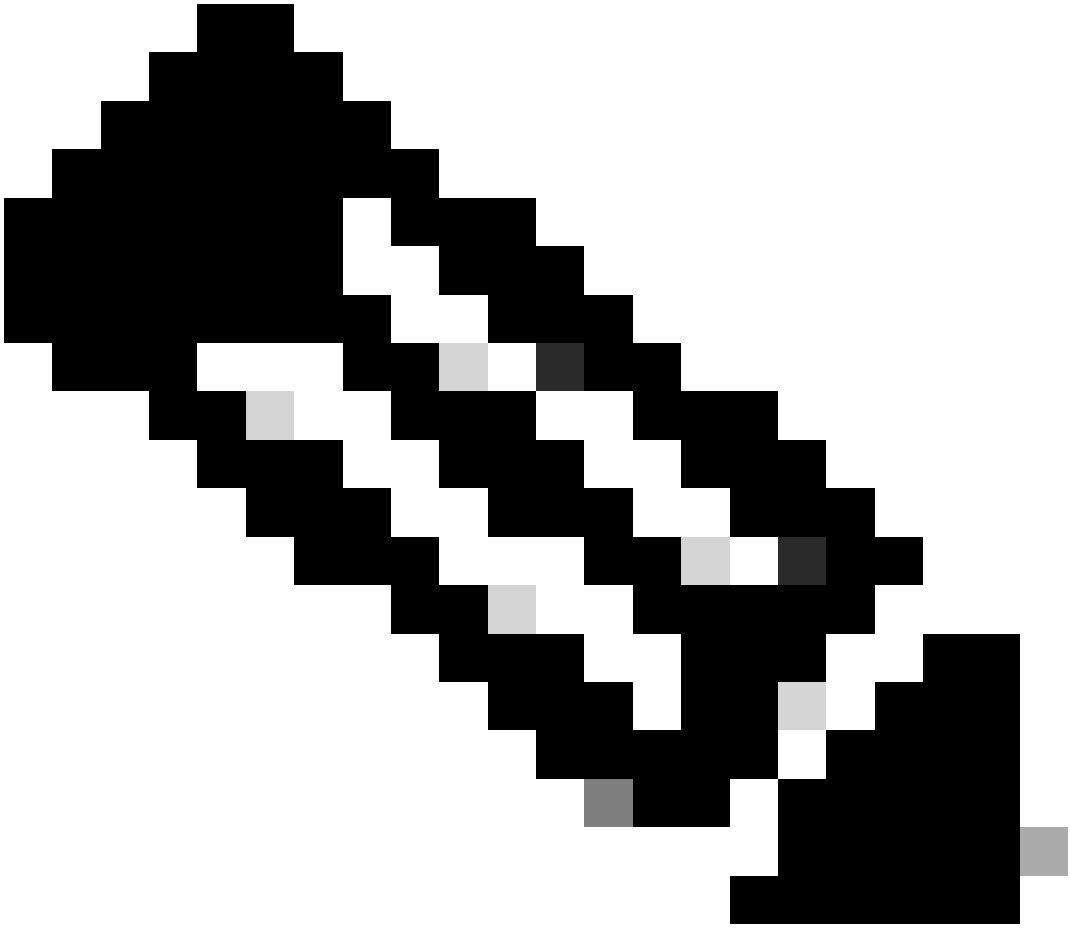
In diesem Beispiel wird die Verwendung des folgenden Befehls veranschaulicht:




```
RTA#
router bgp 100
 neighbor 10.195.225.11 remote-as 100
 neighbor 10.195.225.11 update-source loopback 1
```

```
RTB#
router bgp 100
 neighbor 10.212.1.1 remote-as 100
```

In diesem Beispiel wird iBGP in AS100 von RTA und RTB ausgeführt. Im neighbor Befehl verwendet RTB die Loopback-Schnittstelle von RTA, 10.212.1.1. In diesem Fall muss die RTA das BGP zwingen, die Loopback-IP-Adresse als Quelle in der TCP-Nachbarverbindung zu verwenden. Um diese Aktion zu erzwingen, fügt RTA **update-source interface-type interface-number** neighbor 10.195.225.11 update-source loopback 1 so hinzu, dass der Befehl ausgeführt wird. Diese Anweisung erzwingt, dass BGP die IP-Adresse der Loopback-Schnittstelle verwendet, wenn BGP mit dem Nachbarn 10.195.225.11 kommuniziert. Diese Anweisung erzwingt, dass BGP die IP-Adresse der Loopback-Schnittstelle verwendet.

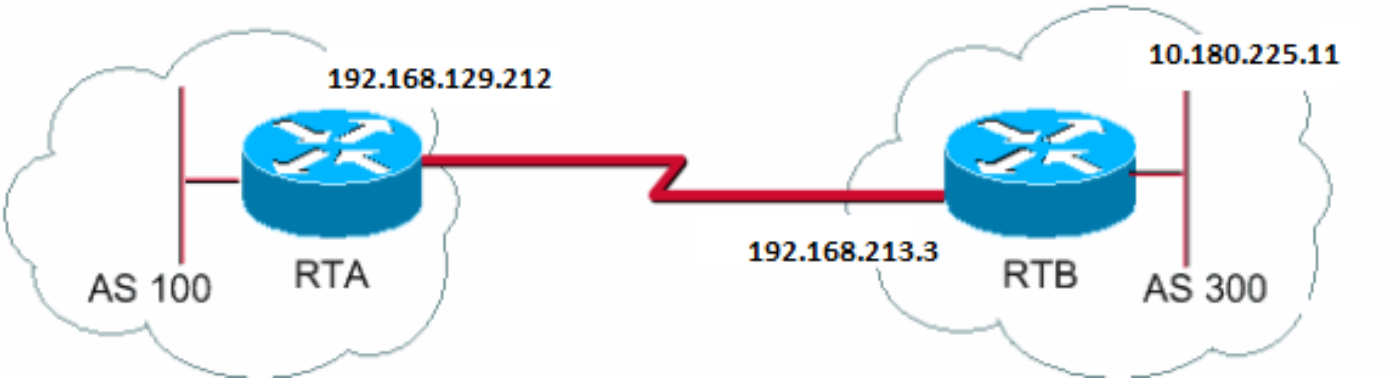


Hinweis: Die RTA hat die IP-Adresse der physischen Schnittstelle von RTB (10.195.225.11) als Nachbar verwendet. Die Verwendung dieser IP-Adresse macht eine spezielle Konfiguration der RTB überflüssig. Eine vollständige Beispielkonfiguration für das Netzwerkszenario finden Sie unter Beispielkonfiguration für iBGP und eBGP mit oder ohne Loopback-Adresse.

---

eBGP-Multihop

In einigen Fällen kann ein Cisco Router eBGP mit einem Drittanbieter-Router ausführen, der keine direkte Verbindung der beiden externen Peers zulässt. Um die Verbindung herzustellen, können Sie eBGP-Multihop verwenden. Der eBGP-Multihop ermöglicht eine Nachbarverbindung zwischen zwei externen Peers ohne direkte Verbindung. Der Multihop ist nur für eBGP und nicht für iBGP. Dieses Beispiel zeigt den eBGP-Multihop:

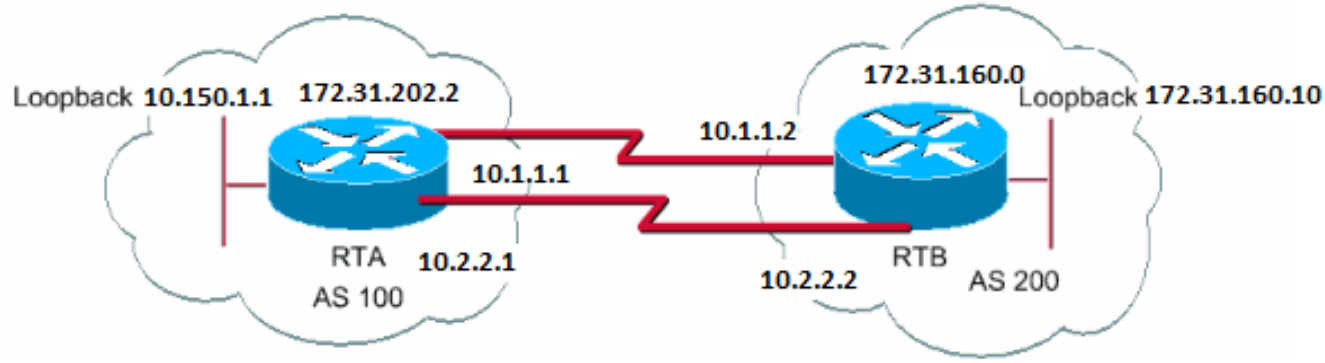



```
RTA#
router bgp 100
 neighbor 10.180.225.11 remote-as 300
 neighbor 10.180.225.11 ebgp-multihop

RTB#
router bgp 300
 neighbor 192.168.129.212 remote-as 100
```

RTA gibt einen externen Nachbarn an, der keine direkte Verbindung hat. RTA muss die Verwendung des Befehls neighbor ebgp-multihop angeben. Andererseits gibt RTB einen Nachbarn an, der eine direkte Verbindung hat, nämlich 192.168.129.212. Aufgrund dieser direkten Verbindung benötigt die RTB den neighbor ebgp-multihop Befehl nicht. Sie müssen auch ein IGP oder statisches Routing konfigurieren, damit Nachbarn ohne Verbindung sich gegenseitig erreichen können.

Das Beispiel im Abschnitt eBGP-Multihop (Load Balancing) zeigt, wie ein Load Balancing mit BGP durchgeführt wird, wenn das eBGP über parallele Leitungen betrieben wird.

eBGP Multihop (Lastenausgleich)




```
RTA#
int loopback 0
 ip address 10.150.1.1 255.255.255.0

router bgp 100
 neighbor 172.31.160.10 remote-as 200
 neighbor 172.31.160.10 ebgp-multihop
 neighbor 172.31.160.10 update-source loopback 0
 network 172.31.202.2

ip route 172.31.160.0 255.255.0.0 10.1.1.2
ip route 172.31.160.0 255.255.0.0 10.2.2.2

RTB#
int loopback 0
 ip address 172.31.160.10 255.255.255.0

router bgp 200
 neighbor 10.150.1.1 remote-as 100
 neighbor 10.150.1.1 update-source loopback 0
 neighbor 10.150.1.1 ebgp-multihop
 network 172.31.160.0

ip route 172.31.202.2 255.255.0.0 10.1.1.1
ip route 172.31.202.2 255.255.0.0 10.2.2.1
```

In diesem Beispiel wird die Verwendung von Loopback-Schnittstellen update-source, und ebgp-multihop veranschaulicht. Das Beispiel dient als Workaround, um einen Lastenausgleich zwischen zwei eBGP-Routern über parallele serielle Leitungen zu erreichen. In normalen Situationen wählt das BGP eine der Leitungen aus, an die Pakete gesendet werden sollen, und Load Balancing findet nicht statt. Mit der Einführung von Loopback-Schnittstellen ist der nächste Hop für eBGP die Loopback-Schnittstelle. Sie verwenden statische Routen oder ein IGP, um zwei kostengünstige Pfade zum Ziel einzuführen. Für den nächsten Hop 172.31.160.10 hat die RTA zwei Möglichkeiten: einen Pfad über 10.1.1.2 und den anderen über 10.2.2.2. Die RTB hat die gleichen Optionen.

Routenübersichten

Routing-Karten werden beim BGP intensiv verwendet. Im BGP-Kontext ist die Routing-Zuordnung eine Methode zur Kontrolle und Änderung von Routing-Informationen. Die Steuerung und Änderung von Routing-Informationen erfolgt durch die Definition von Bedingungen für die Routen-Neuverteilung von einem Routing-Protokoll zu einem anderen. Die Steuerung der Routing-Informationen kann auch beim Einschleusen

in das BGP oder beim Verlassen des BGP erfolgen. Das Format der Routenübersicht ist also wie folgt:

<#root>

route-map map-tag [[permit | deny] | [sequence-number]]

Das Map-Tag ist einfach ein Name, den Sie der Route Map geben. Sie können mehrere Instanzen derselben Routenübersicht oder desselben Namenskennzeichens definieren. Die Sequenznummer ist lediglich ein Hinweis auf die Position, die eine neue Routenübersicht in der Liste der Routenübersichten haben soll, die Sie bereits mit dem gleichen Namen konfiguriert haben.

In diesem Beispiel werden zwei Instanzen der Routenzuordnung mit dem Namen MYMAP definiert. Die erste Instanz hat eine Sequenznummer von 10, und die zweite eine Sequenznummer von 20.

- **route-map MYMAP permit 10**(Der erste Satz von Bedingungen geht hier.)
- **route-map MYMAP permit 20**(Der zweite Satz von Bedingungen geht hier.)

Wenn Sie die Routenübersicht MYMAP auf eingehende oder ausgehende Routen anwenden, werden die ersten Bedingungen über Instanz 10 angewendet. Wenn die ersten Bedingungen nicht erfüllt sind, fahren Sie mit einer höheren Instanz der Routenübersicht fort.

Zuordnen und Festlegen von Konfigurationsbefehlen

Jede Route Map besteht aus einer Liste von match und set Konfigurationsbefehlen. Die Übereinstimmung gibt ein match Kriterium an und set gibt eine set Aktion an, wenn die vom match Befehl erzwungenen Kriterien erfüllt sind.

Sie können beispielsweise eine Routenübersicht definieren, die ausgehende Updates überprüft. Wenn eine Übereinstimmung mit der IP-Adresse 10.1.1.1 vorliegt, wird die Metrik für dieses Update auf 5 gesetzt. Diese Befehle veranschaulichen das Beispiel:

<#root>

```
match ip address 10.1.1.1
```

```
set metric 5
```

Wenn nun die Anpassungskriterien erfüllt sind und Sie über einen verfügen, permit gibt es eine Neuverteilung oder Steuerung der Routen, wie die Aktion set angibt. Sie brechen aus der Liste aus.

Wenn die Übereinstimmungskriterien erfüllt sind und Sie über einen verfügendeny, erfolgt keine Neuverteilung oder Steuerung der Route. Sie brechen aus der Liste aus.

Wenn die Anpassungskriterien nicht erfüllt sind und Sie über ein permit oder deny verfügen, wird die nächste Instanz der Routenübersicht überprüft. Beispiel: Instanz 20 wird geprüft. Diese Prüfung der nächsten Instanz wird fortgesetzt, bis Sie entweder alle Instanzen der Routenübersicht aufheben oder beenden. Wenn Sie die Liste ohne Übereinstimmung beenden, wird die Route not accepted nor forwarded.

Wenn Sie in Cisco IOS®-Softwareversionen vor Version 11.2 der Cisco IOS-Software Routing-Zuordnungen zum Filtern von BGP-Updates statt zur Neuverteilung zwischen Protokollen verwenden, können Sie bei Verwendung eines **Match-**Befehls für die IP-Adresse nicht nach eingehendem Datenverkehr filtern. Ein Filter für ausgehende Anrufe ist akzeptabel. Für Cisco IOS Software, Version 11.2 und höher, gelten diese Einschränkungen nicht.

Die zugehörigen Befehle für match sind:

- matchas-path
- match community
- matchclns

- match interface
- matchip address
- matchip nexthop
- matchip route-source
- matchmetric
- match route-type
- match tag

Die zugehörigen Befehle für set sind:

- set as-path
- set clns
-

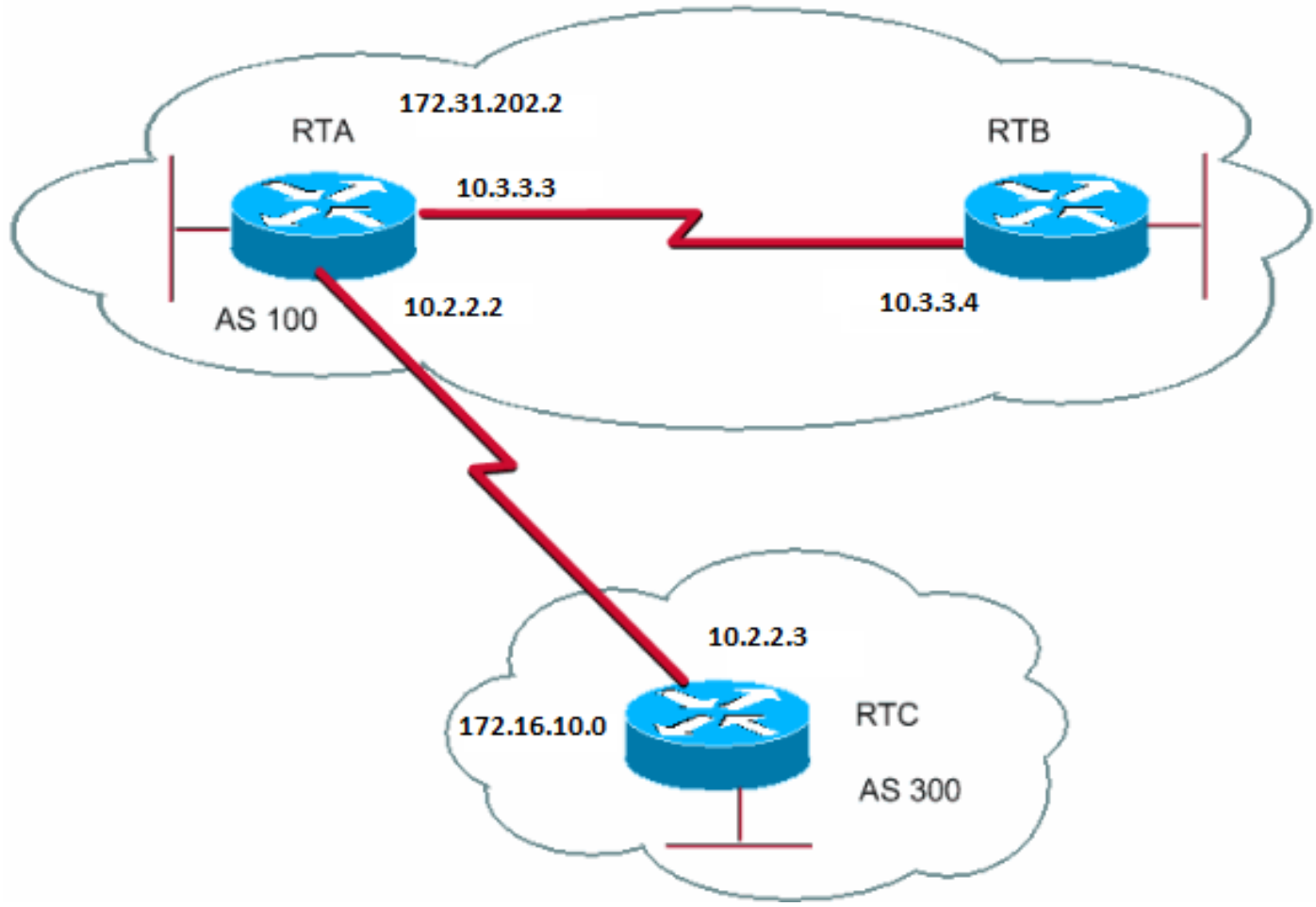
set automatic-tag

- set community
- set interface
- set default interface
- set ip default nexthop
- set level
- set local-preference
- set metric
- set metric-type
- set nexthop
- set origin

- set tag

- set weight

Sehen Sie sich einige Routenplanbeispiele an:



Beispiele für Routenübersichten

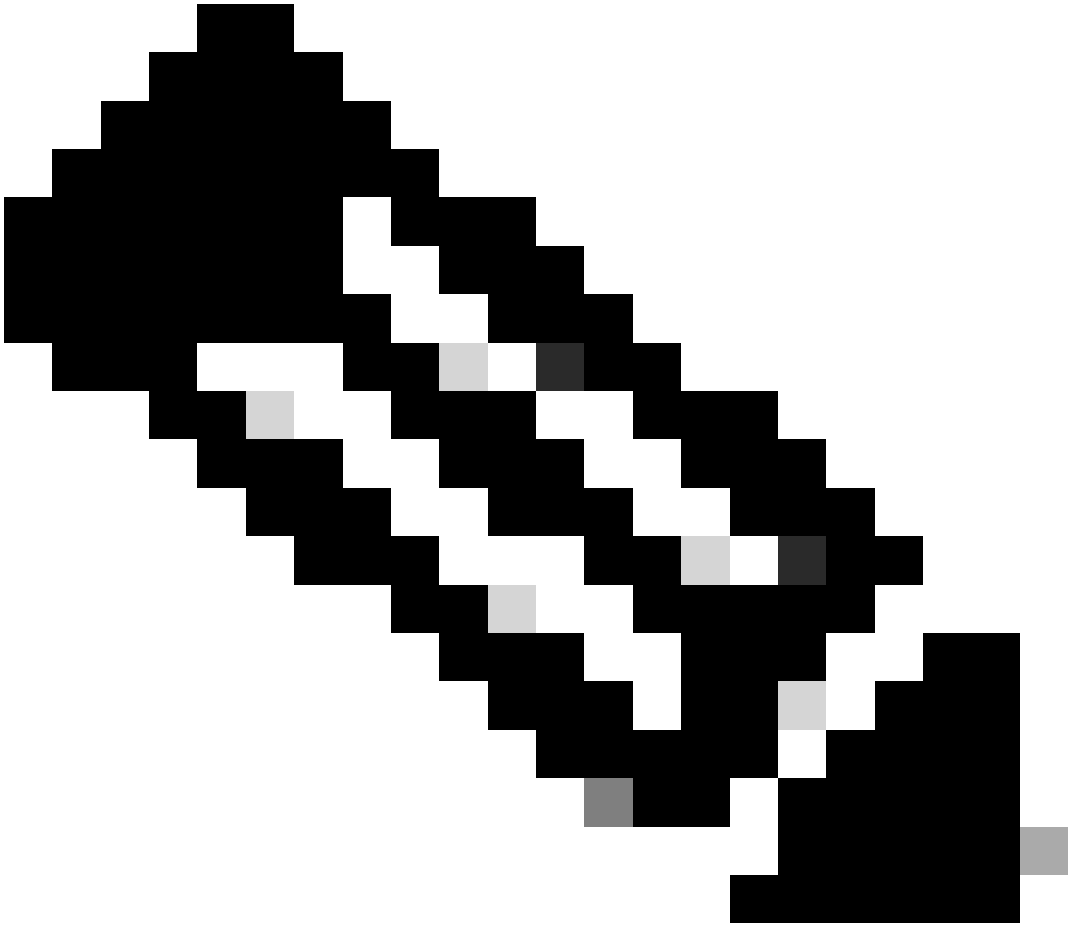
Beispiel 1

Angenommen, RTA und RTB führen das Routing Information Protocol (RIP) aus, und RTA und RTC führen BGP aus. RTA erhält Updates über BGP und verteilt die Updates auf RIP weiter. Angenommen, die RTA möchte die Routen um 172.16.10.0 mit einer Metrik von 2 und alle anderen Routen mit einer Metrik von 5 auf RTB umverteilen. In diesem Fall können Sie die folgende Konfiguration verwenden:

```
RTA#
router rip
 network 10.3.0.0
 network 10.2.0.0
```

```
 network 172.31.202.2
 passive-interface Serial0
 redistribute bgp 100 route-map SETMETRIC

router bgp 100
 neighbor 10.2.2.3 remote-as 300
 network 172.31.202.2

route-map SETMETRIC permit 10
 match ip-address 1
 set metric 2

route-map SETMETRIC permit 20
 set metric 5

access-list 1 permit 172.16.10.0 0.0.255.255
```

Wenn in diesem Beispiel eine Route mit der IP-Adresse 172.16.10.0 übereinstimmt, hat die Route die Metrik 2. Anschließend werden Sie aus der Routenplanliste ausgeschlossen. Wenn keine Übereinstimmung gefunden wird, fahren Sie in der Routenübersichtsliste fort, die angibt, dass alle anderen Elemente auf Metrik 5 gesetzt sind.



Hinweis: Stellen Sie sich immer die Frage "Was geschieht mit Routen, die keiner der Übereinstimmungsanweisungen entsprechen?" Diese Routen werden standardmäßig gelöscht.

Beispiel 2

Angenommen, Sie möchten in Beispiel 1 nicht, dass AS100 Updates zu 172.16.10.0 akzeptiert. Sie können keine Routenzuordnungen auf den eingehenden Datenverkehr anwenden, wenn Sie eine Übereinstimmung mit einer IP-Adresse als Grundlage haben. Daher müssen Sie eine Outbound-Routing-Map auf RTC verwenden:

```
RTC#
router bgp 300
```

```
 network 172.16.10.0
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 route-map STOPUPDATES out

route-map STOPUPDATES permit 10
 match ip address 1

access-list 1 deny 172.16.10.0 0.0.255.255
access-list 1 permit 0.0.0.0 255.255.255.255
```

Nachdem Sie sich nun mit dem Starten von BGP und dem Definieren eines Nachbarn besser vertraut gemacht haben, sehen Sie sich an, wie der Austausch von Netzwerkinformationen gestartet wird.

Es gibt mehrere Möglichkeiten, Netzwerkinformationen mithilfe des BGP zu senden. In diesen Abschnitten werden die einzelnen Methoden erläutert:

- Netzwerkbefehl
- Neuverteilung
- Statische Routen und Neuverteilung

Netzwerkbefehl

Das Format des network Befehls ist:

<#root>

network <network-number> mask <network-mask>

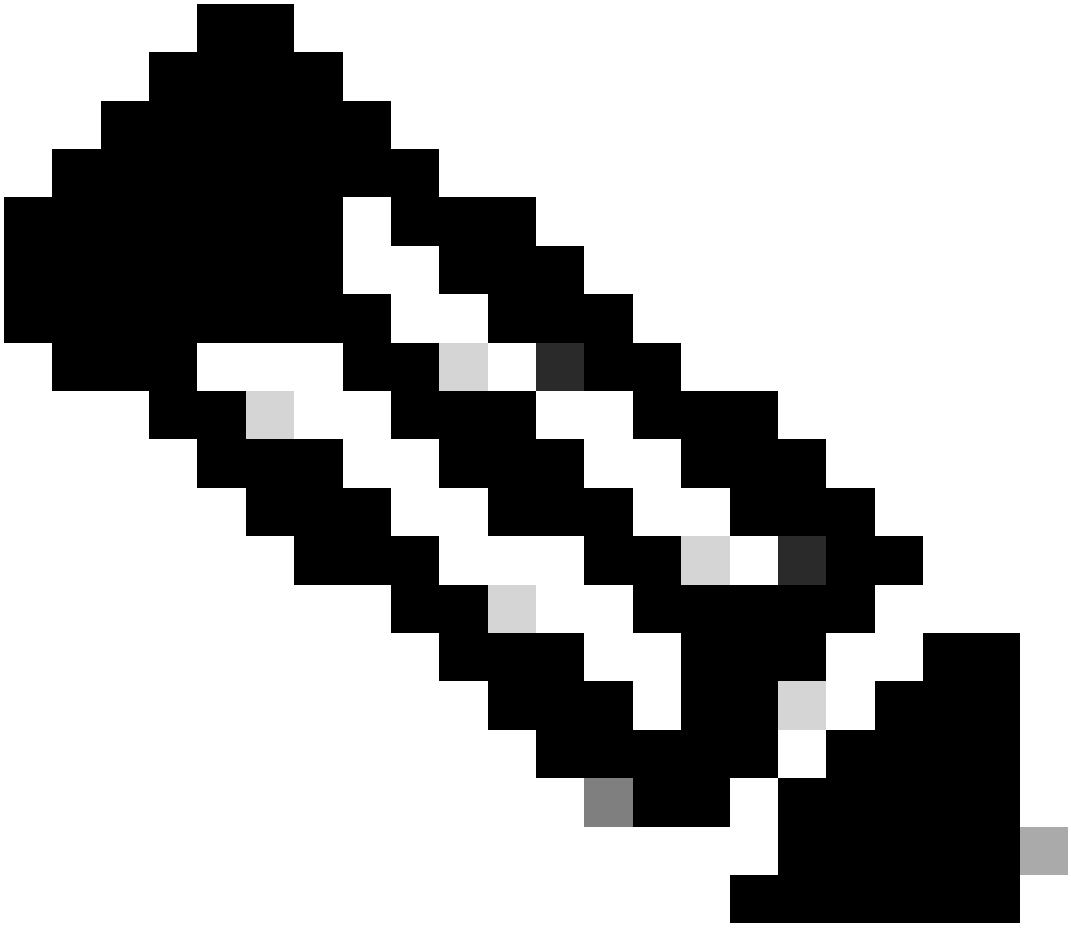
Der network Befehl steuert die von diesem Feld stammenden Netzwerke. Dieses Konzept unterscheidet sich von der bekannten Konfiguration mit Interior Gateway Routing Protocol (IGRP) und RIP. Mit diesem Befehl wird nicht versucht, BGP auf einer bestimmten Schnittstelle auszuführen. Stattdessen versuchen Sie, dem BGP mitzuteilen, welche Netzwerke BGP von diesem Feld stammen muss. Der Befehl verwendet einen Maskenteil, da BGP Version 4 (BGP4) Subnetting und Supernetting verarbeiten kann. Es sind maximal 200 Einträge für den network Befehl zulässig.

Der network Befehl funktioniert, wenn der Router das Netzwerk kennt, das Sie ankündigen möchten, unabhängig davon, ob es sich um ein verbundenes, statisches oder ein dynamisches Netzwerk handelt.

Ein Beispiel für den Befehl **network** ist:

```
RTA#
router bgp 1
 network 192.168.213.0 mask 255.255.0.0

ip route 192.168.213.0 255.255.0.0 null 0
```

Dieses Beispiel zeigt, dass Router A einen Netzwerkeintrag für 192.168.213.0/16 generiert. /16 gibt an, dass Sie ein Supernet der Klasse C-Adresse verwenden und die ersten beiden Oktette oder die ersten 16 Bit ankündigen.

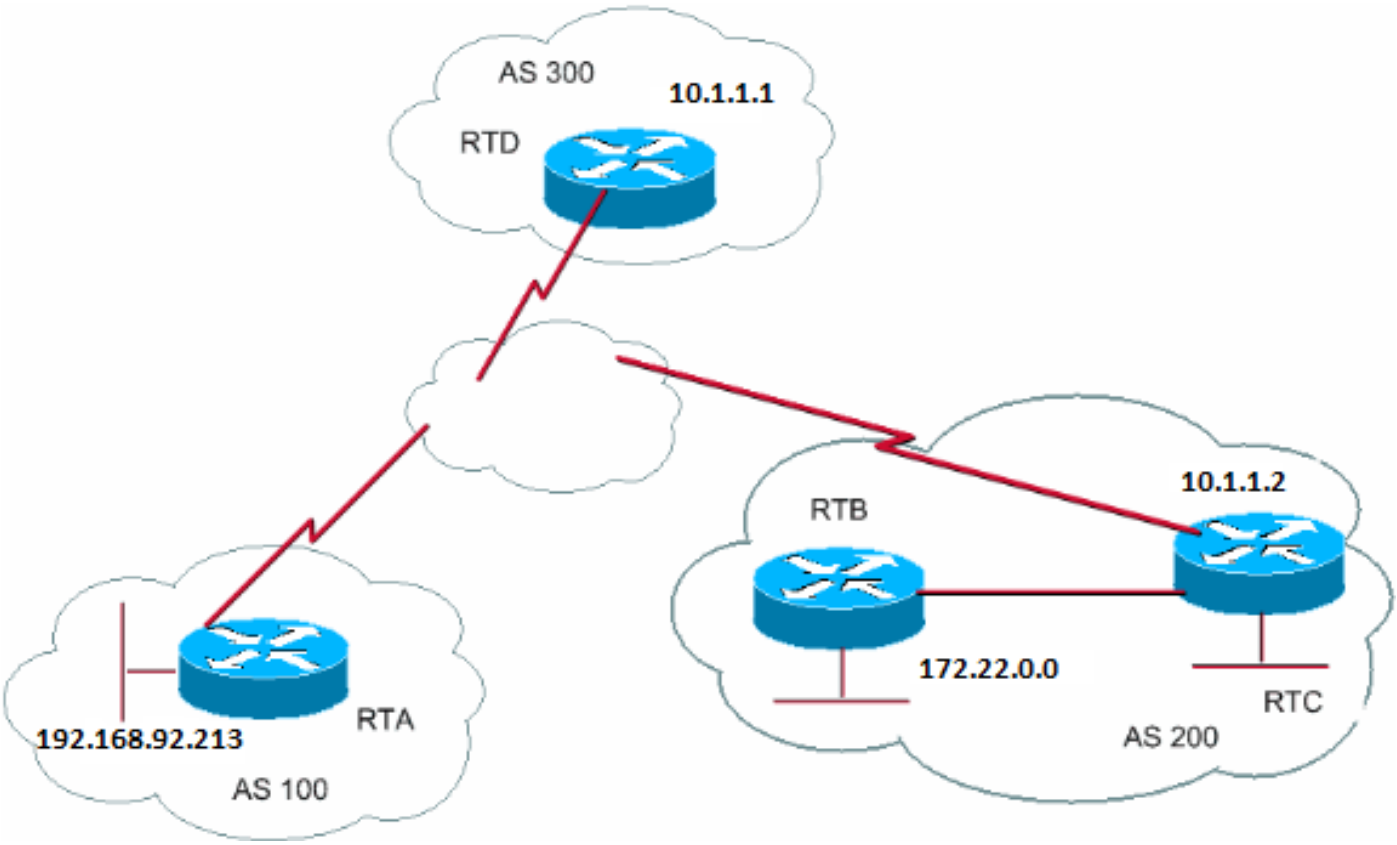


Hinweis: Sie benötigen die statische Route, damit der Router 192.168.213.0 generiert, da die statische Route einen passenden Eintrag in die Routing-Tabelle einfügt.

Neuverteilung

Mit diesem network Befehl können Sie Ihre Netzwerke über BGP ankündigen. Eine weitere Möglichkeit ist die Verteilung des IGP in das BGP. Das IGP kann IGRP, das OSPF-Protokoll (Open Shortest Path First), RIP, EIGRP (Enhanced Interior Gateway Routing Protocol) oder ein anderes Protokoll sein. Diese Neuverteilung kann beängstigend erscheinen, da Sie nun alle internen Routen in BGP kopieren. Einige dieser Routen wurden möglicherweise über BGP gelernt, und Sie müssen sie nicht erneut senden. Achten Sie beim Filtern darauf, dass Sie nur Routen ins Internet senden, die Sie ankündigen möchten, und nicht alle Routen, die Sie haben. Hier ein Beispiel.

RTA gibt den 192.168.92.213 und RTC den 172.22.0.0 bekannt. Sehen Sie sich die RTC-Konfiguration an:



Wenn Sie den networkBefehl ausgeben, haben Sie Folgendes:

```
RTC#
router eigrp 10
 network 172.22.0.0
 redistribute bgp 200
 default-metric 1000 100 250 100 1500

router bgp 200
 neighbor 10.1.1.1 remote-as 300
 network 172.22.0.0 mask 255.255.0.0

!--- This limits the networks that your AS originates to 172.22.0.0.
```

Wenn Sie stattdessen Umverteilung verwenden, haben Sie Folgendes:

```
RTC#
router eigrp 10
network 172.22.0.0
redistribute bgp 200
default-metric 1000 100 250 100 1500

router bgp 200
neighbor 10.1.1.1 remote-as 300
redistribute eigrp 10
```

```
!--- EIGRP injects 192.168.92.213 again into BGP.
```

Diese Neuverteilung bewirkt den Ursprung von 192.168.92.213 durch Ihr AS. Sie sind nicht die Quelle von 192.168.92.213; AS100 ist die Quelle. Daher müssen Sie Filter verwenden, um zu verhindern, dass die Quelle durch Ihr AS aus dem Netzwerk austritt. Die richtige Konfiguration lautet:

```
RTC#
router eigrp 10
 network 172.22.0.0
 redistribute bgp 200
 default-metric 1000 100 250 100 1500

router bgp 200
 neighbor 10.1.1.1 remote-as 300
 neighbor 10.1.1.1 distribute-list 1 out
 redistribute eigrp 10

access-list 1 permit 172.22.0.0 0.0.255.255
```

Verwenden Sie den access-listBefehl, um die Netzwerke zu steuern, die von AS200 stammen.

Die Neuverteilung von OSPF in BGP unterscheidet sich geringfügig von der Neuverteilung für andere IGPs. Die einfache Frage von redistribute ospf 1unter funktioniert router bgpnicht. Spezifische Schlüsselwörter wie internal, externalund **nssa-external** sind erforderlich, um die jeweiligen Routen neu zu verteilen. Weitere Informationen zur Neuverteilung von OSPF-Routen in BGP finden Sie unter Verstehen.

Statische Routen und Neuverteilung

Sie können statische Routen immer verwenden, um ein Netzwerk oder ein Subnetz zu initiieren. Der einzige Unterschied besteht darin, dass das BGP diese Routen als unvollständig oder unbekannt betrachtet. Sie können das gleiche Ergebnis erzielen wie im Beispiel im Abschnitt "Neuverteilung":

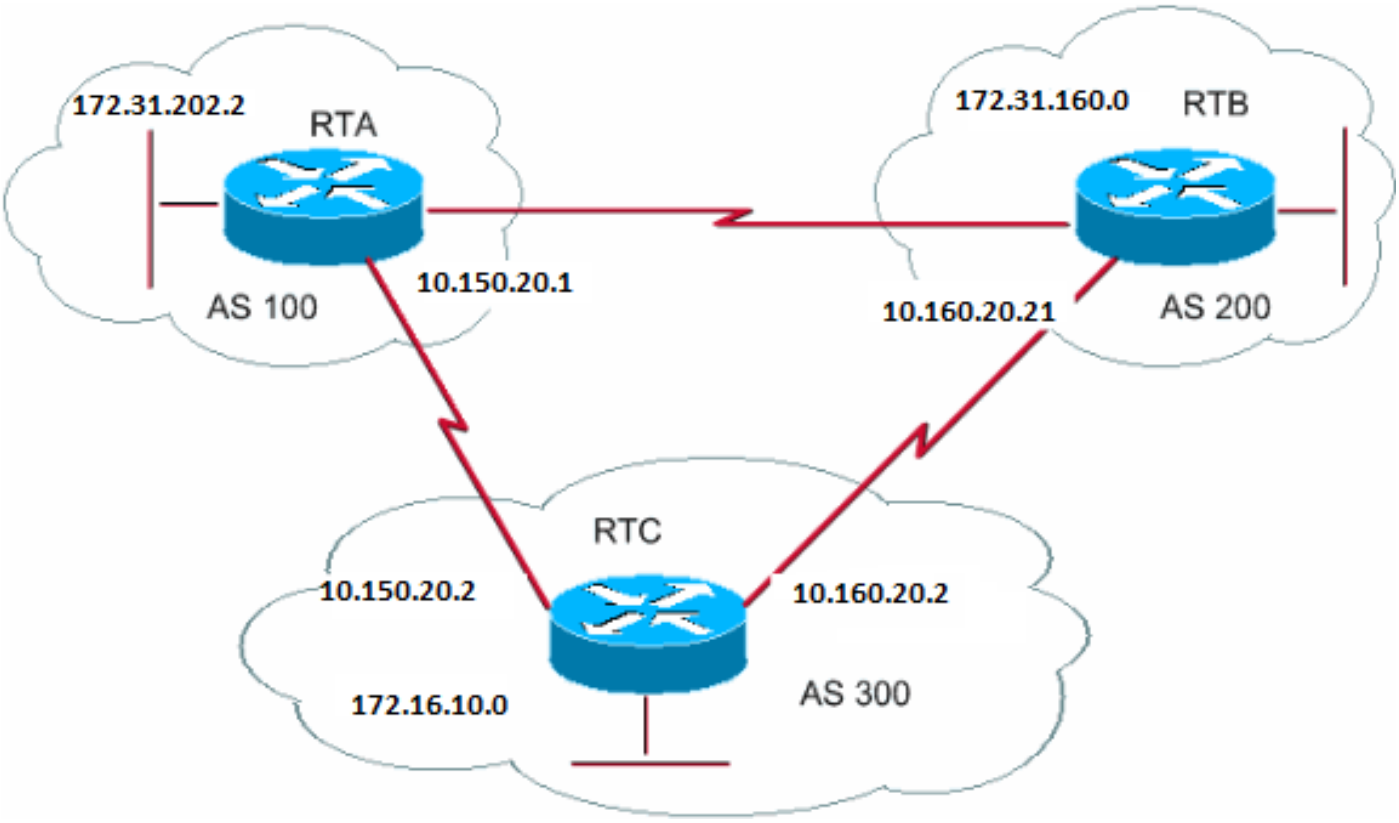
```
RTC#
router eigrp 10
 network 172.22.0.0
 redistribute bgp 200
 default-metric 1000 100 250 100 1500

router bgp 200
 neighbor 10.1.1.1 remote-as 300
 redistribute static

ip route 172.22.0.0 255.255.255.0 null0
```

Die null0 Schnittstelle ignoriert das Paket. Wenn Sie das Paket erhalten und eine spezifischere Übereinstimmung als 172.22.0.0 besteht, sendet der Router das Paket an die spezifische Übereinstimmung. Andernfalls ignoriert der Router das Paket. Diese Methode ist eine gute Möglichkeit, ein Supernet anzukündigen.

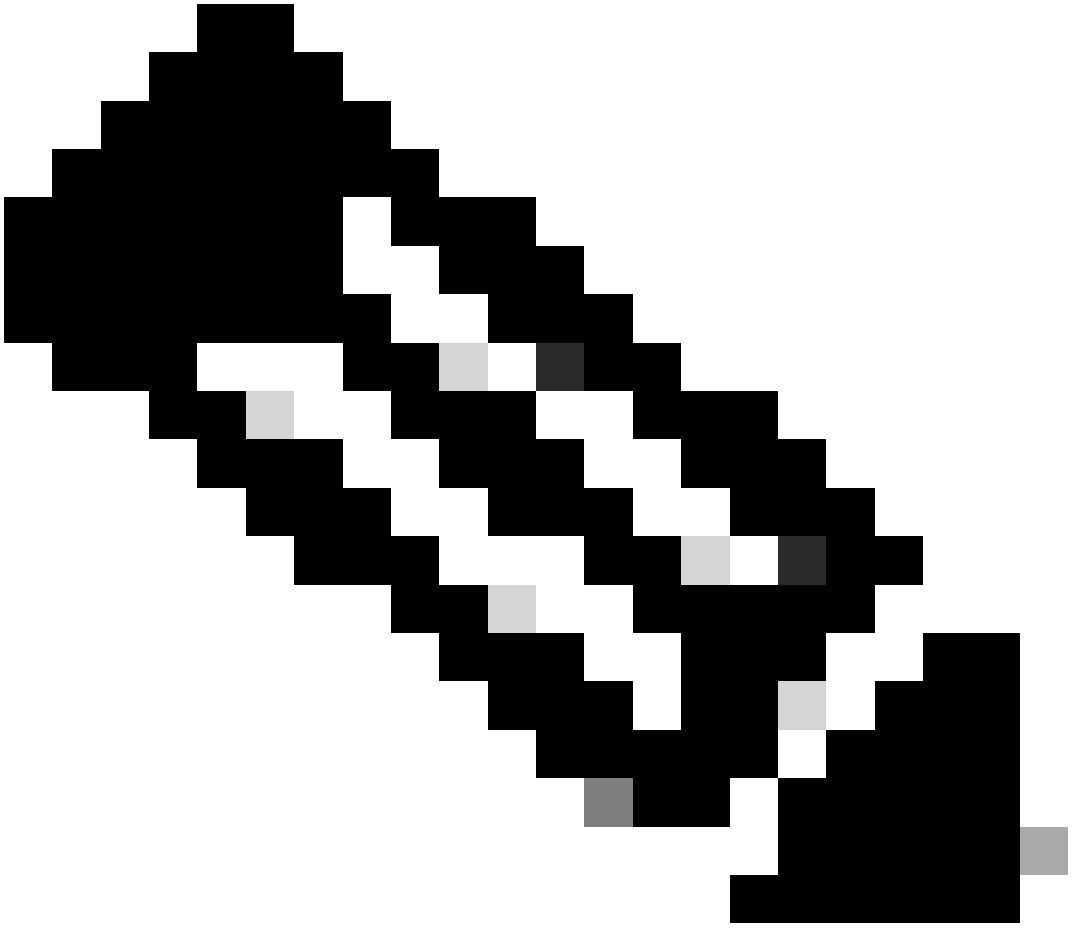
In diesem Dokument wurde erläutert, wie Sie verschiedene Methoden verwenden können, um Routen aus dem AS zu generieren. Denken Sie daran, dass diese Routen zusätzlich zu anderen BGP-Routen generiert werden, die das BGP über Nachbarn (intern oder extern) bezogen hat. Das BGP gibt Informationen, die das BGP von einem Peer erhält, an andere Peers weiter. Der Unterschied besteht darin, dass Routen, die aus dem network Befehl, der Neuverteilung oder dem statischen Befehl generiert werden, anzeigen, dass Ihr AS der Ursprung dieser Netzwerke ist.

Die Umverteilung ist immer die Methode zur Einspeisung von BGP in IGP.

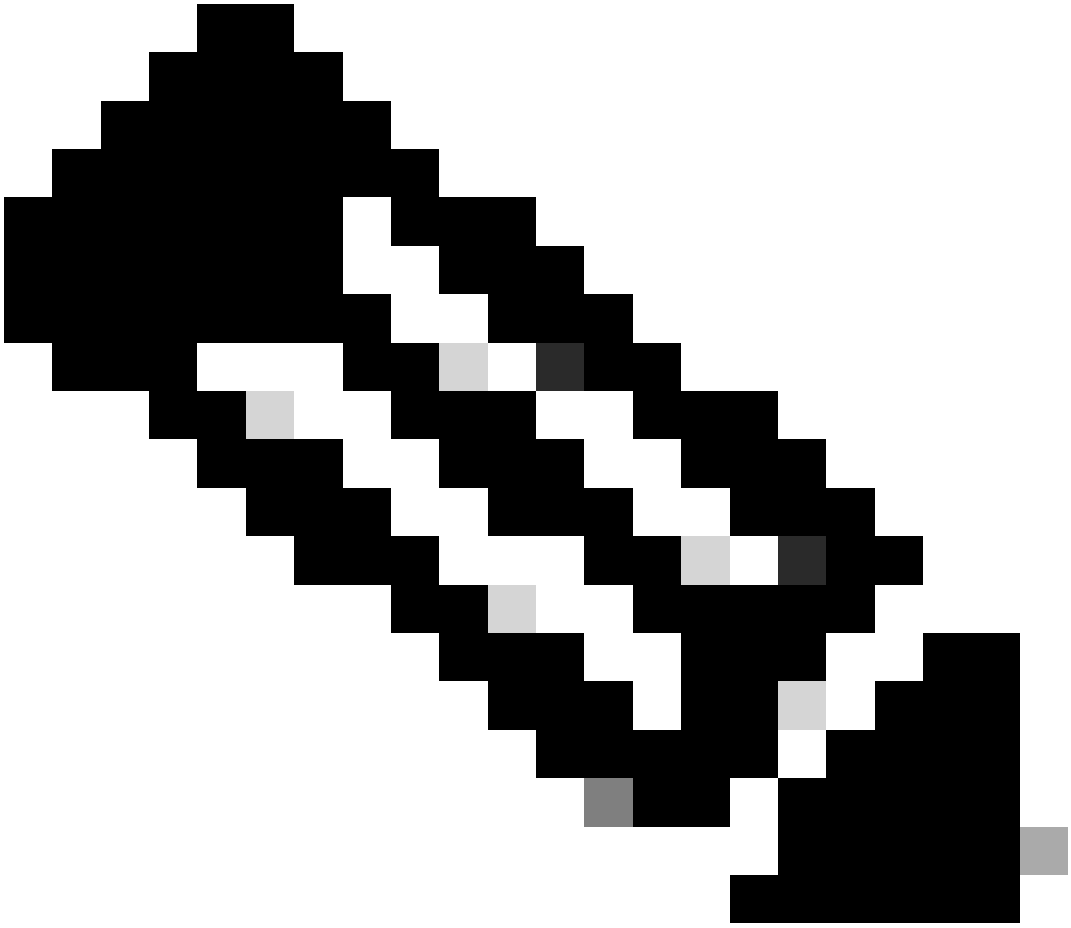
Hier ein Beispiel:




```
RTA#
router bgp 100
 neighbor 10.150.20.2 remote-as 300
 network 172.31.202.2

RTB#
router bgp 200
 neighbor 10.160.20.2 remote-as 300
 network 172.31.160.0

RTC#
router bgp 300
 neighbor 10.150.20.1 remote-as 100
 neighbor 10.160.20.21 remote-as 200
 network 170.10.00
```



Hinweis: Sie benötigen das Netzwerk 172.31.202.2 oder das Netzwerk 172.31.160.0 in RTC nicht, es sei denn, Sie möchten, dass RTC diese Netzwerke generiert und diese Netzwerke weitergibt, sobald sie von AS100 und AS200 kommen. Auch hier besteht der Unterschied darin, dass der Netzwerkbefehl eine zusätzliche Ankündigung für dieselben Netzwerke hinzufügt, was darauf hinweist, dass AS300 auch ein Ausgangspunkt für diese Routen ist.



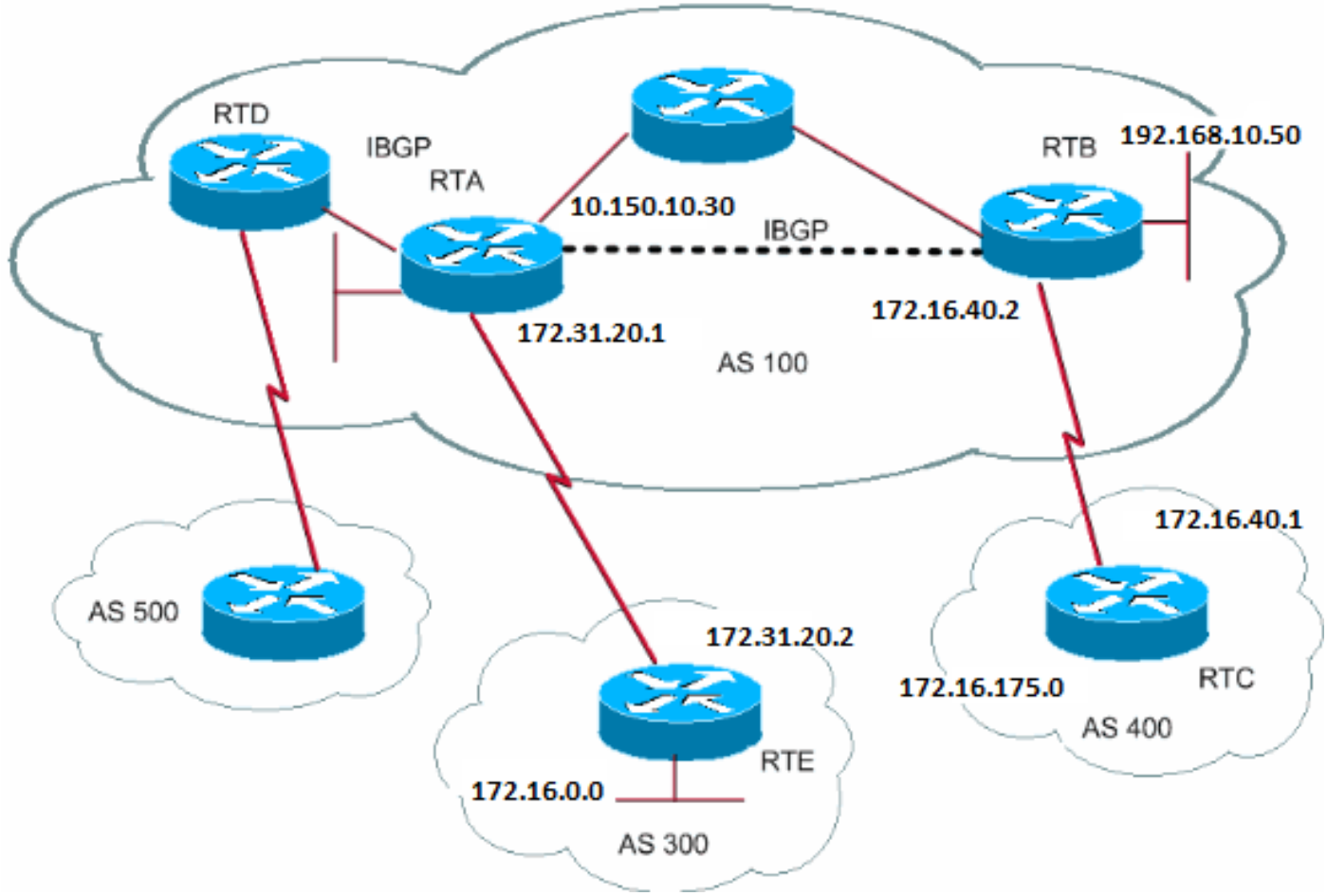
Hinweis: Beachten Sie, dass das BGP keine Updates akzeptiert, die von seinem eigenen AS stammen. Diese Weigerung gewährleistet eine schleifenfreie Interdomänentopologie.

Nehmen wir beispielsweise an, dass AS200 gemäß dem Beispiel in diesem Abschnitt über eine direkte BGP-Verbindung mit AS100 verfügt. Die RTA generiert die Route 172.31.202.2 und sendet sie an AS300. Anschließend übergibt RTC diese Route an AS200 und behält den Ursprung als AS100 bei. RTB übergibt 172.31.202.2 an AS100, wobei der Ursprung noch AS100 ist. Die RTA stellt fest, dass das Update von ihrem eigenen AS stammt, und ignoriert das Update.

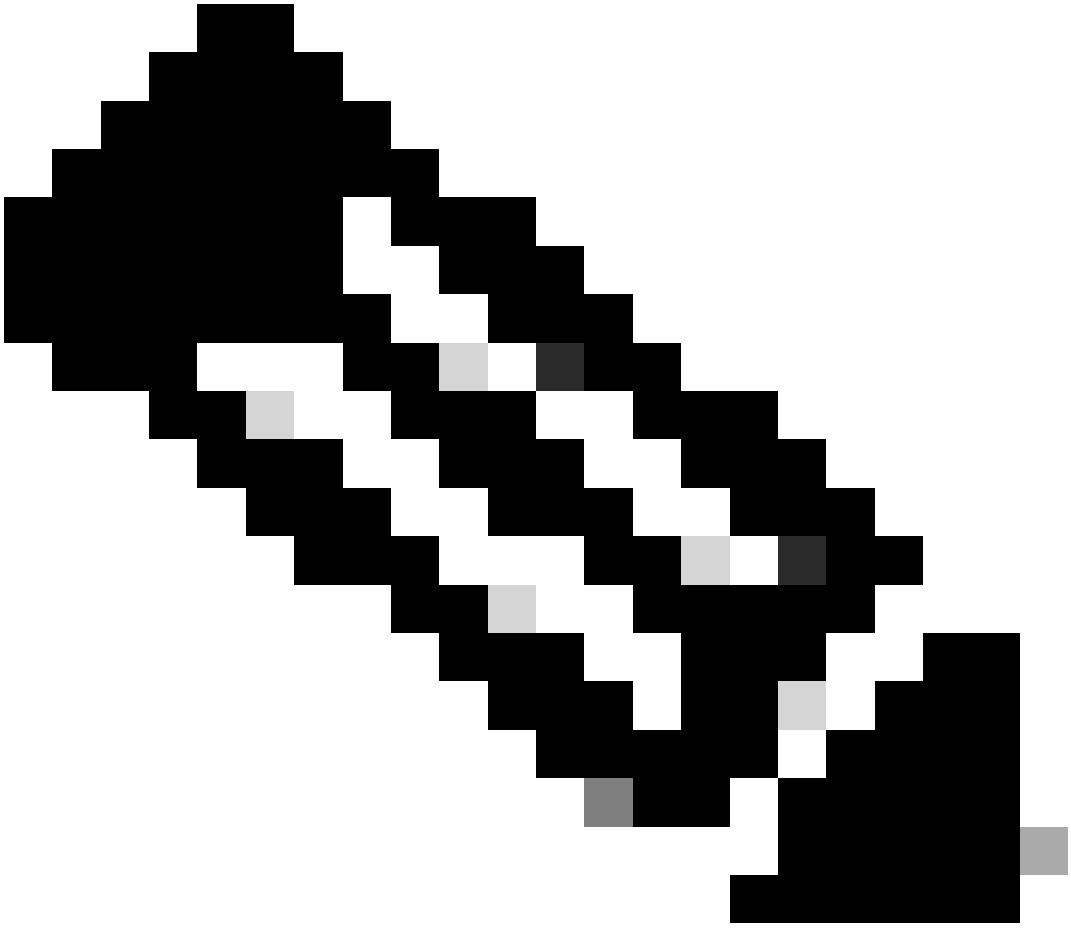
iBGP

Sie verwenden iBGP, wenn ein AS als Übertragungssystem zu einem anderen AS fungieren möchte. Dasselbe können Sie tun, wenn Sie über eBGP lernen, in IGP umverteilen und dann wieder in ein anderes AS umverteilen. iBGP bietet jedoch mehr Flexibilität und effizientere

Möglichkeiten für den Informationsaustausch innerhalb eines AS. So bietet das iBGP beispielsweise Möglichkeiten, den besten Austrittspunkt aus dem AS mithilfe lokaler Präferenzen zu steuern. Weitere Informationen zu lokalen Einstellungen finden Sie im Abschnitt Lokales Voreinstellungsattribut.




```
RTA#
router bgp 100
 neighbor 192.168.10.50 remote-as 100
 neighbor 172.31.20.2 remote-as 300
 network 172.31.202.2

RTB#
router bgp 100
 neighbor 10.150.10.30 remote-as 100
 neighbor 172.16.40.1 remote-as 400
 network 192.168.10.150

RTC#
router bgp 400
 neighbor 172.16.40.2 remote-as 100
 network 172.16.0.0
```



Hinweis: Wenn ein BGP-Sprecher ein Update von anderen BGP-Sprechern in seinem eigenen AS (iBGP) empfängt, verteilt der BGP-Sprecher, der das Update empfängt, diese Informationen nicht an andere BGP-Sprecher in seinem eigenen AS. Der BGP-Router, der das Update erhält, verteilt die Informationen an andere BGP-Router außerhalb des AS. Daher sollte eine vollständige Vermaschung zwischen den iBGP-Routern innerhalb eines AS erfolgen.

Die RTA und der RTB führen iBGP aus. RTA und RTD führen auch iBGP aus. Die BGP-Updates von RTB zu RTA übertragen Daten an RTE, das sich außerhalb des AS befindet. Die Updates werden nicht an RTD übertragen, das sich im AS befindet. Erstellen Sie daher ein iBGP-Peering zwischen RTB und RTD, um den Fluss der Updates nicht zu unterbrechen.

Der BGP-Entscheidungsalgorithmus

Nachdem das BGP Updates zu verschiedenen Zielen von verschiedenen autonomen Systemen erhalten hat, muss das Protokoll Pfade wählen,

um ein bestimmtes Ziel zu erreichen. BGP wählt nur einen Pfad, um ein bestimmtes Ziel zu erreichen.

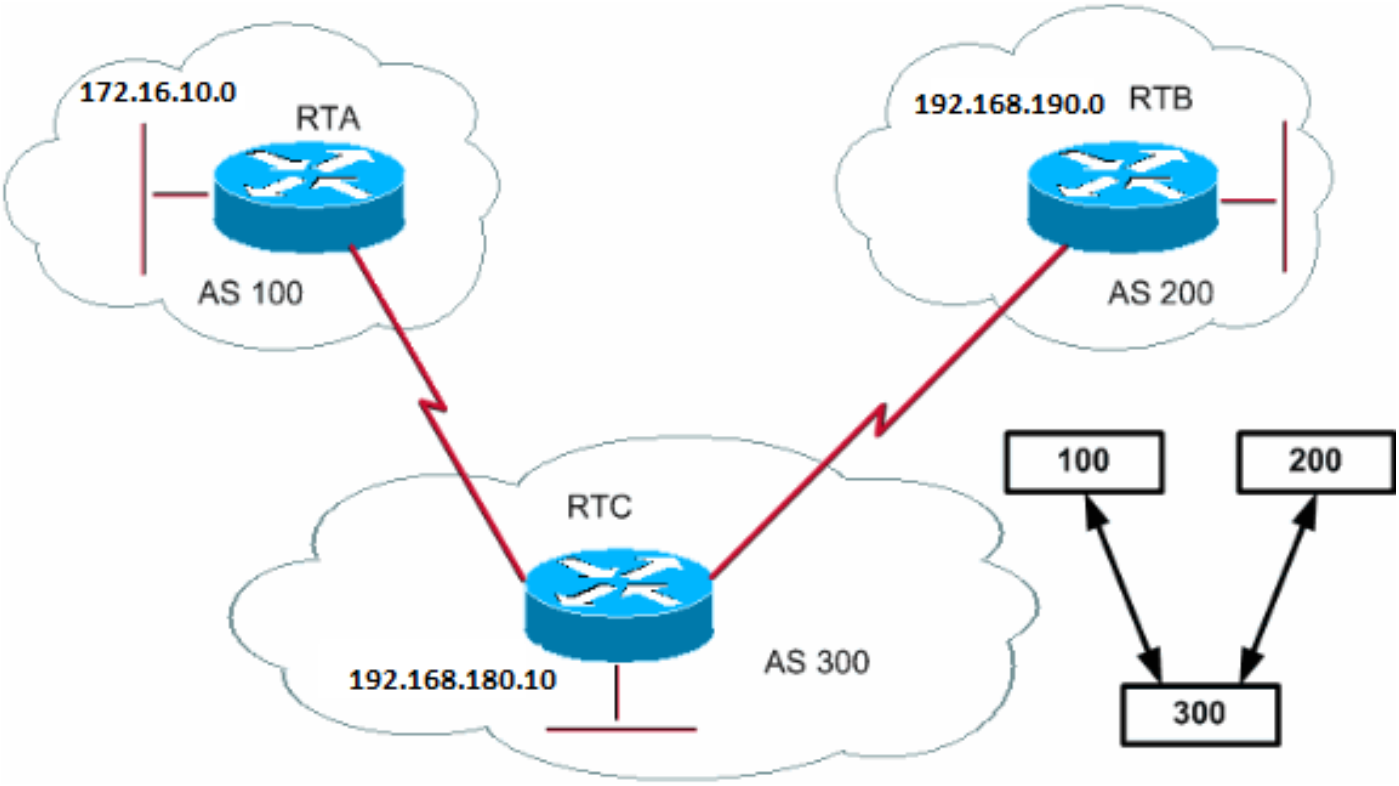
BGP basiert die Entscheidung auf verschiedenen attributes Attributen, z. B. Next Hop, administrativen Gewichtungen, lokaler Präferenz, Routenursprung, Pfadlänge, Ursprungscode, Metrik und anderen.

BGP leitet immer den besten Pfad an seine Nachbarn weiter. Weitere Informationen finden Sie unter BGP Best Path Selection Algorithm (Algorithmus zur Auswahl des besten Pfads).

Im nächsten Abschnitt werden diese Attribute und ihre Verwendung erläutert.

BGP-Anwenderberichte 2

AS_PATH-Attribut



Wenn eine Routenaktualisierung ein AS durchläuft, wird dieser Aktualisierung die AS-Nummer vorangestellt. Das AS_PATH-Attribut ist die Liste der AS-Nummern, die eine Route durchlaufen hat, um ein Ziel zu erreichen. Ein AS_SET ist ein geordneter mathematischer Satz {} aller durchlaufenen ASs. Der Abschnitt "CIDR Example 2 (as-set)" dieses Dokuments enthält ein Beispiel für AS_SET.

In dem Beispiel in diesem Abschnitt kündigt RTB das Netzwerk 192.168.190.0 in AS200 an. Wenn diese Route AS300 durchläuft, hängt RTC seine eigene AS-Nummer an das Netzwerk an. Wenn 192.168.190.0 die RTA erreicht, sind zwei AS-Nummern an das Netzwerk angeschlossen: zuerst 200 und dann 300. Für die RTA ist der Weg bis zu 192.168.190.0 (300, 200).

Das gleiche Verfahren gilt für 172.16.10.0 und 192.168.180.10. RTB muss Pfad einnehmen (300, 100); RTB durchläuft AS300 und dann AS100, um 172.16.10.0 zu erreichen. RTC muss Pfad (200) durchlaufen, um 192.168.190.0 zu erreichen, und Pfad (100), um 172.16.10.0 zu erreichen.

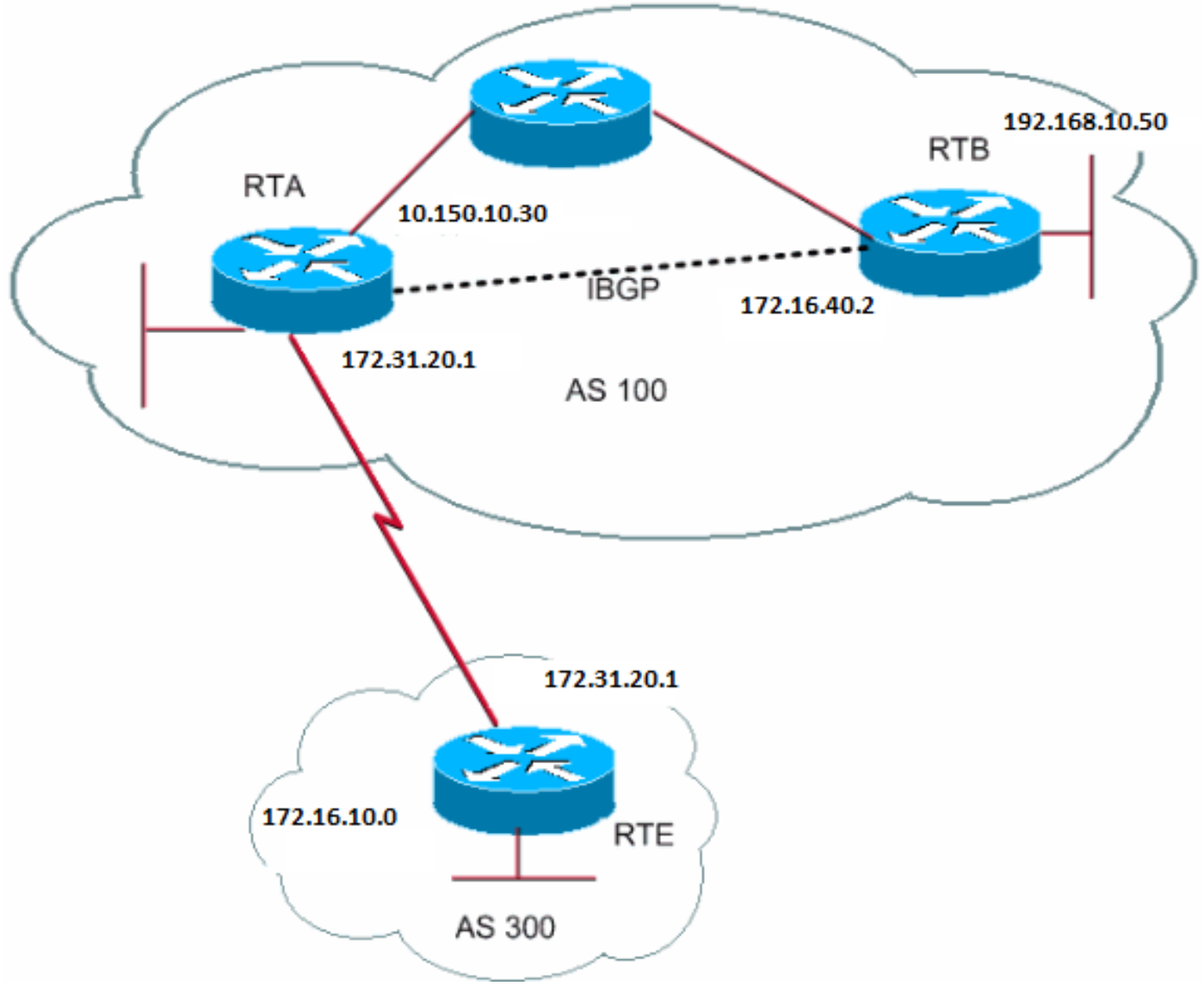
Ursprüngliches Attribut

Der Ursprung ist ein obligatorisches Attribut, das den Ursprung der Pfadinformationen definiert. Das Ursprungsattribut kann drei Werte annehmen:

- IGP - Network Layer Reachability Information (NLRI) befindet sich innerhalb des AS des Ursprungs. Dies geschieht normalerweise, wenn Sie den **bgp network**Befehl ausgeben. In der BGP-Tabelle wird "IGP" ignoriert.

- EGP: NLRI wird über das External Gateway Protocol (EGP) gelernt. Eine in der BGP-Tabelle zeigt den EGP an.

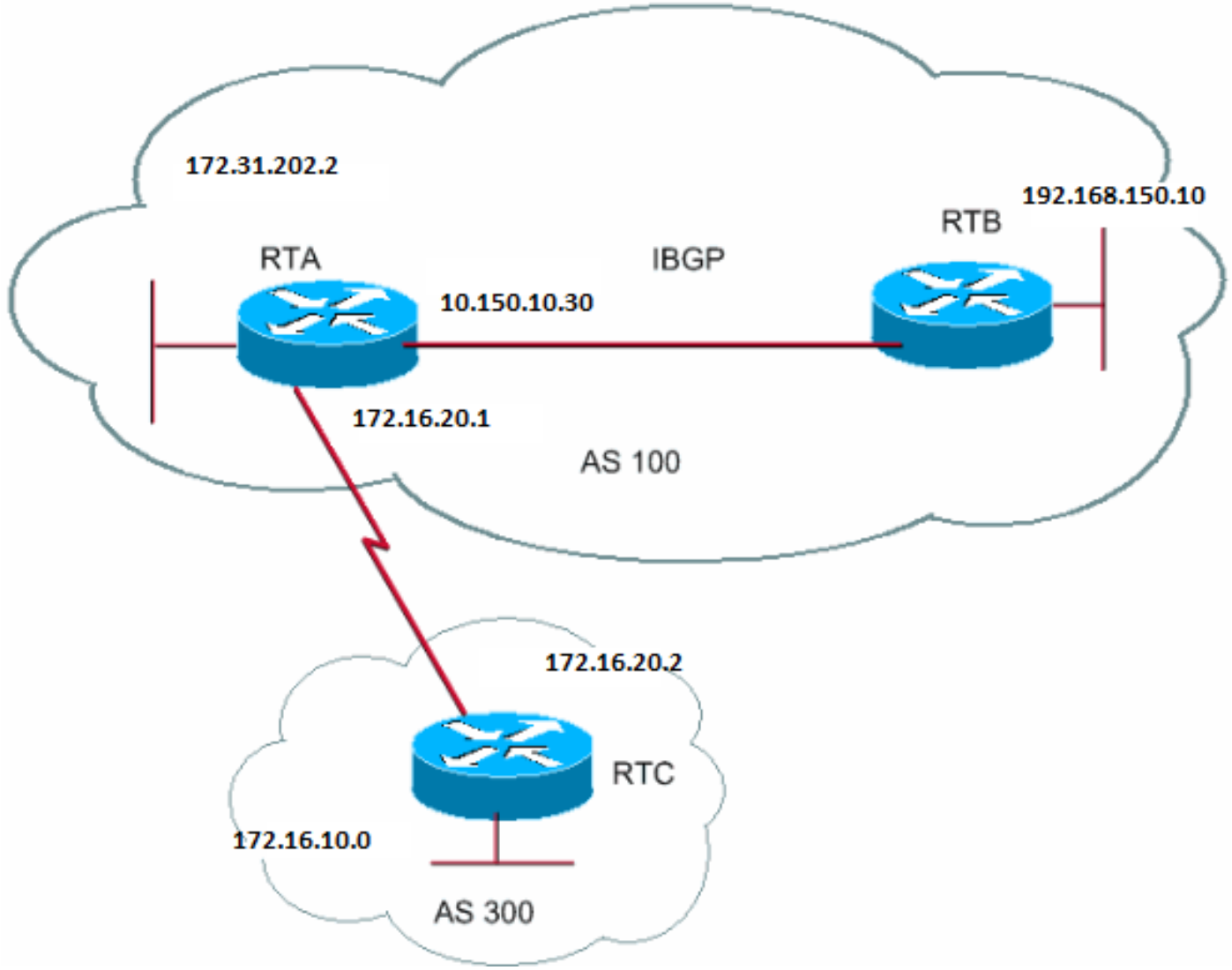
- INCOMPLETE (UNVOLLSTÄNDIG): NLRI ist unbekannt oder wurde auf andere Weise erlernt. UNVOLLSTÄNDIG tritt in der Regel dann auf, wenn Sie Routen von anderen Routing-Protokollen in das BGP umverteilen und der Ursprung der Route unvollständig ist. Ein? in der BGP-Tabelle bedeutet "INCOMPLETE".




```
RTA#
router bgp 100
 neighbor 192.168.10.50 remote-as 100
 neighbor 172.31.20.2 remote-as 300
 network 172.31.202.2
 redistribute static

ip route 192.168.190.0 255.255.0.0 null0

RTB#
router bgp 100
 neighbor 10.150.10.30 remote-as 100
 network 192.168.10.150

RTE#
router bgp 300
 neighbor 172.31.20.1 remote-as 100
 network 172.16.10.0
```

RTA erreicht 172.16.10.0 via 300 i. "300 i" bedeutet, dass der nächste AS-Pfad 300 ist und der Ursprung der Route das IGP ist. Über i erreicht die RTA auch den 192.168.10.150. Dieses "i" bedeutet, dass sich der Eintrag im selben AS befindet und der Ursprung das IGP ist. RTE erreicht 172.31.202.2 über 100 i. "100 i" bedeutet, dass das nächste AS 100 ist und der Ursprung IGP ist. RTE erreicht ebenfalls 192.168.190.0 via 100 ?. Die "100 ?" bedeutet, dass das nächste AS 100 ist und dass der Ursprung unvollständig ist und von einer statischen Route stammt.

BGP Next-Hop-Attribut



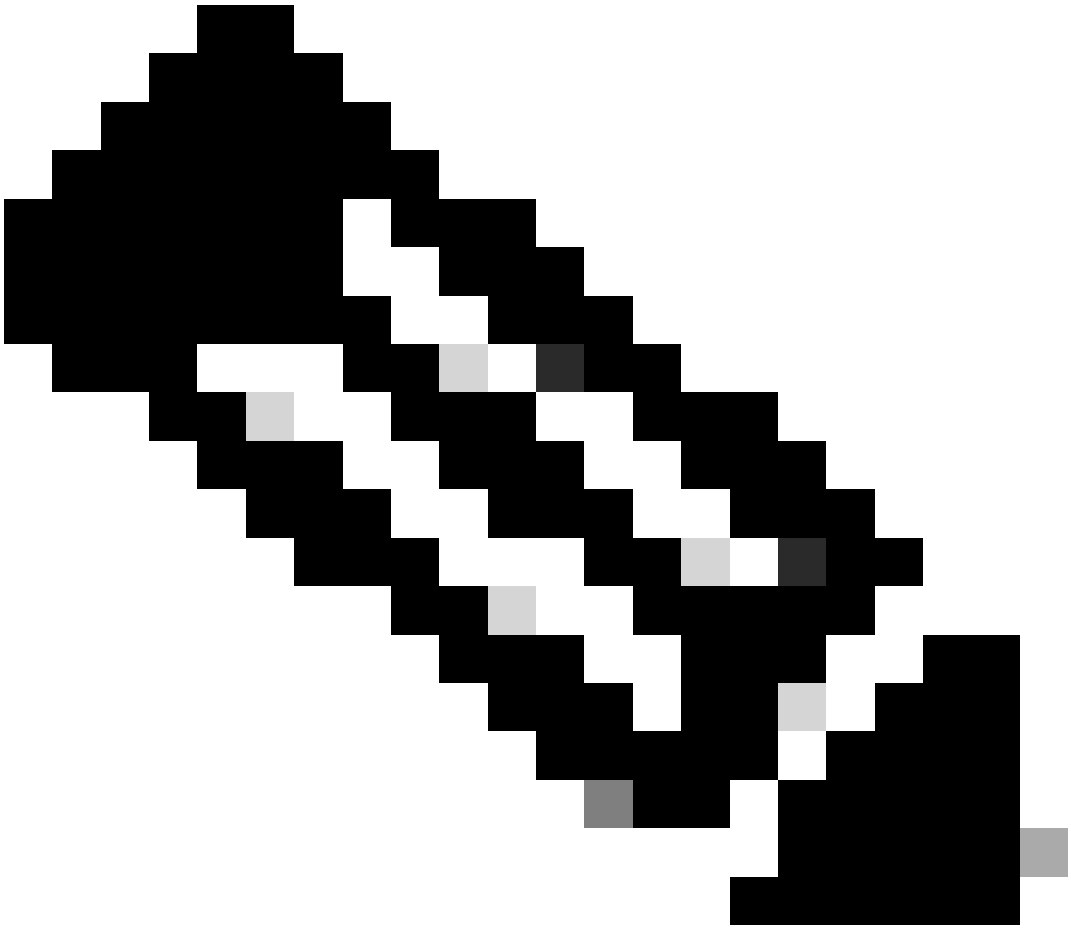
BGP Next-Hop-Attribut

Das BGP Next Hop-Attribut ist die nächste Hop-IP-Adresse, die verwendet wird, um ein bestimmtes Ziel zu erreichen.

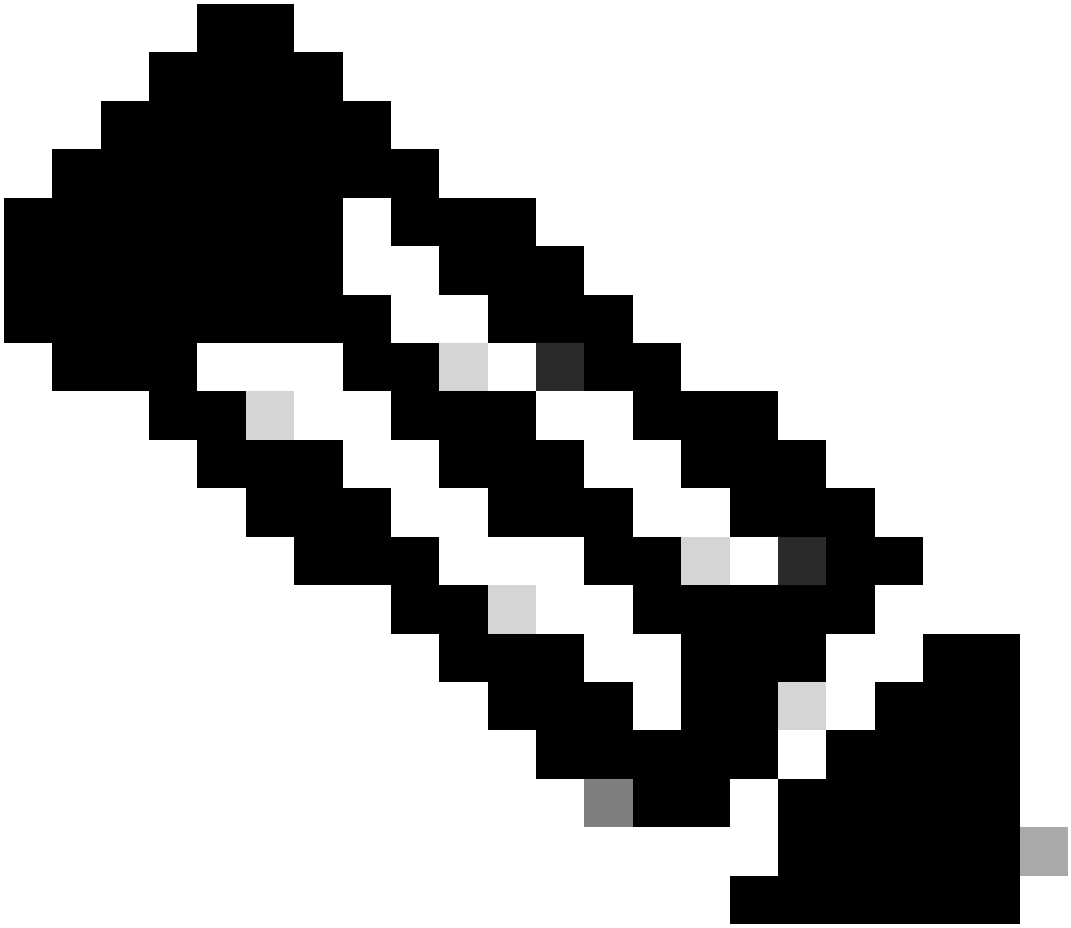
Bei eBGP ist der nächste Hop immer die IP-Adresse des Nachbarn, die der neighbor Befehl angibt. Im Beispiel in diesem Abschnitt kündigt RTC 172.16.10.0 der RTA mit dem nächsten Hop 172.31.20.2 an. RTA kündigt RTC 172.31.202.2 mit dem nächsten Hop 172.31.20.1 an. Für iBGP sieht das Protokoll vor, dass der nächste Hop, den eBGP ankündigt, in iBGP übertragen werden muss. Aufgrund dieser Regel kündigt die RTA 172.16.10.0 ihrem iBGP-Peer RTB mit dem nächsten Hop 172.31.20.2 an. Basierend auf RTB ist der nächste Hop bis zu 172.16.10.0 172.31.20.2 *und nicht* 10.150.10.30.

Stellen Sie sicher, dass die RTB über IGP die Adresse 172.31.20.2 erreichen kann. Andernfalls verwirft die RTB Pakete mit dem Ziel 172.16.10.0, da auf die nächste Hop-Adresse nicht zugegriffen werden kann. Wenn RTB beispielsweise iGRP ausführt, können Sie iGRP auch im RTA-Netzwerk 172.16.10.0 ausführen. Sie möchten iGRP auf der Verbindung zu RTC passiv machen, sodass BGP nur ausgetauscht wird.

```
RTA#
router bgp 100
 neighbor 172.31.20.2 remote-as 300
 neighbor 192.168.150.10 remote-as 100
 network 172.31.202.2

RTB#
router bgp 100
 neighbor 10.150.10.30 remote-as 100

RTC#
router bgp 300
 neighbor 172.31.20.1 remote-as 100
 network 172.16.10.0
```



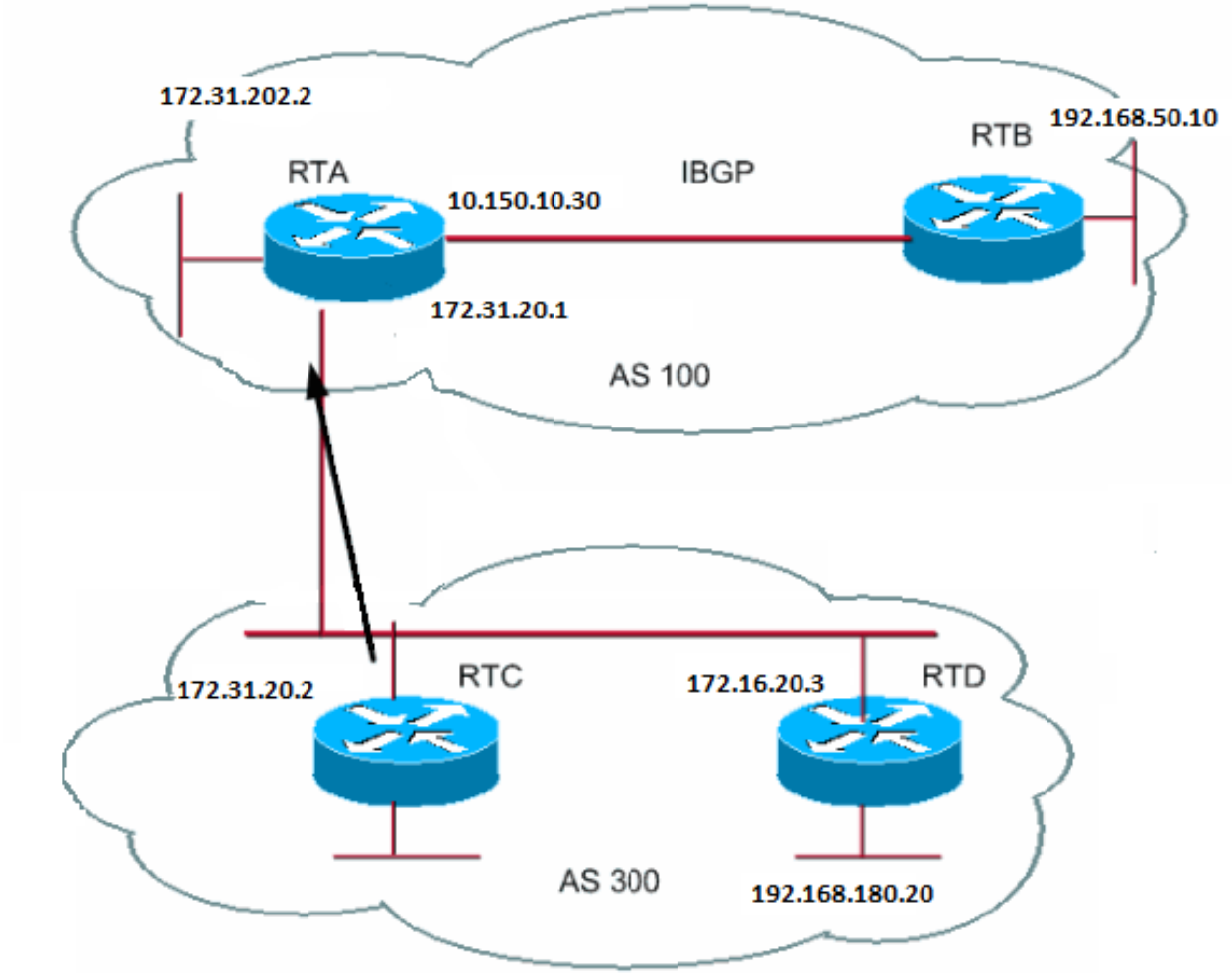
Hinweis: RTC kündigt der RTA 172.16.10.0 mit einem Next Hop gleich 172.31.20.2 an.



Hinweis: RTA informiert RTB über 172.16.10.0 mit einem Next Hop von 172.31.20.2. Der nächste eBGP-Hop wird in iBGP übertragen.

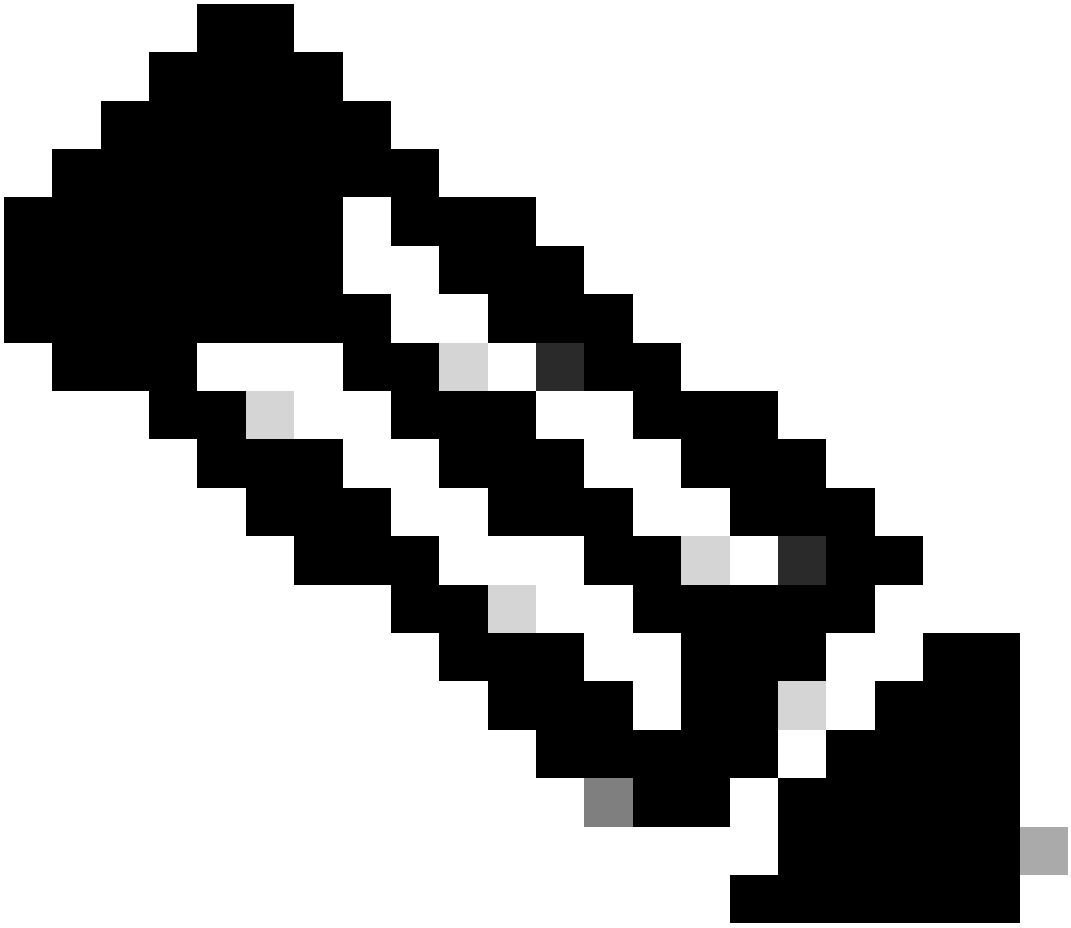
Besondere Vorsicht bei der Verwendung von NBMA-Netzwerken (Multiaccess and Non-Broadcast Multiaccess) ist erforderlich. Weitere Informationen finden Sie in den Abschnitten BGP Next Hop (Multiaccess Networks) und BGP Next Hop (NBMA).

BGP Next-Hop (Multiaccess-Netzwerke)



Dieses Beispiel zeigt, wie sich der nächste Hop in einem Mehrzugriffsnetzwerk wie Ethernet verhält.

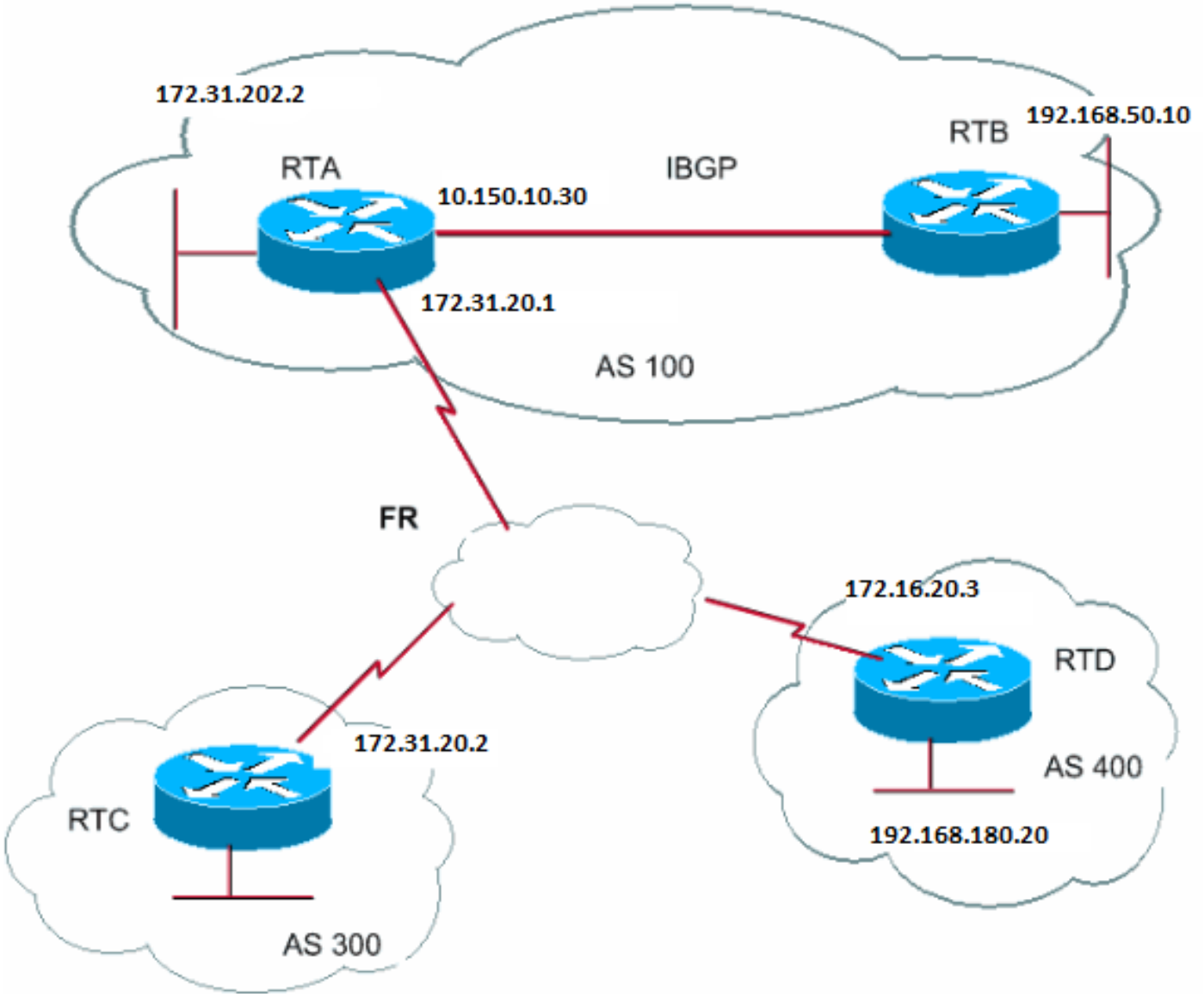
Angenommen, RTC und RTD in AS300 führen OSPF aus. RTC führt BGP mit RTA aus. RTC kann das Netzwerk 192.168.180.20 über 172.16.20.3 erreichen. Wenn RTC ein BGP-Update bezüglich 192.168.180.20 an RTA sendet, verwendet RTC als nächsten Hop 172.16.20.3. RTC verwendet keine eigene IP-Adresse, 172.31.20.2. RTC verwendet diese Adresse, da es sich bei dem Netzwerk zwischen RTA, RTC und RTD um ein Multizugriffsnetzwerk handelt. Die RTA Nutzung von RTD als Next Hop um 192.168.180.20 zu erreichen ist sinnvoller als der Extra Hop via RTC.



Hinweis: RTC kündigt der RTA 192.168.180.20 mit dem nächsten Hop 172.16.20.3 an.

Wenn das übliche Medium für RTA, RTC und RTD nicht Multiaccess, sondern NBMA ist, treten weitere Komplikationen auf.

BGP Next-Hop (NBMA)



Das gemeinsame Medium erscheint im Diagramm als Wolke. Wenn es sich bei dem gängigen Medium um ein Frame-Relay oder eine beliebige NBMA-Cloud handelt, verhält sich das Gerät genauso, als ob eine Verbindung über Ethernet besteht. RTC kündigt der RTA 192.168.180.20 mit dem nächsten Hop 172.16.20.3 an.

Das Problem besteht darin, dass die RTA keinen direkten permanenten Virtual Circuit (PVC) zu RTD hat und den nächsten Hop nicht erreichen kann. In diesem Fall schlägt das Routing fehl.

Der next-hop-selfBefehl behebt diese Situation.

Next-Hop-Self-Befehl

In Situationen mit dem nächsten Hop, wie im Beispiel des BGP Next Hop (NBMA), können Sie den next-hop-self Befehl verwenden. Die Syntax lautet:

<#root>

```
neighbor {ip-address | peer-group-name} next-hop-self
```

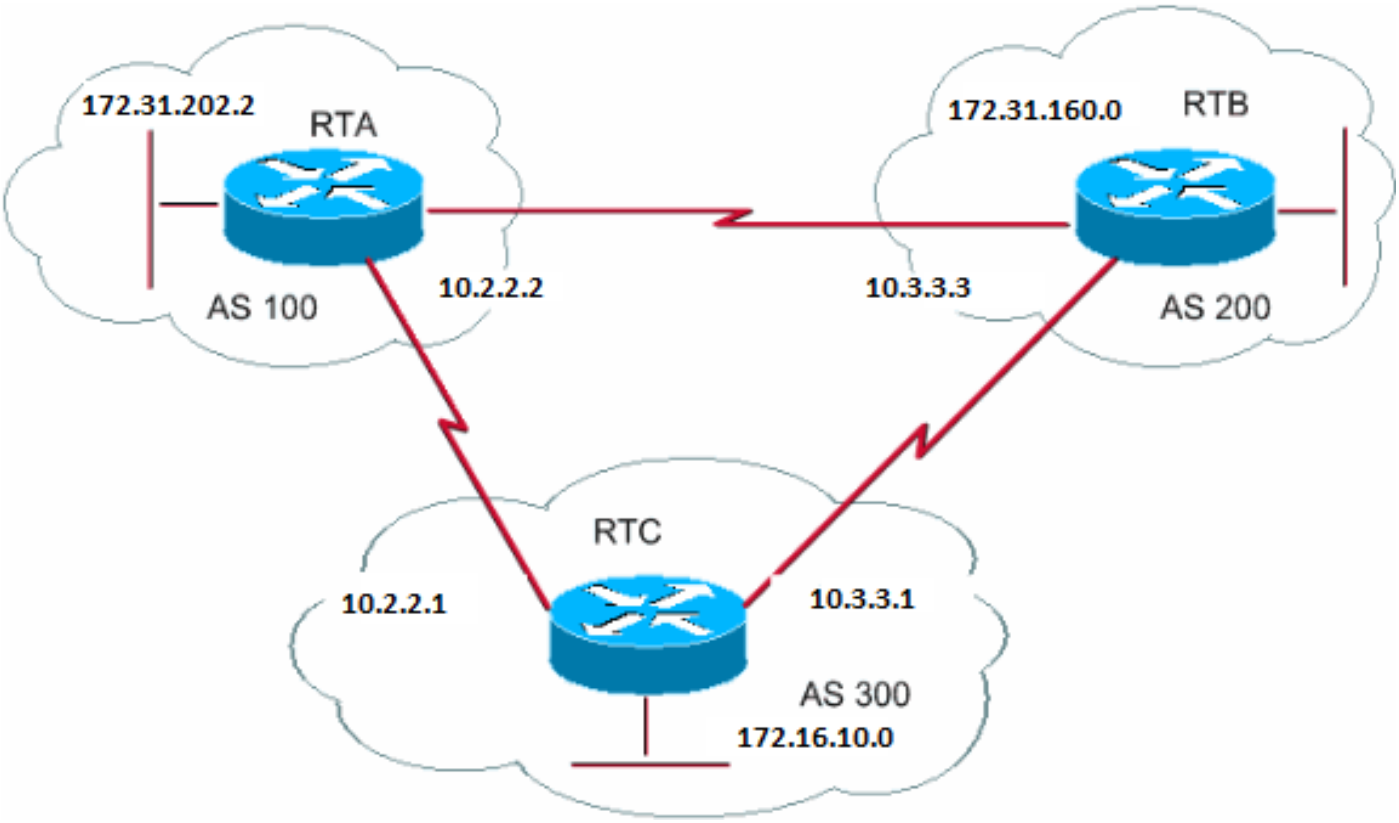
Mit diesem next-hop-selfBefehl können Sie das BGP zwingen, eine bestimmte IP-Adresse als nächsten Hop zu verwenden.

Im Beispiel für BGP Next Hop (NBMA) löst diese Konfiguration das Problem:

```
RTC#
router bgp 300
 neighbor 172.31.20.1 remote-as 100
 neighbor 172.31.20.1 next-hop-self
```

RTC kündigt 192.168.180.20 mit einem Next Hop gleich 172.31.20.2 an.

BGP-Backdoor



Im vorherigen Diagramm wurde eBGP auf RTA und RTC ausgeführt. RTB und RTC führen eBGP aus. RTA und RTB führen eine Art IGP aus,

entweder RIP, IGRP oder ein anderes Protokoll. eBGP-Updates haben per Definition eine Entfernung von 20, was kleiner ist als die IGP-Entfernungen. Die Standardabstände sind:

- 120 für RIP
- 100 für IGRP
- 90 für EIGRP
- 10 für OSPF

RTA empfängt Updates zu 172.31.160.0 über zwei Routing-Protokolle:

- eBGP mit einer Entfernung von 20
- IGP mit einer Entfernung größer als 20

BGP hat standardmäßig die folgenden Entfernungen:

- Externe Entfernung - 20
- Interner Abstand - 200

- Lokale Entfernung - 200

Sie können jedoch den distance Befehl verwenden, um die Standardabstände zu ändern:

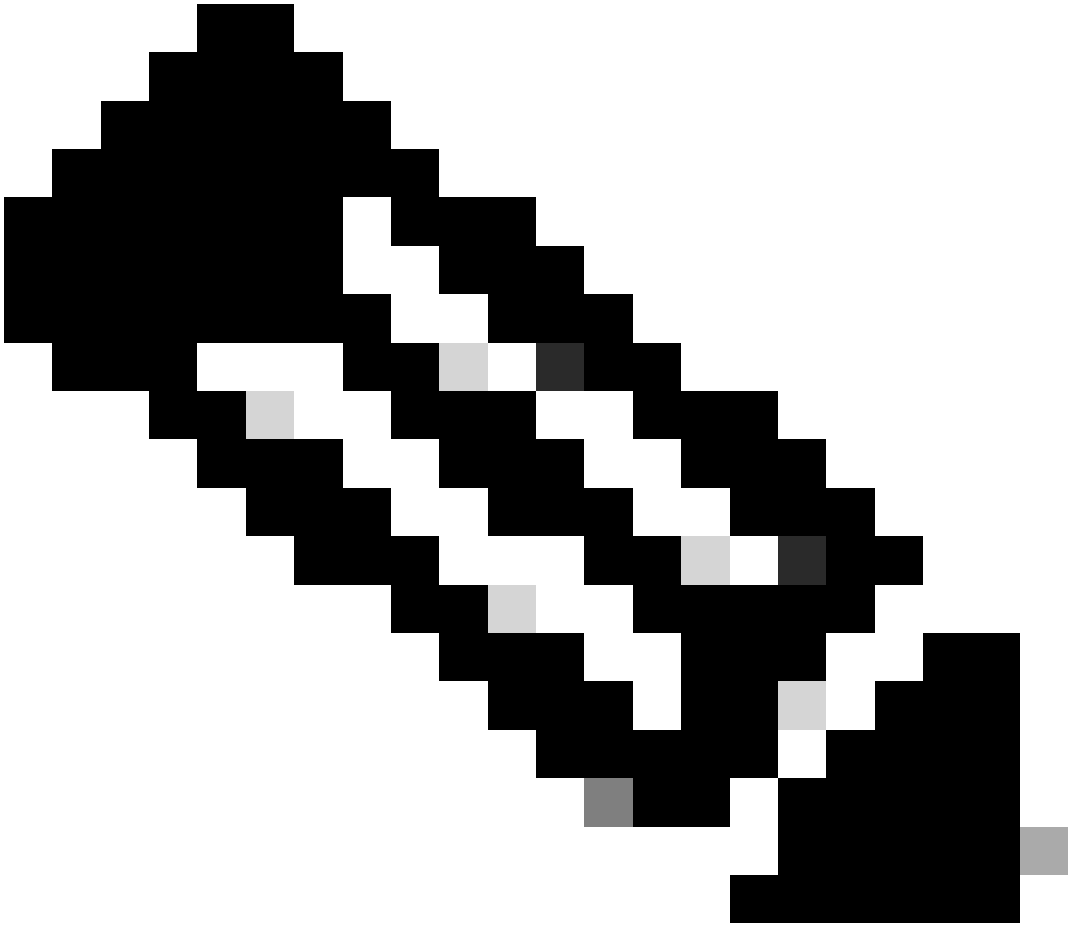
<#root>

distance bgp <external-distance> <internal-distance> <local-distance>

Aufgrund der kürzeren Distanz wählt die RTA eBGP via RTC aus.

Wenn Sie möchten, dass die RTA mehr über 172.31.160.0 via RTB (IGP) erfahren soll, dann haben Sie zwei Möglichkeiten:

- Ändern Sie die externe Distanz des eBGP oder die IGP-Distanz.



Hinweis: Diese Änderung wird nicht empfohlen.

- BGP Backdoor verwenden.

Durch die BGP-Backdoor wird die IGP-Route zur bevorzugten Route.

Geben Sie den Befehl **networkAddressbackdoor** ein.

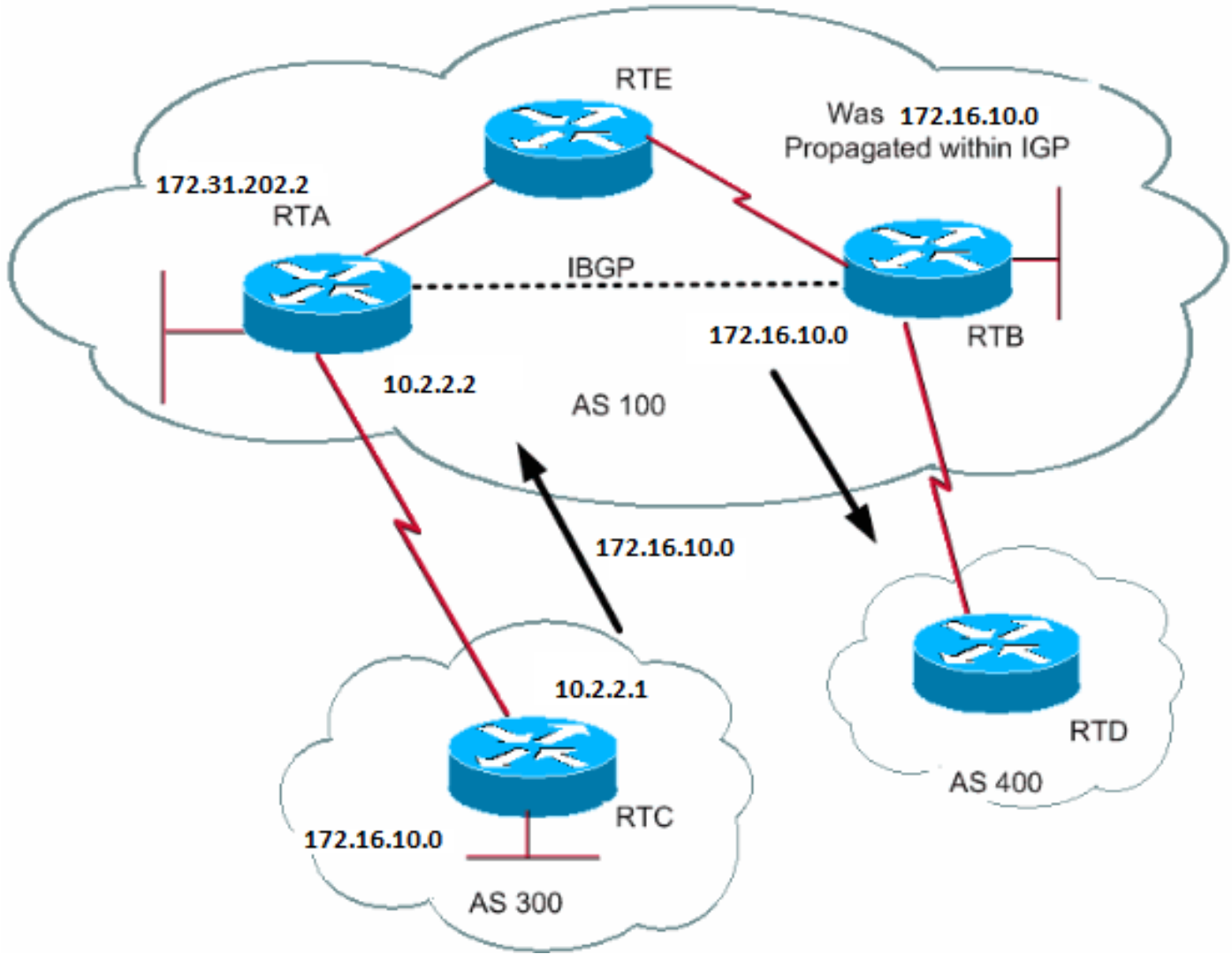
Das konfigurierte Netzwerk ist das Netzwerk, das Sie über IGP erreichen möchten. Beim BGP wird dieses Netzwerk genauso behandelt wie ein lokal zugewiesenes Netzwerk, es sei denn, BGP-Updates kündigen dieses Netzwerk nicht an.

```
RTA#
router eigrp 10
 network 172.31.202.2

router bgp 100
 neighbor 10.2.2.1 remote-as 300
 network 172.31.160.0 backdoor
```

Netzwerk 172.31.160.0 wird als lokaler Eintrag behandelt, aber nicht als normaler Netzwerkeintrag angekündigt.

RTA lernt 172.31.160.0 von RTB via EIGRP mit Distanz 90. Die RTA erfährt die Adresse vom RTC auch via eBGP mit der Distanz 20. In der Regel wird eBGP bevorzugt, aber aufgrund des **Network Backdoor**-Befehls EIGRP.

Synchronisierung



Bevor Sie auf die Synchronisierung eingehen, sollten Sie sich dieses Szenario ansehen. RTC in AS300 sendet Updates über 172.16.10.0. RTA

und RTB laufen iBGP, sodass RTB das Update bekommt und 172.16.10.0 über den nächsten Hop 10.2.2.1 erreichen kann. Denken Sie daran, dass der nächste Hop über iBGP übertragen wird. Um den nächsten Hop zu erreichen, muss die RTB den Datenverkehr an RTE senden.

Angenommen, die RTA hat das Netzwerk 172.16.10.0 nicht in IGP umverteilt. An diesem Punkt hat RTE keine Ahnung, dass 172.16.10.0 überhaupt existiert.

Wenn RTB gegenüber AS400 ankündigt, dass RTB 172.16.10.0 erreichen kann, fließt Datenverkehr, der von RTD zu RTB mit dem Ziel 172.16.10.0 kommt, ein und fällt bei RTE.

Bei der Synchronisierung wird festgelegt, dass das BGP, wenn das AS Datenverkehr von einem anderen AS an ein drittes AS weiterleitet, keine Route ankündigen darf, bevor alle Router im AS von der Route über IGP erfahren haben. BGP wartet, bis IGP die Route innerhalb des AS propagiert hat. Anschließend kündigt das BGP die Route externen Peers an.

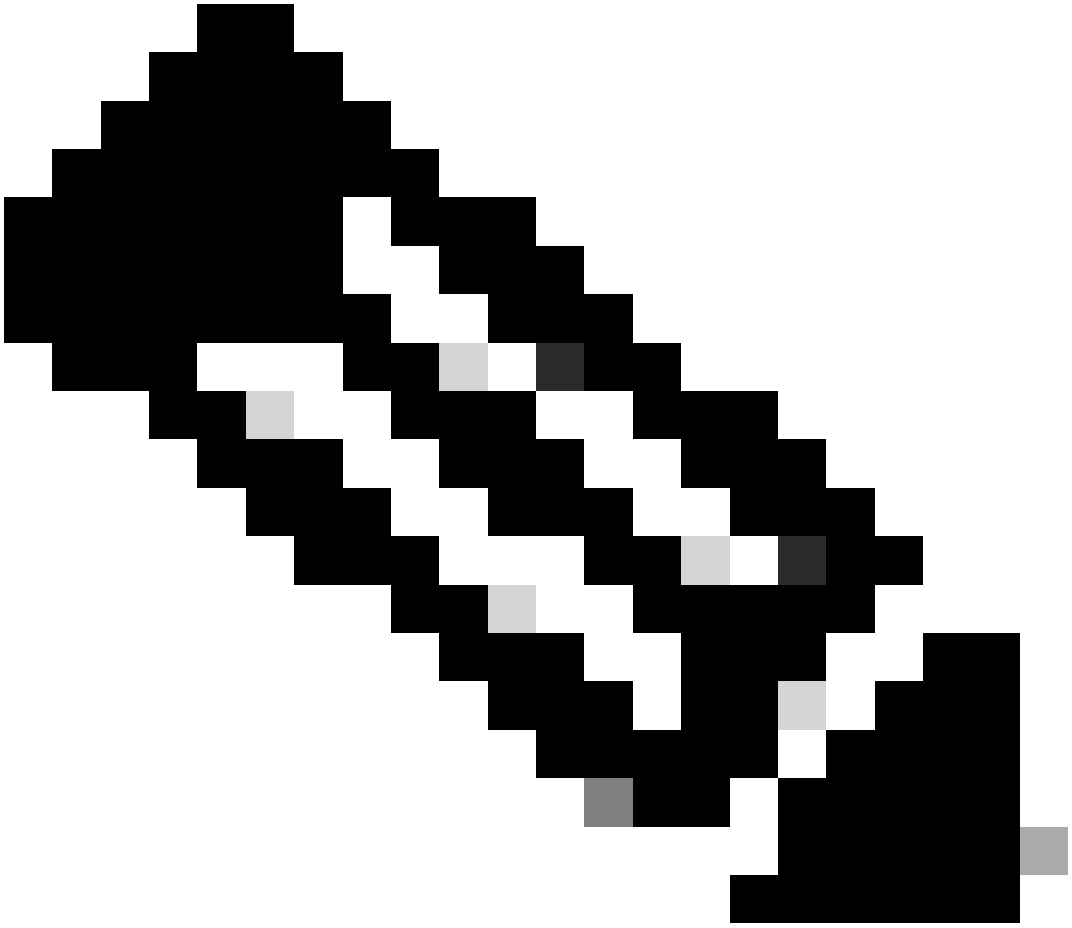
Im Beispiel in diesem Abschnitt wartet die RTB auf Informationen zu 172.16.10.0 über IGP. Anschließend sendet RTB das Update an RTD. Sie können RTB glauben machen, dass IGP die Informationen propagiert hat, wenn Sie eine statische Route in RTB hinzufügen, die auf 172.16.10.0 verweist. Stellen Sie sicher, dass andere Router 172.16.10.0 erreichen können.

Synchronisierung deaktivieren

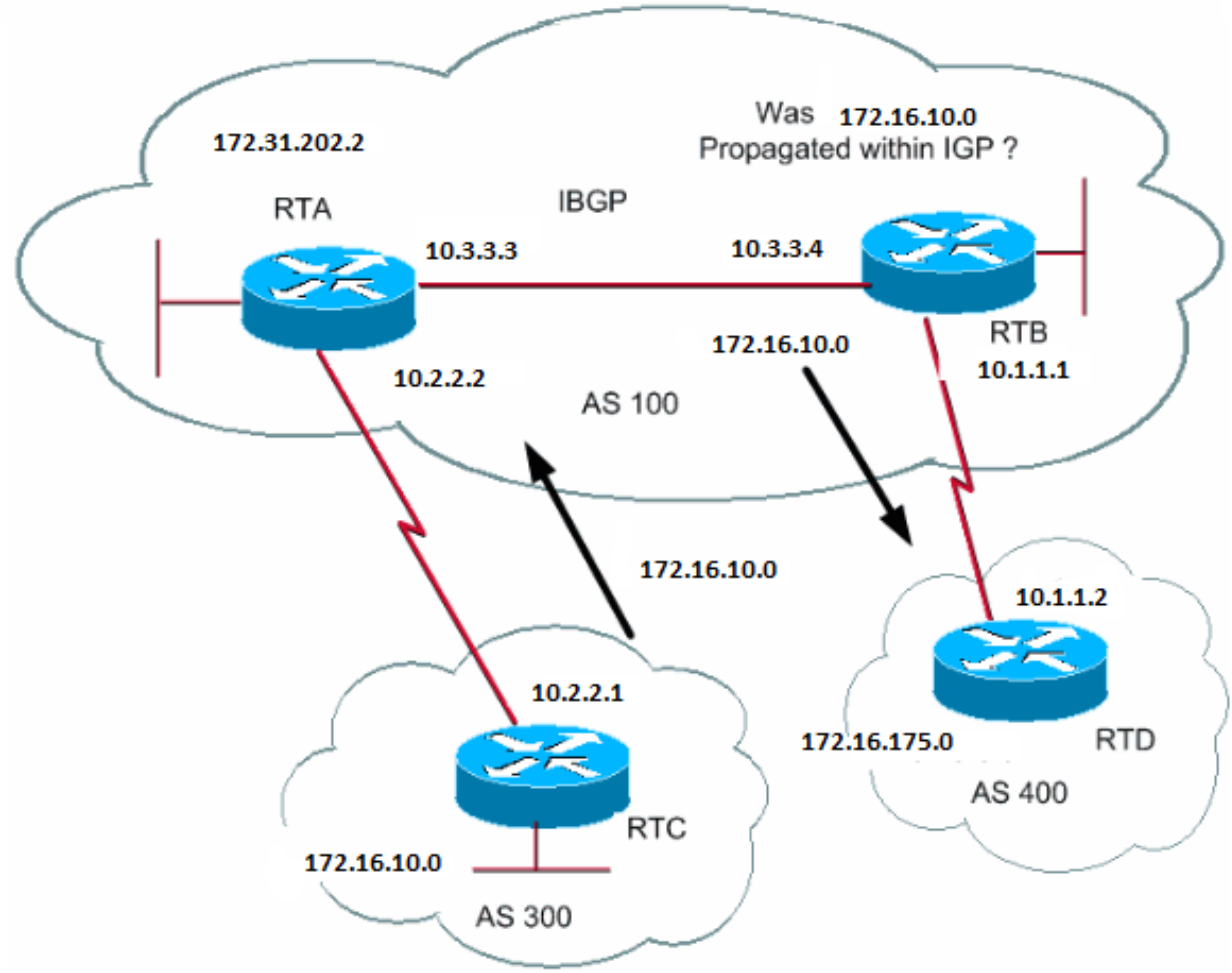
In einigen Fällen ist keine Synchronisierung erforderlich. Wenn Sie keinen Datenverkehr von einem anderen AS durch das AS leiten, können Sie die Synchronisierung deaktivieren. Sie können die Synchronisierung auch deaktivieren, wenn auf allen Routern in Ihrem AS BGP ausgeführt wird. Durch die Deaktivierung dieser Funktion können Sie im IGP weniger Routen übertragen und die Konvergenz des BGP beschleunigen.

Die Deaktivierung der Synchronisierung erfolgt nicht automatisch. Wenn auf all Ihren Routern im AS BGP und gar kein IGP ausgeführt wird, kann der Router davon nichts wissen. Ihr Router wartet unbegrenzt auf ein IGP-Update zu einer bestimmten Route, bevor der Router die Route an externe Peers sendet. In diesem Fall müssen Sie die Synchronisierung manuell deaktivieren, damit das Routing ordnungsgemäß funktioniert:

```
router bgp 100
 no synchronization
```

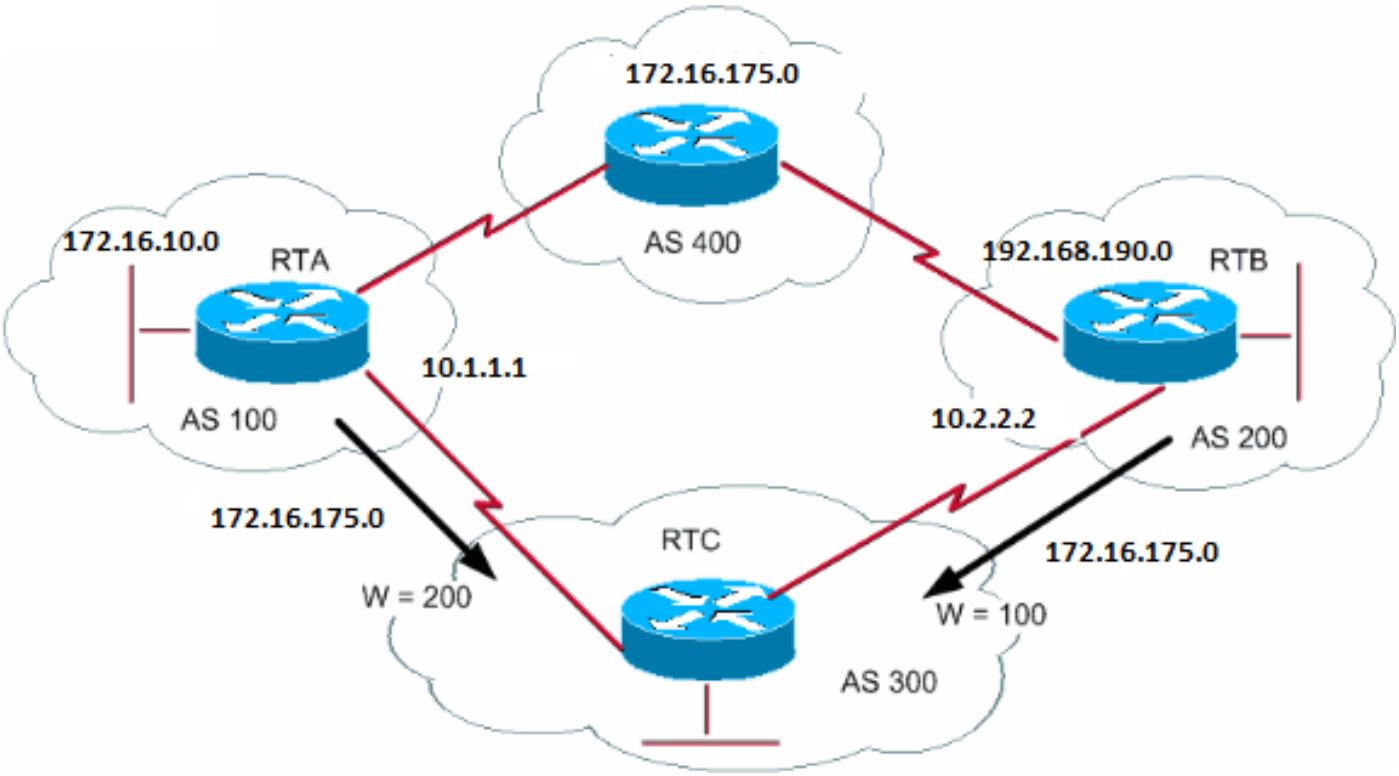


Hinweis: Stellen Sie sicher, dass Sie den Befehl clear ip bgp address eingeben, um die Sitzung zurückzusetzen.




```
RTB#
router bgp 100
 network 172.31.202.2
 neighbor 10.1.1.2 remote-as 400
 neighbor 10.3.3.3 remote-as 100
 no synchronization

!--- RTB puts 172.16.10.0 in its IP routing table and advertises the network
!--- to RTD, even if RTB does not have an IGP path to 172.16.10.0.

RTD#
router bgp 400
 neighbor 10.1.1.1 remote-as 100
 network 172.16.0.0

RTA#
router bgp 100
 network 172.31.202.2
 neighbor 10.3.3.4 remote-as 100
```

Gewichtungsattribut



Das weight-Attribut ist ein von Cisco definiertes Attribut. Dieses Attribut verwendet die Gewichtung, um einen besten Pfad auszuwählen. Das Gewicht wird lokal dem Router zugewiesen. Der Wert ist nur für den jeweiligen Router sinnvoll. Der Wert wird nicht propagiert oder über die Routen-Updates übertragen. Bei einem Gewicht kann es sich um eine Zahl zwischen 0 und 65.535 handeln. Pfade, die vom Router stammen, haben standardmäßig ein Gewicht von 32.768, andere Pfade haben ein Gewicht von 0.

Routen mit einer höheren Gewichtung haben Vorrang, wenn mehrere Routen zum gleichen Ziel vorhanden sind. Sehen Sie sich das Beispiel in diesem Abschnitt an. Die RTA hat von AS4 Informationen zum Netzwerk 172.16.0.0 erhalten. RTA leitet das Update an RTC weiter. Die RTB hat außerdem von AS4 Informationen zum Netzwerk 172.16.0.0 erhalten. RTB meldet das Update an RTC. RTC hat nun zwei Möglichkeiten, um 172.16.0.0 zu erreichen und muss entscheiden, welchen Weg es geht. Wenn Sie das Gewicht der Updates auf RTC, die von RTA kommen, so einstellen, dass das Gewicht größer ist als das Gewicht der Updates, die von RTB kommen, zwingen Sie RTC, RTA als nächsten Hop zu verwenden, um 172.16.0.0 zu erreichen. Mehrere Methoden erreichen diese Gewichtsgruppe:

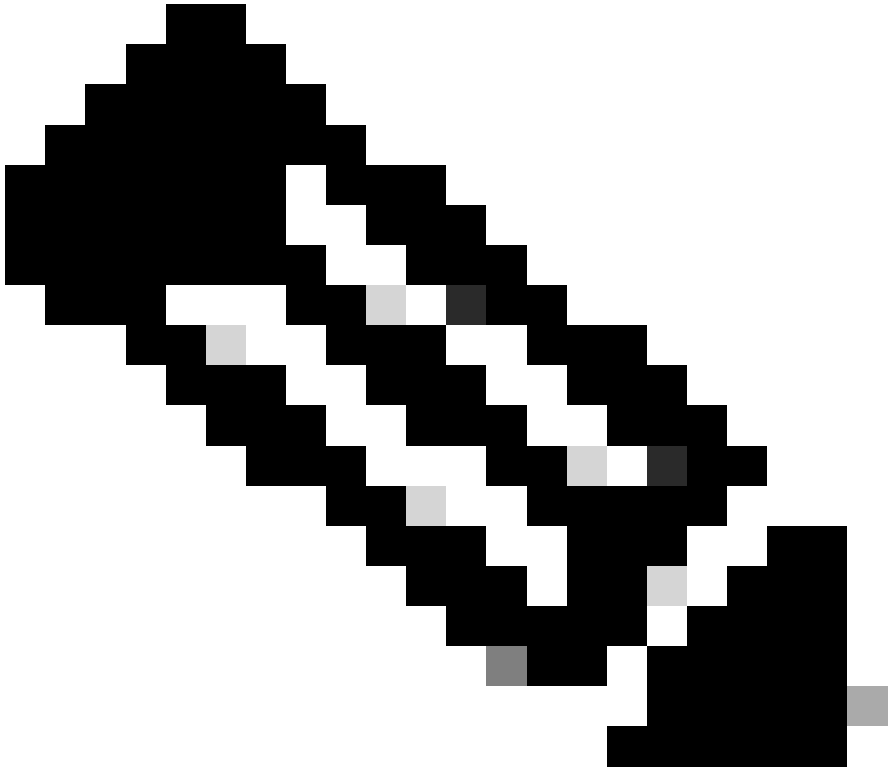
- Verwenden Sie den Befehl **neighbor**.

  neighbor {ip-address|peer-group} weight <weight>

- AS_PATH-Zugriffslisten verwenden.

- **ip as-path access-list <access-list-number>{permit | deny} <as-regular-expression>**

- **neighbor <ip-address>filterliste <access-list-number>weight <weight>**



Hinweis: In einigen Szenarien können sehr wenige Befehle vorhanden sein, die in einigen Softwareversionen nicht verfügbar sind.

- Routenpläne verwenden.

```
RTC#
router bgp 300
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 weight 200

!--- The route to 172.16.0.0 from RTA has a 200 weight.

 neighbor 10.2.2.2 remote-as 200
 neighbor 10.2.2.2 weight 100

!--- The route to 172.16.0.0 from RTB has a 100 weight.
```

Die RTA, die einen höheren Gewichtungswert hat, hat die Präferenz als nächster Hop.

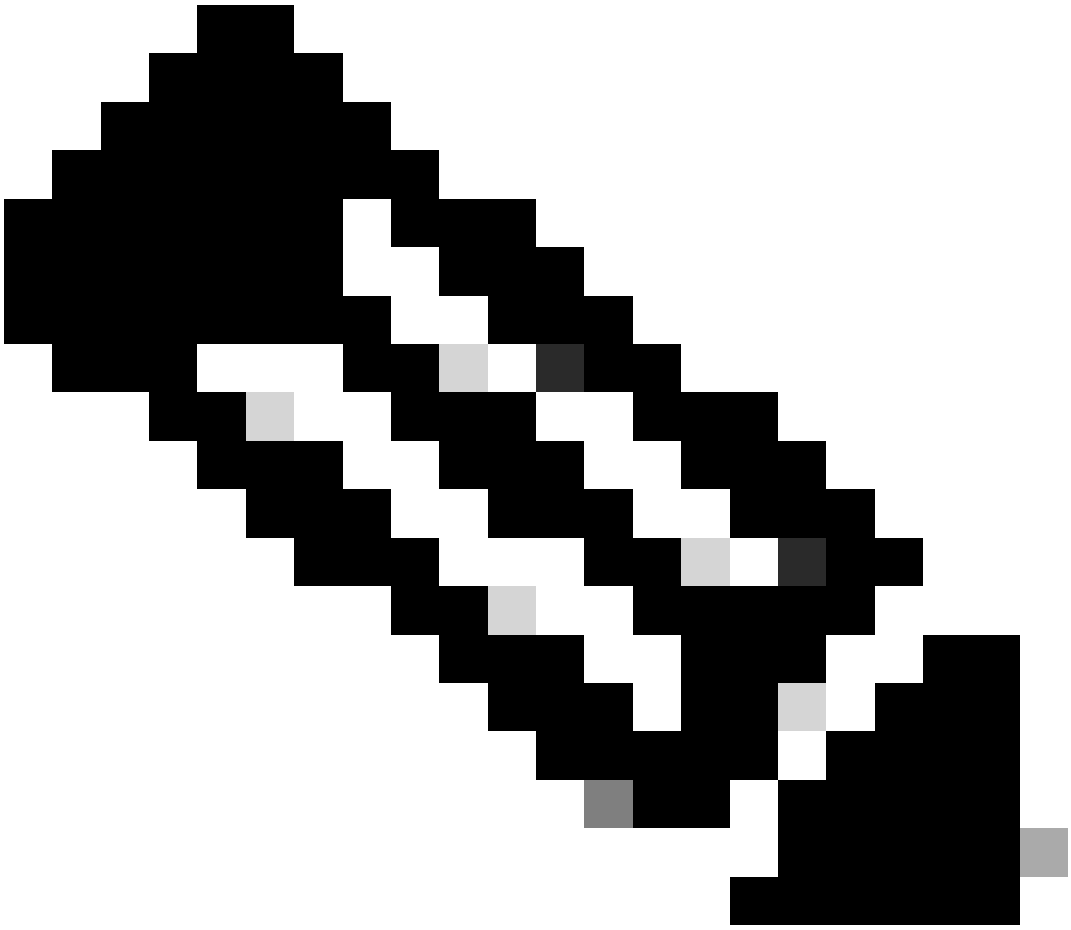
Dasselbe Ergebnis können Sie mit IP AS_PATH und Filterlisten erreichen.

```
RTC#
router bgp 300
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 filter-list 5 weight 200
 neighbor 10.2.2.2 remote-as 200
 neighbor 10.2.2.2 filter-list 6 weight 100
...
ip as-path access-list 5 permit ^100$

!--- This only permits path 100.

ip as-path access-list 6 permit ^200$
...
```

Dasselbe Ergebnis können Sie auch mit Routenplänen erzielen.

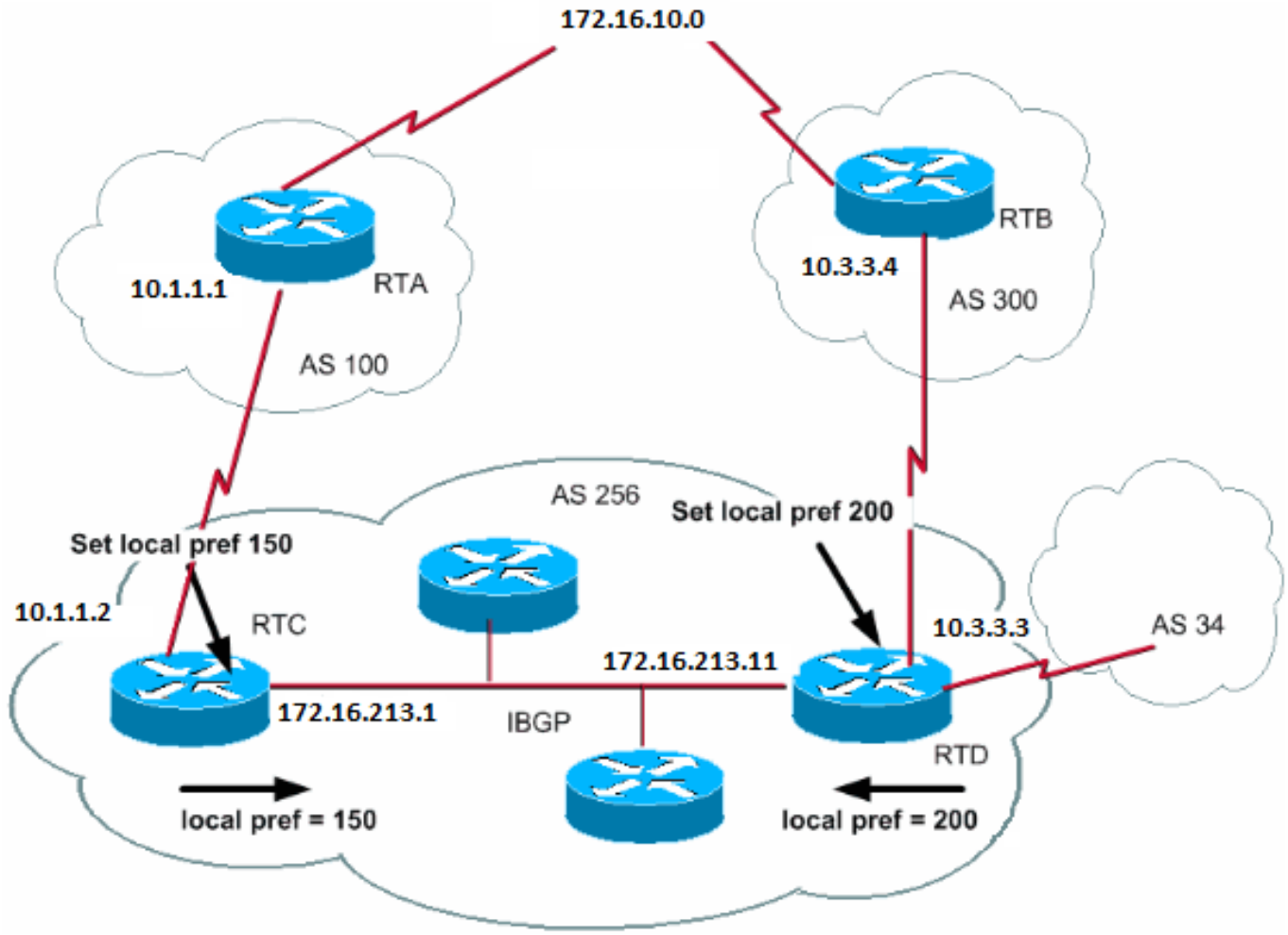
```
RTC#
router bgp 300
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 route-map setweightin in
 neighbor 10.2.2.2 remote-as 200
 neighbor 10.2.2.2 route-map setweightin in
...
```

```
ip as-path access-list 5 permit ^100$
...

route-map setweightin permit 10
 match as-path 5
 set weight 200

!--- Anything that applies to access list 5, such as packets from AS100, has weight 200.

route-map setweightin permit 20
 set weight 100

!--- Anything else has weight 100.
```



Hinweis: Sie können die Gewichtung ändern, um den MPLS-VPN-BGP-Pfad mit dem IGP-Pfad als Backup vorzuziehen.

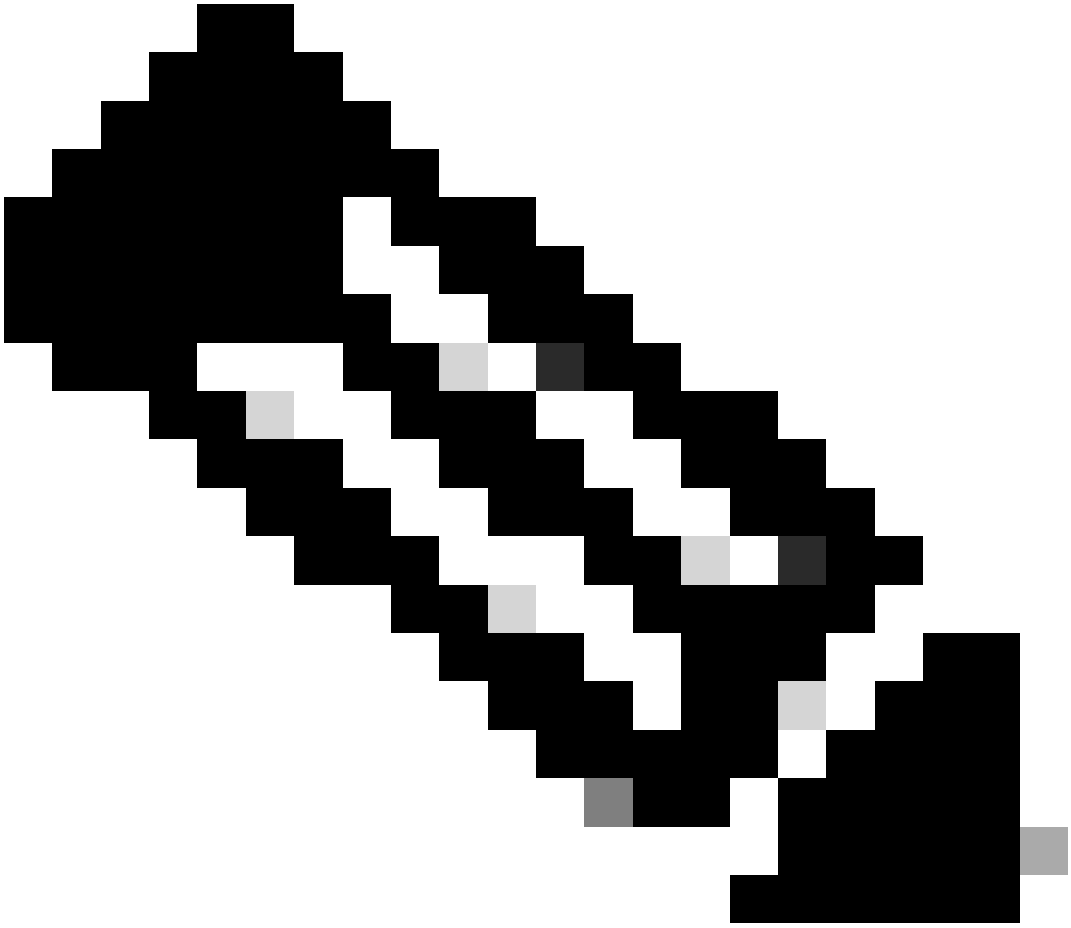
Lokales Voreinstellungsattribut



Die lokale Voreinstellung ist ein Hinweis für das AS darauf, welcher Pfad bevorzugt das AS verlässt, um ein bestimmtes Netzwerk zu erreichen. Ein Pfad mit einer höheren lokalen Präferenz wird weiter bevorzugt. Der Standardwert für die lokale Voreinstellung ist 100.

Im Gegensatz zum weight-Attribut, das nur für den lokalen Router relevant ist, ist die lokale Präferenz ein Attribut, das Router im selben AS austauschen.

Mit dem Befehl **bgp default local-preference value** können Sie die lokale Voreinstellung festlegen. Sie können auch lokale Präferenzen für Routing-Zuordnungen festlegen, wie im Beispiel in diesem Abschnitt veranschaulicht:



Hinweis: Ein Soft-Reset (d. h. Löschen des BGP-Prozesses auf dem Router) ist erforderlich, damit die Änderungen berücksichtigt werden. Um den BGP-Prozess zu löschen, verwenden Sie den clear ip bgp [soft][in/out] Befehl, bei dem "soft" ein Soft Reset anzeigt und die Sitzung nicht beendet wird, und [in/out] die eingehende oder ausgehende Konfiguration angibt. Wenn kein In/Out angegeben ist, werden sowohl eingehende als auch ausgehende Sitzungen zurückgesetzt.

Mit dem Befehl **bgp default local-preference** wird die lokale Einstellung für die Updates des Routers festgelegt, die zu Peers im selben AS geleitet werden. Im Diagramm in diesem Abschnitt erhält AS256 Updates zu 172.16.10.0 von zwei verschiedenen Seiten der Organisation. Mithilfe der lokalen Einstellungen können Sie bestimmen, wie das AS256 beendet werden soll, um das Netzwerk zu erreichen. Es wird angenommen, dass als Ausgangspunkt RTD verwendet wird. Mit dieser Konfiguration werden die lokalen Einstellungen für Updates von AS300 bis 200 und für Updates von AS100 bis 150 festgelegt:

```
RTC#
router bgp 256
 neighbor 10.1.1.1 remote-as 100
  neighbor 10.213.11.2 remote-as 256
 bgp default local-preference 150

RTD#
router bgp 256
 neighbor 10.3.3.4 remote-as 300
 neighbor 10.213.11.1 remote-as 256
 bgp default local-preference 200
```

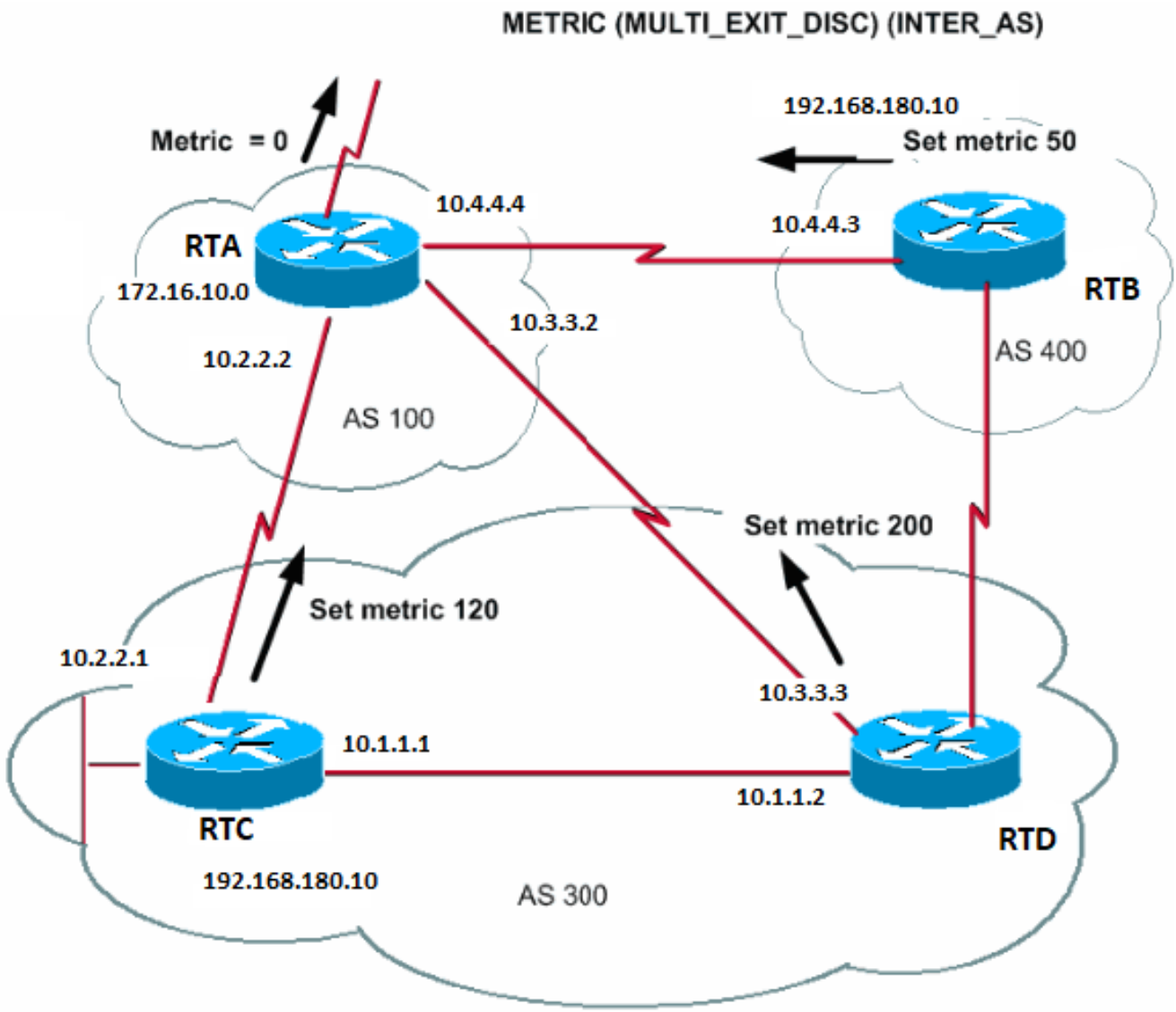
In dieser Konfiguration setzt RTC die lokale Voreinstellung aller Updates auf 150. Mit der gleichen RTD-Einstellung wird die lokale Einstellung aller Aktualisierungen auf 200 festgelegt. Innerhalb von AS256 findet ein Austausch lokaler Präferenzen statt. Daher erkennen sowohl RTC als auch RTD, dass das Netzwerk 172.16.10.0 eine höhere lokale Präferenz hat, wenn Updates von AS300 anstatt von AS100 stammen. Der gesamte Datenverkehr im AS256, der dieses Netzwerk als Ziel hat, wird mit RTD als Ausgangspunkt übertragen.

Die Verwendung von Routing-Karten bietet mehr Flexibilität. Im Beispiel in diesem Abschnitt werden alle Updates, die RTD erhält, mit der lokalen Präferenz 200 markiert, wenn die Updates RTD erreichen. Updates von AS34 werden ebenfalls mit der lokalen Präferenz von 200 gekennzeichnet. Dieses Tag kann unnötig sein. Aus diesem Grund können Sie Routenzuordnungen verwenden, um die spezifischen Updates anzugeben, die mit einer spezifischen lokalen Präferenz versehen werden müssen. Hier ein Beispiel:

```
RTD#
router bgp 256
 neighbor 10.3.3.4 remote-as 300
 neighbor 10.3.3.4 route-map setlocalin in
 neighbor 10.213.11.1 remote-as 256
....
ip as-path access-list 7 permit ^300$
...

route-map setlocalin permit 10
 match as-path 7
 set local-preference 200

route-map setlocalin permit 20
 set local-preference 150
```

Bei dieser Konfiguration haben alle Updates von AS300 eine lokale Präferenz von 200. Alle anderen Updates, z. B. Updates von AS34, haben einen Wert von 150.

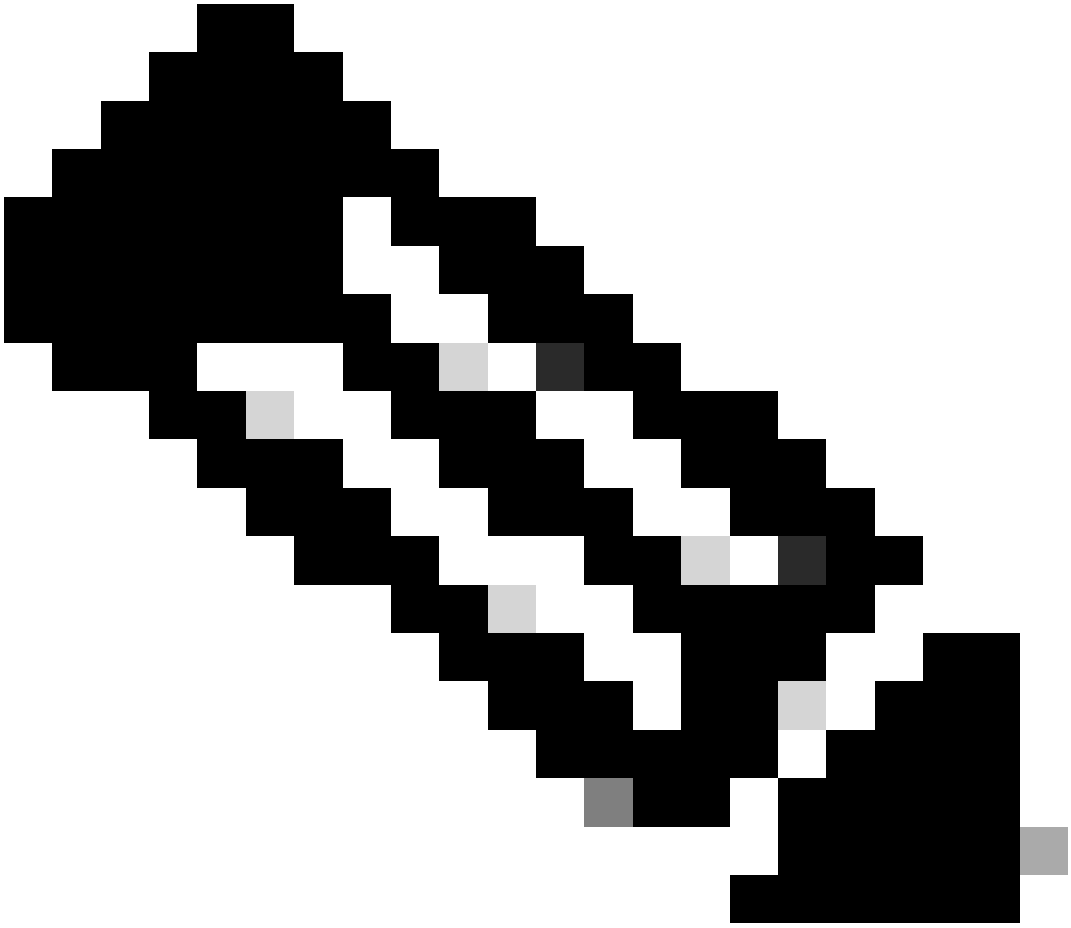
Metrisches Attribut



Das metrische Attribut hat auch den Namen MULTI_EXIT_DISCRIMINATOR, MED (BGP4) oder INTER_AS (BGP3). Das Attribut ist ein Hinweis an externe Nachbarn bezüglich der Pfadpräferenz zu einem AS. Das Attribut ermöglicht eine dynamische Beeinflussung eines anderen AS bei der Erreichung einer bestimmten Route, wenn mehrere Eintrittspunkte in dieses AS vorhanden sind. Ein niedrigerer Metrikwert wird mehr bevorzugt.

Anders als bei der lokalen Präferenz werden die Metriken zwischen den AS ausgetauscht. Eine Metrik wird in ein AS übernommen, verlässt das AS jedoch nicht. Wenn eine Aktualisierung mit einer bestimmten Metrik in das AS eingeht, wird diese Metrik verwendet, um Entscheidungen innerhalb des AS zu treffen. Wenn dieselbe Aktualisierung an ein drittes AS weitergeleitet wird, wird dieser Wert auf 0 zurückgesetzt. Das Diagramm in diesem Abschnitt zeigt die Metrik. Der metrische Standardwert ist 0.

Sofern ein Router keine anderen Anweisungen erhält, vergleicht er die Kennzahlen für die Pfade von Nachbarn im selben AS. Damit der Router Metriken von Nachbarn vergleichen kann, die von verschiedenen ASs stammen, müssen Sie den speziellen Konfigurationsbefehl **bgp always-compare-med** auf dem Router eingeben.



Hinweis: Es gibt zwei BGP-Konfigurationsbefehle, die die MED-basierte Pfadauswahl beeinflussen können. Die Befehle sind der Befehl bgp deterministic-med und der Befehl bgp always-compare-med. Eine Ausgabe des Befehls "bgp deterministic-med" stellt den Vergleich der MED-Variablen bei der Routenauswahl sicher, wenn verschiedene Peers im gleichen AS angeben. Eine Ausgabe des Befehls bgp always-compare-med stellt den Vergleich der MED für Pfade von Nachbarn in verschiedenen ASs sicher. Der Befehl bgp always-compare-med ist nützlich, wenn sich mehrere Service Provider oder Unternehmen auf eine einheitliche Richtlinie für die MED-Einstellung einigen. Unter Unterschiede zwischen dem deterministischen BGP-Med-Befehl und dem BGP Always-Compare-Med-Befehl erfahren Sie, wie sich diese Befehle auf die BGP-Pfadauswahl auswirken.

Im Diagramm in diesem Abschnitt erhält AS100 Informationen zum Netzwerk 192.168.180.10 über drei verschiedene Router: RTC, RTD und RTB. RTC und RTD gehören zum AS300, RTB zum AS400.

In diesem Beispiel wird der AS-Path-Vergleich auf RTA mit dem Befehl **bgp bestpath as-path ignore** ignoriert. Er ist so konfiguriert, dass BGP beim Routenvergleich auf das nächste Attribut (in diesem Fall Metrik oder MED) hereinfallen muss. Wenn der Befehl ausgelassen wird,

kann das BGP die Route 192.168.180.10 vom Router-RTC installieren, da dieser den kürzesten AS-Pfad aufweist.

Angenommen, Sie haben die von RTC kommende Kennzahl auf 120, die von RTD auf 200 und die von RTB auf 50 gesetzt. Standardmäßig vergleicht ein Router Metriken, die von Nachbarn im selben AS stammen. Daher kann RTA nur die von RTC stammende Kennzahl mit der von RTD vergleichen. RTA wählt RTC als besten Next Hop, da 120 weniger als 200 ist. Wenn RTA ein Update von RTB mit Metrik 50 erhält, kann RTA die Metrik nicht mit 120 vergleichen, da RTC und RTB sich in unterschiedlichen ASs befinden. Die RTA muss anhand einiger anderer Attribute auswählen.

Um RTA zu zwingen, die Kennzahlen zu vergleichen, müssen Sie den Befehl **bgp always-compare-med** auf RTA ausführen. Diese Konfigurationen veranschaulichen diesen Prozess:

```
RTA#
router bgp 100
 neighbor 10.2.2.1 remote-as 300
 neighbor 10.3.3.3 remote-as 300
 neighbor 10.4.4.3 remote-as 400
 bgp bestpath as-path ignore

RTC#
router bgp 300
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 route-map setmetricout out
 neighbor 10.1.1.2 remote-as 300

route-map setmetricout permit 10
   set metric 120

RTD#
router bgp 300
 neighbor 10.3.3.2 remote-as 100
 neighbor 10.3.3.2 route-map setmetricout out
 neighbor 10.1.1.1 remote-as 300

route-map setmetricout permit 10
 set metric 200

RTB#
router bgp 400
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.4.4.4 route-map setmetricout out

route-map setmetricout permit 10
 set metric 50
```

Bei diesen Konfigurationen wählt RTA RTC als nächsten Hop, da alle anderen Attribute gleich sind. Um RTB in den Metrikvergleich aufzunehmen, muss die RTA wie folgt konfiguriert werden:

```
RTA#
router bgp 100
 neighbor 2.2.21 remote-as 300
 neighbor 10.3.3.3 remote-as 300
```

```
 neighbor 10.4.4.3 remote-as 400
 bgp always-compare-med
```

In diesem Fall wählt die RTA RTB als besten Next Hop, um das Netzwerk 192.168.180.10 zu erreichen.

Wenn Sie den Befehl **default-metricnumber** ausführen, können Sie auch während der Neuverteilung von Routen in das BGP Metriken festlegen.

Angenommen, in dem Beispiel in diesem Abschnitt injiziert die RTB ein Netzwerk über statische Verbindungen in AS100. Hier die Konfiguration:

```
RTB#
router bgp 400
 redistribute static
 default-metric 50

ip route 192.168.180.10 255.255.0.0 null 0

!--- This causes RTB to send out 192.168.180.10 with a metric of 50.
```

Community-Attribut

Das Community-Attribut ist ein transitives, optionales Attribut im Bereich von 0 bis 4.294.967.200. Mit dem Communityattribut können Ziele in einer bestimmten Community gruppiert und Routing-Entscheidungen angewendet werden, die zu diesen Communitys passen. Die Routing-Entscheidungen lauten unter anderem "Annehmen", "Bevorzugen" und "Umverteilen".

Sie können Routing Maps verwenden, um die Community-Attribute festzulegen. Der Befehl route map **set** hat folgende Syntax:

<#root>

set community community-number [additive] [well-known-community]

Einige vordefinierte, bekannte Communitys zur Verwendung in diesem Befehl sind:

- **no-export** - Nicht an eBGP-Peers weitergeben. Bewahren Sie diese Route innerhalb eines AS auf.
- **no-advertise**: Diese Route darf keinem Peer (intern oder extern) gemeldet werden.
- **internet** - Gebt diese Route der Internet-Community bekannt. Jeder Router gehört zu dieser Community.
- **local-as** - Verwenden Sie in Konföderationsszenarien, um die Übertragung von Paketen außerhalb des lokalen AS zu verhindern.

Hier zwei Beispiele für Routing-Karten, die die Community prägen:

```
route-map communitymap
 match ip address 1
 set community no-advertise
```

Oder

```
route-map setcommunity
 match as-path 1
 set community 200 additive
```

Wenn Sie das **additive** Schlüsselwort nicht festlegen, ersetzt 200 jede alte Community, die bereits existiert. Wenn Sie das Schlüsselwort **additive** verwenden, eine Zugabe von 200 zur Gemeinschaft erfolgt. Selbst wenn Sie das Community-Attribut festlegen, wird dieses Attribut standardmäßig nicht an Nachbarn übertragen. Um das Attribut an einen Nachbarn zu senden, müssen Sie den folgenden Befehl verwenden:

neighbor {ip-address | peer-group-name} send-community

Hier ein Beispiel:

```
RTA#
router bgp 100
 neighbor 10.3.3.3 remote-as 300
 neighbor 10.3.3.3 send-community
 neighbor 10.3.3.3 route-map setcommunity out
```

In Cisco IOS Software, Version 12.0 und höher, können Sie Communitys in drei verschiedenen Formaten konfigurieren: dezimal, hexadezimal und AA:NN. Standardmäßig verwendet die Cisco IOS Software das ältere Dezimalformat. Um die Konfiguration und Anzeige in AA:NN zu ermöglichen, geben Sie den Befehl **ip bgp-community new-**global configuration format ein. Der erste Teil von AA:NN steht für die AS-Nummer und der zweite Teil für eine 2-Byte-Nummer.

Hier ein Beispiel:

Ohne den Befehl **ip bgp-community new-format** in der globalen Konfiguration wird der Wert des Community-Attributs mit dem Befehl **show ip bgp 10.6.0.0** im Dezimalformat angezeigt. In diesem Beispiel wird der Community-Attributwert als6553620 angezeigt.

<#root>

```
Router#
```

show ip bgp 10.6.0.0

```
BGP routing table entry for 10.6.0.0/8, version 7
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  1
    10.10.10.1 from 10.10.10.1 (10.255.255.1)
      Origin IGP, metric 0, localpref 100, valid, external, best
```

Community: 6553620

Geben Sie nun den Befehl **ip bgp-community new-format** global auf diesem Router ein.

```
<#root>

Router#


configure terminal


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


ip bgp-community new-format


Router(config)#


exit
```

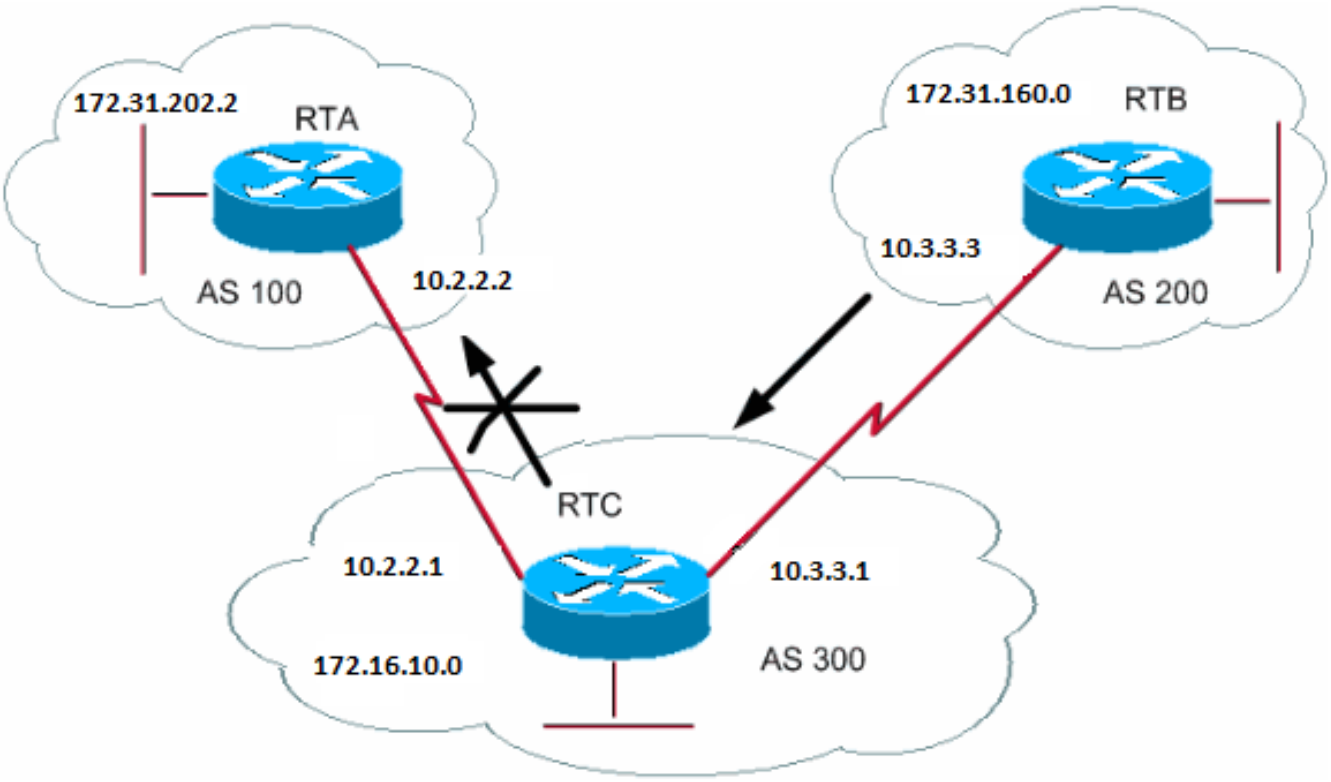
Mit dem Befehl **eip bgp-community new**-formatglobal configuration wird der Community-Wert im Format AA:NN angezeigt. Der Wert wird in der Ausgabe des Befehls **how ip bgp 10.6.0.0** als`100:20` angezeigt. In diesem Beispiel:

```
    <#root>

Router#


show ip bgp 10.6.0.0


BGP routing table entry for 10.6.0.0/8, version 9
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  1
    10.10.10.1 from 10.10.10.1 (10.255.255.1)
      Origin IGP, metric 0, localpref 100, valid, external, best


Community: 100:20
```


BGP-Filter

Eine Reihe unterschiedlicher Filtermethoden ermöglicht Ihnen, das Senden und Empfangen von BGP-Updates zu steuern. Sie können BGP-Updates nach Routeninformationen oder nach Pfadinformationen oder Communitys filtern. Alle Methoden erzielen die gleichen Ergebnisse. Die Wahl einer Methode gegenüber einer anderen Methode hängt von der jeweiligen Netzwerkkonfiguration ab.

Routenfilter



Um die Routing-Informationen einzuschränken, die der Router erhält oder ankündigt, können Sie BGP mithilfe von Routing-Updates für oder von einem bestimmten Nachbarn filtern. Sie definieren eine Zugriffsliste und wenden die Zugriffsliste auf die Aktualisierungen an oder von einem Nachbarn an. Führen Sie diesen Befehl im Router-Konfigurationsmodus aus:

<#root>

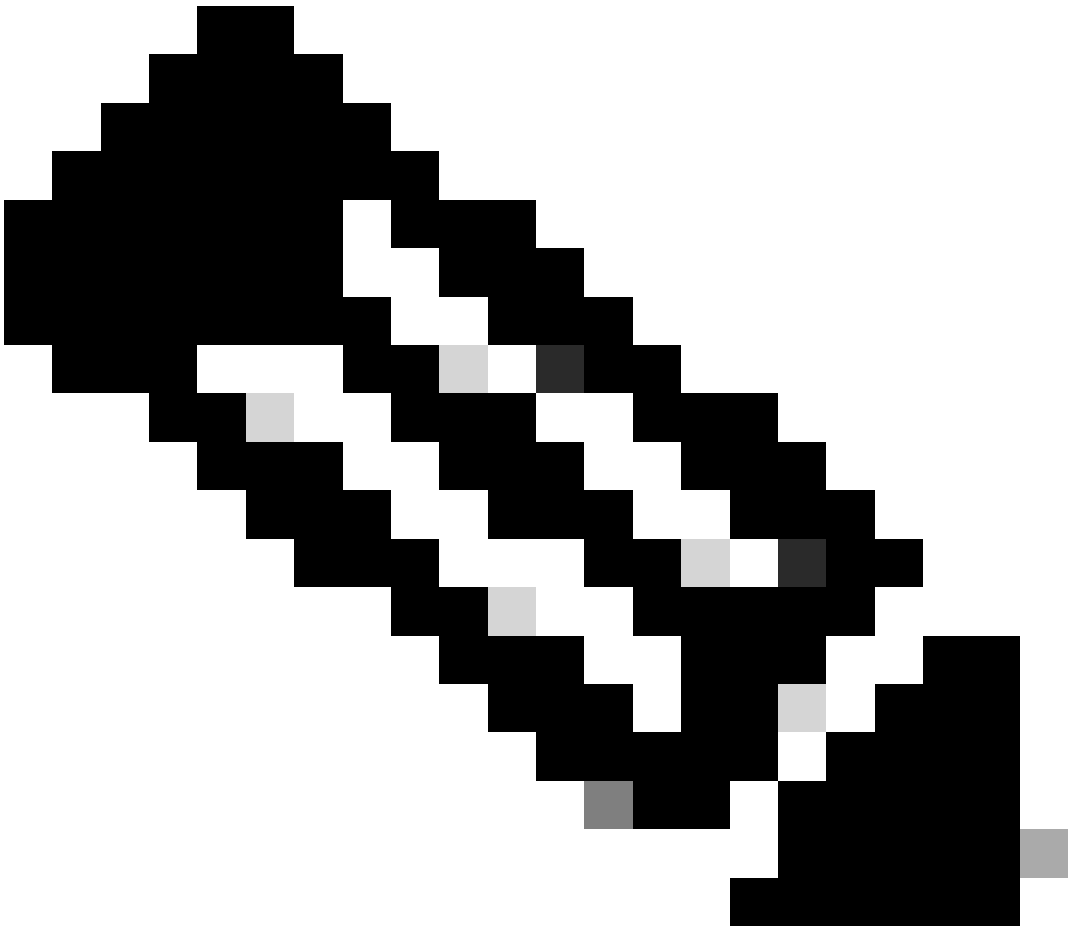
neighbor {ip-address | peer-group-name} distribute-list access-list-number {in | out}

In diesem Beispiel stammt das Netzwerk 172.31.160.0 von RTB und sendet das Update an RTC. Wenn RTC die Weitergabe der Updates an AS100 stoppen möchte, müssen Sie eine Zugriffsliste definieren, um diese Updates zu filtern, und die Zugriffsliste während der Kommunikation mit RTA anwenden:

```
RTC#
router bgp 300
 network 172.16.10.0
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 distribute-list 1 out

access-list 1 deny 172.31.160.0 0.0.255.255

access-list 1 permit 0.0.0.0 255.255.255.255

!--- Filter out all routing updates about 160.10.x.x.
```

Die Verwendung von Zugriffslisten ist etwas kompliziert, wenn Sie mit Supernets arbeiten, die zu Konflikten führen können.

Angenommen, in dem Beispiel in diesem Abschnitt weist die RTB unterschiedliche Subnetze auf: 160.10.x.x. Ihr Ziel ist es, Updates zu filtern und nur 192.168.160.0/8 anzukündigen.



Hinweis: Die Notation /8 bedeutet, dass Sie 8 Bit der Subnetzmaske verwenden, die von der äußersten linken Seite der IP-Adresse ausgeht. Diese Adresse entspricht 192.168.160.0 255.0.0.0.

Der Befehl access-list 1 permit 192.168.160.0 0.255.255.25 5erlaubt 192.168.160.0/8, 192.168.160.0/9 usw. Um das Update auf 192.168.160.0/8 zu beschränken, müssen Sie eine erweiterte Zugriffsliste dieses Formats verwenden:

<#root>

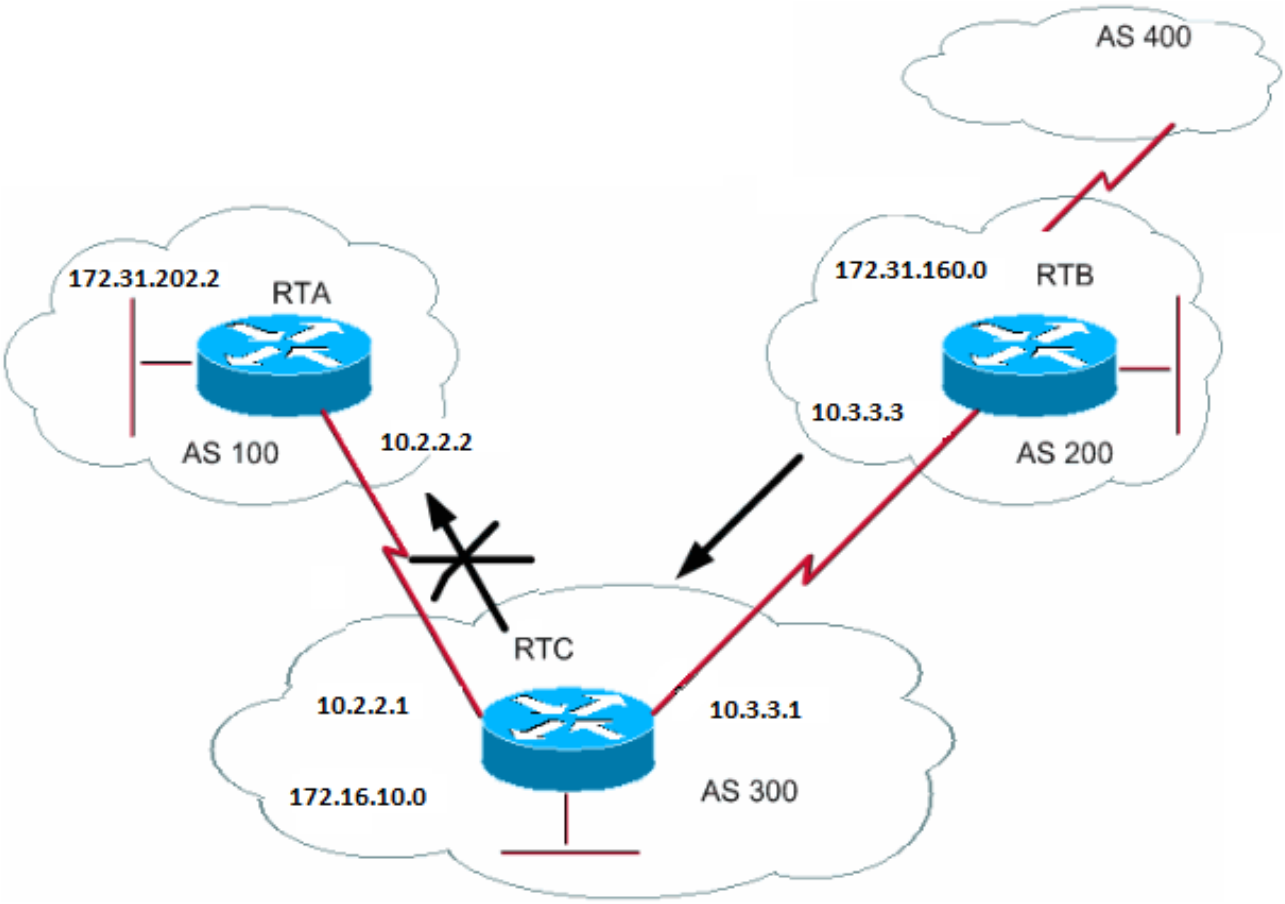
access-list 101 permit ip 192.168.160.0 0.255.255.255 255.0.0.0 0.0.0.0.

Diese Liste erlaubt nur 192.168.160.0/8.

Beispielkonfigurationen für die Filterung von Netzwerken von BGP-Peers finden Sie unter Ein oder mehrere Netzwerke von einem BGP-Peer blockieren. Die Methode verwendet den Befehl **distribute-list** mit standardmäßigen und erweiterten Zugriffskontrolllisten (ACLs) sowie die Möglichkeit, die Präfixliste zu filtern.

Pfadfilter

Sie können auch Pfade filtern.



Mithilfe der BGP-AS-Pfade können Sie eine Zugriffsliste für eingehende und ausgehende Updates festlegen. In dem Diagramm in diesem Abschnitt können Sie Updates zu 172.31.160.0 blockieren, sodass sie nicht zu AS100 wechseln. Um die Updates zu blockieren, definieren Sie eine Zugriffsliste auf RTC, die die Übertragung von Updates, die von AS200 stammen, an AS100 verhindert. Führen Sie folgende Befehle aus:

<#root>

ip as-path access-list access-list-number {permit | deny} as-regular-expression

<#root>

neighbor {ip-address | peer-group-name} filter-list access-list-number {in | out}

In diesem Beispiel wird der RTC-Versand von Updates über 172.31.160.0 an RTA beendet:

```
RTC#
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 filter-list 1 out

!--- The 1 is the access list number below.

ip as-path access-list 1 deny ^200$
ip as-path access-list 1 permit .*
```

Der access-list 1Befehl in diesem Beispiel erzwingt das Verweigern von Updates mit Pfadinformationen, die mit 200 beginnen und mit 200 enden. ^200$im Befehl ist ein "regulärer Ausdruck", in dem^bedeutet "beginnt mit" und$ bedeutet "endet mit". Da RTB Updates über 172.31.160.0 mit Pfadinformationen sendet, die mit 200 beginnen und mit 200 enden, entsprechen die Updates der Zugriffsliste. Die Zugriffsliste verweigert diese Aktualisierungen.

Der.*Ausdruck ist ein weiterer regulärer Ausdruck, in dem die.bedeutet "beliebiges Zeichen" und die* bedeutet "die Wiederholung dieses Zeichens". Also.* stellt alle Pfadinformationen dar, die notwendig sind, um die Übertragung aller anderen Updates zu ermöglichen.

Was passiert, wenn Sie anstelle von^200$^200 verwenden? Bei einem AS400 enthalten Updates, die vom AS400 stammen, wie im Diagramm in diesem Abschnitt, Pfadinformationen des Formulars (200, 400). In dieser Pfadangabe stehen 200 als Erstes und 400 als Letztes. Diese Aktualisierungen entsprechen der Zugriffsliste^200, da die Pfadinformationen mit 200 beginnen. Die Zugriffsliste verhindert die Übertragung dieser Updates an die RTA, was nicht zwingend erforderlich ist.

Um zu überprüfen, ob Sie den richtigen regulären Ausdruck implementiert haben, geben Sie den Befehl **show ip bgp regexpregular-expression ein**. Dieser Befehl zeigt alle Pfade an, die mit der Konfiguration für reguläre Ausdrücke übereinstimmen.

AS-Regulärer Ausdruck

In diesem Abschnitt wird die Erstellung eines regulären Ausdrucks erläutert.

Ein regulärer Ausdruck ist ein Muster, das mit einer Eingabezeichenfolge verglichen werden muss. Wenn Sie einen regulären Ausdruck erstellen, geben Sie eine Zeichenfolge an, deren Eingabe übereinstimmen muss. Im Fall von BGP geben Sie eine Zeichenfolge an, die aus Pfadinformationen besteht, mit denen eine Eingabe übereinstimmen muss.

Im Beispiel im Abschnitt **Path Filter** haben Sie die Zeichenfolge`^200$` angegeben. Sie wollten, dass Pfadinformationen, die in Updates enthalten sind, mit der Zeichenfolge übereinstimmen, um zu entscheiden.

Ein regulärer Ausdruck umfasst Folgendes:

- **Bereich**

  Ein Bereich ist eine Folge von Zeichen innerhalb der linken und rechten eckigen Klammern. Ein Beispiel ist `[abcd]`.

- **Atom**

  Ein Atom ist ein einzelnes Zeichen. Hier einige Beispiele:

  `.`

  - Die `.`entspricht einem beliebigen Zeichen.

  `^`

  - Der `Wert` ^entspricht dem Anfang der Eingabezeichenfolge.

  `$`

- Der $ entspricht dem Ende der Eingabezeichenfolge.

\

- Der \entspricht dem Zeichen.

_

- Der_entspricht einem Komma (`,`), der linken Klammer (`{`), der rechten Klammer (`}`), dem Anfang der Eingabezeichenfolge, dem Ende der Eingabezeichenfolge oder einem Leerzeichen.

•

Stück

Ein Stück ist eines dieser Symbole, das nach einem Atom kommt:

*

- Das *entspricht 0 oder mehr Sequenzen des Atoms.

+

- Das +entspricht 1 oder mehr Sequenzen des Atoms.

?

- Das ?entspricht dem Atom oder der Nullzeichenfolge.

- **Zweigstelle**

  Eine Verzweigung besteht aus 0 oder mehr verketteten Teilen.

Hier einige Beispiele für reguläre Ausdrücke:

a*

- Dieser Ausdruck gibt jedes Vorkommen des Buchstabens "a" an, der keines enthält.

```
a+
```

- Dieser Ausdruck gibt an, dass mindestens ein Vorkommen des Buchstabens "a" vorhanden sein muss.

```
ab?a
```

- Dieser Ausdruck entspricht "aa" oder "aba".

```
_100_
```

- Dieser Ausdruck bedeutet über AS100.

```
_100$
```

- Dieser Ausdruck gibt einen Ursprung von AS100 an.

```
^100 .*
```

- Dieser Ausdruck gibt die Übertragung von AS100 an.

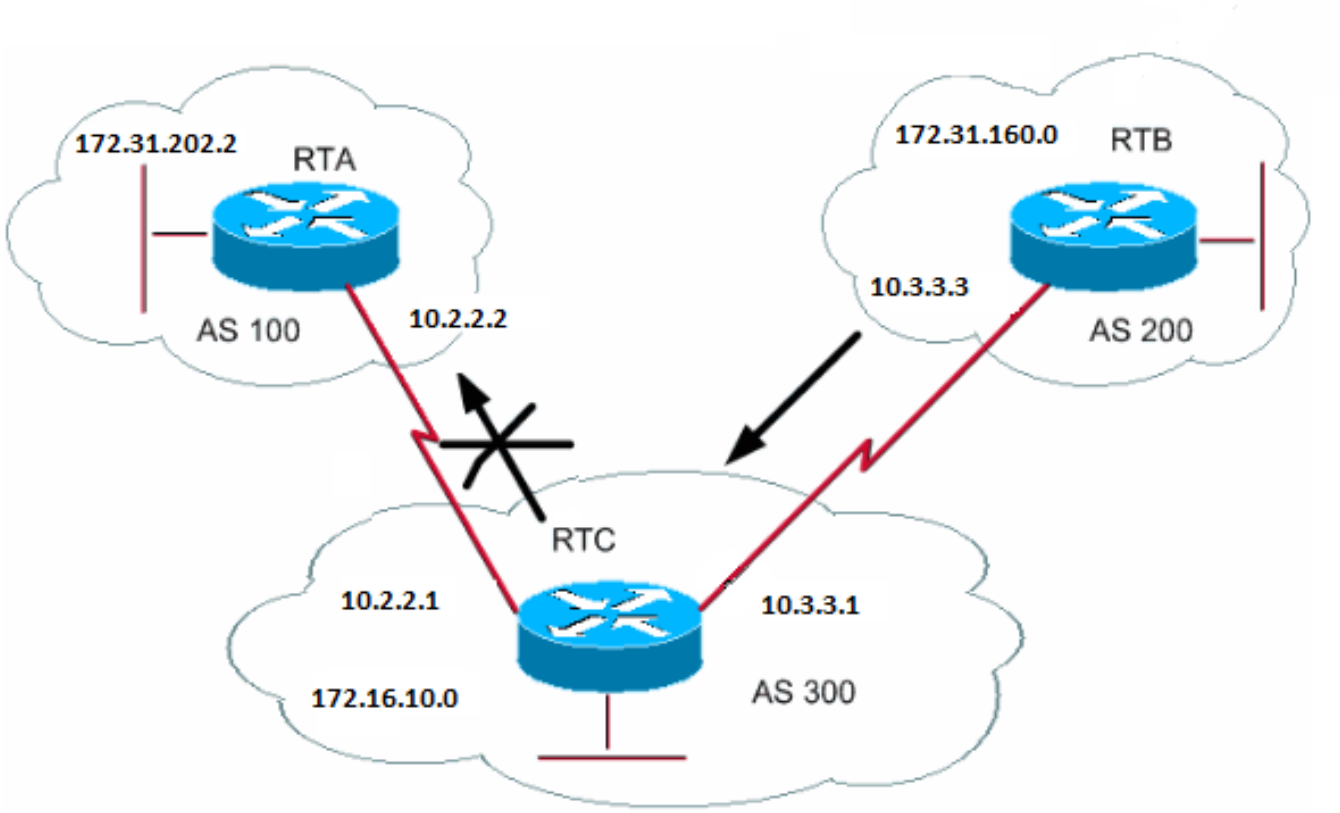
```
^$
```

- Dieser Ausdruck gibt den Ursprung dieses AS an.

Beispielkonfigurationen für die Filterung regulärer Ausdrücke finden Sie unter Reguläre Ausdrücke in BGP verwenden.

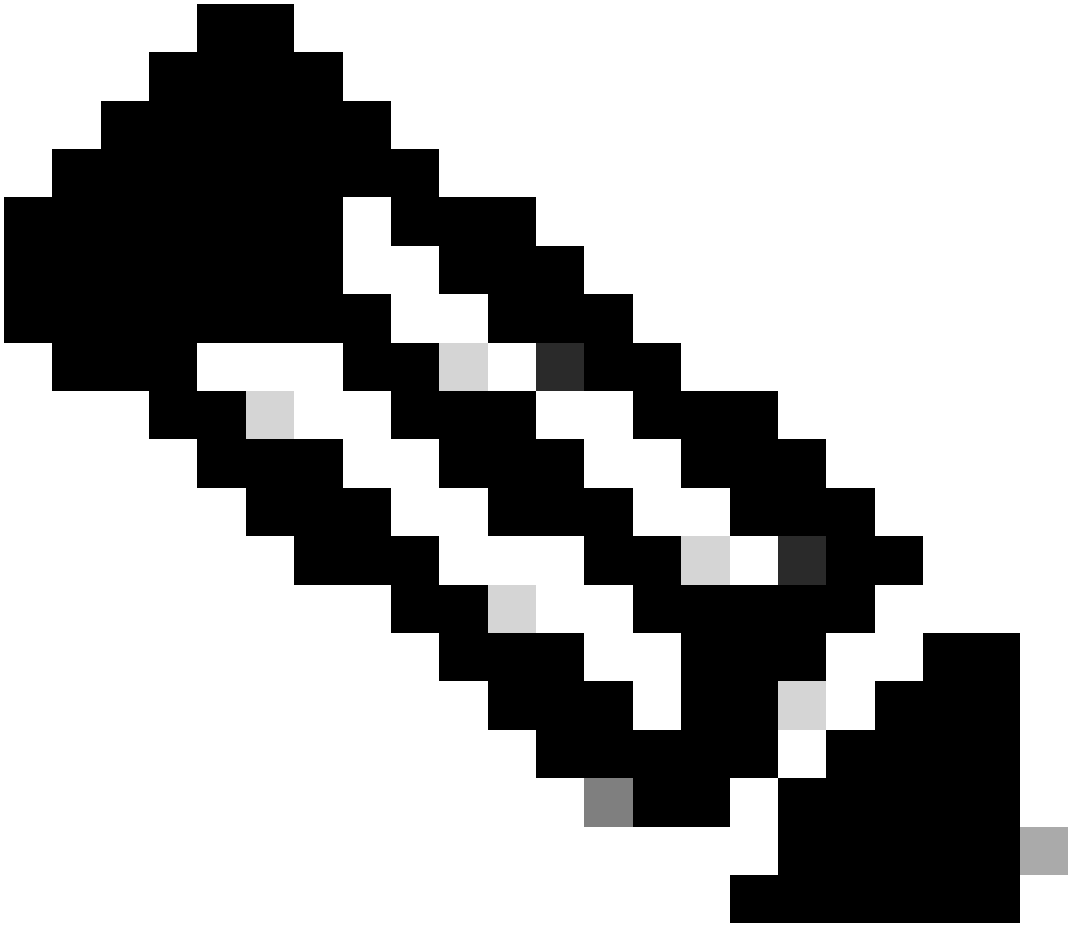
BGP-Community-Filter

In diesem Dokument werden die Routenfilterung und die AS-Pfad-Filterung behandelt. Eine weitere Methode ist die Community-Filterung. Im AbschnittCommunity-Attribut wird Community behandelt. Dieser Abschnitt enthält einige Beispiele für die Verwendung von Community.

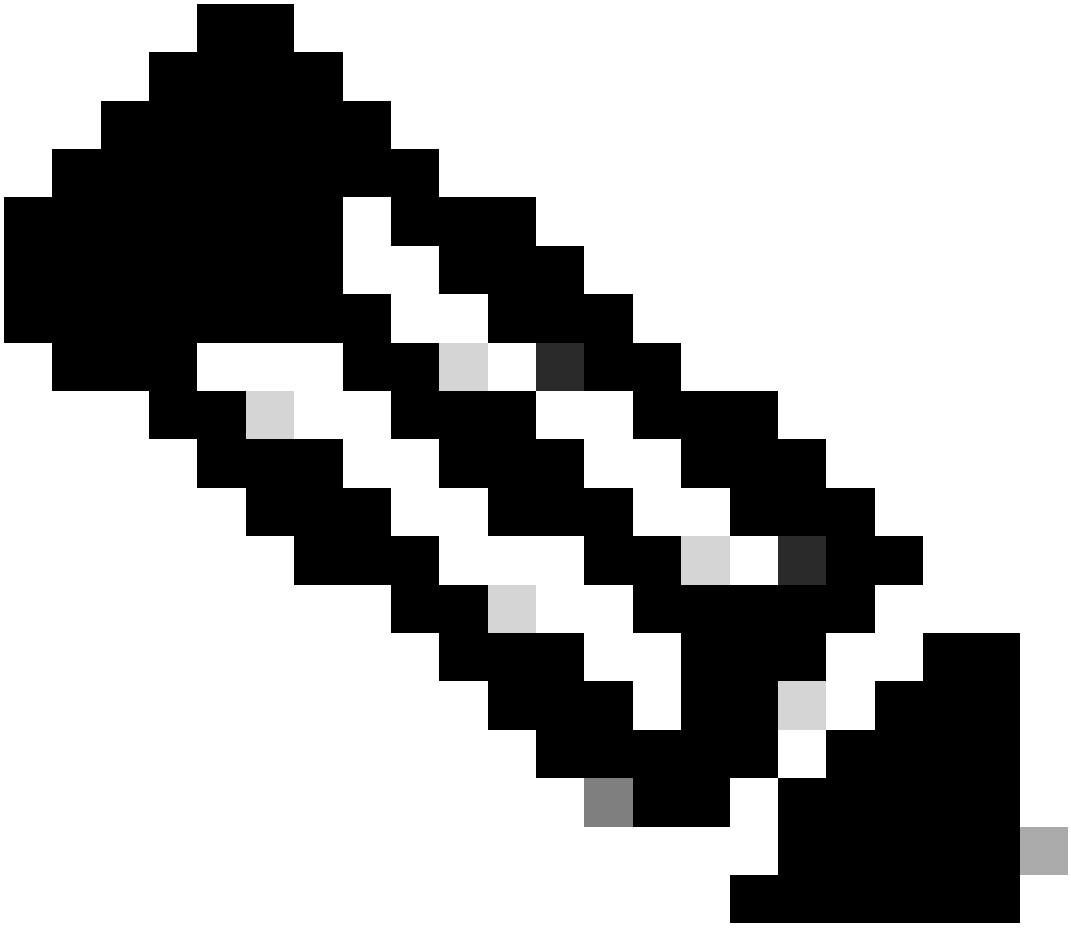


In diesem Beispiel soll das Community-Attribut auf die BGP-Routen festgelegt werden, die von der RTB angekündigt werden, sodass diese Routen von der RTC nicht an die externen Peers propagiert werden. Verwenden Sie das no-exportCommunity-Attribut.

```
RTB#
router bgp 200
 network 172.31.160.0
 neighbor 10.3.3.1 remote-as 300
 neighbor 10.3.3.1 send-community
 neighbor 10.3.3.1 route-map setcommunity out

route-map setcommunity
 match ip address 1
 set community no-export

access-list 1 permit 0.0.0.0 255.255.255.255
```



Hinweis: In diesem Beispiel wird der route-map setcommunity Befehl verwendet, um die Community auf "no-export" zu setzen.



Hinweis: Der **neighbor send-community** Befehl ist erforderlich, um dieses Attribut an RTC zu senden.

Wenn RTC die Updates mit dem Attribut NO_EXPORT erhält, propagiert RTC die Updates nicht an externe Peer-RTA.

In diesem Beispiel hat RTB das Community-Attribut auf**100 200 additive** festgelegt. Durch diese Aktion wird vor der Übertragung an RTC der Wert 100 bis 200 zu jedem aktuellen Community-Wert hinzugefügt.

```
RTB#
router bgp 200
 network 172.31.160.0
 neighbor 10.3.3.1 remote-as 300
```

```
 neighbor 10.3.3.1 send-community
 neighbor 10.3.3.1 route-map setcommunity out

route-map setcommunity
 match ip address 2
 set community 100 200 additive

access-list 2 permit 0.0.0.0 255.255.255.255
```

Eine Community-Liste ist eine Gruppe von Communitys, die Sie in einer **Match-**Klausel einer Routenübersicht verwenden. Mit der Community-Liste können Sie Attribute mit verschiedenen Listen von Community-Nummern filtern oder festlegen.

<#root>

ip community-list <community-list-number> {permit | deny} <community-number>

Sie können z. B. diese Routenübersicht **für "Match-on-Community"** definieren:

```
route-map match-on-community
 match community 10

!--- The community list number is 10.

 set weight 20
 ip community-list 10 permit 200 300

!--- The community number is 200 300.
```

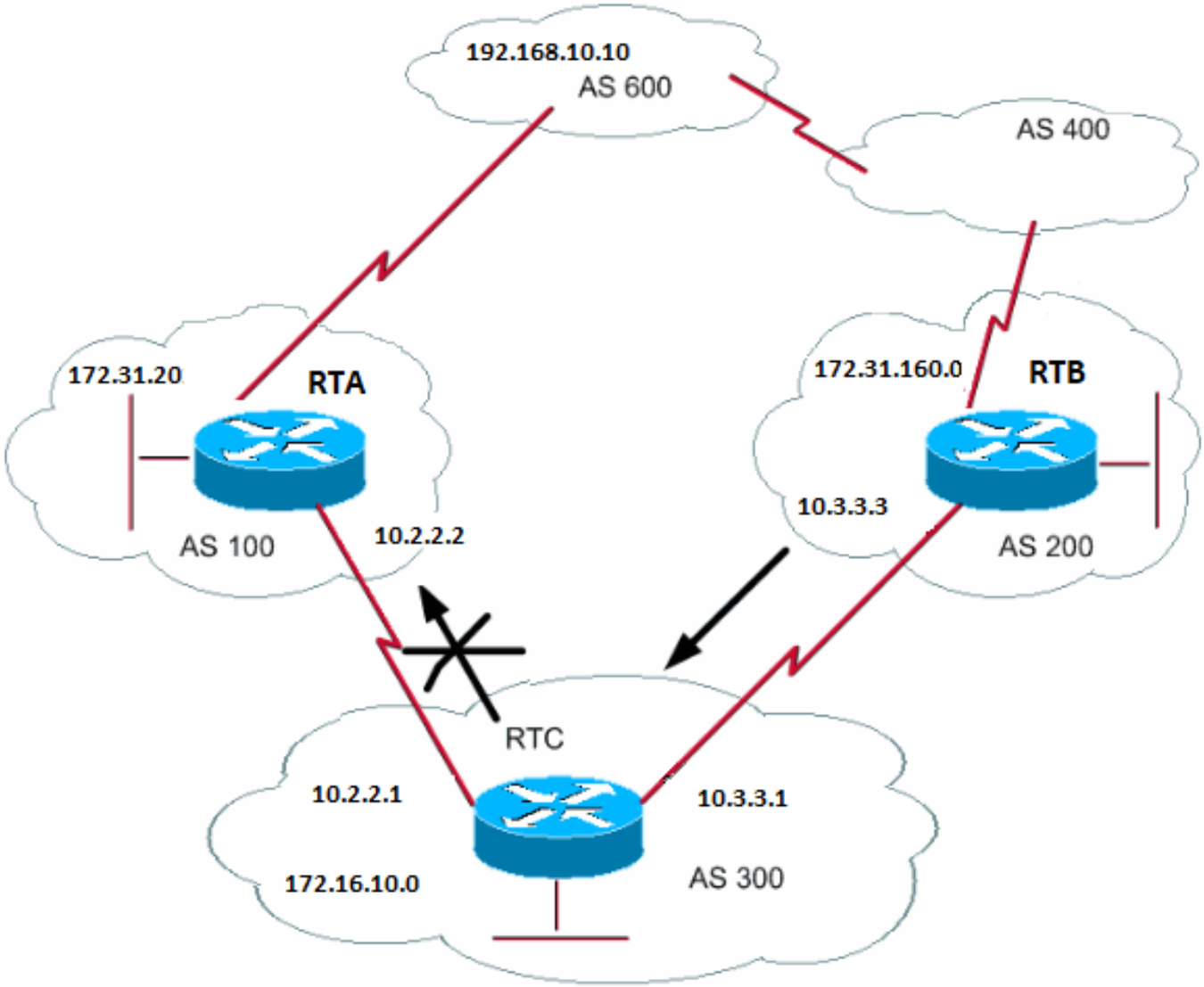
Sie können die Community-Liste verwenden, um bestimmte Parameter wie Gewicht und Metrik in bestimmten Updates mit dem Community-Wert als Grundlage zu filtern oder festzulegen. Im zweiten Beispiel in diesem Abschnitt hat RTB Updates an RTC mit einer Community von 100 200 gesendet. Wenn RTC die Gewichtung anhand dieser Werte festlegen möchte, können Sie Folgendes tun:

```
RTC#
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.3.3.3 route-map check-community in

route-map check-community permit 10
 match community 1
 set weight 20

route-map check-community permit 20
 match community 2 exact
 set weight 10

route-map check-community permit 30
 match community 3

ip community-list 1 permit 100
ip community-list 2 permit 200
ip community-list 3 permit internet
```

In diesem Beispiel stimmt jede Route mit 100 im Community-Attribut mit Liste 1 überein. Die Gewichtung dieser Route wird auf 20 festgelegt. Jede Route, die nur 200 als Community hat, stimmt mit Liste 2 überein und hat ein Gewicht von 20. Das Schlüsselwort **genau** besagt, dass die Gemeinschaft nur aus 200 besteht und nichts anderes. Die letzte Community-Liste soll sicherstellen, dass keine anderen Updates verloren gehen. Beachten Sie, dass alle nicht übereinstimmenden Elemente automatisch verworfen werden. Das Schlüsselwort **internet** gibt alle Routen an, da alle Routen Mitglieder der Internet-Community sind.

Weitere Informationen finden Sie unter Konfigurieren und Steuern eines Upstream-Anbieternetzwerks mit BGP-Community-Werten.

BGP-Nachbarn und Routenzuordnungen



Sie können den Befehl **neighbor** in Verbindung mit Routenübersichten verwenden, um Parameter für eingehende und ausgehende Updates zu filtern oder festzulegen.

Mit der **Neighbor**-Anweisung verknüpfte Routenzuordnungen haben keine Auswirkungen auf eingehende Updates, wenn Sie anhand der IP-Adresse eine Übereinstimmung herstellen:

<#root>

neighbor <ip-address> route-map <route-map-name>

Angenommen, in dem Diagramm in diesem Abschnitt soll RTC von AS200 etwas über Netzwerke lernen, die lokal für AS200 sind, und nichts anderes. Außerdem möchten Sie die Gewichtung der akzeptierten Routen auf 20 festlegen. Verwenden Sie eine Kombination aus **Nachbar**- und **As-Path**-Zugriffslisten:

```
RTC#
router bgp 300
 network 172.16.10.0
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.3.3.3 route-map stamp in

route-map stamp
 match as-path 1
 set weight 20

ip as-path access-list 1 permit ^200$
```

Alle von AS200 stammenden Updates enthalten Pfadinformationen, die mit 200 beginnen und mit 200 enden. Diese Aktualisierungen sind zulässig. Alle anderen Updates werden gelöscht.

Angenommen, Sie möchten:

- Akzeptanz von Updates, die von AS200 stammen und ein Gewicht von 20 aufweisen
- Die Verwerfung von Updates, die von AS400 stammen
- Ein Gewicht von 10 für andere Updates

```
RTC#
router bgp 300
 network 172.16.10.0
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.3.3.3 route-map stamp in

route-map stamp permit 10
 match as-path 1
 set weight 20

route-map stamp permit 20
 match as-path 2
```

```
 set weight 10

ip as-path access-list 1 permit ^200$
ip as-path access-list 2 permit ^200 600 .*
```

Mit dieser Anweisung wird eine Gewichtung von 20 für Updates festgelegt, die lokal für AS200 sind. Die Anweisung legt außerdem ein Gewicht von 10 für Updates fest, die sich hinter AS400 befinden, und verwirft Updates, die von AS400 stammen.

Verwenden des Befehls "Als Pfad festlegen"

In einigen Fällen müssen Sie die Pfadinformationen ändern, um den BGP-Entscheidungsprozess zu manipulieren. Der Befehl, den Sie mit einer Routenübersicht verwenden, lautet:

<#root>

set as-path prepend <as-path#> <as-path#>

Angenommen, RTC kündigt im Diagramm im Abschnitt "BGP Neighbors and Route Maps" sein eigenes Netzwerk 172.16.10.0 zwei verschiedenen ASs, AS100 und AS200, an. Wenn die Informationen an AS600 weitergeleitet werden, verfügen die Router in AS600 über Informationen zur Netzwerkerreichbarkeit über 172.16.10.0. Diese Informationen werden über zwei verschiedene Routen bereitgestellt. Die erste Route verläuft über AS100 mit Pfad (100, 300), die zweite über AS400 mit Pfad (400, 200, 300). Wenn alle anderen Attribute identisch sind, wählt AS600 den kürzesten Pfad und die Route über AS100 aus.

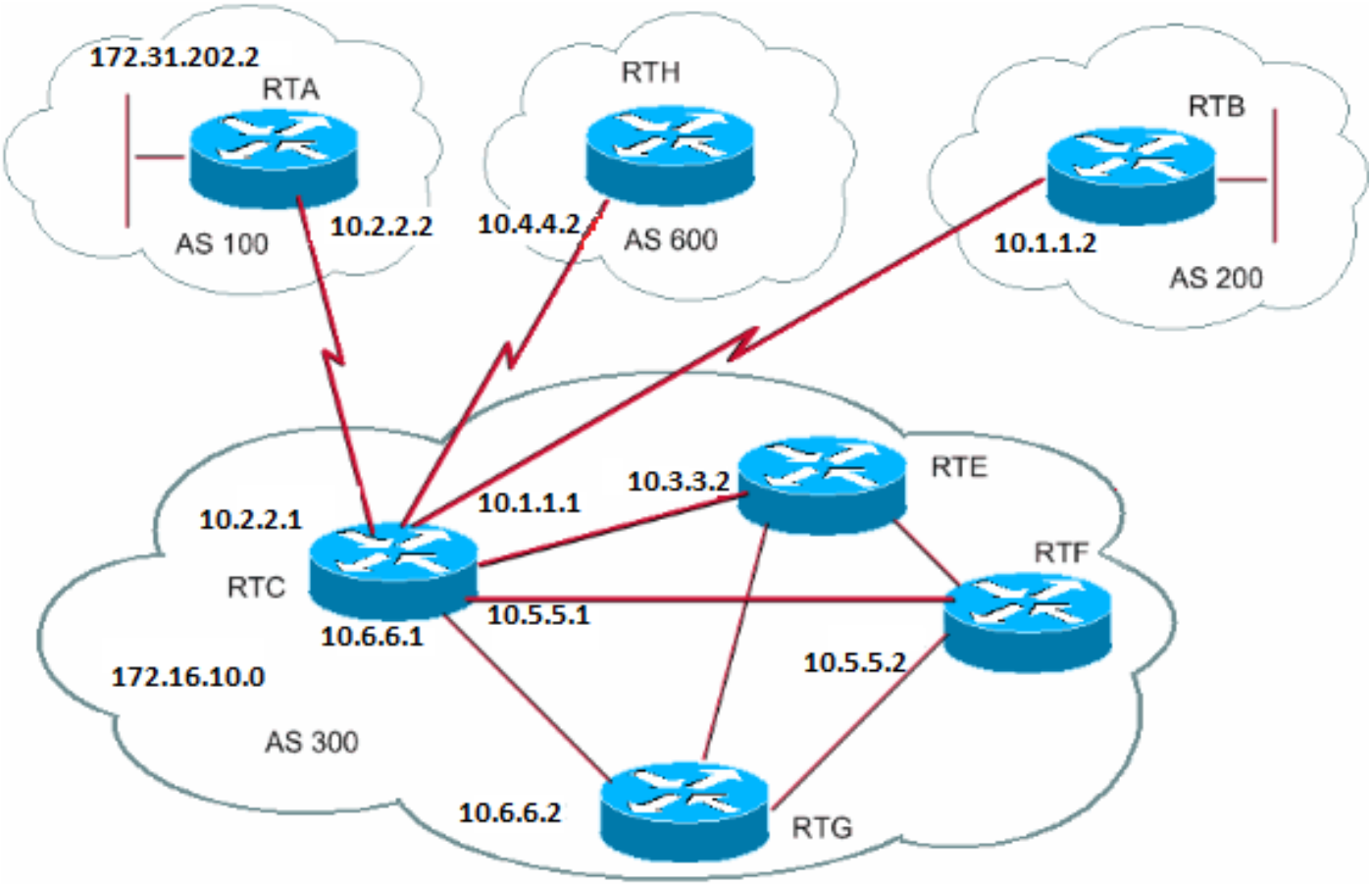
AS300 empfängt den gesamten Datenverkehr über AS100. Wenn Sie diese Entscheidung vom AS300-Ende aus beeinflussen möchten, können Sie festlegen, dass der Pfad durch AS100 länger ist als der Pfad durch AS400. Dies ist möglich, wenn Sie AS-Nummern den aktuellen Pfadinformationen voranstellen, die AS100 angekündigt werden. Es ist gängige Praxis, Ihre eigene AS-Nummer auf diese Weise zu wiederholen:

```
RTC#
router bgp 300
 network 172.16.10.0
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 route-map SETPATH out
```

```
route-map SETPATH
 set as-path prepend 300 300
```

Aufgrund dieser Konfiguration erhält AS600 Updates um 172.16.10.0 über AS100 mit Pfadinformationen von: (100, 300, 300, 300). Diese Pfadinformationen sind länger als die vom AS400 empfangenen (400, 200, 300).

BGP-Peer-Gruppen



Eine BGP-Peer-Gruppe ist eine Gruppe von BGP-Nachbarn mit denselben Update-Richtlinien. Routingzuordnungen, Verteilungs- und Filterlisten legen in der Regel Aktualisierungsrichtlinien fest. Sie definieren nicht für jeden einzelnen Nachbarn die gleichen Richtlinien, sondern definieren stattdessen einen Namen für die Peergruppe und weisen diese Richtlinien der Peergruppe zu.

Mitglieder der Peer-Gruppe übernehmen alle Konfigurationsoptionen der Peer-Gruppe. Sie können Mitglieder auch so konfigurieren, dass diese Optionen überschrieben werden, wenn die Optionen keine Auswirkungen auf ausgehende Updates haben. Sie können nur Optionen überschreiben, die für den eingehenden Datenverkehr festgelegt wurden.

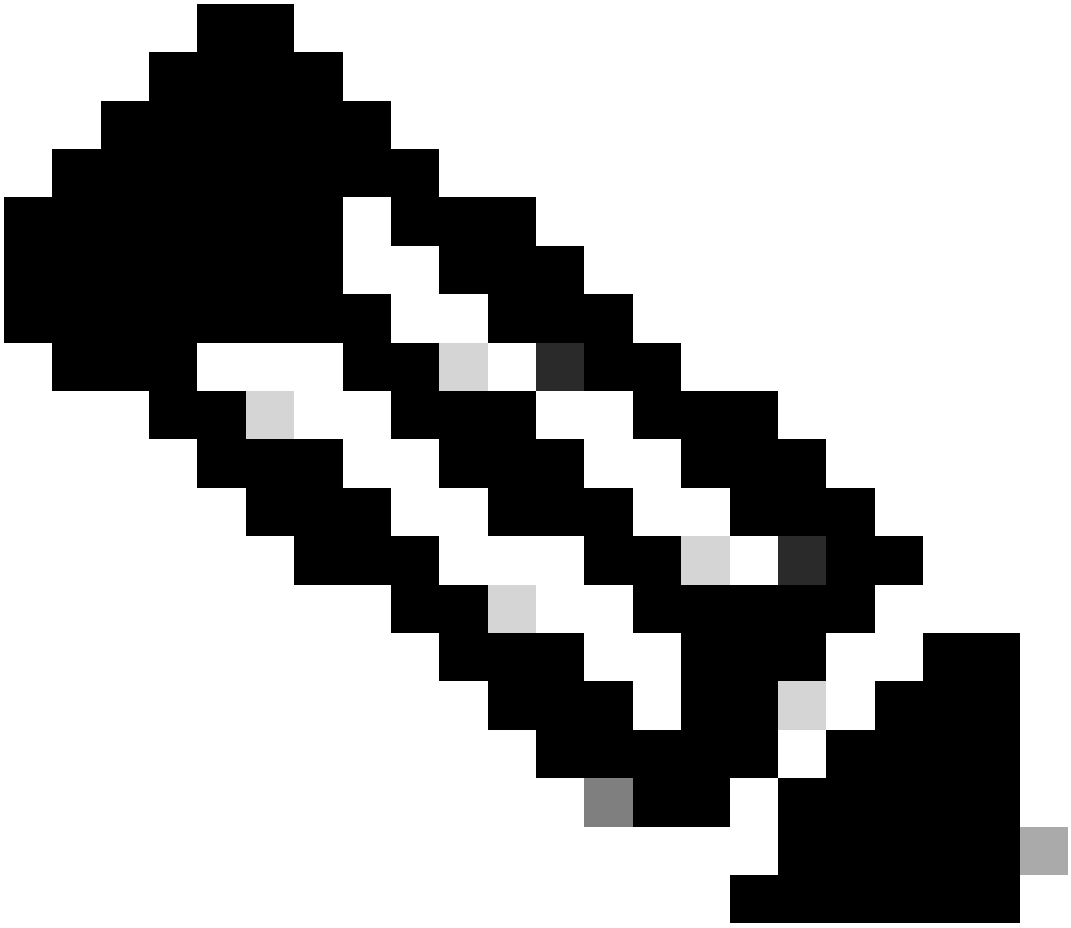
Führen Sie den folgenden Befehl aus, um eine Peer-Gruppe zu definieren:

neighbor peer-group-name peer-group

In diesem Beispiel werden Peer-Gruppen auf interne und externe BGP-Nachbarn angewendet:

```
RTC#
router bgp 300
 neighbor internalmap peer-group
 neighbor internalmap remote-as 300
 neighbor internalmap route-map SETMETRIC out
 neighbor internalmap filter-list 1 out
 neighbor internalmap filter-list 2 in
 neighbor 10.5.5.2 peer-group internalmap
 neighbor 10.6.6.2 peer-group internalmap
 neighbor 10.3.3.2 peer-group internalmap
 neighbor 10.3.3.2 filter-list 3 in
```

Diese Konfiguration definiert eine Peer-Gruppe mit dem Namen **internalmap**. In der Konfiguration werden einige Richtlinien für die Gruppe definiert, z. B. eine Routenübersicht **SETMETRIC** zum Festlegen der Metrik auf 5 und zwei verschiedene Filterlisten, 1 und 2. Bei der Konfiguration wird die Peer-Gruppe auf alle internen Nachbarn, RTE, RTF und RTG, angewendet. Außerdem wird in der Konfiguration eine separate Filterliste 3 für die benachbarte RTE definiert. Diese Filterliste überschreibt die Filterliste 2 innerhalb der Peer-Gruppe.

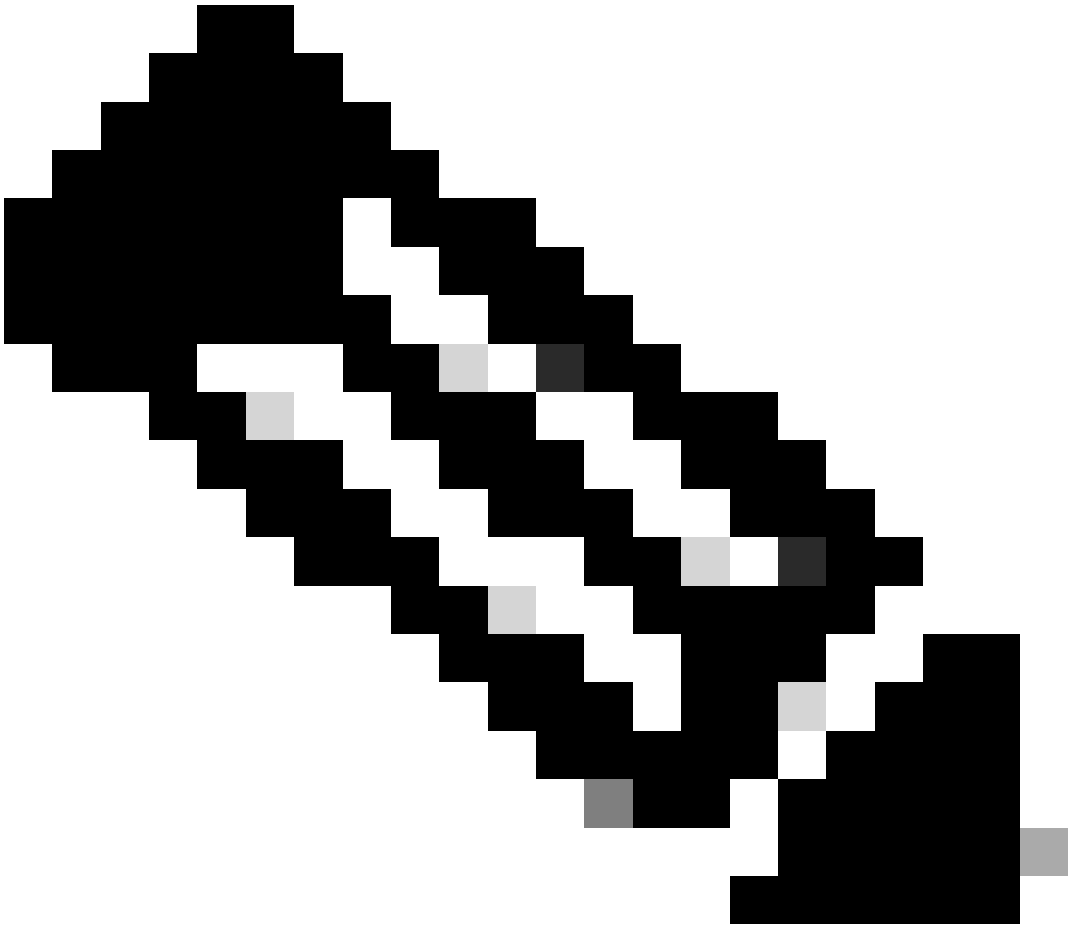


Hinweis: Sie können nur Optionen außer Kraft setzen, die sich auf eingehende Updates auswirken.

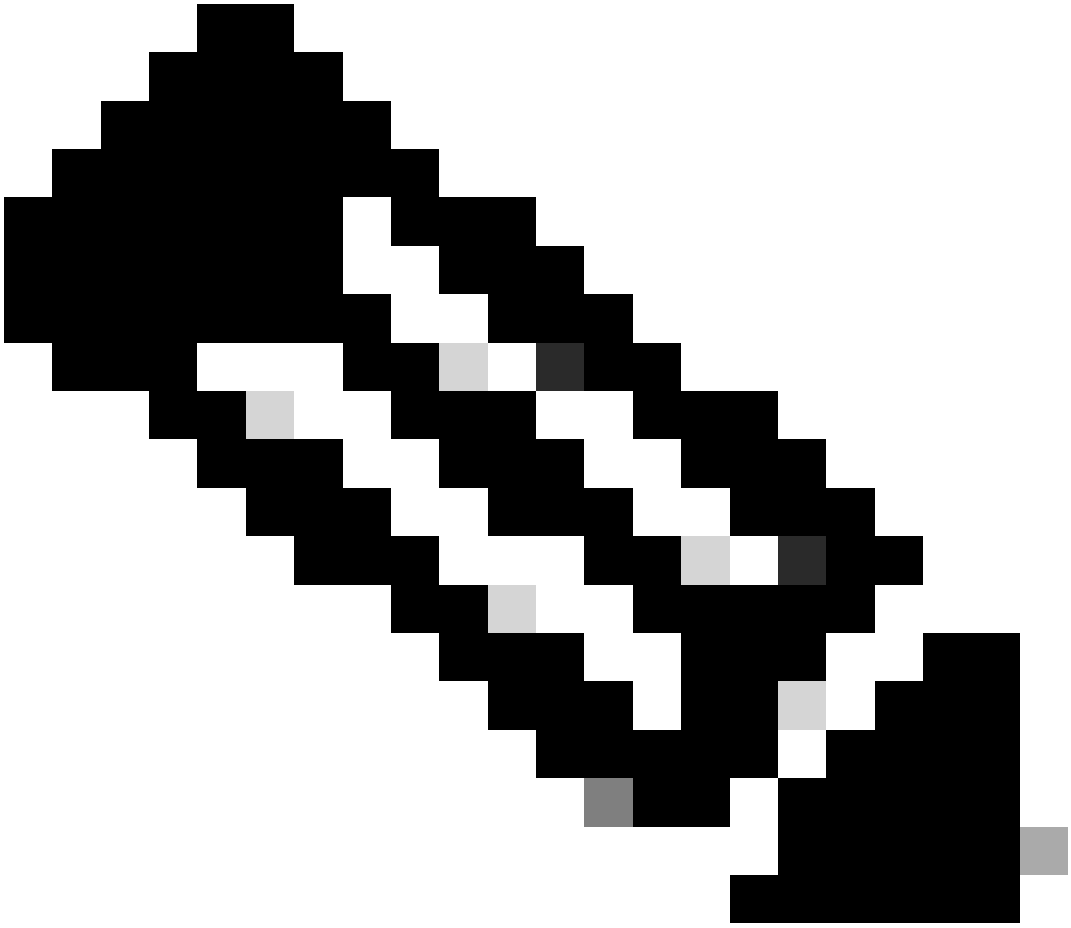
Sehen Sie sich nun an, wie Sie Peer-Gruppen mit externen Nachbarn verwenden können. Mit dem gleichen Diagramm in diesem Abschnitt konfigurieren Sie RTC mit einer externen **Zuordnung der** Peergruppe und wenden die Peergruppe auf externe Nachbarn an.

```
RTC#
router bgp 300
 neighbor externalmap peer-group
 neighbor externalmap route-map SETMETRIC
 neighbor externalmap filter-list 1 out
 neighbor externalmap filter-list 2 in
 neighbor 10.2.2.2 remote-as 100
```

```
neighbor 10.2.2.2 peer-group externalmap
neighbor 10.4.4.2 remote-as 600
neighbor 10.4.4.2 peer-group externalmap
neighbor 10.1.1.2 remote-as 200
neighbor 10.1.1.2 peer-group externalmap
neighbor 10.1.1.2 filter-list 3 in
```



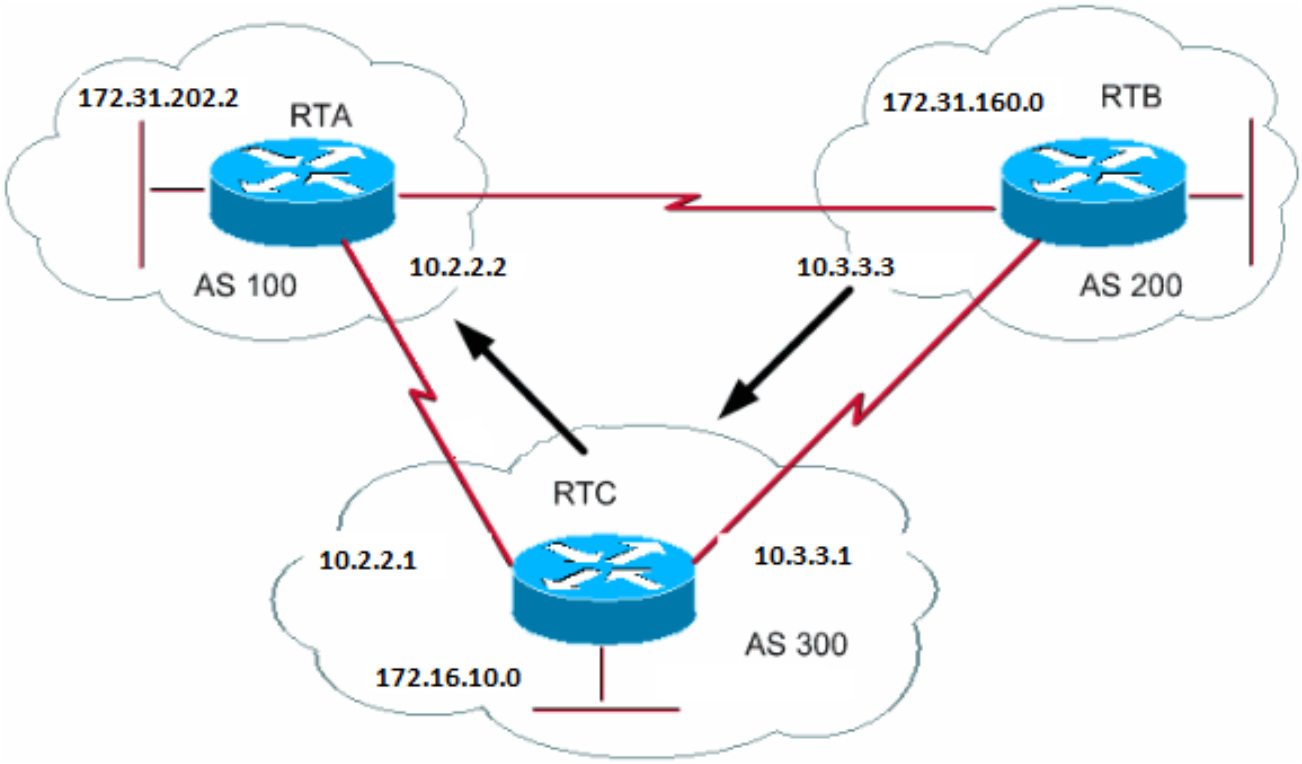
Hinweis: In diesen Konfigurationen definieren Sie die Remote-as-Anweisungen außerhalb der Peer-Gruppe, da Sie unterschiedliche externe ASs definieren müssen. Außerdem überschreiben Sie die eingehenden Updates von Nachbar 10.1.1.2 mit der Zuweisung von Filterliste 3. Weitere Informationen zu Peer-Gruppen finden Sie unter BGP Peer Groups.



Hinweis: In Version 12.0(24)S der Cisco IOS-Software hat Cisco die Funktion "BGP Dynamic Update Peer Groups" eingeführt. Diese Funktion steht auch in späteren Versionen der Cisco IOS Software zur Verfügung. Mit dieser Funktion wird ein neuer Algorithmus eingeführt, der Update-Gruppen von Nachbarn, die dieselben Richtlinien für ausgehenden Datenverkehr nutzen, dynamisch berechnet und optimiert. Diese Nachbarn können dieselben Aktualisierungsnachrichten gemeinsam nutzen. In früheren Versionen der Cisco IOS-Software basierte die Gruppe der BGP-Update-Nachrichten auf Peer-Group-Konfigurationen. Mit dieser Methode werden Richtlinien für ausgehenden Datenverkehr und bestimmte Sitzungskonfigurationen aktualisiert. Die Funktion "BGP Dynamic Update Peer Group" (Dynamische BGP-Update-Peer-Gruppe) trennt die Update-Gruppenreplikation von der Konfiguration der Peer-Gruppe. Diese Trennung verbessert die Konvergenzzeit und die Flexibilität der Nachbarkonfiguration. Weitere Informationen finden Sie unter BGP Dynamic Update Peer-Groups (Peer-Gruppen für dynamische BGP-Updates).

BGP-Anwenderberichte 4

CIDR- und Aggregatadressen



Eine der wichtigsten Verbesserungen von BGP4 gegenüber BGP3 ist das klassenlose Interdomain Routing (CIDR). CIDR oder Supernetting ist eine neue Art, IP-Adressen zu betrachten. Bei CIDR gibt es keine Vorstellung von Klassen wie Klasse A, B oder C. Das Netzwerk 192.168.213.0 war beispielsweise früher ein illegales Netzwerk der Klasse C. Nun, das Netzwerk ist ein legales Supernet, 192.168.213.0/16. Die 16 steht für die Anzahl der Bit in der Subnetzmaske, wenn Sie die IP-Adresse ganz links zählen. Diese Darstellung ist ähnlich wie 192.168.213.0 255.255.0.0.

Sie verwenden Aggregate, um die Größe der Routing-Tabellen zu minimieren. Bei der Aggregation werden die Merkmale mehrerer verschiedener Routen so kombiniert, dass eine Ankündigung einer einzelnen Route möglich ist. In diesem Beispiel generiert die RTB das Netzwerk 172.31.160.0. Sie konfigurieren RTC so, dass ein Supernet dieser Route 192.168.160.0 an RTA propagiert wird:

```
RTB#
router bgp 200
 neighbor 10.3.3.1 remote-as 300
 network 172.31.160.0

#RTC
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 network 172.16.10.0
 aggregate-address 192.168.160.0 255.0.0.0
```

RTC übermittelt die Sammeladresse 192.168.160.0 an RTA.

Aggregatbefehle

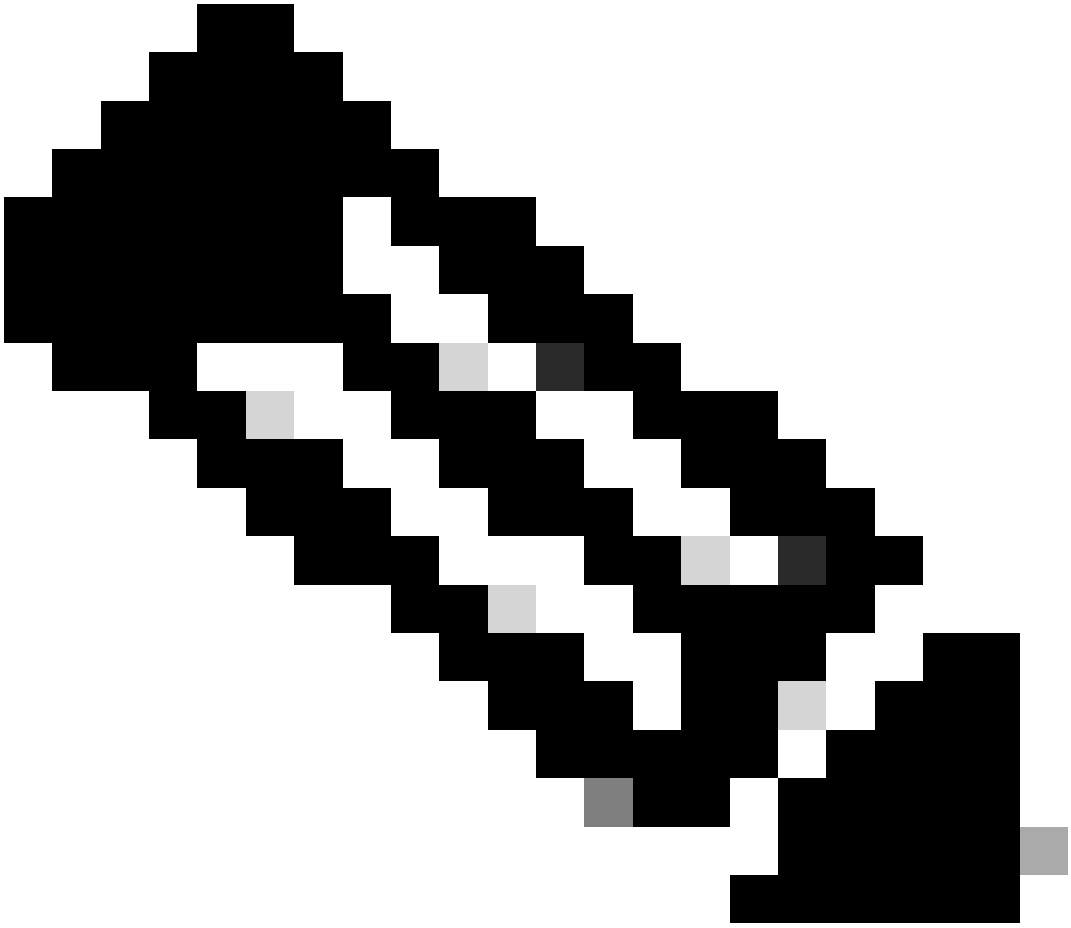
Es gibt eine Vielzahl von Aggregatbefehlen. Sie müssen wissen, wie jede einzelne funktioniert, um das gewünschte Aggregationsverhalten zu erzielen.

Der erste Befehl stammt aus dem Beispiel im Abschnitt CIDR und Aggregate Addresses:

<#root>

aggregate-address address-mask

Dieser Befehl kündigt die Präfixroute und alle spezifischeren Routen an. Der Befehl **aggregate-address 192.168.160.0** propagiert ein zusätzliches Netzwerk 192.168.160.0, verhindert jedoch nicht die Weitergabe von 172.31.160.0 an RTA. Das Ergebnis ist die Übertragung der beiden Netzwerke 192.168.160.0 und 172.31.160.0 an die RTA. Dabei handelt es sich um die Meldung des Präfix und der spezifischeren Route.



Hinweis: Sie können keine Adressen aggregieren, wenn diese in der BGP-Routing-Tabelle keine spezifischere Route aufweisen.

RTB kann beispielsweise kein Aggregat für 192.168.160.0 generieren, wenn RTB keinen spezifischeren Eintrag 192.168.160.0 in der BGP-Tabelle hat. Eine Einschleusung der spezifischeren Route in die BGP-Tabelle ist möglich. Die Injektion kann erfolgen über:

- Eingehende Updates von anderen ASs
-

Umverteilung eines IGP oder statischen IGPs in BGP

-

Der Befehl **network**, z. B. **network 172.31.160.0**

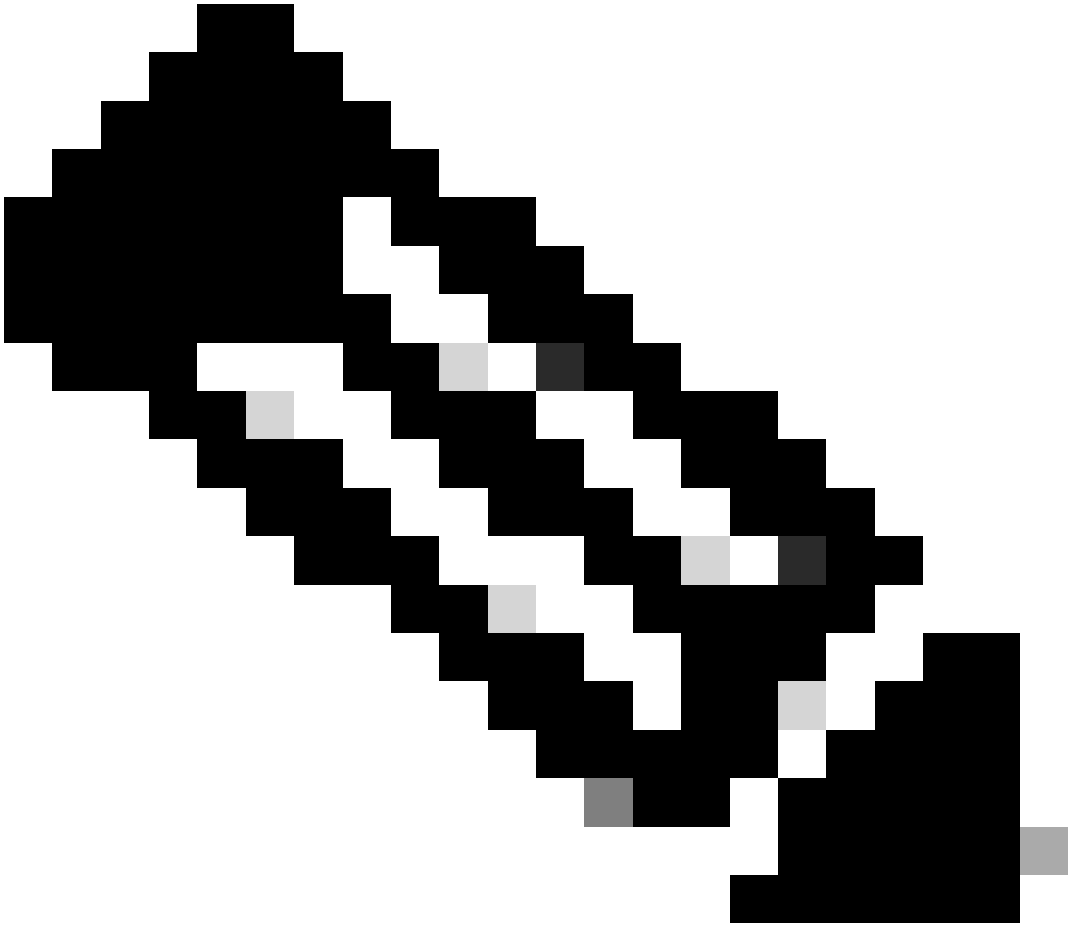
Wenn RTC nur das Netzwerk 192.168.160.0 propagieren soll und nicht die spezifischere Route, geben Sie den folgenden Befehl ein:

<#root>

aggregate-address <address> <mask> summary-only

Dieser Befehl gibt nur das Präfix an. Der Befehl unterdrückt alle spezifischeren Routen.

Der Befehl "**aggregate 192.168.160.0 255.0.0.0 summary-only"** propagiert das Netzwerk 192.168.160.0 und unterdrückt die spezifischere Route 172.31.160.0.



Hinweis: Wenn Sie ein Netzwerk aggregieren, das über die Netzwerk-Anweisung in das BGP eingespeist wurde, fügt der Netzwerkeintrag immer BGP-Updates hinzu. Diese Injektion erfolgt auch dann, wenn Sie den Befehl aggregate summary-only verwenden. Das Beispiel im Abschnitt CIDR. Beispiel 1 behandelt diese Situation.

<#root>

aggregate-address <address> <mask> as-set

Dieser Befehl kündigt das Präfix und die spezifischeren Routen an. Der Befehl enthält jedoch **als festgelegt** Informationen in den Pfadinformationen der Routing-Updates.

<#root>

aggregate 192.168.0.0 255.0.0.0 as-set

Im Abschnitt CIDR Beispiel 2 (as-set) wird dieser Befehl erläutert.

Wenn Sie bei der Aggregation spezifischere Routen unterdrücken möchten, definieren Sie eine Routenübersicht, und wenden Sie die Routenübersicht auf die Aggregate an. Mit dieser Aktion können Sie auswählen, welche spezifischeren Routen unterdrückt werden sollen.

<#root>

aggregate-address <address> <mask> suppress-map <map-name>

Dieser Befehl kündigt das Präfix und die spezifischeren Routen an. Der Befehl unterdrückt jedoch die Ankündigung auf Routing-Map-Basis. Angenommen, Sie möchten mit dem Diagramm im Abschnitt CIDR und Aggregate-Adressen 192.168.160.0 aggregieren, die spezifischere Route 192.168.160.20 unterdrücken und die Übertragung von 172.31.160 zulassen. 0. Diese Routenübersicht verwenden:

```
route-map CHECK permit 10
 match ip address 1

access-list 1 permit 192.168.160.20 0.0.255.255
access-list 1 deny 0.0.0.0 255.255.255.255
```

Durch Definition von **suppress-map** werden alle Pakete, die die Zugriffsliste zulässt, bei der Aktualisierung unterdrückt.

Wenden Sie dann die Routenübersicht auf die **aggregierte** Anweisung an.

```
RTC#
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 remote-as 100
 network 172.16.10.0
 aggregate-address 192.168.160.0 255.0.0.0 suppress-map CHECK
```

Hier ist eine weitere Variante:

<#root>

aggregate-address <address> <mask> attribute-map <map-name>

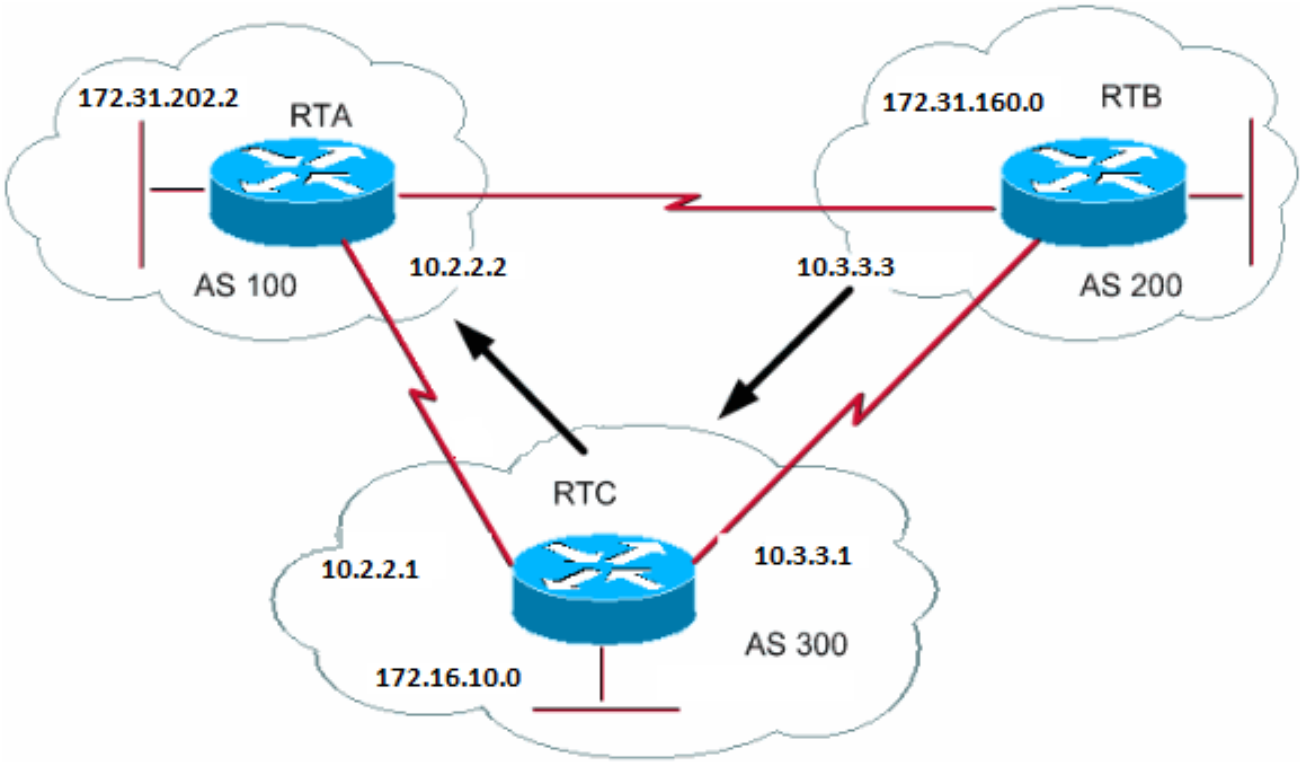
Mit diesem Befehl können Sie die Attribute (z. B. Metrik) zum Zeitpunkt des Sendens der Aggregate festlegen. Um den Ursprung der Aggregate auf IGP festzulegen, wenden Sie diese Routenzuordnung auf den Befehl **aggregate attribute-map** an:

```
route-map SETMETRIC
 set origin igp
```

```
aggregate-address 192.168.160.0 255.0.0.0 attribute-map SETORIGIN
```

Weitere Informationen finden Sie unter Understanding Route Aggregation in BGP.

CIDR Beispiel 1



Anforderung: RTB darf das Präfix 192.168.160.0 ankündigen und alle spezifischeren Routen unterdrücken. Das Problem mit dieser Anforderung ist, dass das Netzwerk 172.31.160.0 lokal für AS200 ist, was bedeutet, dass AS200 der Urheber von 172.31.160.0 ist. Sie können nicht zulassen, dass RTB ein Präfix für 192.168.160.0 generiert, ohne dass ein Eintrag für 172.31.160.0 generiert wird, selbst wenn Sie den Befehl **aggregate summary-only** verwenden. RTB generiert beide Netzwerke, da RTB der Urheber von 172.31.160.0 ist. Es gibt zwei Lösungen für dieses Problem.

Die erste Lösung besteht in der Verwendung einer statischen Route und der Neuverteilung über das BGP. Das Ergebnis ist, dass RTB das Aggregat mit einem Ursprung von unvollständig (?).

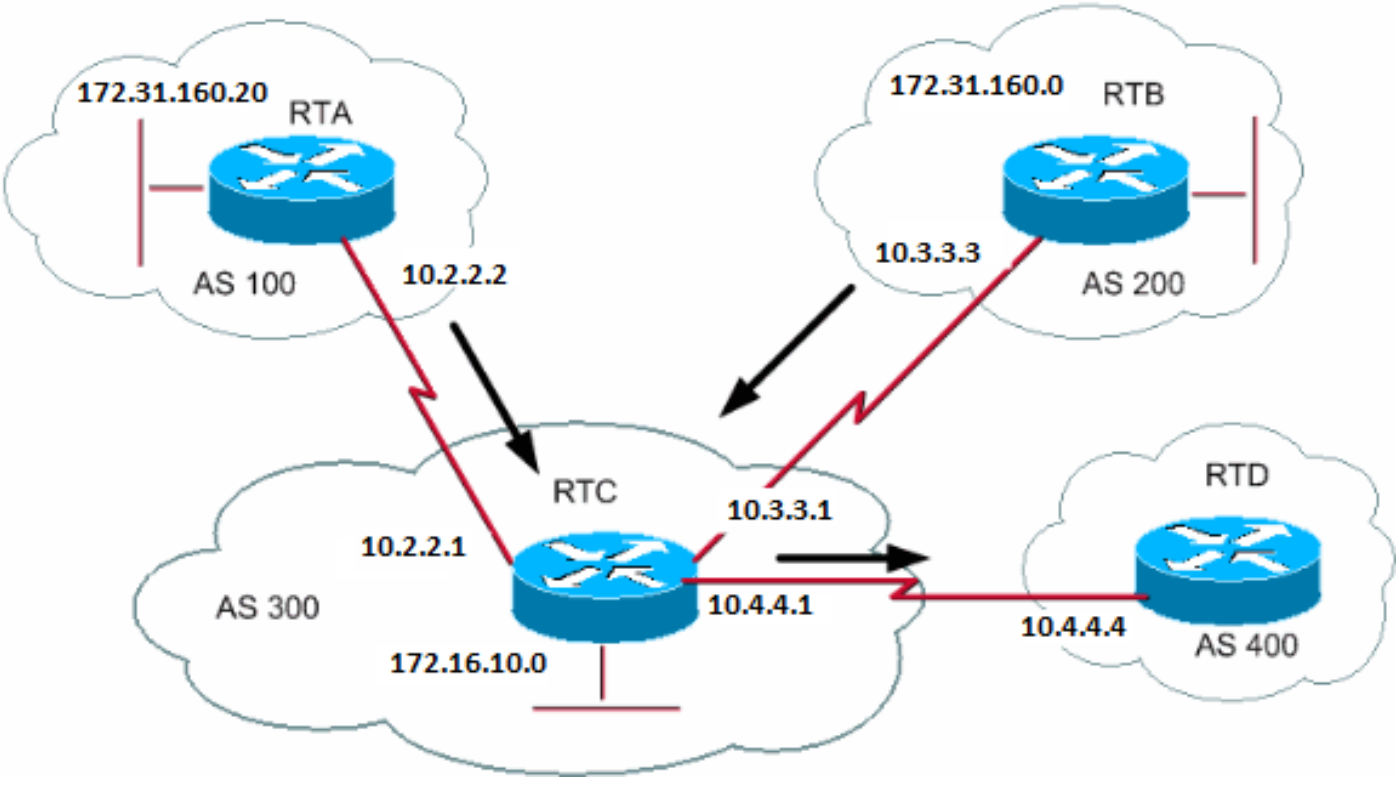
```
RTB#
router bgp 200
 neighbor 10.3.3.1 remote-as 300
 redistribute static

!--- This generates an update for 192.168.160.0 !--- with the origin path as "incomplete".

ip route 192.168.160.0 255.0.0.0 null0
```

Bei der zweiten Lösung fügen Sie zusätzlich zur statischen Route einen Eintrag für den Befehl **network** hinzu. Dieser Eintrag hat den gleichen Effekt, mit der Ausnahme, dass er den Ursprung des Updates auf IGP festlegt.

```
RTB#
router bgp 200
 network 192.168.160.0 mask 255.0.0.0

!--- This entry marks the update with origin IGP.

 neighbor 10.3.3.1 remote-as 300
 redistribute static

ip route 192.168.160.0 255.0.0.0 null0
```

CIDR Beispiel 2 (as-set)

Sie verwenden die Anweisung **wie** in der Aggregation **festgelegt**, um die Größe der Pfadinformationen zu reduzieren. Bei "**as-set**" wird die AS-Nummer nur einmal aufgeführt, unabhängig davon, wie oft die AS-Nummer in mehreren aggregierten Pfaden auftauchte. Der Befehl **aggregate-as-set** wird in Situationen verwendet, in denen die Aggregation von Informationen einen Informationsverlust in Bezug auf das path-Attribut verursacht. In diesem Beispiel bezieht RTC Updates von RTA über 192.168.160.20 und Updates über 172.31.160.0 von RTB. Angenommen, RTC aggregiert das Netzwerk 192.168.160.0/8 und sendet das Netzwerk an RTD. Die FTE kennt den Ursprung dieser Strecke nicht. Wenn Sie die **Aggregat-as-set-**Anweisung hinzufügen, erzwingen Sie RTC, Pfadinformationen in Form eines set{} zu generieren. Dieser Satz enthält alle Pfadinformationen, unabhängig davon, welcher Pfad zuerst angegeben wurde.




```
RTB#
```

```
router bgp 200
 network 172.31.160.0
 neighbor 10.3.3.1 remote-as 300

RTA#
router bgp 100
 network 192.168.160.20
 neighbor 10.2.2.1 remote-as 300
```

Fall 1:

RTC hat keine **As-Set**-Anweisung. RTC sendet ein Update 192.168.160.0/8 mit Pfadinformationen (300) an RTD, als ob die Route von AS300 stammt.

```
RTC#
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.4.4.4 remote-as 400
 aggregate 192.168.160.0 255.0.0.0 summary-only

!--- This command causes RTC to send RTD updates about 192.168.160.0/8
!--- with no indication that 192.168.160.0 actually comes from two different ASs.
!--- This may create loops if RTD has an entry back into AS100 or AS200.
```

Fall 2:

```
RTC#
router bgp 300
 neighbor 10.3.3.3 remote-as 200
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.4.4.4 remote-as 400
 aggregate 192.168.160.0 255.0.0.0 summary-only
 aggregate 192.168.160.0 255.0.0.0 as-set

!--- This command causes RTC to send RTD updates about 192.168.160.0/8
!--- with an indication that 192.168.160.0 belongs to a set {100 200}.
```

Die nächsten beiden Themen, BGP Confederation und Route Reflectors, richten sich an Internet Service Provider (ISPs), die eine weitere Kontrolle der explosionsartigen Zunahme von iBGP-Peering in ihren ASs anstreben.

BGP-Konföderation

Die Implementierung der BGP-Konföderation reduziert die iBGP-Mesh-Größe innerhalb eines AS. Der Trick besteht darin, ein AS in mehrere ASs zu unterteilen und die gesamte Gruppe einer einzelnen Konföderation zuzuweisen. Jedes AS allein verfügt über ein vollständiges iBGP-Mesh und hat Verbindungen zu anderen AS innerhalb des Verbunds. Obwohl diese ASs innerhalb der Konföderation über eBGP-Peers zu ASs verfügen, tauschen die ASs das Routing so aus, als würden sie iBGP verwenden. Auf diese Weise erhält der Verbund Informationen zu Next Hop, Metrik und lokalen Einstellungen aufrecht. Nach außen scheint der Verbund ein einzelnes AS zu sein.

Führen Sie den folgenden Befehl aus, um eine BGP-Konföderation zu konfigurieren:

```
<#root>
```

bgp confederation identifier <autonomous-system>

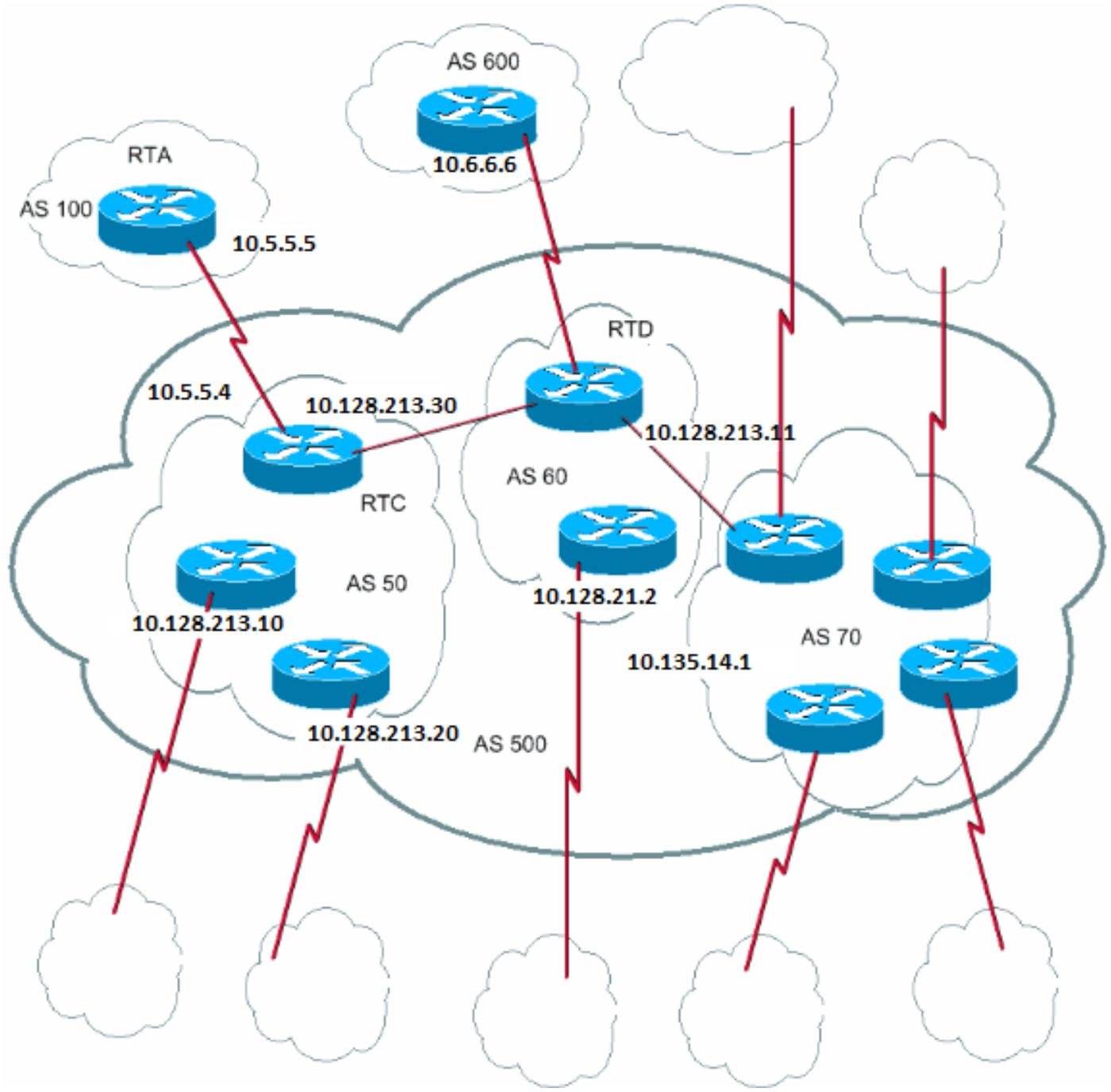
Der Confederation Identifier ist die AS-Nummer der Confederation Group.

Die Ausgabe dieses Befehls führt Peering zwischen mehreren ASs innerhalb der Konföderation aus:

```
<#root>
```

bgp confederation peers <autonomous-system> <autonomous-system>

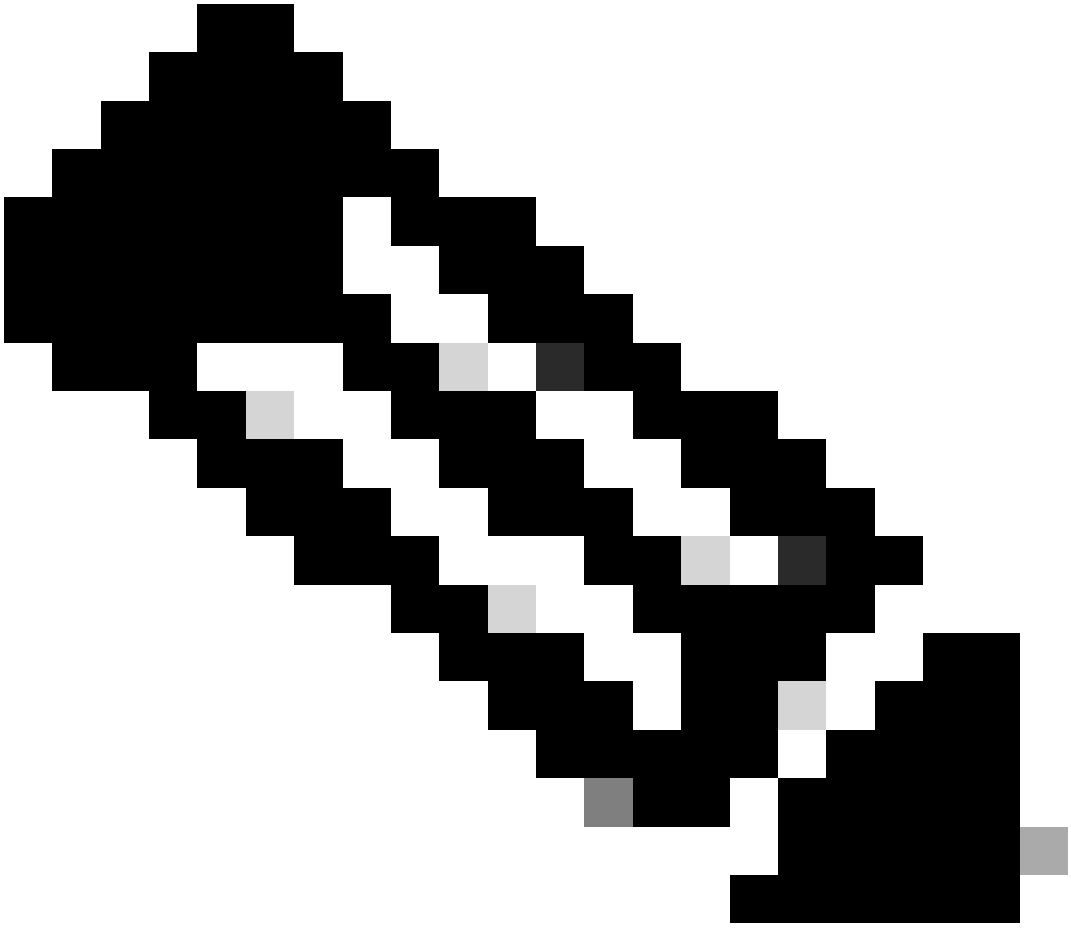
Hier ein Beispiel für einen Verbund:



Angenommen, Sie haben einen AS500, der aus neun BGP-Routern besteht. Es gibt auch andere Nicht-BGP-Router, aber Sie interessieren sich nur für die BGP-Router, die über eBGP-Verbindungen mit anderen ASs verfügen. Wenn Sie ein vollständiges iBGP-Mesh in AS500 erstellen möchten, benötigen Sie neun Peer-Verbindungen für jeden Router. Sie benötigen acht iBGP-Peers und einen eBGP-Peer zu externen AS.

Wenn Sie die Konföderation verwenden, können Sie AS500 in mehrere ASs unterteilen: AS50, AS60 und AS70. Sie geben dem AS eine Konföderationskennung von 500. Die Außenwelt sieht nur ein AS, AS500. Für AS50, AS60 und AS70 definieren Sie jeweils ein vollständiges Netz von iBGP-Peers und die Liste der Confederation Peers mit dem Befehl **bgp confederation peers**.

Nachfolgend finden Sie eine Beispielkonfiguration der Router RTC, RTD und RTA:



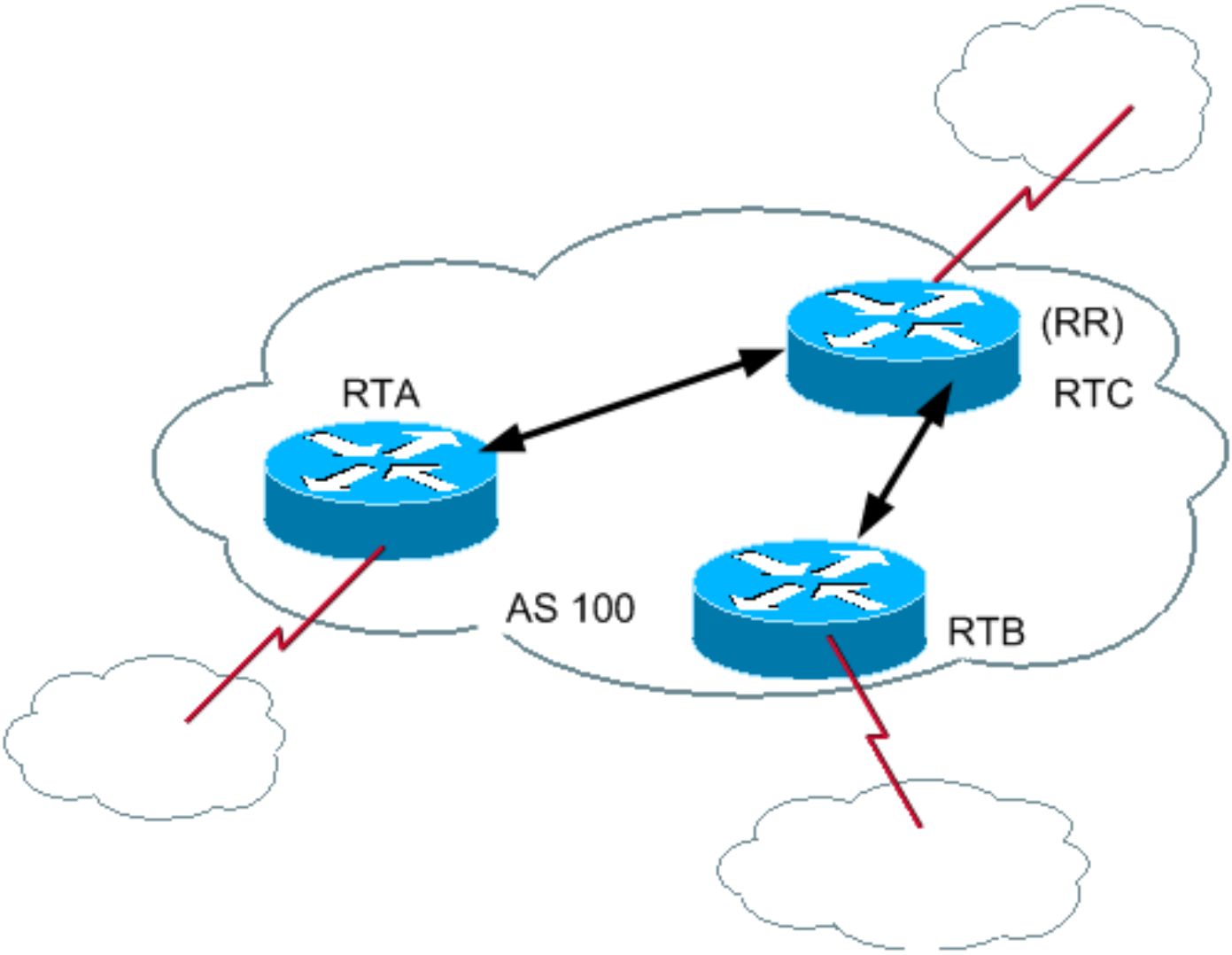
Hinweis: RTA hat keine Kenntnisse über AS50, AS60 oder AS70. Die RTA kennt nur den AS500.

```
RTC#
router bgp 50
 bgp confederation identifier 500
 bgp confederation peers 60 70
 neighbor 10.128.213.10 remote-as 50 (IBGP connection within AS50)
 neighbor 10.128.213.20 remote-as 50 (IBGP connection within AS50)
 neighbor 10.128.213.11 remote-as 60 (BGP connection with confederation peer 60)
 neighbor 10.128.213.14 remote-as 70 (BGP connection with confederation peer 70)
 neighbor 10.5.5.5 remote-as 100 (EBGP connection to external AS100)

RTD#
router bgp 60
```

```
 bgp confederation identifier 500
 bgp confederation peers 50 70
 neighbor 10.128.210.2 remote-as 60 (IBGP connection within AS60)
 neighbor 10.128.213.30 remote-as 50(BGP connection with confederation peer 50)
 neighbor 10.128.213.14 remote-as 70 (BGP connection with confederation peer 70)
 neighbor 10.6.6.16 remote-as 600 (EBGP connection to external AS600)

RTA#
router bgp 100
 neighbor 10.5.5.4 remote-as 500 (EBGP connection to confederation 500)
```

Routen-Reflektoren

Eine weitere Lösung für die Explosion von iBGP-Peering in einem AS sind Routen-Reflektoren (RRs). Wie der iBGP-Abschnitt zeigt, kündigt ein BGP-Sprecher keine Route an, die der BGP-Sprecher über einen anderen iBGP-Sprecher zu einem dritten iBGP-Sprecher gelernt hat. Sie können diese Einschränkung etwas lockern und zusätzliche Kontrolle bereitstellen, die es einem Router ermöglicht, vom iBGP bezogene Routen anderen iBGP-Routern anzukündigen oder diese zu reflektieren. Diese Routen-Reflektion verringert die Anzahl der iBGP-Peers innerhalb eines AS.



In normalen Fällen muss eine vollständige iBGP-Vermaschung zwischen RTA, RTB und RTC innerhalb von AS100 aufrechterhalten werden. Wenn Sie das RR-Konzept nutzen, kann RTC als RR gewählt werden. Auf diese Weise verfügt RTC über ein partielles iBGP-Peering mit RTA

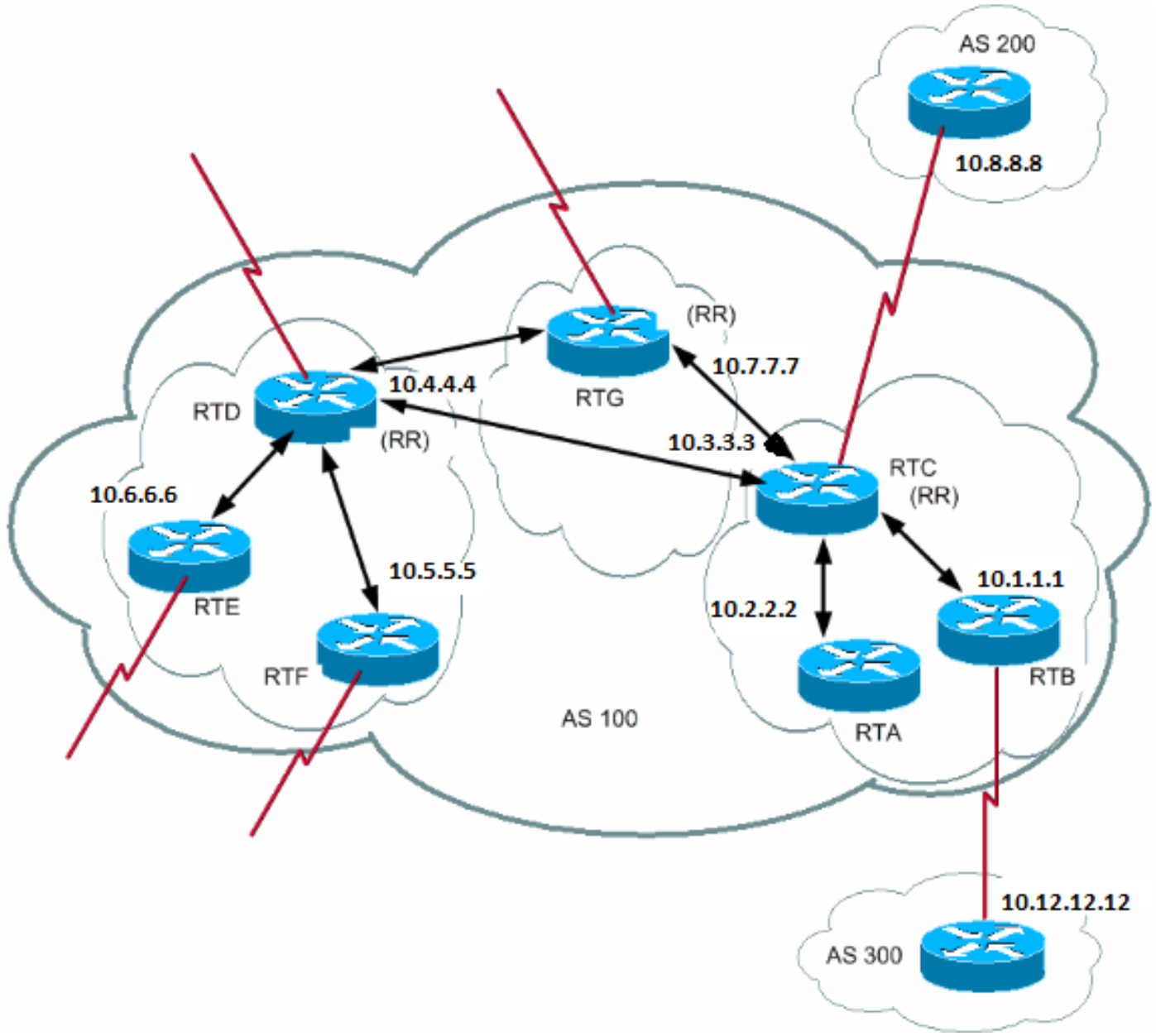
und RTB. Peering zwischen RTA und RTB ist nicht erforderlich, da RTC ein RR für die Updates von RTA und RTB ist.

<#root>

neighbor <ip address> route-reflector-client

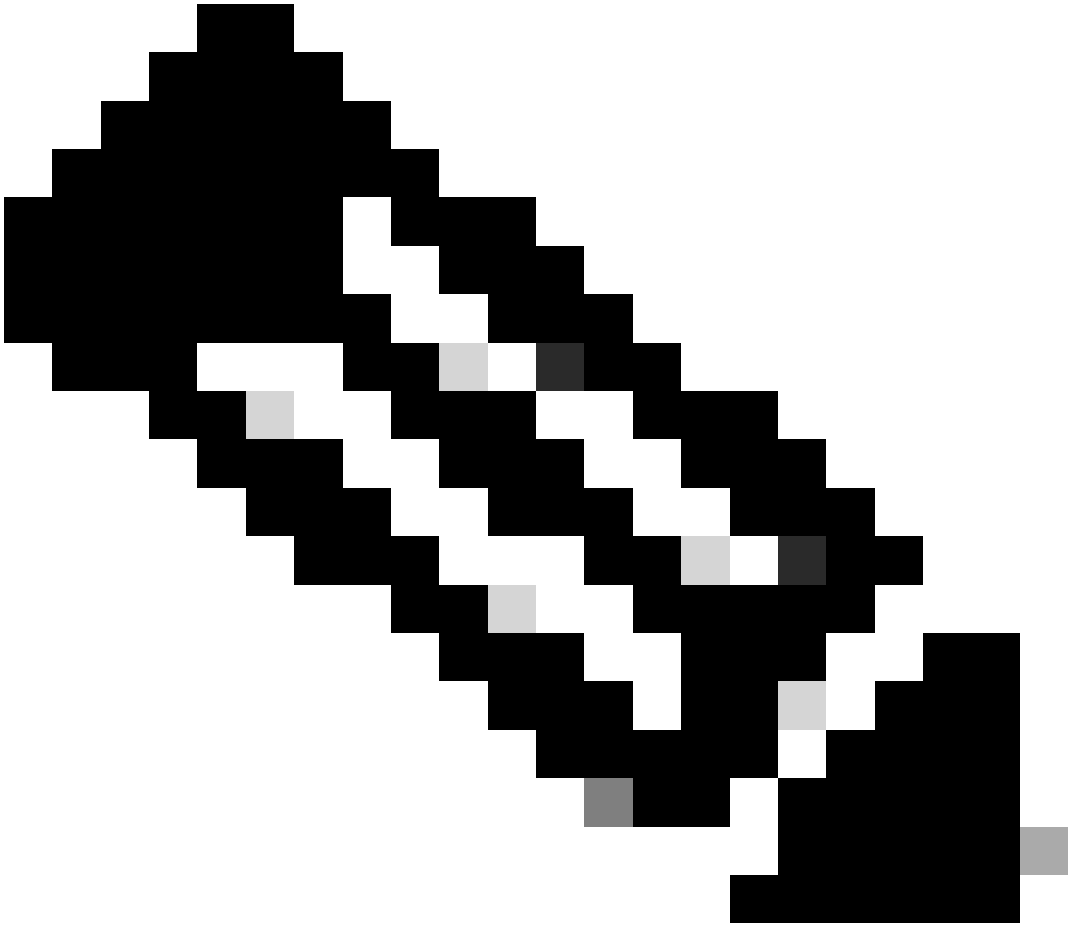
Der Router mit diesem Befehl ist der RR und die Nachbarn, bei denen die Befehlspunkte die Clients dieses RR sind. Im Beispiel enthält die RTC-Konfiguration den Befehl **neighbor route-reflektor-client,** der auf die RTA- und RTB-IP-Adressen zeigt. Die Kombination aus RR und Clients ist ein "Cluster". In diesem Beispiel bilden RTA, RTB und RTC einen Cluster mit einem einzelnen RR innerhalb von AS100.

Andere iBGP-Peers des RR, die keine Clients sind, sind Nicht-Clients.



Ein AS kann mehr als einen RR haben. In dieser Situation behandelt ein RR andere RRs genau wie jeden anderen iBGP-Sprecher. Andere RRs können demselben Cluster (Client-Gruppe) oder anderen Clustern angehören. In einer einfachen Konfiguration können Sie das AS in mehrere Cluster aufteilen. Sie konfigurieren jeden RR mit anderen RRs als Nicht-Client-Peers in einer vollständig vernetzten Topologie. Clients dürfen keine Peer-Verbindung mit iBGP-Routern außerhalb des Client-Clusters herstellen.

Im vorherigen Diagramm bilden RTA, RTB und RTC einen Cluster. RTC ist der RR. Für RTC sind RTA und RTB Clients und alles andere ist ein Nonclient. Denken Sie daran, dass der **Nachbar-Route-Reflektor-Client-**Befehl auf Clients eines RR verweist. Derselbe RTD ist der RR für die Clients RTE und RTF. RTG ist ein RR in einem dritten Cluster.



Hinweis: RTD, RTC und RTG sind vollständig vermascht, Router in einem Cluster jedoch nicht.

Wenn ein RR eine Route empfängt, werden die RR-Routen wie in dieser Liste dargestellt weitergeleitet. Diese Aktivität hängt jedoch vom Peer-Typ ab:

- Routes from a non client peer (Routen von einem Nicht-Client-Peer): Reflektiert alle Clients im Cluster.
-

Routes from a Client peer (Routen von einem Client-Peer) - Reflektiert alle Nicht-Client-Peers und auch die Client-Peers.

- Routes from a eBGP peer (Routen von einem eBGP-Peer): Sendet das Update an alle Client- und Nicht-Client-Peers.

Die relative BGP-Konfiguration der Router RTC, RTD und RTB sieht wie folgt aus:

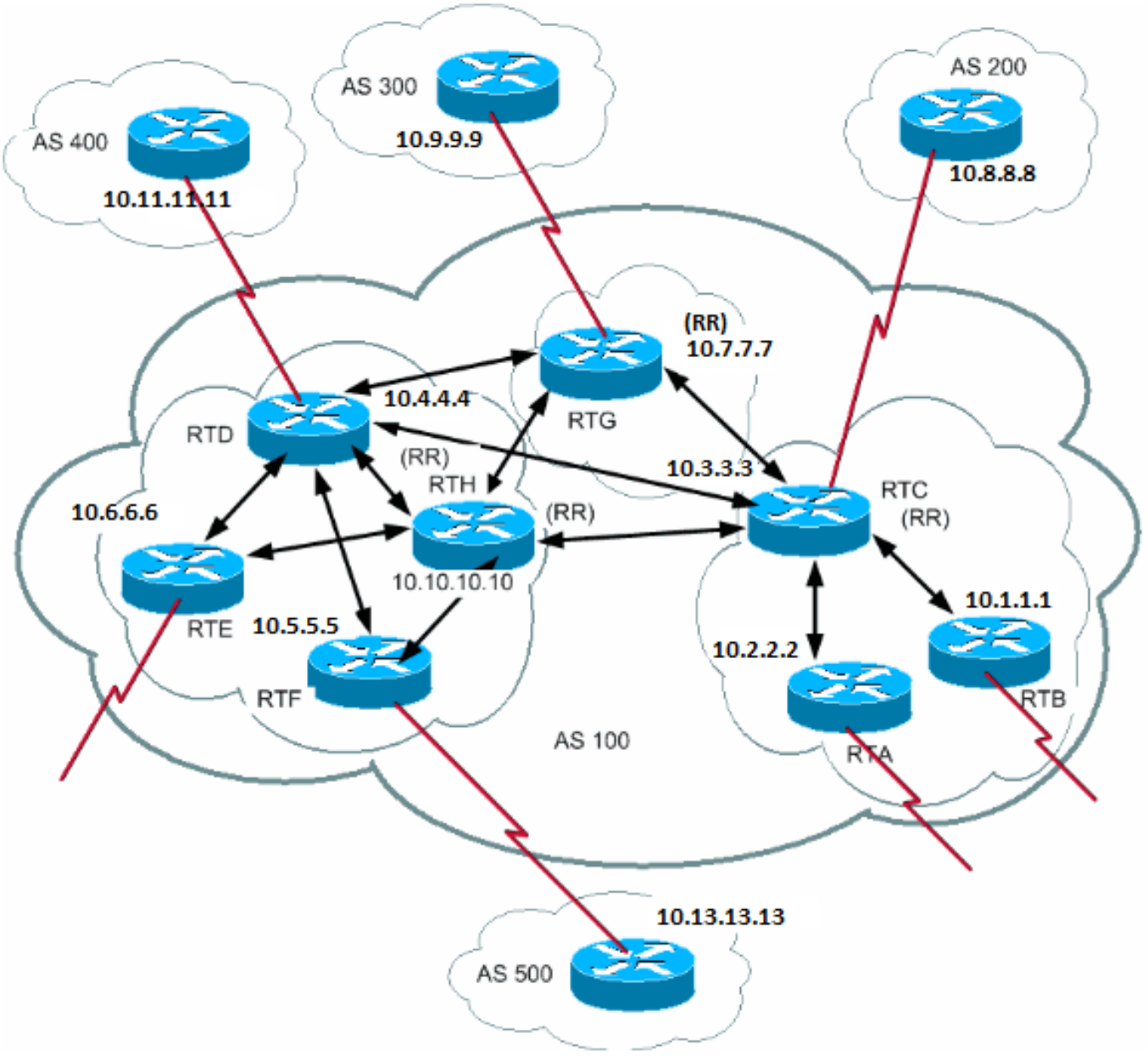
```
RTC#
router bgp 100
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 route-reflector-client
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 route-reflector-client
 neighbor 10.7.7.7 remote-as 100
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.8.8.8 remote-as 200

RTB#
router bgp 100
 neighbor 10.3.3.3 remote-as 100
 neighbor 10.12.12.12 remote-as 300

RTD#
router bgp 100
 neighbor 10.6.6.16 remote-as 100
 neighbor 10.6.6.16 route-reflector-client
 neighbor 10.5.5.5 remote-as 100
 neighbor 10.5.5.5 route-reflector-client
 neighbor 10.7.7.7 remote-as 100
 neighbor 10.3.3.3 remote-as 100
```

Da die vom iBGP ermittelten Routen reflektiert werden, kann eine Routing-Informationsschleife vorhanden sein. Das RR-Schema verfügt über einige Methoden, um diese Schleife zu vermeiden:

- **originator-id** - Dies ist ein optionales, nicht transitives BGP-Attribut mit 4 Byte Länge. Ein RR erstellt dieses Attribut. Das Attribut enthält die Router-ID (RID) des Routenursprungs im lokalen AS. Wenn die Routing-Informationen aufgrund einer fehlerhaften Konfiguration an den Ausgangspunkt zurückgegeben werden, werden sie ignoriert.

- **cluster-list** - Der Abschnitt Mehrere RRs innerhalb eines Clusters deckt die Cluster-Liste ab.

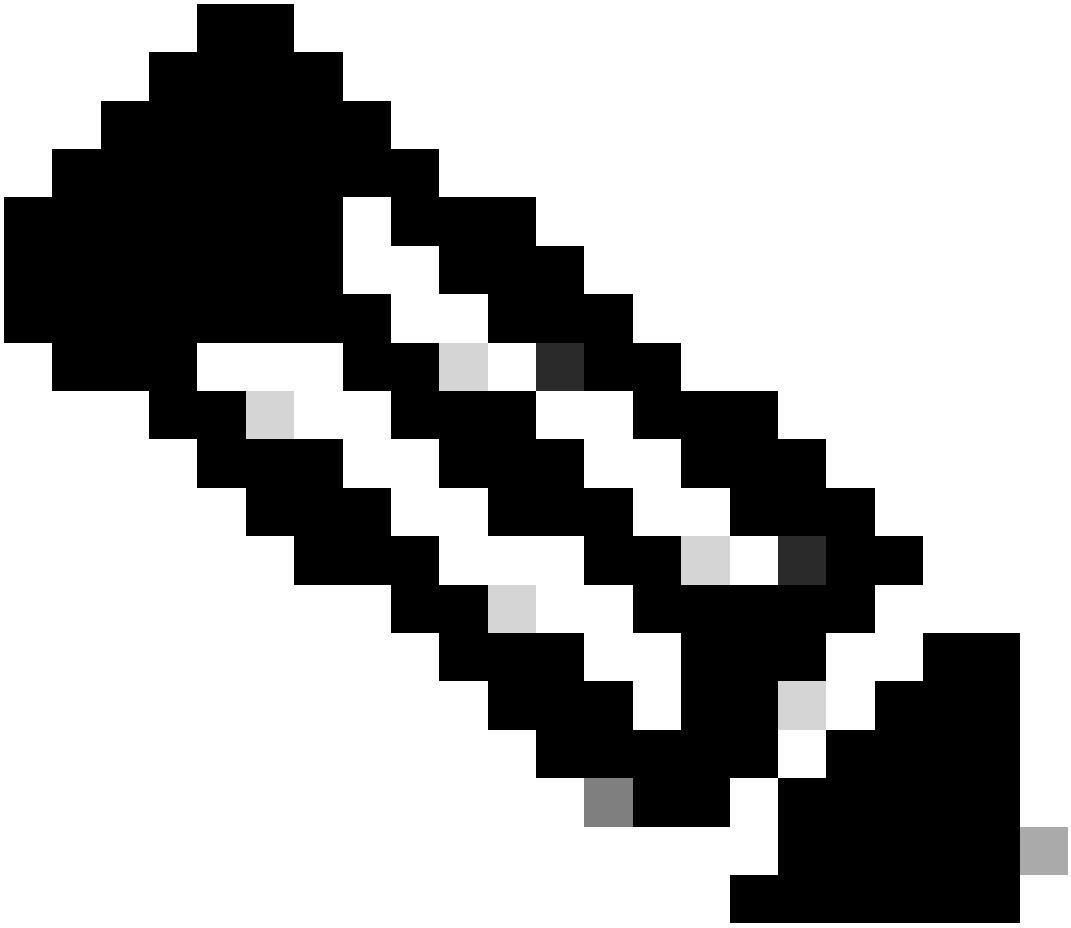
Mehrere RRs innerhalb eines Clusters



In der Regel verfügt ein Client-Cluster über einen einzigen RR. In diesem Fall identifiziert die Router-ID des RR den Cluster. Um die Redundanz zu erhöhen und Single-Points-of-Failure zu vermeiden, kann ein Cluster mehrere RRs haben. Sie müssen alle RRs im gleichen Cluster mit einer Cluster-ID von 4 Byte konfigurieren, damit ein RR Aktualisierungen von RRs im gleichen Cluster erkennen kann.

Eine Clusterliste ist eine Folge von Cluster-IDs, die von der Route weitergeleitet wurden. Wenn ein RR eine Route von den RR-Clients zu Nicht-Clients außerhalb des Clusters widerspiegelt, hängt der RR die lokale Cluster-ID an die Cluster-Liste an. Wenn dieses Update eine leere Clusterliste enthält, erstellt der RR eine. Mit diesem Attribut kann ein RR feststellen, ob die Routing-Informationen aufgrund einer schlechten Konfiguration zu demselben Cluster zurückgeleitet wurden. Wenn die lokale Cluster-ID in der Clusterliste gefunden wird, wird die Meldung ignoriert.

Im Diagramm in diesem Abschnitt gehören RTD, RTE, RTF und RTH zu einem Cluster. RTD und RTH sind RRs für dasselbe Cluster.

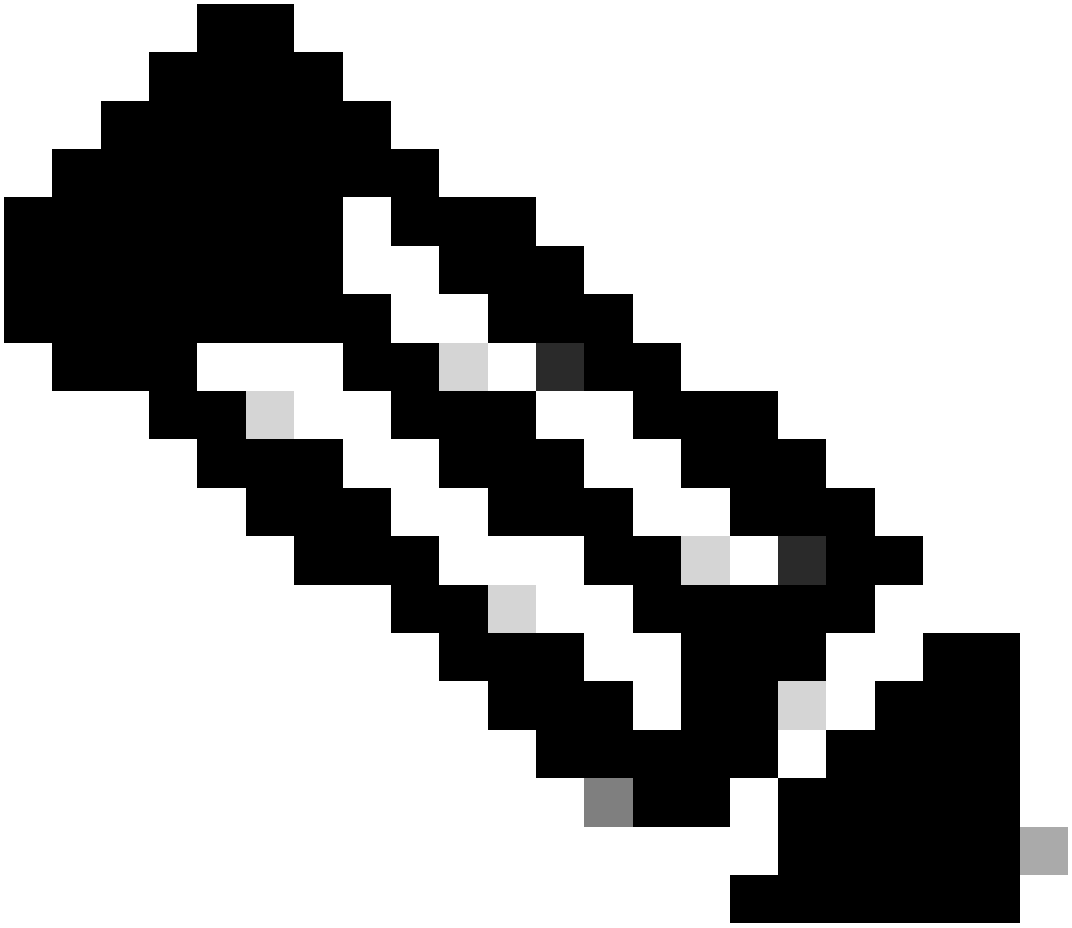


Hinweis: Es besteht Redundanz, da RTH über vollständig vernetztes Peering mit allen RRs verfügt. Wenn RTD ausfällt, ersetzt RTH RTD.

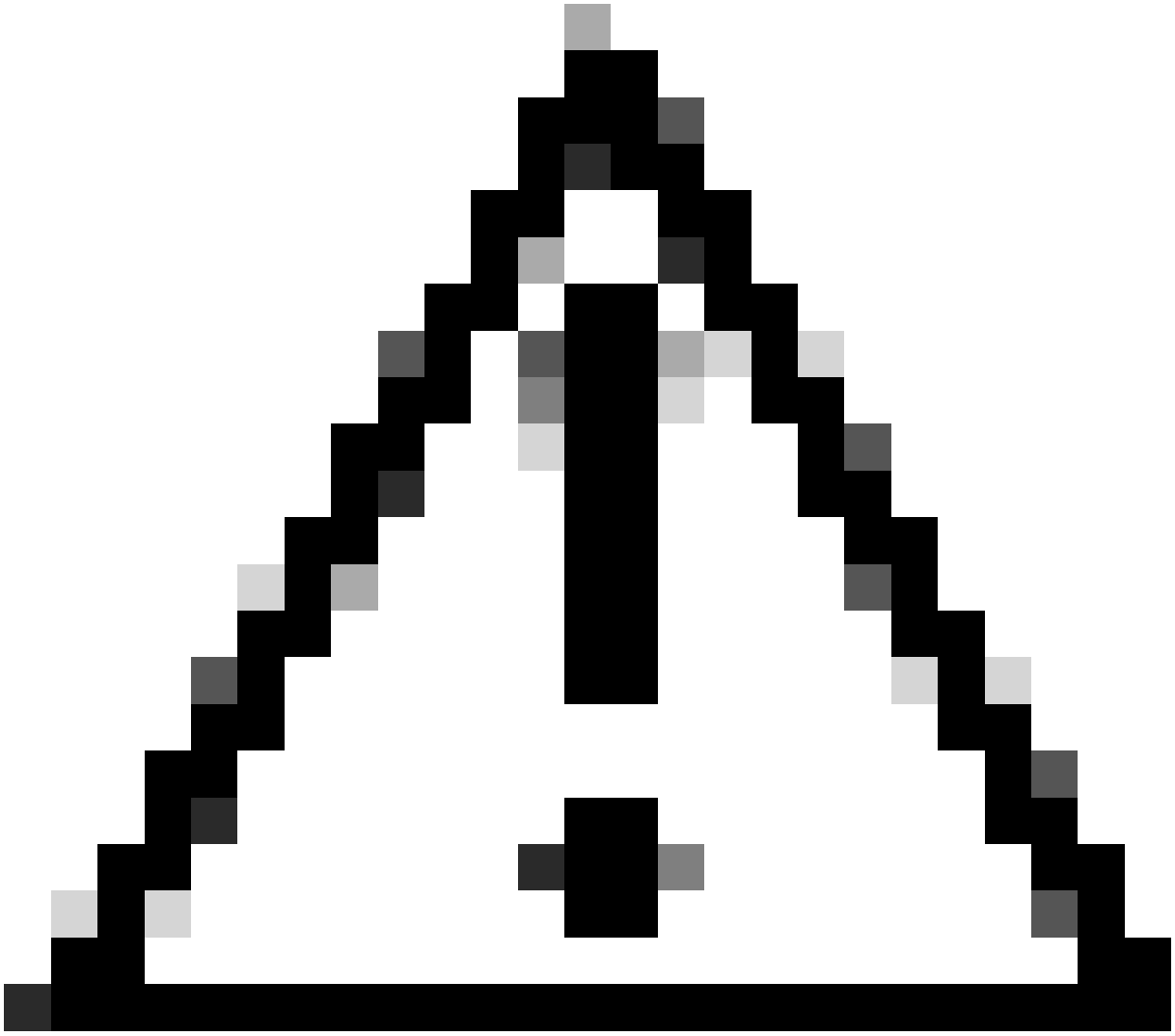
Nachfolgend finden Sie die Konfiguration von RTH, RTD, RTF und RTC:

```
RTH#
router bgp 100
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.5.5.5 remote-as 100
 neighbor 10.5.5.5 route-reflector-client
 neighbor 10.6.6.16 remote-as 100
 neighbor 10.6.6.16 route-reflector-client
 neighbor 10.7.7.7 remote-as 100
```

```
 neighbor 10.3.3.3 remote-as 100
 neighbor 10.9.9.9 remote-as 300
 bgp cluster-id 10

RTD#
router bgp 100
 neighbor 10.10.10.10 remote-as 100
 neighbor 10.5.5.5 remote-as 100
 neighbor 10.5.5.5 route-reflector-client
 neighbor 10.6.6.16 remote-as 100
 neighbor 10.6.6.16 route-reflector-client
 neighbor 10.7.7.7 remote-as 100
 neighbor 10.3.3.3 remote-as 100
 neighbor 10.11.11.11 remote-as 400
 bgp cluster-id 10

RTF#
router bgp 100
 neighbor 10.10.10.10 remote-as 100
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.13.13.13 remote-as 500

RTC#
router bgp 100
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 route-reflector-client
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.2.2.2 route-reflector-client
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.7.7.7 remote-as 100
 neighbor 10.10.10.10 remote-as 100
 neighbor 10.8.8.8 remote-as 200
```



Hinweis: Sie benötigen den Befehl "bgp cluster-id" für RTC nicht, da in diesem Cluster nur ein RR vorhanden ist.



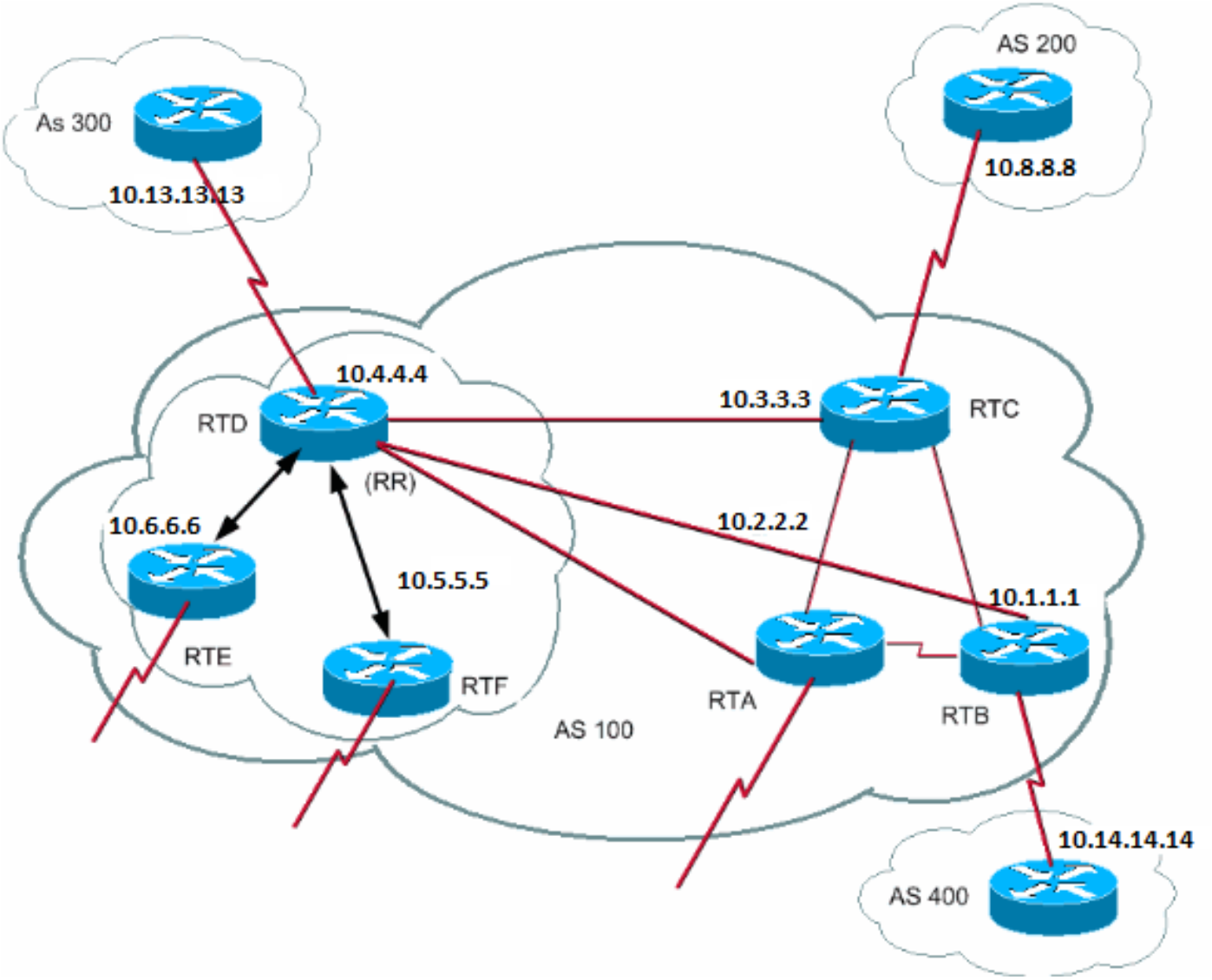
Achtung: Bei dieser Konfiguration werden keine Peer-Gruppen verwendet. Verwenden Sie keine Peer-Gruppen, wenn die Clients innerhalb eines Clusters untereinander keine direkten iBGP-Peers haben und die Clients Updates über den RR austauschen. Wenn Sie Peer-Gruppen konfigurieren, wird ein potenzieller Abbruch von der Quelle einer Route auf dem RR an alle Clients im Cluster übertragen. Diese Übertragung kann Probleme verursachen.

Der Router-Unterbefehl **bgp client-to-client mirror** ist auf dem RR standardmäßig aktiviert. Wenn Sie die BGP-Client-to-Client-Reflektion auf dem RR deaktivieren und redundantes BGP-Peering zwischen den Clients durchführen, können Sie Peer-Gruppen sicher verwenden. Weitere Informationen finden Sie unter Einschränkungen von Peer-Gruppen.

RR- und konventionelle BGP-Lautsprecher

Ein AS kann über BGP-Router verfügen, die das Konzept der RRs nicht verstehen. Im vorliegenden Dokument werden diese Router als

konventionelle BGP-Router bezeichnet. Das RR-Schema ermöglicht die parallele Nutzung solcher herkömmlichen BGP-Router. Diese Router können entweder Mitglieder einer Client-Gruppe oder einer Nicht-Client-Gruppe sein. Diese Router ermöglichen eine einfache und schrittweise Migration vom aktuellen iBGP- zum RR-Modell. Sie können Cluster erstellen, wenn Sie einen einzelnen Router als RR konfigurieren und andere RRs und RR-Clients zu normalen iBGP-Peers machen. Anschließend können Sie schrittweise weitere Cluster erstellen.



In diesem Diagramm haben RTD, RTE und RTF das Konzept der Routen-Reflektion. RTC, RTA und RTB sind konventionelle Router. Sie können diese Router nicht als RRs konfigurieren. Sie können normale iBGP-Mesh-Verbindungen zwischen diesen Routern und RTD herstellen. Später, wenn Sie bereit sind zu aktualisieren, können Sie RTC ein RR mit Clients RTA und RTB machen. Clients müssen das Routen-Reflexionsschema nicht verstehen; nur die RRs erfordern das Upgrade.

Nachfolgend finden Sie die Konfiguration von RTD und RTC:

```
RTD#
router bgp 100
 neighbor 10.6.6.16 remote-as 100
 neighbor 10.6.6.16 route-reflector-client
 neighbor 10.5.5.5 remote-as 100
 neighbor 10.5.5.5 route-reflector-client
 neighbor 10.3.3.3 remote-as 100
 neighbor 10.2.2.2 remote-as 100
```

```
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.13.13.13 remote-as 300

RTC#
router bgp 100
 neighbor 10.4.4.4 remote-as 100
 neighbor 10.2.2.2 remote-as 100
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.14.14.14 remote-as 400
```

Wenn Sie bereit sind, RTC zu aktualisieren und RTC zu einem RR zu machen, entfernen Sie das iBGP-Full-Mesh, und lassen Sie RTA und RTB zu Clients von RTC werden.

Vermeidung der Schleife von Routing-Informationen

Bisher wurden in diesem Dokument zwei Attribute genannt, mit denen Sie Schleifen potenzieller Informationen verhindern können: **Originator-ID** und **Cluster-Liste**.

Eine weitere Möglichkeit zur Steuerung von Loops besteht darin, die **festgelegte** Klausel von Outbound-Routing-Maps stärker zu beschränken. Die **set**-Klausel für ausgehende Routenzuordnungen wirkt sich nicht auf Routen aus, die iBGP-Peers entsprechen.

Sie können auch weitere Einschränkungen für "**next-hop-self"** festlegen, das heißt eine Konfigurationsoption pro Nachbar. Wenn Sie **Next-Hop**-Selfo auf RRs verwenden, wirkt sich die Klausel nur auf den nächsten Hop der vom eBGP bezogenen Routen aus, da der nächste Hop der reflektierten Routen nicht geändert werden darf.

Streckendämpfung

In Version 11.0 der Cisco IOS-Software wurde eine Routen-Dämpfung eingeführt. Routen-Dampening ist ein Mechanismus zur Minimierung der Instabilität, die durch Fluktuationen der Routen verursacht wird. Die Routendämpfung reduziert außerdem Schwingungen über das Netzwerk. Sie definieren Kriterien, um schlecht funktionierende Routen zu identifizieren. Eine Route, die flattert, bekommt eine Strafe von 1000 für jede Klappe. Sobald die kumulative Strafe einen vordefinierten Unterdrückungsgrenzwert erreicht, wird die Routenankündigung unterdrückt. Die Strafe fällt aufgrund einer vorkonfigurierten Halbwertszeit exponentiell ab. Sobald die Penalty unter einem vordefinierten Wiederverwendungslimit abnimmt, wird die Routenankündigung nicht mehr unterdrückt.

Routen-Dampening gilt nicht für Routen, die sich außerhalb eines AS befinden und vom iBGP bezogen werden. Auf diese Weise verhindert das Routing-Dampening eine höhere Benachteiligung der iBGP-Peers bei Routen außerhalb des AS.

Die Strafe verfällt mit einer Granularität von 5 Sekunden. Die Routen werden mit einer Genauigkeit von 10 Sekunden nicht unterdrückt. Der Router behält die Dämpfungsinformationen bei, bis die Strafe unter der Hälfte des Wiederverwendungs-Grenzwerts liegt. An diesem Punkt löscht der Router die Informationen.

Anfänglich ist die Dämpfung standardmäßig ausgeschaltet. Bei Bedarf kann diese Funktion in Zukunft standardmäßig aktiviert werden. Diese Befehle steuern die Routendämpfung:

- **BGP-Dämpfung** - Schaltet die Dämpfung ein.

- **keine BGP-Dämpfung** - Schaltet die Dämpfung ab.

- **bgp dämpfeninghalf-life-time (Halbwertszeit**) - Ändert die Halbwertszeit.

Ein Befehl, der alle Parameter gleichzeitig festlegt, ist:

- **BGP-DämpfungHalbwertszeitVerwendetUnterdrückenmaximale Unterdrückungszeit**

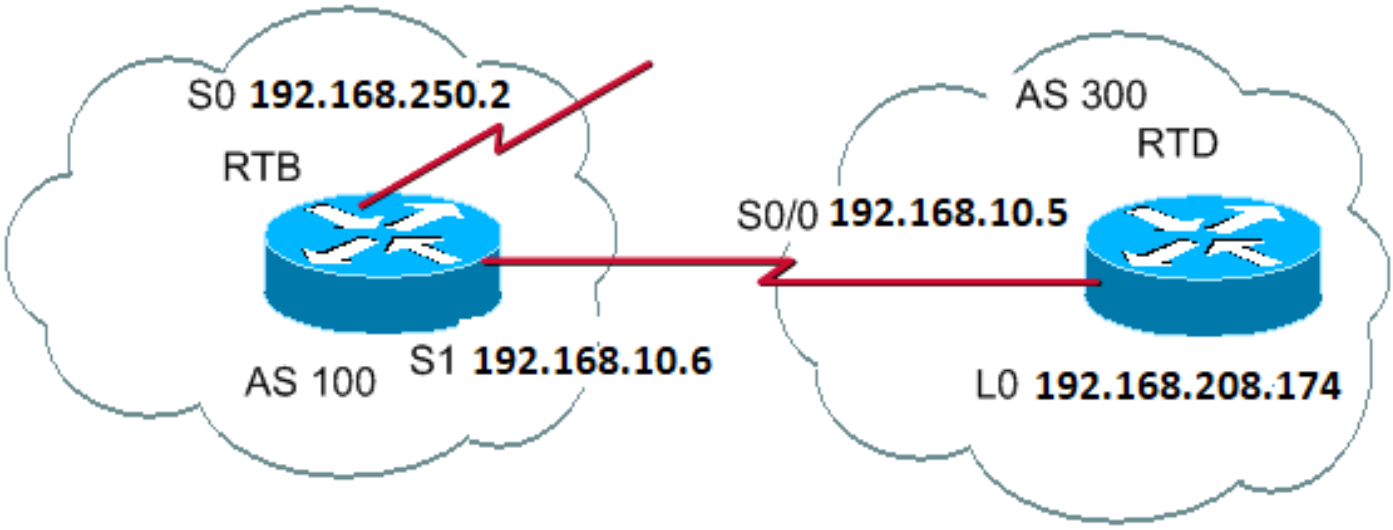
Diese Liste enthält Details zur Syntax:

- **half-life-time (Halbwertszeit**) - Der Bereich liegt zwischen 1 und 45 Minuten, und der aktuelle Standardwert ist 15 Minuten.

- **reuse-value** - Der Bereich liegt zwischen 1 und 20.000 und der Standardwert ist 750.

- **suppress-value** - Der Bereich liegt zwischen 1 und 20.000 und der Standardwert ist 2000.

- **max-suppress-time**: Die maximale Dauer für die Unterdrückung einer Route. Der Bereich liegt zwischen 1 und 255 Minuten, der Standardwert ist das Vierfache der Halbwertszeit.




```
RTB#
hostname RTB

interface Serial0
 ip address 192.168.250.2 255.255.255.252

interface Serial1
 ip address 192.168.10.6 255.255.255.252

router bgp 100
 bgp dampening
 network 192.168.250.15
 neighbor 192.168.10.5 remote-as 300
```

```
RTD#
hostname RTD

interface Loopback0
 ip address 192.168.208.174 255.255.255.192

interface Serial0/0
 ip address 192.168.10.5 255.255.255.252

router bgp 300
 network 192.168.10.0
 neighbor 192.168.10.6 remote-as 100
```

Die RTB-Konfiguration dient zur Routen-Dämpfung mit Standardparametern. Wenn Sie davon ausgehen, dass die eBGP-Verbindung zu RTD stabil ist, sieht die RTB-BGP-Tabelle folgendermaßen aus:

<#root>

```
RTB#
```

show ip bgp

```
BGP table version is 24, local router ID is 192.168.250.2 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
*> 192.168.10.0     192.168.10.5           0             0 300 i
*> 192.168.250.15     0.0.0.0                0         32768 i
```

Um eine Routing-Klappe zu simulieren, geben Sie den Befehl **clear ip bgp 192.168.10.6** auf RTD ein. Die RTB-BGP-Tabelle sieht wie folgt aus:

<#root>

```
RTB#
```

show ip bgp

```
BGP table version is 24, local router ID is 192.168.250.2 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
 h 192.168.10.0     192.168.10.5           0             0 300 i
*> 192.168.250.15     0.0.0.0                0         32768 i
```

Der BGP-Eintrag für 192.168.10.0 befindet sich im ahistorystate-Status. Diese Platzierung bedeutet, dass Sie keinen besten Pfad zur Route haben, aber Informationen über das Flapping der Route weiterhin vorhanden sind.

<#root>

```
RTB#
```

show ip bgp 192.168.10.0

```
BGP routing table entry for 192.168.10.0 255.255.255.0, version 25
Paths: (1 available, no best path)
300 (history entry)
    192.168.10.5 from 192.168.10.5 (192.168.208.174)
Origin IGP, metric 0, external
Dampinfo: penalty 910, flapped 1 times in 0:02:03
```

Die Route hat eine Strafe für Flapping erhalten, aber die Strafe ist immer noch unter dem Unterdrückungslimit. Der Standardwert ist 2000. Es ist noch keine Routenunterdrückung aufgetreten. Wenn die Route noch ein paar Mal klappt, sehen Sie:

```
    <#root>

RTB#
```

show ip bgp

```
BGP table version is 32, local router ID is 192.168.250.2 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
*d 192.168.10.0     192.168.10.5           0             0  300 i
*> 192.168.250.15     0.0.0.0               0       32768  i

RTB#
```

show ip bgp 192.168.10.0

```
BGP routing table entry for 192.168.10.0 255.255.255.0, version 32
```

```
Paths: (1 available, no best path)
300, (suppressed due to dampening)
192.168.10.5 from 192.168.10.5 (192.168.208.174)
      Origin IGP, metric 0, valid, external
Dampinfo: penalty 2615, flapped 3 times in 0:05:18 , reuse in 0:27:00
```

Die Route wurde gedämpft oder unterdrückt. Die Route wird wiederverwendet, wenn die Strafe den "Wiederverwendungswert" erreicht. In diesem Fall ist der Wiederverwendungswert der Standardwert 750. Die dämpfenden Informationen werden gelöscht, wenn die Strafe unter der Hälfte des Wiederverwendungslimits liegt. In diesem Fall erfolgt die Löschung, wenn die Strafe 375 wird (750/2 = 375). Mit diesen Befehlen werden Informationen zu Klappenstatistiken angezeigt und gelöscht:

- **show ip bgp flat-statistics** - Zeigt Flapping-Statistiken für alle Pfade an.
- **show ip bgp flatch-statistics regulärer Ausdruck** - Zeigt Flatch-Statistiken für alle Pfade an, die mit dem regulären Ausdruck übereinstimmen.
- **show ip bgp flat-statistics filter-listlist** - Zeigt Flapping-Statistiken für alle Pfade an, die den Filter passieren.
- **show ip bgp flaps-statisticsA.B.C.D m.m.m.m**— Zeigt Flaps für einen einzigen Eintrag an.
- **show ip bgp klappe-statisticsA.B.C.D m.m.m.mlonger-prefix** - Zeigt die Flap-Statistik für spezifischere Einträge an.
- **show ip bgp neighbor [gedämpfte-routen] | [Klappenstatistik]** - Zeigt Klappenstatistiken für alle Pfade von einem Nachbarn an.
- **clear ip bgp flatch-statistics** - Löscht Flatch-Statistiken für alle Routen.
-

clear ip bgp flatch-statistics regulärer Ausdruck - Löscht Flatch-Statistiken für alle Pfade, die mit dem regulären Ausdruck übereinstimmen.

- **clear ip bgp flatch-statistics filter-listlist** - Löscht Flatch-Statistiken für alle Pfade, die den Filter passieren.

- **clear ip bgp flaps-statisticsA.B.C.D m.m.m.m**— Löscht die Flapping-Statistiken für einen einzelnen Eintrag.

- **clear ip bgpA.B.C.Dklappe-statistics -** Löscht die Flapping-Statistiken für alle Pfade eines Nachbarn.

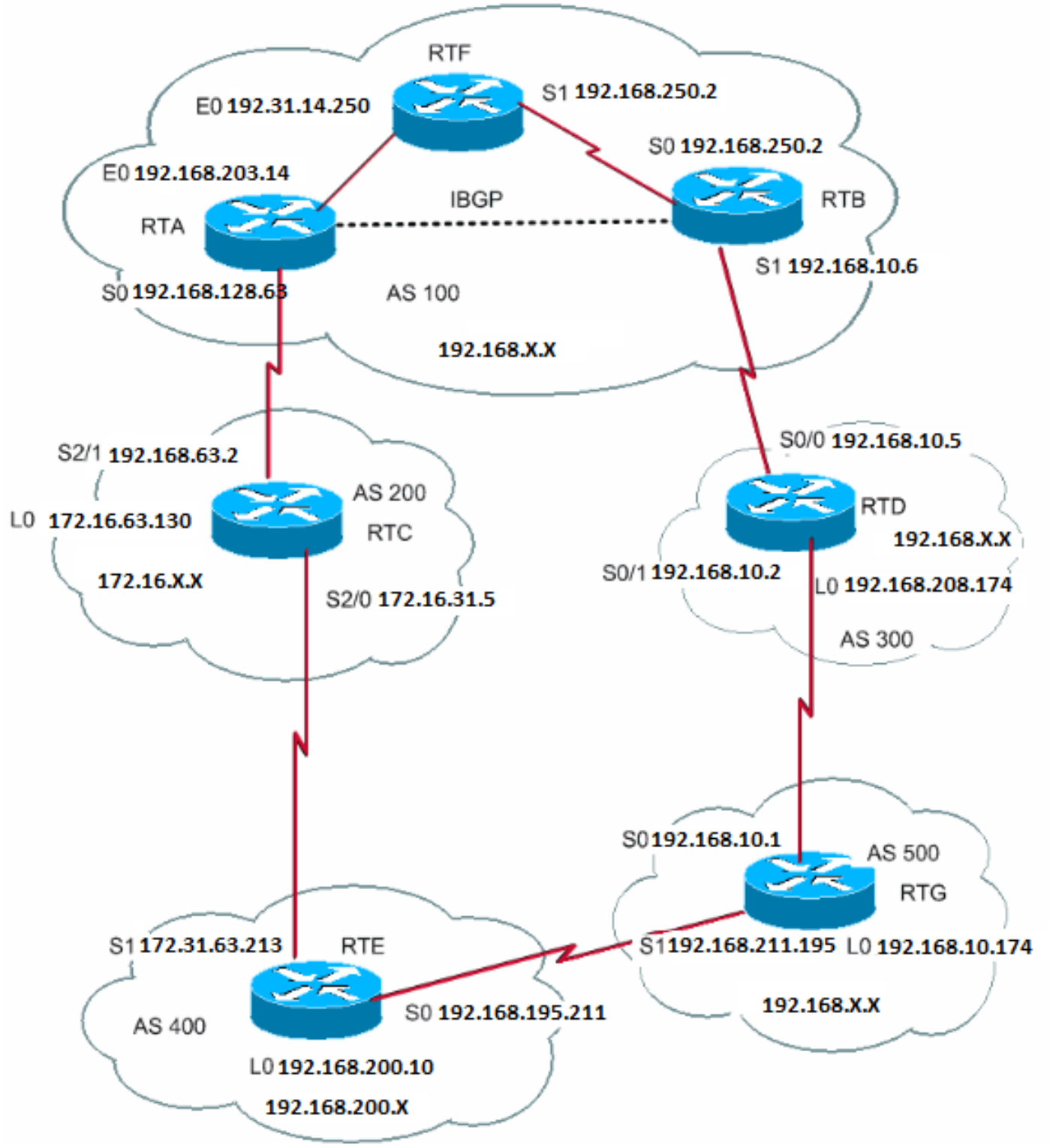
Wie BGP einen Pfad auswählt

Nachdem Sie nun mit den BGP-Attributen und der Terminologie vertraut sind, lesen Sie den Abschnitt BGP Best Path Selection Algorithm (Algorithmus zur Auswahl des besten BGP-Pfads).

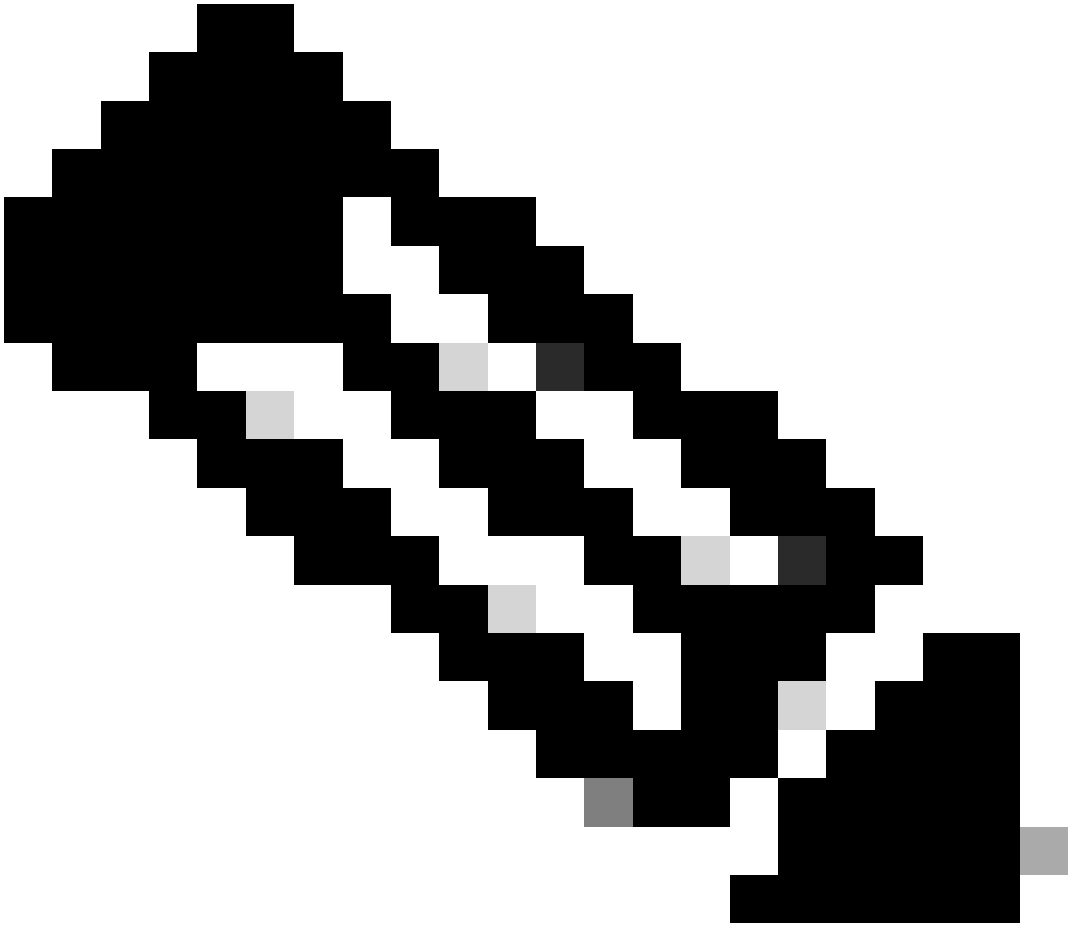
BGP-Anwenderberichte 5

Praxisbeispiel

Dieser Abschnitt enthält ein Designbeispiel mit den Konfigurations- und Routing-Tabellen, wie sie auf Cisco Routern tatsächlich angezeigt werden.



Dieser Abschnitt zeigt, wie Sie diese Konfiguration Schritt für Schritt erstellen und was dabei schief gehen kann. Wenn Sie ein AS haben, das sich über eBGP mit zwei ISPs verbindet, führen Sie iBGP immer innerhalb Ihres AS aus, um eine bessere Kontrolle über Ihre Routen zu erhalten. In diesem Beispiel wird iBGP im AS100 zwischen RTA und RTB ausgeführt, und OSPF wird als IGP ausgeführt. Angenommen, Sie stellen eine Verbindung zu zwei ISPs her, AS200 und AS300. Dies ist die erste Konfigurationsausführung für alle Router:



Hinweis: Diese Konfigurationen sind nicht die endgültigen.

```
RTA#
hostname RTA

ip subnet-zero

interface Loopback0
 ip address 192.168.203.250 255.255.255.0

interface Ethernet0
ip address 192.168.203.14 255.255.255.0

interface Serial0
```

```
 ip address 192.168.128.63 255.255.255.252

router ospf 10
 network 192.168.203.25 0.0.255.255 area 0

router bgp 100
 network 192.168.203.13
 network 192.168.250.14
 neighbor 172.31.63.250 remote-as 200
 neighbor 192.168.250.2 remote-as 100
 neighbor 192.168.250.2 update-source Loopback0

RTF#
hostname RTF

ip subnet-zero

interface Ethernet0
 ip address 172.31.14.250 255.255.255.0

interface Serial1
 ip address 172.16.15.250 255.255.255.252

router ospf 10
 network 192.168.203.25 0.0.255.255 area 0

RTB#
hostname RTB

ip subnet-zero

interface Serial0
 ip address 192.168.250.2 255.255.255.252

interface Serial1
 ip address 192.168.10.6 255.255.255.252

router ospf 10
 network 192.168.203.25 0.0.255.255 area 0

router bgp 100
network 192.168.250.15
 neighbor 192.168.10.5 remote-as 300
 neighbor 192.168.203.250 remote-as 100

RTC#
hostname RTC

ip subnet-zero

interface Loopback0
 ip address 192.168.128.6330 255.255.255.192

interface Serial2/0
 ip address 172.16.31.5 255.255.255.252
!
interface Serial2/1
 ip address 172.31.63.250 255.255.255.252

router bgp 200
 network 172.31.10.0
 neighbor 192.168.128.63 remote-as 100
```

```
 neighbor 172.31.63.213 remote-as 400

RTD#
hostname RTD

ip subnet-zero

interface Loopback0
ip address 192.168.208.174 255.255.255.192

interface Serial0/0
 ip address 192.168.10.5 255.255.255.252
!
interface Serial0/1
 ip address 192.168.10.2 255.255.255.252

router bgp 300
 network 192.168.10.0
 neighbor 192.168.10.1 remote-as 500
 neighbor 192.168.10.6 remote-as 100

RTE#
hostname RTE

ip subnet-zero

interface Loopback0
ip address 192.168.200.10 255.255.255.0

interface Serial0
 ip address 192.168.195.211 255.255.255.252

interface Serial1
 ip address 172.31.63.213 255.255.255.252
 clockrate 1000000

router bgp 400
 network 192.168.10.10
 neighbor 172.16.31.5 remote-as 200
 neighbor 192.168.211.195 remote-as 500

RTG#
hostname RTG

ip subnet-zero

interface Loopback0
 ip address 192.168.211.19574 255.255.255.192

interface Serial0
 ip address 192.168.10.1 255.255.255.252

interface Serial1
 ip address 192.168.211.195 255.255.255.252

router bgp 500
 network 192.168.211.10
 neighbor 192.168.10.2 remote-as 300
 neighbor 192.168.195.211 remote-as 400
```

Verwenden Sie immer den network Befehl, oder verteilen Sie statische Einträge im BGP, um Netzwerke anzukündigen. Diese Methode ist besser als eine Umverteilung von IGP in BGP. In diesem Beispiel wird der network Befehl zum Einschleusen von Netzwerken in das BGP verwendet.

Hier beginnt man mit der s1-Schnittstelle bei RTB-Shutdown, als ob die Verbindung zwischen RTB und RTD nicht existiert. Dies ist die RTB-BGP-Tabelle:

```
    <#root>

RTB#

show ip bgp BGP


table version is 4, local router ID is 192.168.250.2 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
*i172.31.10.0      172.31.63.250           0    100      0 200 i
*i192.168.10.0     172.31.63.250                100      0 200 400 500
300 i
*i192.168.211.10     172.31.63.250                100      0 200 400 500 i
*i192.168.10.10     172.31.63.250                100      0 200 400 i
*>i192.168.203.13    192.168.203.250          0    100      0 i
*>i192.168.250.14    192.168.203.250          0    100      0 i
*>192.168.250.15     0.0.0.0                  0         32768 i
```

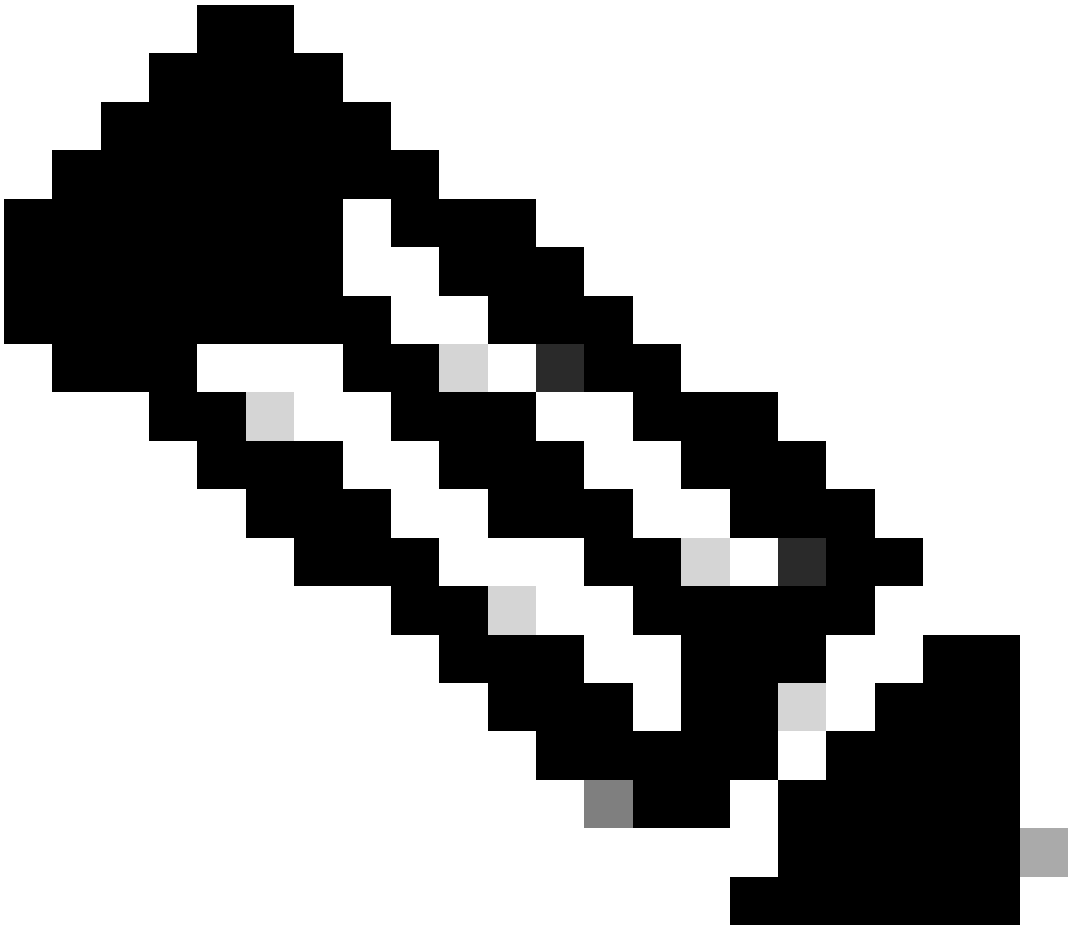
In dieser Tabelle werden folgende Notierungen angezeigt:

- Aniat the begin (Aniat am Anfang) - Zeigt an, dass der Eintrag über einen iBGP-Peer abgerufen wurde.

- Aniat the end (Aniat am Ende): Zeigt an, dass die Pfadinformationen vom IGP stammen.

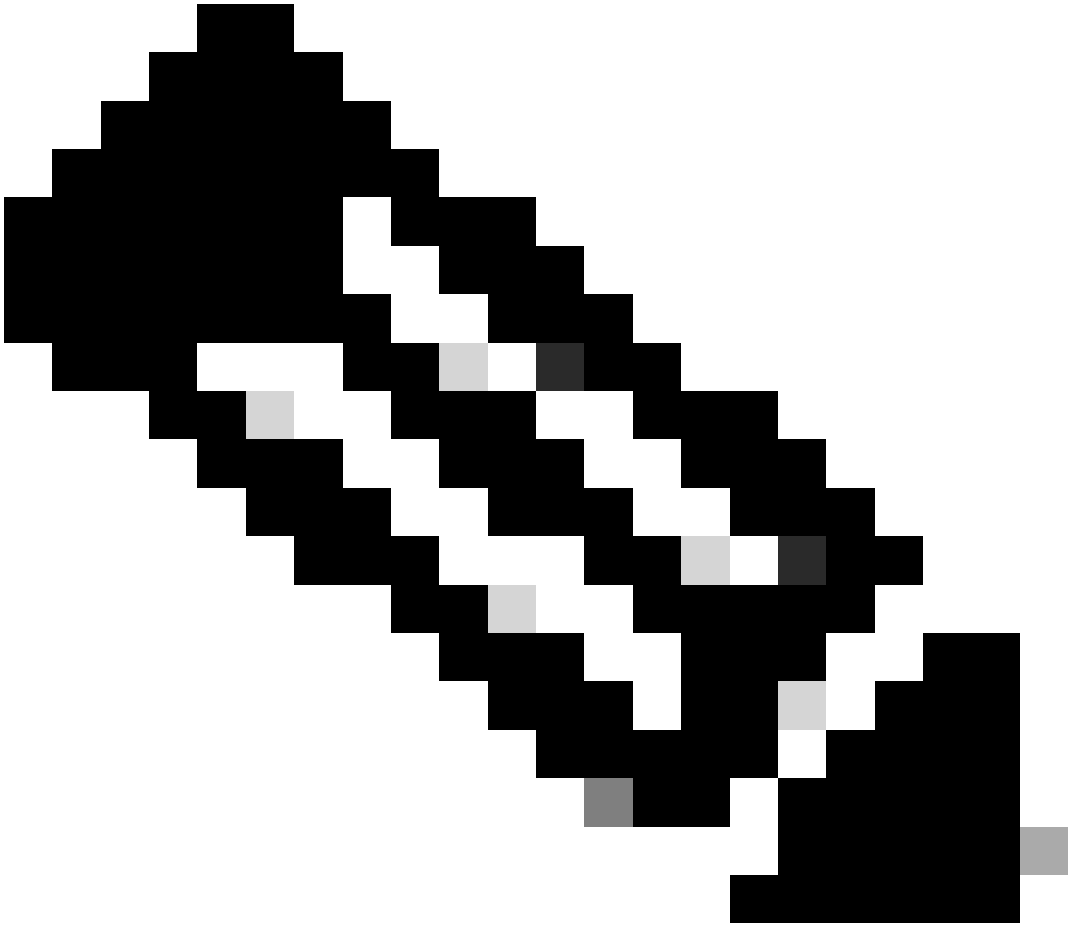
- Pathinformation: Diese Informationen sind intuitiv. Beispielsweise wird das Netzwerk 172.31.10.0 über Pfad 200 mit dem nächsten Hop

172.31.63.250 erfasst.



Hinweis: Jeder lokal generierte Eintrag, z. B. 192.168.250.15, hat den nächsten Hop 0.0.0.0.

- An**>**symbol - gibt an, dass das BGP die beste Route gewählt hat. BGP verwendet die Entscheidungsschritte, die im Dokument BGP Best Path Selection Algorithm umrissen werden. Das BGP wählt einen besten Pfad zum Erreichen eines Ziels aus, installiert den Pfad in der IP-Routing-Tabelle und kündigt den Pfad anderen BGP-Peers an.



Hinweis: Beachten Sie das Next Hop-Attribut. Die RTB kennt den Wert 172.31.10.0 über einen Next Hop von 172.31.63.250, dem eBGP Next Hop, der in das iBGP übertragen wird.

Sehen Sie sich die IP-Routing-Tabelle an:

<#root>

RTB#

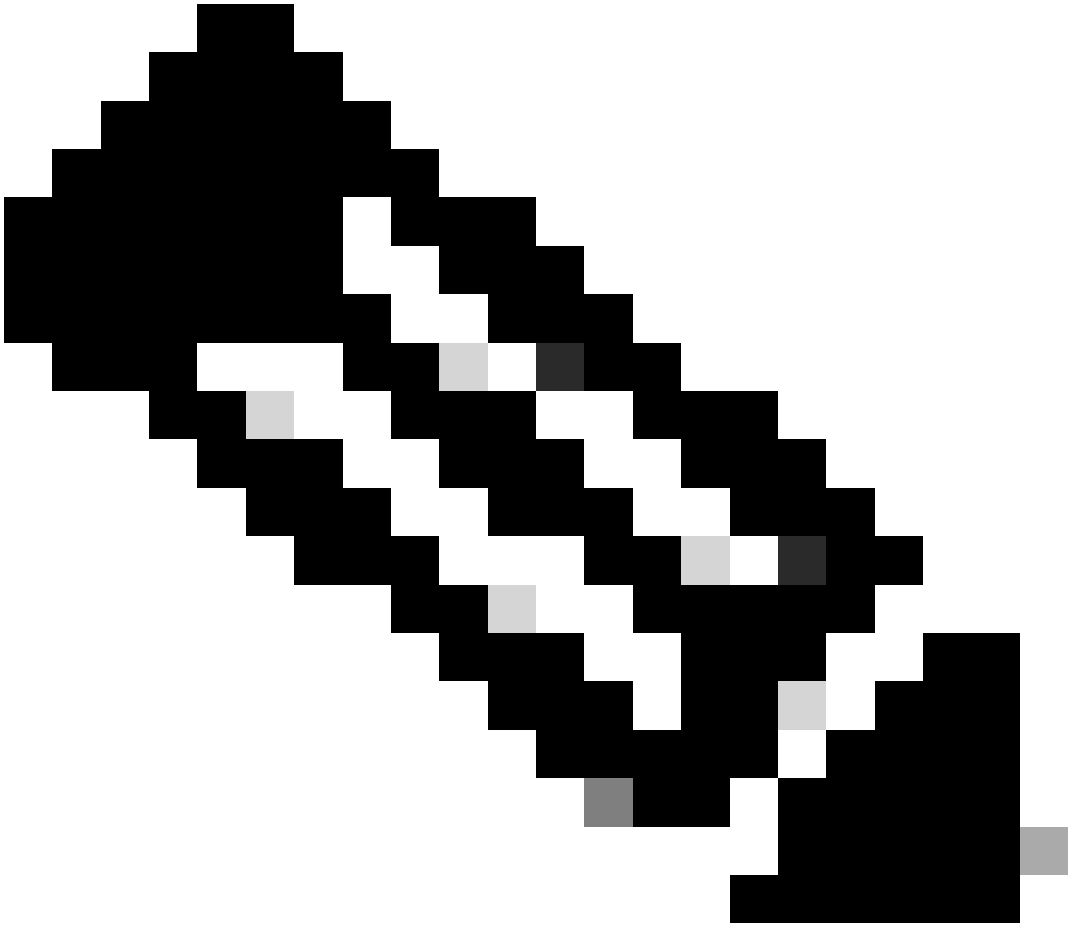
show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate
default

Gateway of last resort is not set

     192.168.203.13 255.255.255.255 is subnetted, 1 subnets
O       192.168.203.250 [110/75] via 172.16.15.250, 02:50:45, Serial0
     192.168.250.15 255.255.255.252 is subnetted, 1 subnets
C       192.168.250.15 is directly connected, Serial0
O    192.168.250.14 [110/74] via 172.16.15.250, 02:50:46, Serial0
```

Offensichtlich hat keiner der BGP-Einträge die Routing-Tabelle erreicht. Hier gibt es zwei Probleme.

Das erste Problem besteht darin, dass der nächste Hop für diese Einträge, 172.31.63.250, nicht erreichbar ist. Der nächste Hop kann über dieses IGP (OSPF) nicht erreicht werden. Die RTB hat nicht über OSPF von 192.168.213.63 erfahren. Sie können OSPF auf der RTA s0-Schnittstelle ausführen und passiv machen. So weiß RTB, wie man den nächsten Hop erreicht: 172.31.63.250. Diese RTA-Konfiguration wird hier angezeigt:

```
RTA#
hostname RTA

ip subnet-zero

interface Loopback0
 ip address 192.168.203.250 255.255.255.0

interface Ethernet0
ip address 192.168.203.14 255.255.255.0

interface Serial0
 ip address 192.168.128.63 255.255.255.252

router ospf 10
 passive-interface Serial0
 network 192.168.203.25 0.0.255.255 area 0
 network 172.31.10.0 0.0.255.255 area 0

router bgp 100
 network 192.168.203.25 mask 255.255.0.0
 neighbor 172.31.63.250 remote-as 200
 neighbor 192.168.250.2 remote-as 100
 neighbor 192.168.250.2 update-source Loopback0
```



Hinweis: Sie können den bgp nexthop self Befehl zwischen RTA und RTB ausgeben, um den nächsten Hop zu ändern.

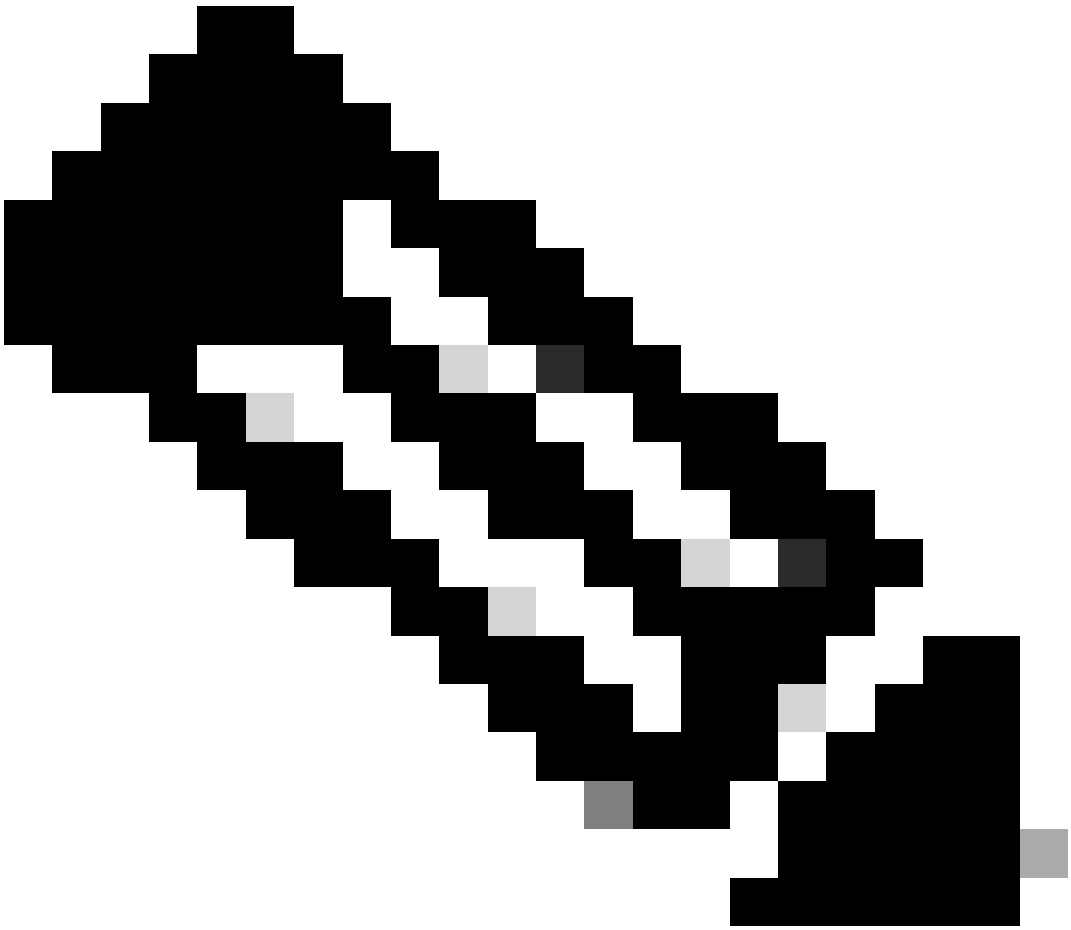
Die neue BGP-Tabelle für RTB sieht wie folgt aus:

<#root>

RTB#

show ip bgp

```
BGP table version is 10, local router ID is 192.168.250.2
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
*>i172.31.10.0      172.31.63.250           0    100      0 200 i
*>i192.168.10.0     172.31.63.250                100      0 200 400 500
300 i
*>i192.168.211.10     172.31.63.250                100      0 200 400 500 i
*>i192.168.10.10     172.31.63.250                100      0 200 400 i
*>i192.168.203.13     192.168.203.250          0    100      0 i
*>i192.168.250.14     192.168.203.250          0    100      0 i
*> 192.168.250.15     0.0.0.0                  0         32768 i
```



Hinweis: Alle Einträge haben >, d. h. BGP kann den nächsten Hop erreichen.

Sehen Sie sich die Routing-Tabelle an:

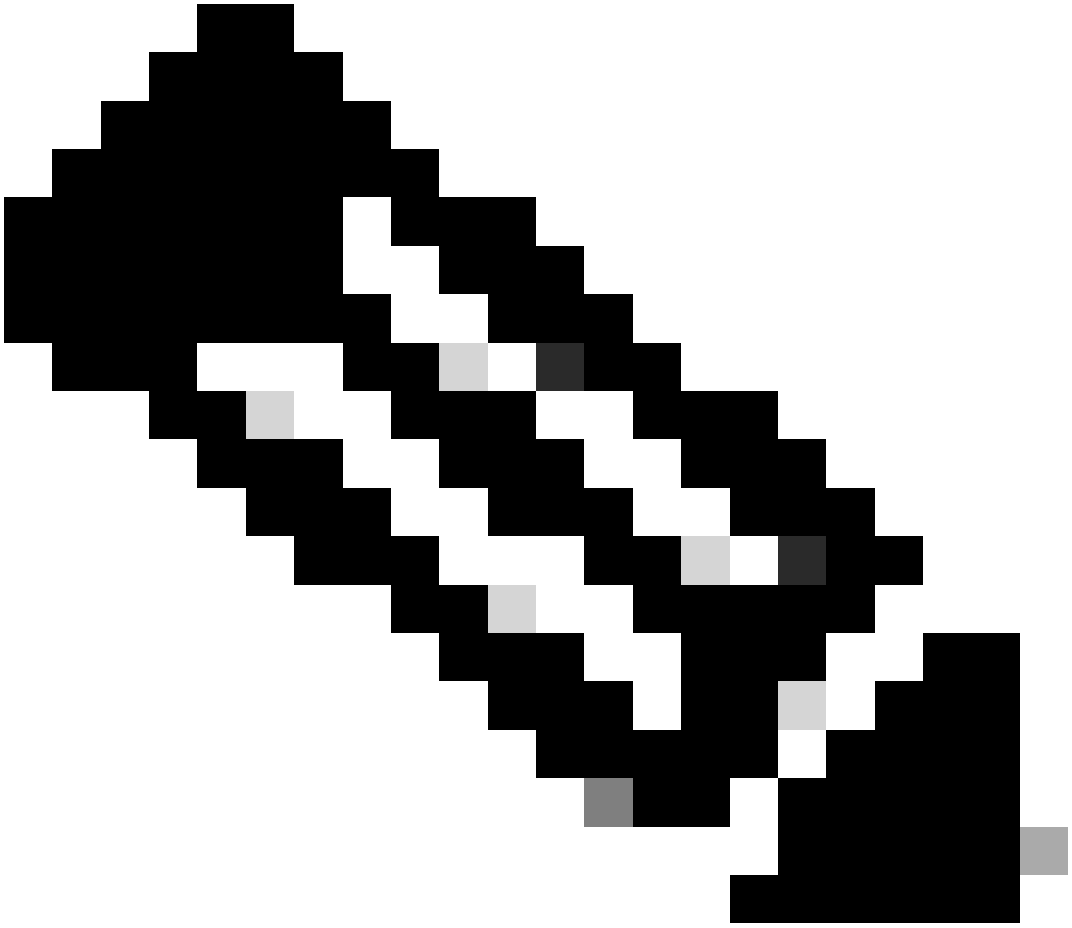
<#root>

RTB#

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is not set

     192.168.203.13 255.255.255.255 is subnetted, 1 subnets
O       192.168.203.250 [110/75] via 172.16.15.250, 00:04:46, Serial0
     192.168.250.15 255.255.255.252 is subnetted, 1 subnets
C       192.168.250.15 is directly connected, Serial0
O    192.168.250.14 [110/74] via 172.16.15.250, 00:04:46, Serial0
     172.31.10.0 255.255.255.252 is subnetted, 1 subnets
O       192.168.213.63 [110/138] via 172.16.15.250, 00:04:47, Serial0
```

Das zweite Problem besteht darin, dass Sie die BGP-Einträge in der Routing-Tabelle immer noch nicht sehen. Der einzige Unterschied besteht darin, dass 192.168.213.63 jetzt über OSPF erreichbar ist. Dieses Problem ist ein Synchronisierungsproblem. Das BGP stellt diese Einträge nicht in die Routing-Tabelle und sendet sie auch nicht in BGP-Updates, da keine Synchronisierung mit dem IGP möglich ist.



Hinweis: RTF hat keine Vorstellung von den Netzwerken 192.168.10.0 und 192.168.211.10, da Sie BGP noch nicht in OSPF umverteilt haben.

Wenn Sie in diesem Szenario die Synchronisierung deaktivieren, werden die Einträge in der Routing-Tabelle angezeigt. Dennoch ist die Anbindung weiterhin unterbrochen.

Wenn Sie die Synchronisierung auf RTB deaktivieren, geschieht Folgendes:

<#root>

```
RTB#
```

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is not set

B    192.168.10.10 [200/0] via 172.31.63.250, 00:01:07
B    192.168.211.10 [200/0] via 172.31.63.250, 00:01:07
B    192.168.10.0 [200/0] via 172.31.63.250, 00:01:07
     192.168.203.13 is variably subnetted, 2 subnets, 2 masks
O       192.168.203.250 255.255.255.255
           [110/75] via 172.16.15.250, 00:12:37, Serial0
B       192.168.203.13 255.255.255.0 [200/0] via 192.168.203.250, 00:01:08
     192.168.250.15 255.255.255.252 is subnetted, 1 subnets
C       192.168.250.15 is directly connected, Serial0
O    192.168.250.14 [110/74] via 172.16.15.250, 00:12:37, Serial0
     172.31.10.0 is variably subnetted, 2 subnets, 2 masks
B       172.31.10.0 255.255.0.0 [200/0] via 172.31.63.250, 00:01:08
O       192.168.213.63 255.255.255.252
           [110/138] via 172.16.15.250, 00:12:37, Serial0
```

Die Routing-Tabelle sieht gut aus, aber es gibt keine Möglichkeit, diese Netzwerke zu erreichen. RTF in der Mitte weiß nicht, wie man die Netzwerke erreicht:

<#root>

```
RTF#
```

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
```

```
candidate default

Gateway of last resort is not set

     192.168.203.13 255.255.255.255 is subnetted, 1 subnets
O       192.168.203.250 [110/11] via 192.168.203.14, 00:14:15, Ethernet0
     192.168.250.15 255.255.255.252 is subnetted, 1 subnets
C       192.168.250.15 is directly connected, Serial1
C    192.168.250.14 is directly connected, Ethernet0
     172.31.10.0 255.255.255.252 is subnetted, 1 subnets
O       192.168.213.63 [110/74] via 192.168.203.14, 00:14:15, Ethernet0
```

Wenn Sie in dieser Situation die Synchronisierung deaktivieren, besteht das Problem weiterhin. Für andere Probleme muss die Synchronisierung jedoch später erfolgen. Verteilung von BGP in OSPF auf RTA mit einer Metrik von 2000:

```
RTA#
hostname RTA

ip subnet-zero

interface Loopback0
 ip address 192.168.203.250 255.255.255.0

interface Ethernet0
ip address 192.168.203.14 255.255.255.0

interface Serial0
 ip address 192.168.128.63 255.255.255.252

router ospf 10
 redistribute bgp 100 metric 2000 subnets
 passive-interface Serial0
 network 192.168.203.25 0.0.255.255 area 0
 network 172.31.10.0 0.0.255.255 area 0

router bgp 100
 network 192.168.203.25 mask 255.255.0.0
 neighbor 172.31.63.250 remote-as 200
 neighbor 192.168.250.2 remote-as 100
 neighbor 192.168.250.2 update-source Loopback0
```

Die Routing-Tabelle sieht wie folgt aus:

<#root>

```
RTB#
```

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is not set

O E2 192.168.10.10 [110/2000] via 172.16.15.250, 00:00:14, Serial0
O E2 192.168.211.10 [110/2000] via 172.16.15.250, 00:00:14, Serial0
O E2 192.168.10.0 [110/2000] via 172.16.15.250, 00:00:14, Serial0
     192.168.203.13 is variably subnetted, 2 subnets, 2 masks
O       192.168.203.250 255.255.255.255
           [110/75] via 172.16.15.250, 00:00:15, Serial0
O E2    192.168.203.13 255.255.255.0
           [110/2000] via 172.16.15.250, 00:00:15, Serial0
     192.168.250.15 255.255.255.252 is subnetted, 2 subnets
C       172.31.250.8 is directly connected, Loopback1
C       192.168.250.15 is directly connected, Serial0
O    192.168.250.14 [110/74] via 172.16.15.250, 00:00:15, Serial0
     172.31.10.0 is variably subnetted, 2 subnets, 2 masks
O E2    172.31.10.0 255.255.0.0 [110/2000] via 172.16.15.250,
00:00:15,Serial0
O       192.168.213.63 255.255.255.252
           [110/138] via 172.16.15.250, 00:00:16, Serial0
```

Die BGP-Einträge wurden entfernt, da OSPF eine größere Distanz als iBGP aufweist. Die OSPF-Distanz beträgt 110 und die iBGP-Distanz 200.

Deaktivieren Sie die Synchronisierung auf RTA, damit RTA 192.168.250.15 ankündigen kann. Diese Aktion ist erforderlich, da RTA aufgrund der unterschiedlichen Masken nicht mit OSPF synchronisiert wird. Deaktivieren Sie die Synchronisierung für RTB, damit RTB die Adresse 192.168.203.13 angeben kann. Diese Maßnahme ist für RTB aus dem gleichen Grund erforderlich.

Rufen Sie nun die RTB s1-Schnittstelle auf, um zu sehen, wie die Routen aussehen. Aktivieren Sie außerdem OSPF für serielle 1 von RTB, um sie passiv zu machen. Mit diesem Schritt kann die RTA über IGP den nächsten Hop 192.168.10.5 ermitteln. Wenn Sie diesen Schritt nicht ausführen, treten Routing-Schleifen auf, da Sie, um den nächsten Hop 192.168.10.5 zu erreichen, über eBGP in die andere Richtung gehen müssen. Dies sind die neuen Konfigurationen von RTA und RTB:

```
RTA#
hostname RTA

ip subnet-zero

interface Loopback0
 ip address 192.168.203.250 255.255.255.0

interface Ethernet0
```

```
ip address 192.168.203.14 255.255.255.0

interface Serial0
 ip address 192.168.128.63 255.255.255.252

router ospf 10
 redistribute bgp 100 metric 2000 subnets
 passive-interface Serial0
 network 192.168.203.25 0.0.255.255 area 0
 network 172.31.10.0 0.0.255.255 area 0

router bgp 100
 no synchronization
 network 192.168.203.13
 network 192.168.250.14
 neighbor 172.31.63.250 remote-as 200
 neighbor 192.168.250.2 remote-as 100
 neighbor 192.168.250.2 update-source Loopback0

RTB#
hostname RTB

ip subnet-zero

interface Serial0
 ip address 192.168.250.2 255.255.255.252

interface Serial1
 ip address 192.168.10.6 255.255.255.252

router ospf 10
 redistribute bgp 100 metric 1000 subnets
 passive-interface Serial1
 network 192.168.203.25 0.0.255.255 area 0
 network 192.168.208.0 0.0.255.255 area 0

router bgp 100
 no synchronization
 network 192.168.250.15
 neighbor 192.168.10.5 remote-as 300
 neighbor 192.168.203.250 remote-as 100
```

Die BGP-Tabellen sehen wie folgt aus:

<#root>

```
RTA#
```

show ip bgp

```
BGP table version is 117, local router ID is 192.168.203.250
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
*> 172.31.10.0      172.31.63.250           0             0 200 i
*>i192.168.10.0     192.168.10.5            0    100      0 300 i
*>i192.168.211.10     192.168.10.5                100      0 300 500 i
*                   172.31.63.250                         0 200 400 500 i
*> 192.168.10.10     172.31.63.250                        0 200 400 i
*> 192.168.203.13     0.0.0.0                 0         32768 i
*> 192.168.250.14     0.0.0.0                 0         32768 i
*>i192.168.250.15     192.168.250.2             0    100      0 i

RTB#
```

show ip bgp

```
BGP table version is 12, local router ID is 172.16.15.2500
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
*>i172.31.10.0      172.31.63.250           0    100      0 200 i
*                   192.168.10.5                          0 300 500 400
200 i
*> 192.168.10.0     192.168.10.5            0             0 300 i
*> 192.168.211.10     192.168.10.5                          0 300 500 i
*>i192.168.10.10     172.31.63.250                100      0 200 400 i
*                   192.168.10.5                          0 300 500 400 i
*>i192.168.203.13     192.168.203.250           0    100      0 i
*>i192.168.250.14     192.168.203.250           0    100      0 i
*> 192.168.250.15     0.0.0.0                 0         32768 i
```

Es gibt mehrere Möglichkeiten, Ihr Netzwerk für die Kommunikation mit den beiden verschiedenen ISPs, AS200 und AS300, zu konzipieren. Eine Möglichkeit besteht darin, einen primären ISP und einen Backup-ISP zu haben. Sie können partielle Routen von einem der ISPs und Standardrouten zu beiden ISPs abrufen. In diesem Beispiel erhalten Sie partielle Routen von AS200 und nur lokale Routen von AS300. Sowohl RTA als auch RTB generieren Standard-Routen in OSPF, wobei aufgrund der niedrigeren Kennzahl RTB als Präferenz gilt. Auf diese Weise können Sie den ausgehenden Datenverkehr zwischen den beiden ISPs ausgleichen.

Eine potenzielle Asymmetrie kann auftreten, wenn der Datenverkehr, der die RTA verlässt, über die RTB zurückkehrt. Diese Situation kann auftreten, wenn Sie den gleichen Pool von IP-Adressen verwenden, das gleiche große Netz, wenn Sie mit den beiden ISPs sprechen. Aufgrund der Aggregation kann Ihr gesamtes AS nach außen wie eine Einheit aussehen. Eintrittspunkte in Ihr Netzwerk können über RTA oder RTB erfolgen. Sie können feststellen, dass der gesamte eingehende Datenverkehr zu Ihrem AS über einen einzigen Punkt eingeht, obwohl Sie über mehrere Punkte im Internet verfügen. Im Beispiel haben Sie zwei verschiedene Hauptnetze, wenn Sie mit den beiden ISPs sprechen.

Ein weiterer potenzieller Grund für die Asymmetrie ist die unterschiedlich angekündigte Pfadlänge, um Ihr AS zu erreichen. Möglicherweise ist

ein Service Provider einem bestimmten Ziel näher als ein anderer. Im Beispiel wird Datenverkehr von AS400, der Ihr Netzwerk als Ziel hat, aufgrund des kürzeren Pfads immer über RTA empfangen. Du kannst versuchen, diese Entscheidung zu treffen. Sie können den Befehl **set as-path prepend** verwenden, um den Updates Pfadnummern vorzustellen und die Pfadlänge zu verlängern. Mit Attributen wie lokaler Präferenz, Metrik oder Gewicht kann AS400 den Austrittspunkt jedoch auf AS200 gesetzt haben. In diesem Fall gibt es nichts, was du tun kannst.

Diese Konfiguration ist die Endkonfiguration für alle Router:

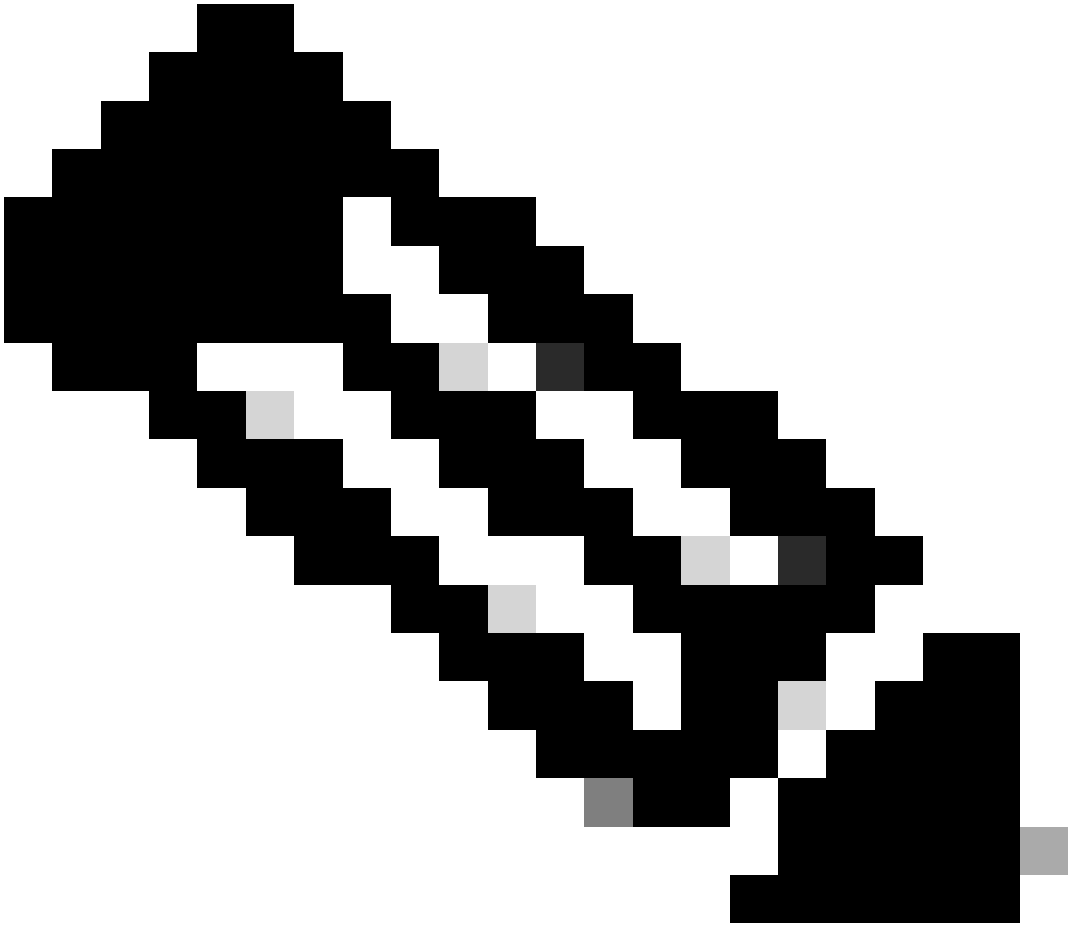
```
RTA#
hostname RTA

ip subnet-zero

interface Loopback0
 ip address 192.168.203.250 255.255.255.0

interface Ethernet0
 ip address 192.168.203.14 255.255.255.0

interface Serial0
 ip address 192.168.128.63 255.255.255.252

router ospf 10
 redistribute bgp 100 metric 2000 subnets
 passive-interface Serial0
 network 192.168.203.25 0.0.255.255 area 0
 network 172.31.10.0 0.0.255.255 area 0
 default-information originate metric 2000

router bgp 100
 no synchronization
 network 192.168.203.13
 network 192.168.250.14
 neighbor 172.31.63.250 remote-as 200
 neighbor 172.31.63.250 route-map setlocalpref in
 neighbor 192.168.250.2 remote-as 100
 neighbor 192.168.250.2 update-source Loopback0

ip classless
ip default-network 172.31.200.200

route-map setlocalpref permit 10
 set local-preference 200
```

Auf der RTA wird die lokale Präferenz für Routen, die vom AS200 ausgehen, auf 200 festgelegt. Darüber hinaus wird network 172.31.200.200 als mögliche Standardeinstellung ausgewählt. Mit dem Befehl **ip default-network** können Sie den Standardwert auswählen.

In diesem Beispiel wird bei Verwendung des Befehls **default-information originate** mit OSPF die Standardroute in die OSPF-Domäne eingefügt. In diesem Beispiel wird dieser Befehl auch mit dem Intermediate System-to-Intermediate System Protocol (IS-IS-Protokoll) und BGP verwendet. Für RIP wird ohne zusätzliche Konfiguration eine automatische Neuverteilung in RIP von 0.0.0.0 vorgenommen. Bei IGRP und EIGRP werden die Standardinformationen nach der Neuverteilung von BGP in IGRP und EIGRP in die IGP-Domäne eingefügt. Mit IGRP und EIGRP können Sie außerdem eine statische Route zu 0.0.0.0 in die IGP-Domäne umverteilen.

```
RTF#
hostname RTF

ip subnet-zero

interface Ethernet0
 ip address 172.31.14.250 255.255.255.0

interface Serial1
 ip address 172.16.15.250 255.255.255.252

router ospf 10
 network 192.168.203.25 0.0.255.255 area 0

ip classless

RTB#
hostname RTB

ip subnet-zero

interface Loopback1
 ip address 172.16.15.2500 255.255.255.252

interface Serial0
 ip address 192.168.250.2 255.255.255.252
!
interface Serial1
 ip address 192.168.10.6 255.255.255.252

router ospf 10
 redistribute bgp 100 metric 1000 subnets
 passive-interface Serial1
 network 192.168.203.25 0.0.255.255 area 0
 network 192.168.10.6 0.0.0.0 area 0
 default-information originate metric 1000
!
router bgp 100
 no synchronization
 network 192.168.250.15
 neighbor 192.168.10.5 remote-as 300
 neighbor 192.168.10.5 route-map localonly in
 neighbor 192.168.203.250 remote-as 100
!
ip classless
ip default-network 192.168.10.0
ip as-path access-list 1 permit ^300$

route-map localonly permit 10
 match as-path 1
 set local-preference 300
```

Für RTB ist die lokale Präferenz für Updates, die von AS300 stammen, auf 300 festgelegt. Dieser Wert ist höher als der Wert für die lokale Voreinstellung von iBGP-Updates, die von der RTA bereitgestellt werden. Auf diese Weise wählt AS100 die RTB für die lokalen Routen von AS300 aus. Alle anderen Routen auf der RTB, sofern andere Routen vorhanden sind, werden intern mit einer lokalen Präferenz von 100 übertragen. Dieser Wert ist niedriger als die lokale Präferenz von 200, die von der RTA abgeleitet wird. RTA ist die Präferenz.



Hinweis: Sie haben nur die lokalen AS300-Routen angekündigt. Alle Pfadinformationen, die nicht mit ^300$ übereinstimmen, werden gelöscht. Wenn Sie die lokalen Routen und die benachbarten Routen, die Kunden des ISP sind, ankündigen möchten, verwenden Sie ^300_[0-9]*.

Die Ausgabe des regulären Ausdrucks gibt die lokalen AS300-Routen an:

<#root>

RTB#

show ip bgp regexp ^300$

```
BGP table version is 14, local router ID is 172.16.15.2500
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
*> 192.168.10.0     192.168.10.5           0    300      0 300
```

```
RTC#
hostname RTC

ip subnet-zero

interface Loopback0
 ip address 192.168.128.6330 255.255.255.192

interface Serial2/0
 ip address 172.16.31.5 255.255.255.252
!
interface Serial2/1
 ip address 172.31.63.250 255.255.255.252

router bgp 200
 network 172.31.10.0
 neighbor 192.168.128.63 remote-as 100
 neighbor 192.168.128.63 distribute-list 1 out
 neighbor 172.31.63.213 remote-as 400

ip classless
access-list 1 deny   192.168.211.0 0.0.255.255
access-list 1 permit any
```

Unter RTC aggregieren Sie 172.31.10.0/16 und geben die spezifischen Routen für die Injektion in AS100 an. Wenn der ISP sich weigert, diese Aufgabe auszuführen, müssen Sie am eingehenden Ende von AS100 filtern.

```
RTD#
hostname RTD

ip subnet-zero

interface Loopback0
 ip address 192.168.208.174 255.255.255.192
!
interface Serial0/0
 ip address 192.168.10.5 255.255.255.252
!
interface Serial0/1
 ip address 192.168.10.2 255.255.255.252
```

```
router bgp 300
 network 192.168.10.0
 neighbor 192.168.10.1 remote-as 500
 neighbor 192.168.10.6 remote-as 100

RTG#
hostname RTG

ip subnet-zero

interface Loopback0
 ip address 192.168.211.19574 255.255.255.192

interface Serial0
 ip address 192.168.10.1 255.255.255.252

interface Serial1
 ip address 192.168.211.195 255.255.255.252

router bgp 500
 network 192.168.211.10
 aggregate-address 192.168.211.0 255.255.0.0 summary-only
 neighbor 192.168.10.2 remote-as 300
 neighbor 192.168.10.2 send-community
 neighbor 192.168.10.2 route-map setcommunity out
 neighbor 192.168.195.211 remote-as 400
!
ip classless
access-list 1 permit 192.168.211.0 0.0.255.255
access-list 2 permit any
route-map setcommunity permit 20
 match ip address 2
!
route-map setcommunity permit 10
 match ip address 1
 set community no-export
```

Eine Demonstration der Verwendung von Community-Filterung befindet sich auf RTG. Sie fügen eine no-export Community zu 192.168.211.0-Updates für RTD hinzu. Auf diese Weise exportiert RTD diese Route nicht an RTB. In diesem Fall akzeptiert die RTB diese Routen jedoch ohnehin nicht.

```
RTE#
hostname RTE

ip subnet-zero

interface Loopback0
 ip address 192.168.200.10 255.255.255.0

interface Serial0
 ip address 192.168.195.211 255.255.255.252

interface Serial1
 ip address 172.31.63.213 255.255.255.252
```

```
router bgp 400
 network 192.168.10.10
 aggregate-address 172.31.200.200 255.255.0.0 summary-only
 neighbor 172.16.31.5 remote-as 200
 neighbor 192.168.211.195 remote-as 500

ip classless
```

RTE aggregiert 172.31.200.200/16. Nachfolgend sind die endgültigen BGP- und Routing-Tabellen für RTA, RTF und RTB aufgeführt:

<#root>

```
RTA#
```

show ip bgp

```
BGP table version is 21, local router ID is 192.168.203.250
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
*> 172.31.10.0      172.31.63.250           0    200      0 200 i
*>i192.168.10.0     192.168.10.5            0    300      0 300 i
*> 172.31.200.200/16   172.31.63.250               200      0 200 400 i
*> 192.168.203.13     0.0.0.0                  0         32768 i
*> 192.168.250.14     0.0.0.0                  0         32768 i
*>i192.168.250.15     192.168.250.2             0    100      0 i

RTA#
```

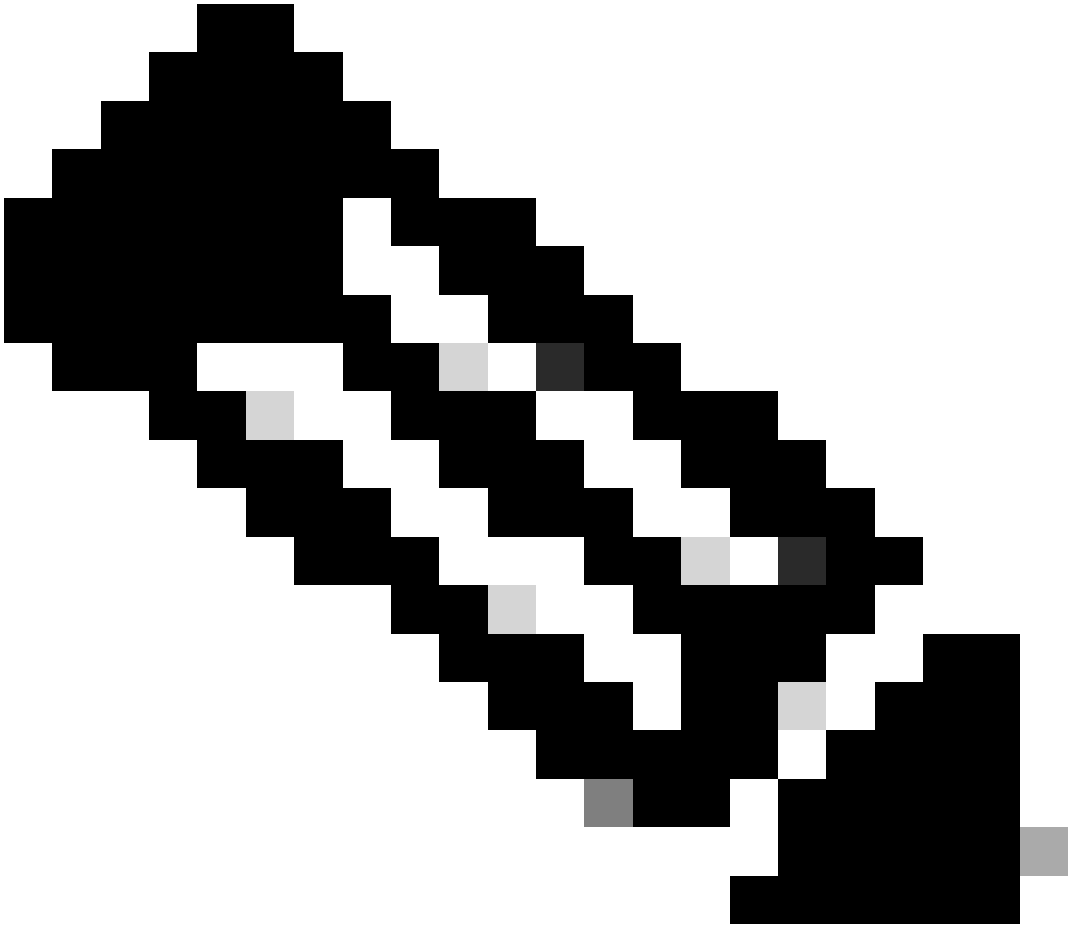
show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is 172.31.63.250 to network 172.31.200.200
```

```
     192.168.10.0 is variably subnetted, 2 subnets, 2 masks
O E2    192.168.10.0 255.255.255.0
           [110/1000] via 172.31.14.250, 00:41:25, Ethernet0
O       192.168.10.4 255.255.255.252
           [110/138] via 172.31.14.250, 00:41:25, Ethernet0
C    192.168.203.13 is directly connected, Loopback0
     192.168.250.15 is variably subnetted, 3 subnets, 3 masks
O       172.16.15.2500 255.255.255.255
           [110/75] via 172.31.14.250, 00:41:25, Ethernet0
O       192.168.250.15 255.255.255.252
           [110/74] via 172.31.14.250, 00:41:25, Ethernet0
B       192.168.250.15 255.255.255.0 [200/0] via 192.168.250.2, 00:41:25
C    192.168.250.14 is directly connected, Ethernet0
     172.31.10.0 is variably subnetted, 2 subnets, 2 masks
B       172.31.10.0 255.255.0.0 [20/0] via 172.31.63.250, 00:41:26
C       192.168.213.63 255.255.255.252 is directly connected, Serial0
O*E2 0.0.0.0/0 [110/1000] via 172.31.14.250, Ethernet0/0
B*   172.31.200.200 255.255.0.0 [20/0] via 172.31.63.250, 00:02:38

RTF#
```

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is 192.168.250.2 to network 0.0.0.0

     192.168.10.0 is variably subnetted, 2 subnets, 2 masks
O E2    192.168.10.0 255.255.255.0
           [110/1000] via 192.168.250.2, 00:48:50, Serial1
O       192.168.10.4 255.255.255.252
           [110/128] via 192.168.250.2, 01:12:09, Serial1
     192.168.203.13 is variably subnetted, 2 subnets, 2 masks
O       192.168.203.250 255.255.255.255
           [110/11] via 192.168.203.14, 01:12:09, Ethernet0
O E2    192.168.203.13 255.255.255.0
           [110/2000] via 192.168.203.14, 01:12:09, Ethernet0
     192.168.250.15 is variably subnetted, 2 subnets, 2 masks
O       172.16.15.2500 255.255.255.255
           [110/65] via 192.168.250.2, 01:12:09, Serial1
C       192.168.250.15 255.255.255.252 is directly connected, Serial1
C    192.168.250.14 is directly connected, Ethernet0
     172.31.10.0 is variably subnetted, 2 subnets, 2 masks
O E2    172.31.10.0 255.255.0.0
           [110/2000] via 192.168.203.14, 00:45:01, Ethernet0
O       192.168.213.63 255.255.255.252
           [110/74] via 192.168.203.14, 01:12:11, Ethernet0
O E2 172.31.200.200 255.255.0.0 [110/2000] via 192.168.203.14, 00:03:47, Ethernet0
O*E2 0.0.0.0 0.0.0.0 [110/1000] via 192.168.250.2, 00:03:33, Serial1
```



Hinweis: Die RTF-Routing-Tabelle gibt an, dass die Verbindung zu lokalen Netzwerken des AS300, z. B. 192.168.10.0, über RTB erfolgt. Der Weg zu anderen bekannten Netzwerken, wie 172.31.200.200, führt über RTA. Als Gateway of Last Resort wird RTB festgelegt. Wenn etwas mit der Verbindung zwischen RTB und RTD passiert, wird die von der RTA angekündigte Standardeinstellung mit einer Metrik von 2000 gesetzt.

<#root>

RTB#

show ip bgp

```
BGP table version is 14, local router ID is 172.16.15.2500
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
*>i172.31.10.0      172.31.63.250           0    200      0 200 i
*> 192.168.10.0     192.168.10.5            0    300      0 300 i
*>i172.31.200.200/16   172.31.63.250               200      0 200 400 i
*>i192.168.203.13     192.168.203.250          0    100      0 i
*>i192.168.250.14     192.168.203.250          0    100      0 i
*> 192.168.250.15     0.0.0.0                  0         32768 i

RTB#
```

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default

Gateway of last resort is 192.168.10.5 to network 192.168.10.0

 *   192.168.10.0 is variably subnetted, 2 subnets, 2 masks
B*      192.168.10.0 255.255.255.0 [20/0] via 192.168.10.5, 00:50:46
C       192.168.10.4 255.255.255.252 is directly connected, Serial1
     192.168.203.13 is variably subnetted, 2 subnets, 2 masks
O       192.168.203.250 255.255.255.255
           [110/75] via 172.16.15.250, 01:20:33, Serial0
O E2    192.168.203.13 255.255.255.0
           [110/2000] via 172.16.15.250, 01:15:40, Serial0
     192.168.250.15 255.255.255.252 is subnetted, 2 subnets
C       172.31.250.8 is directly connected, Loopback1
C       192.168.250.15 is directly connected, Serial0
O    192.168.250.14 [110/74] via 172.16.15.250, 01:20:33, Serial0
     172.31.10.0 is variably subnetted, 2 subnets, 2 masks
O E2    172.31.10.0 255.255.0.0 [110/2000] via 172.16.15.250, 00:46:55, Serial0
O       192.168.213.63 255.255.255.252
           [110/138] via 172.16.15.250, 01:20:34, Serial0
O*E2 0.0.0.0/0 [110/2000] via 172.16.15.250, 00:08:33, Serial0
O E2 172.31.200.200 255.255.0.0 [110/2000] via 172.16.15.250, 00:05:42, Serial0
```

Zugehörige Informationen

- **BGP: Häufig gestellte Fragen**
- **Beispielkonfigurationen für BGP über eine PIX-Firewall**
- **So verwenden Sie HSRP zur Bereitstellung von Redundanz in einem Multi-Homed-BGP-Netzwerk**
- **Konfigurieren der Redundanz für den Single-Router-Modus und des BGP auf einer Catalyst 6000 MSFC**
- **Optimales Routing und reduzierte BGP-Speicherauslastung**
- **Fehlerbehebung bei gängigen BGP-Problemen**
- **Fehlerbehebung bei hoher CPU-Auslastung durch den BGP-Scanner- oder Router-Prozess**
- **Analyse der Lastverteilung mit BGP in Single- und Multihomed-Umgebungen**
- **BGP-Unterstützungsseite**
- **Technischer Support und Downloads von Cisco**

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.