Kapitel 2

Border Gateway-Protokoll 4 – Einleitung

Border Gateway-Protokoll (BGP) ist ein besonders wichtiges Thema für alle CCIEs und Sie können erwarten, dass Ihr Wissen über BGP in der Prüfung gründlich abgefragt wird.

In Kapitel 1, »Exterior Gateway-Protokoll,« haben Sie bereits gelesen, wie die Entwickler des ARPANET in den frühen 1980er Jahren erkannten, dass Autonome Systeme und ein inter-AS-Protokoll nötig sein würden, um das Internet zukünftig zu unterstützen. Ihre ursprüngliche Lösung, das Exterior Gateway-Protokoll (EGP), war für das auf Backbones ausgelegte ARPANET nützlich, man merkte aber bald, dass eine stärker vernetzte inter-AS-Topologie doch wesentlich zukunftsträchtiger sein würde. Man merkte außerdem, dass EGP auf Grund der Loop-Probleme, der Langsamkeit und der fehlenden Umsetzungsmöglichkeiten für Routing-Richtlinien nicht dazu fähig ist, eine solche Umgebung zu unterstützen.

Es gab Versuche, EGP zu verbessern, schließlich wurde dann aber doch ein neues inter-AS-Protokoll, ein wahres Routing-Protokoll, geschaffen. Dieses inter-AS Routing-Protokoll, das zuerst 1989 in RFC 1105[1] erschien, ist BGP. Die erste Version von BGP wurde genau ein Jahr später in RFC 1163[2] aktualisiert. BGP wurde 1991 in RFC 1267[3] ein weiteres Mal überholt und seit diesem Update werden die drei Versionen BGP-1, BGP-2 und BGP-3 genannt.

Die heutige Version BGP-4 wurde 1995 in RFC 1771[4] eingeführt. BGP-4 unterscheidet sich merklich von den anderen Versionen. Der wichtigste Unterschied ist, dass BGP-4 classless ist, während die früheren Versionen noch classfull waren. Der Grund für diese große Veränderung ist derselbe, aus dem externe Gateway-Protokolle überhaupt existieren: Routing innerhalb des Internets muss sowohl überschaubar als auch zuverlässig bleiben. Classless Interdomain Routing (CIDR) – ursprünglich 1993 in RFC 1517[5] eingeführt und in RFC 1519[6] komplettiert, wurde in RFC 1520[7] noch einmal umgestellt und existiert ebenfalls aus demselben Grund. BGP-4 wurde erfunden, um CIDR zu unterstützen.

2.1 Classless Interdomain Routing

Die Erfindung des Autonomen Systems und des externen Routing-Protokolls löste die frühen Skalierungsprobleme des Internets in den 80er Jahren. In den frühen 1990er Jahren kamen aber ganz andere Skalierungssprobleme auf das Internet zu:

– Explosion der Internet-Routing-Tabellen. Die exponentiell wachsenden Routing-Tabellen wurden immer mehr zum Problem für die Router und die Leute. die sie verwalteten. Die Größe allein war schon Last genug, hinzu kamen jedoch noch die täglichen Topologieveränderungen und Unsicherheiten.

– Erschöpfung der Class B-Adressen. Januar 1993 waren 7133 der 16.382 möglichen Class B-Adressen vergeben; die damalige Zuwachsrate bedeutete, dass nach einer Prognose in RFC 1519 alle Class B-Adressen innerhalb von zwei Jahren vergeben sein würden.

– Die Erschöpfung der 32-Bit-IP-Adressen.

Classless Interdomain Routing bietet eine schnelle Lösung der ersten beiden Probleme an. Eine weitere schnelle Lösung ist Network Address Translation (NAT). Dies wird in Kapitel 4, »Network Address Translation«, weiter besprochen. Diese Schnelllösungen sollten den Internetexperten genug Zeit verschaffen, eine neue IP-Version zu erschaffen, in der es genügend Adressen für die Zukunft gibt. Diese Initiative, genannt IP Next Generation (IPng), führte zu IPv6 mit einem Adressformat von 128 Bit. IPv6, in Kapitel 8, »IP-Version 6,« genauer beschrieben, ist die langfristige Lösung des dritten Problems. Interessanterweise sind CIDR und NAT so erfolgreich geworden ist, dass viele Leute IPv6 nicht mehr für so wichtig halten wie ursprünglich vorgesehen.

CIDR ist nichts weiter als ein strategisch eingeschränktes System zur Adressen-Zusammenfassung, das sich die hierarchische Struktur des Internets zu Nutzen macht. Bevor also CIDR weiter besprochen wird, sollten kurz Zusammenfassung und classless Routing sowie das moderne Internet besprochen werden.

2.1.1 Eine »Zusammenfassungs-Zusammenfassung« (Summerization)

Zusammenfassung oder *Routen-Verdichtung* (beides wurde in *Routing TCP/IP, Band I* besprochen) geschieht, wenn eine Gruppe von benachbarten Adressen als eine weniger spezifische Adresse veröffentlicht wird. Zusam-

menfassung/Routen-Verdichtung wird erreicht, indem die Länge der Subnet-Maske reduziert wird, bis diese nur noch die Bits verdeckt, die alle zusammengefassten Adressen gemeinsam haben. In Abbildung 2.1 zum Beispiel sind die vier Subnets (172.16.100.192/28, 172.16.100.208/28, 172.16.100.224/28 und 172.16.100.240/28) zusammengefasst durch die Adresse 172.16.100.192/26.

Viele Netzwerker halten die Zusammenfassung für ein schwieriges Thema und sind sehr überrascht, wenn sie erfahren, das sie jeden Tag Zusammenfassungen verwenden. Eine Subnet-Adresse ist schließlich auch nichts anderes als eine Zusammenfassung einer Gruppe benachbarter Hostadressen. Die Subnet-Adresse 192.168.5.224/27 ist zum Beispiel die Zusammenfassung der Hostadressen 192.168.5.224/32 bis 192.168.5.255/32 (die »Hostadresse« 192.168.5.224/32 ist natürlich die Adresse des Data Links selbst). Die Haupteigenschaft einer Zusammenfassungs-Adresse ist, dass ihre Maske kürzer ist als die der Adressen, die sie zusammenfasst. Die ultimative Zusammenfassungs-Adresse ist die default Adresse, 0.0.0.0/0, die normalerweise nur als 0/0 geschrieben wird. Die /0 zeigt, dass die Maske so sehr geschrumpft ist, dass keine Netzwerk-Bits mehr übrig sind – die Adresse ist die Zusammenfassung aller IP-Adressen.

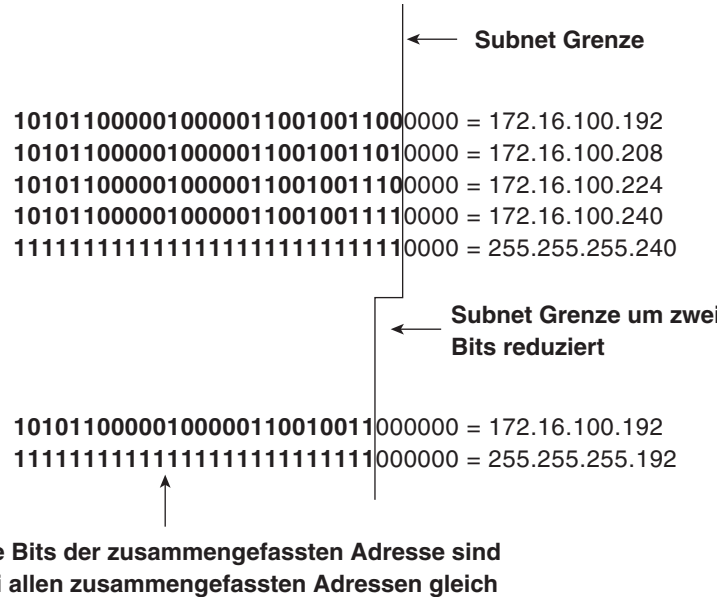

Abb. 2.1: Route Aggregation

Zusammenfassungen können auch über Class-Grenzen hinweggehen. Die vier Class C-Netzwerke (192.168.0.0, 192.168.1.0, 192.168.2.0 und 192.168.3.0) können zum Beispiel alle mit Adresse 192.168.0.0/22 zusammengefasst werden. Die zusammengefasste Adresse mit einer 22-Bit-Maske ist keine Class C-Adresse mehr. Aus diesem Grund muss es für die Zusammenfassung von major Class-Netzwerkadressen eine classless Umgebung geben.

2.1.2 Classless Routing

Classless Routing hat zwei Aspekte:

– Classlessness kann ein Kennzeichen eines Routing-Protokolls sein.

– Classlessness kann ein Kennzeichen eines Routers sein.

Classless Routing-Protokolle tragen als Teil der Routing-Information eine Beschreibung des Netzwerkteils jeder veröffentlichten Adresse. Der Netzwerkteil einer Netzwerkadresse wird oft *Adressen-Präfix* genannt. Ein Adressen-Präfix kann durch eine Adressenmaske beschrieben werden, ein etwas längeres Feld, das anzeigt, wie viele Bits zum Adressen-Präfix gehören, oder es kann definiert werden, indem nur die Präfix-Bits in das Update kommen (siehe Abbildung 2.2). Die classless IP Routing-Protokolle sind RIP-2, EIGRP, OSPF, Integrated IS-IS, und BGP-4.

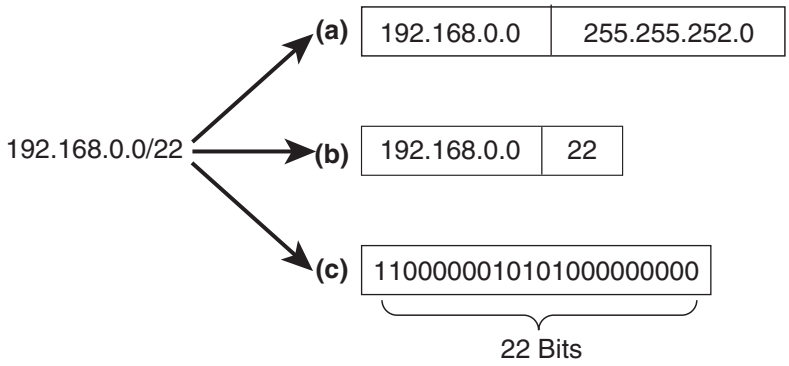

Abb. 2.2: Angabe eines Adress-Präfixes mit einem classless Routing-Protokoll

Ein classfull Router speichert Zieladressen in seiner Routing-Tabelle als major Class-Netzwerke und Subnets dieser Netzwerke. Wenn eine Route nachgeschlagen wird, sucht der Router zuerst bei den major Class-Netzwerkadressen und versucht dann einen entsprechenden Eintrag in der Liste von Subnets unter dieser Adresse zu finden. Ein classless Router ignoriert Adres-

senklassen und versucht nur die »längste Übereinstimmung« zu finden. Das heißt er sucht für eine Zieladresse die Route, die mit den meisten Bits der Adresse übereinstimmt. Die Routing-Tabelle von Beispiel 2.1 zeigt mehrere IP-Netzwerke in unterschiedlichen Subnets. Wenn ein Router classless ist, versucht er die längste Übereinstimmung für jede Zieladresse zu finden.

Beispiel 2.1: Eine Routing-Tabelle, die einige variable subnetted IP-Netzwerke enthält

```
Cleveland#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 192.168.2.130 to network 0.0.0.0

O  E2 192.168.125.0 [110/20] via 192.168.2.2, 00:11:19, Ethernet0
O     192.168.75.0 [110/74] via 192.168.2.130, 00:11:19, Serial0
O  E2 192.168.8.0 [110/40] via 192.168.2.18, 00:11:19, Ethernet1
      192.168.1.0 is variably subnetted, 3 subnets, 3 masks
O E1     192.168.1.64 255.255.255.192
            [110/139] via 192.168.2.134, 00:11:20, Serial1
O E1     192.168.1.0 255.255.255.128
            [110/139] via 192.168.2.134, 00:00:34, Serial1
O E2     192.168.1.0 255.255.255.0
            [110/20] via 192.168.2.2, 00:11:20, Ethernet0
      192.168.2.0 is variably subnetted, 4 subnets, 2 masks
C        192.168.2.0 255.255.255.240 is directly connected, Ethernet0
C        192.168.2.16 255.255.255.240 is directly connected, Ethernet1
C        192.168.2.128 255.255.255.252 is directly connected, Serial0
C        192.168.2.132 255.255.255.252 is directly connected, Serial1
O E2 192.168.225.0 [110/20] via 192.168.2.2, 00:11:20, Ethernet0
O E2 192.168.230.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O E2 192.168.198.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O E2 192.168.215.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O E2 192.168.129.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O E2 192.168.131.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O E2 192.168.135.0 [110/20] via 192.168.2.2, 00:11:21, Ethernet0
O*E2 0.0.0.0 0.0.0.0 [110/1] via 192.168.2.130, 00:11:21, Serial0
O E2 192.168.0.0 255.255.0.0 [110/40] via 192.168.2.18, 00:11:22, Ethernet1
Cleveland#
```

Wenn der Router ein Paket mit einer Zieladresse von 192.168.1.75 bekommt, decken sich mehrere Einträge in der Routing-Tabelle mit der Adresse: 192.168.0.0/16, 192.168.1.0/24, 192.168.1.0/25 und 192.168.1.64/26. Der Eintrag 192.168.1.64/26 wird ausgewählt (siehe Beispiel 2.2), da er mit 26 Bits der Zieladresse übereinstimmt – die längste Übereinstimmung.

Beispiel 2.2: Ein Packet mit der Zieladresse 192.168.1.75 wird von der Schnittstelle Serial1 gesendet.

```
Cleveland#show ip route 192.168.1.75
Routing entry for 192.168.1.64 255.255.255.192
  Known via "ospf 1", distance 110, metric 139, type extern 1
  Redistributing via ospf 1
  Last update from 192.168.2.134 on Serial1, 06:46:52 ago
  Routing Descriptor Blöcke:
  * 192.168.2.134, from 192.168.7.1, 06:46:52 ago, via Serial1
      Route metric is 139, traffic share count is 1
```

Ein Paket mit einer Zieladresse von 192.168.1.217 stimmt nicht mit 192.168.1.64/26 überein und auch nicht mit 192.168.1.0/25. Die längste Übereinstimmung für diese Adresse ist 192.168.1.0/24, wie in Beispiel 2.3 zu sehen ist.

Beispiel 2.3: Der Router kann 192.168.1.217 nicht mit einem genaueren Subnet abgleichen, er verwendet die Netzwerkadresse 192.168.1.0/24.

```
Cleveland#show ip route 192.168.1.217
Routing entry for 192.168.1.0 255.255.255.0
  Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 10
  Redistributing via ospf 1
  Last update from 192.168.2.2 on Ethernet0, 06:48:18 ago
  Routing Descriptor Blöcke:
  * 192.168.2.2, from 10.2.1.1, 06:48:18 ago, via Ethernet0
      Route metric is 20, traffic share count is 1
```

Die längste Übereinstimmung von Zieladresse 192.168.5.3 ist die zusammengefasste Adresse 192.168.0.0/16, dies ist in Beispiel 2.4 zu sehen.

Beispiel 2.4: Die Pakete für 192.168.5.3 entsprechen keinem genaueren Subnet oder Netzwerk und stimmen deshalb mit Supernet 192.168.0.0/16 überein.

```
Cleveland#show ip route 192.168.5.3
Routing entry for 192.168.0.0 255.255.0.0, supernet
  Known via "ospf 1", distance 110, metric 139, type extern 1
  Redistributing via ospf 1
  Last update from 192.168.2.18 on Ethernet1, 06:49:26 ago
  Routing Descriptor Blöcke:
  * 192.168.2.18, from 192.168.7.1, 06:49:26 ago, via Ethernet1
      Route metric is 139, traffic share count is 1
```

Eine Zieladresse von 192.169.1.1 stimmt mit keinem Eintrag in der Routing-Tabelle überein, wie in Beispiel 2.5 klar zu sehen ist. Pakete mit dieser Zieladresse fallen jedoch nicht heraus, da die Routing-Tabelle in Beispiel 2.1 eine default Route enthält. Das Paket wird an den next Hop Router 192.168.2.130 weitergeleitet.

Beispiel 2.5: Kein Eintrag der Routing-Tabelle stimmt mit 192.169.1.1 überein; Pakete für diese Adresse werden an die default Adresse durch Schnittstelle S0 gesendet.

```
Cleveland#show ip route 192.169.1.1
% Network not in table
```

Seit dem IOS 11.3 sind Cisco-Router in der Grundeinstellung classless. Vor dieser Erscheinung war die IOS-Grundeinstellung classfull. Die Grundeinstellung kann mit dem Befehl **ip classless** geändert werden.

Die Routing-Tabelle in Beispiel 2.1 und die dazugehörigen Beispiele zeigen eine weitere Eigenschaft von classless Routing: Eine Route zu einer Verdichtung zusammengefasster Adressen deutet nicht auf alle Mitglieder der Verdichtung hin. Abbildung 2.3 zeigt die Vektoren der Routen in Beispiel 2.2 bis 2.5.

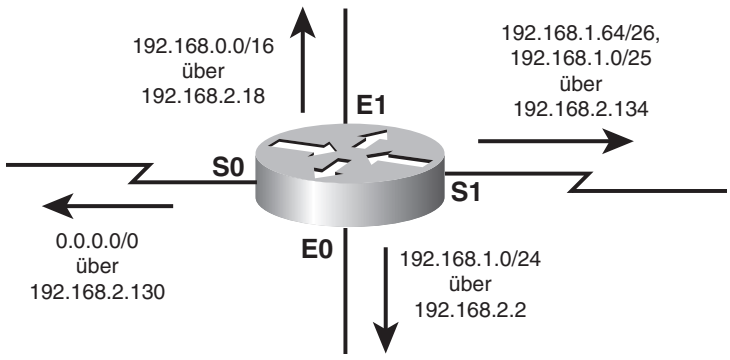

Abb. 2.3: Die Vektoren der Routen in der Routing-Tabelle aus Beispiel 2.1

Das Netzwerk 192.168.1.0/24 kann als Zusammenfassung seiner Subnets gesehen werden; Abbildung 2.3 zeigt, dass die Route zu dieser Netzwerkadresse Pakete durch die Schnittstelle E0 führt. Allerdings führen zwei Routen zu zwei seiner anderen Subnets, 192.168.1.0/25 und 192.168.1.64/26, durch eine andere Schnittstelle, S1.

Anmerkung

192.168.1.64/26 ist selbst ein Mitglied von 192.168.1.0/25. Dass es eigene Routen für beide Adressen gibt, die beide aus S1 herausführen, deutet darauf hin, dass die Routen von einem anderen weiter entfernten Router bekannt gegeben werden.

Genauso ist auch 192.168.1.0/24 ein Mitglied der Zusammenfassung 192.168.0.0/16, deren Route durch E1 verläuft. Die ungenaueste Route, 0.0.0.0/0, die eine Zusammenfassung aller anderen Adressen ist, verläuft durch S0. Wegen der längsten Übereinstimmung werden Pakete für die Subnets 192.168.1.64/26 und 192.168.1.0/25 durch S1 verschickt, während Pakete zu anderen Subnets von Netzwerk 192.168.1.0/24 durch E0 laufen. Pakete, deren Zieladresse mit 192.168 beginnt, werden mit Ausnahme von 192.168.1 durch E1 versendet, während Pakete, deren Zieladresse nicht mit 192.168 beginnt, durch S0 laufen.

2.1.3 Zusammenfassung: Das Gute, das Schlechte und das Asymmetrische

Das Zusammenfassen ist ein sehr nützliches Mittel, um Netzwerk-Ressourcen zu sparen, man braucht viel weniger Speicherplatz zum Speichern der Routing-Tabellen und Bandbreite und Router-Ressourcen, die zum Versand von Routing-Informationen notwendig wären. Die Zusammenfassung von Adressen führt außerdem dazu, dass Instabilitäten im Netzwerk »versteckt« werden.

Das Netzwerk in Abbildung 2.4 hat zum Beispiel eine flapping Route – eine Route, die wegen einer schlechten Verbindung oder Schnittstelle immer wieder den Status zwischen Up und Down wechselt.

Ohne Zusammenfassung müssen bei jeder Statusänderung von Subnet 192.168.1.176/28 alle Router im Netzwerk des Unternehmens informiert werden. Jeder Router muss daraufhin die Informationen verarbeiten und die Routing-Tabelle neu einstellen. Wenn Router Nashville alle upstream Routen über die zusammengefasste Adresse 192.168.1.128/25 informiert, werden Veränderungen in den Subnets der Zusammenfassung von diesem Router nicht weitergegeben. Nashville ist der Verdichtungspunkt und die Verdichtung bleibt stabil, auch wenn einige Mitglieder dies nicht sind.

Der Nachteil einer Zusammenfassung ist eine niedrigere Routing-Genauigkeit. In Beispiel 2.6 funktioniert die Schnittstelle S1 des Routers in Abbildung 2.3 nicht, was dazu führt, dass die an dieser Schnittstelle von

Nachbarn erlernten Routen nicht mehr gültig sind. Statt die Pakete, die normalerweise durch S1 verschickt werden, wie zum Beispiel ein Paket mit Adresse 192.168.1.75, einfach zu ignorieren, wird das Paket nun entlang der nächstbesten Route, 192.168.1.0/24, verschickt und geht durch Schnittstelle E0 (vergleichen Sie mit Beispiel 2.2).

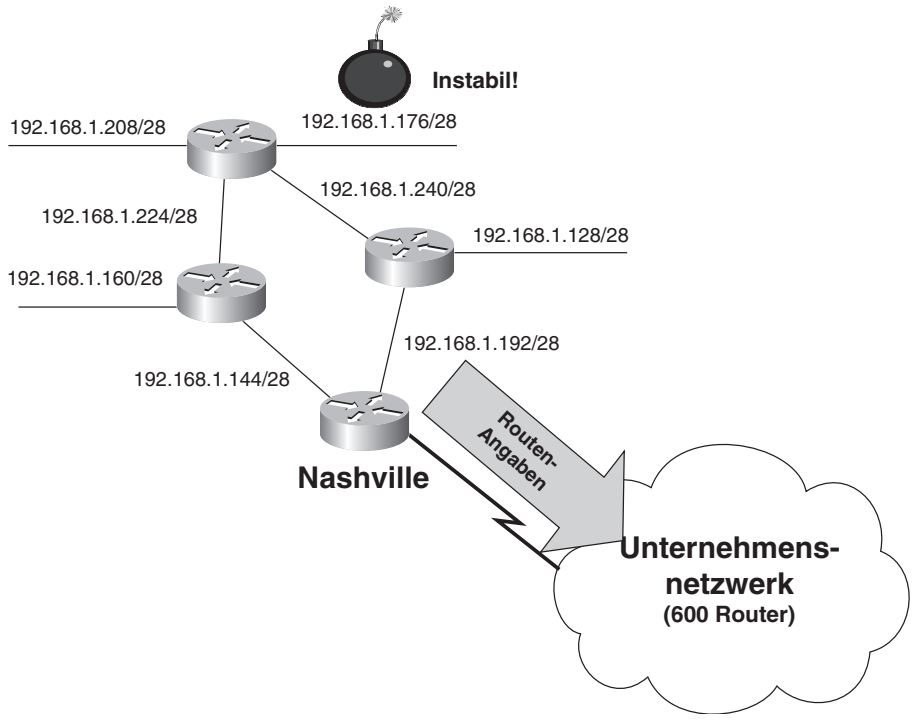

Abb. 2.4: Eine flapping (flatternde) Route kann ein ganzes Netzwerk destabilisieren.

Beispiel 2.6: Eine ausgefallene Route kann zu ungenauem Paketversand führen.

```
Cleveland#
%LINEPROTO-5-UPDOWN: Line-Protokoll on Interface Serial1, changed state to down
%LINK-3-UPDOWN: Interface Serial1, changed state to down
Cleveland#show ip route 192.168.1.75
Routing entry for 192.168.1.0 255.255.255.0
  Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 10
  Redistributing via ospf 1
  Last update from 192.168.2.2 on Ethernet0, 00:00:20 ago
  Routing Descriptor Blöcke:
  * 192.168.2.2, from 10.2.1.1, 00:00:20 ago, via Ethernet0
      Route metric is 20, traffic share count is 1

Cleveland#
```

Diese Ungenauigkeit kann je nach der Struktur des Netzwerks ein Problem darstellen. Nehmen wir beim eben beschriebenen Beispiel an, dass der next Hop Router 192.168.2.2 noch einen Eintrag für die Route zu 192.168.1.64/26 über den Router Cleveland hat, entweder weil das Netzwerk noch nicht konvergiert hat oder weil die Route statisch eingegeben wurde: In diesem Fall entsteht ein Routing Loop. Andererseits kann es sein, dass ein Router, der über die E0-Schnittstelle von Cleveland erreichbar ist, eine »Hintertür«-Route zu Subnet 192.168.1.64/26 hat, die nur dann benutzt wird, wenn die primäre Route über Clevelands S1 ungültig wird. In diesem Fall wurde die Route zu 192.168.1.0/24 als Ersatzroute vorgesehen und der Vorgang in Beispiel 2.6 ist erwünscht.

Abbildung 2.5 zeigt ein Netzwerk, in dem die fehlende Genauigkeit zu einem anderen Problem führen kann. Hier ist die Routing-Domäne 1 durch zwei Router, San Francisco und Atlanta, an Routing-Domäne 2 angeschlossen. Die Definition der Domänen ist für dieses Beispiel unwichtig. Wichtig ist, dass alle Adressen durch die Domäne 1 mit der Adresse 172.16.192.0/18 zusammengefasst werden und dass alle Netzwerke in Domäne 2 durch die Adresse 172.16.128.0/18 zusammengefasst werden.

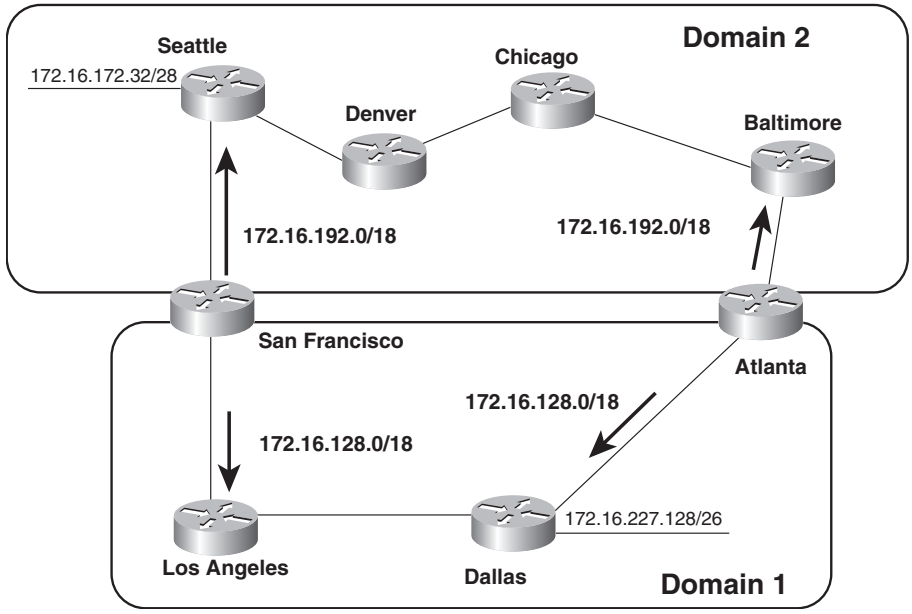

Abb. 2.5: Wenn mehrere Router dieselbe zusammenfassende Adresse bekannt geben, wird der Verlust der Routing-Genauigkeit zum Problem.

Anstatt individuelle Subnets anzugeben, teilen Atlanta und San Francisco den Domänen die zusammenfassenden Adressen mit. Wenn ein Host am Subnet 172.16.227.128/26 von Dallas ein Paket an einen Host an Seattles Subnet 172.16.172.32/28 sendet, wird das Paket höchstwahrscheinlich über Atlanta geroutet, da dies der nächste Router ist, der die zusammengefasste Route von Domäne 2 angibt. Atlanta sendet das Paket an Domäne 2 und es kommt in Seattle an. Wenn der Host an Subnet 172.16.172.32/28 eine Antwort schickt, sendet Seattle das Paket nach San Francisco – dem nächsten Router, der die zusammengefasste Route 172.16.192.0/18 angibt.

Das Problem ist hier, dass der Verkehr zwischen den beiden Subnets nun asymmetrisch geworden ist: Pakete von 172.16.227.128/26 zu dem Subnet 172.16.172.32/28 nehmen einen Weg, während Pakete von 172.16.172.32/28 zu 172.16.227.128/26 einen anderen Weg nehmen. Die Asymmetrie entsteht, weil Dallas und Seattle keine kompletten Routen zu den Subnets des andern besitzen. Sie haben nur Routen zu den Routern, die die Zusammenfassung kennen und müssen ihre Pakete über diese verschicken. Mit anderen Worten hat also die Zusammenfassung bei San Francisco und Atlanta die Details des Netzwerks hinter diesen Routern versteckt.

Asymmetrischer Verkehr kann aus mehreren Gründen unerwünscht sein. Erstens werden die Verkehrsmuster des Netzwerks unberechenbarer, was dazu führt, dass Baselining, Fehlerbeseitigung und das Planen der Kapazität schwieriger werden. Zweitens ist die Benutzung der Links nicht mehr ausgewogen. Die Bandbreite mancher Links kann voll ausgelastet sein, während andere Links zu wenig benutzt bleiben. Drittens kann es einen großen Unterschied zwischen den Verzögerungszeiten bei ankommendem und abgehendem Verkehr geben. Dieser Unterschied kann ein Nachteil für manche verzögerungsempfindlichen Anwendungen wie Voice oder Live-Video werden.

2.1.4 Das Internet: nach all den Jahren immer noch hierarchisch

Obwohl das Internet nicht mehr die Backbone-orientierte Struktur des ARPANETs hat, die in Kapitel 1 beschrieben wurde, bleibt es doch in gewisser Weise ein hierarchisches Netzwerk. Auf dem niedrigsten Level gibt es die Internetbenutzer, die durch einen Internet Service Provider (ISP) an das Internet angeschlossen werden. Meistens ist dieser ISP einer von vielen lokalen Providern in der Region (diese Provider werden *local* ISPs genannt). Es gibt zur Zeit zum Beispiel fast 200 ISPs in Colorados 303 Area Code. Diese lokalen ISPs sind wiederum Kunden von größeren ISPs die sich über eine größere Region ausbreiten, zum Beispiel über einen ganzen Staat oder mehrere nebeneinander liegende Staaten. Diese größeren ISPs werden *regional Ser-*

vice Providers genannt. Beispiele in Colorado sind CSD Internet und Colorado Supernet. Die regional Service Providers sind an große ISPs mit globalen high-speed (DS-3 oder OS-3 oder besser) Backbones angeschlossen. Diese größten Provider sind die *Network Service Provider*, Beispiele sind MCI/WorldCom (UUNET), SprintNet, Cable & Wireless, Concentric Network, PSINet oder auch die Deutsche Telekom. Normalerweise werden diese verschiedenen Arten von Providern als Tier III-, Tier II- und Tier I-Provider bezeichnet.

Abbildung 2.6 zeigt, wie diese verschiedenen Arten von ISPs verwandt sind. Ein Teilnehmer – ob ein Endbenutzer oder ein kleinerer Service Provider – wird mit dem höher eingestuften ISP jeweils immer an dessen ISPs *Point of Presence* (POP) verbunden. Ein POP ist einfach ein nahe gelegener Router, mit dem der Teilnehmer über Dialup oder einen Local Loop verbunden werden kann. In den höheren Schichten sind die Network Service Provider durch *Network Access Points* (NAPs) verbunden. Ein NAP ist ein LAN oder Switch – normalerweise Ethernet, FDDI oder ATM – über den verschiedene Provider Routen und Verkehrsdaten austauschen können.

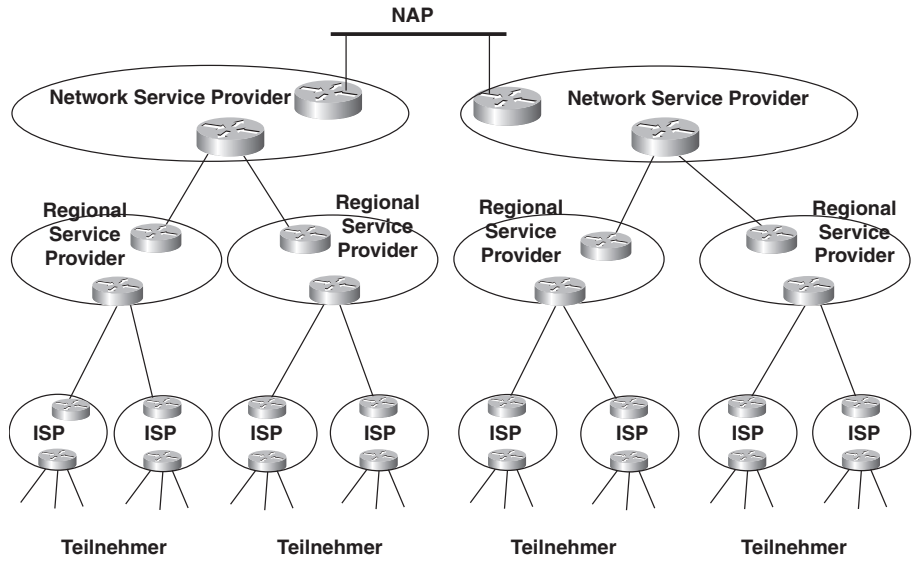

Abb. 2.6: ISP/NAP-Hierarchie

Tabelle 2.1 zeigt, dass manche NAPs Namen wie Commercial Internet Exchange (CIX), Federal Internet Exchange (FIX) und Metropolitan Area Exchange (MAE – ursprünglich Metropolitan Area Ethernets, a Creation of-Metropolitan Fiber Systems, Inc.) haben. CIX, FIX und MAE-East waren

frühe Versuche, um Backbones zu verbinden. Mit der Erfahrung, die durch diese Verbindungspunkte gesammelt wurde, baute die National Science Foundation 1994 die ersten vier NAPs als Teil der Stilllegung des NSFnet.

Tabelle 2.1: Die bekannten Network Access Points in den USA

NAP	Standort	Zugehörigkeit
New York NAP*	Pennsauken, New Jersey	Sprint
Chicago NAP*	Chicago, Illinois	Ameritech and Bellcore
San Francisco NAP*	San Francisco, California	Pacific Bell
Big East NAP	Bohemia, New York	ICS Network Systems
MAE-West	San Jose, California	MCI/WorldCom
MAE-East*	Washington, DC	MCI/WorldCom
MAE-LA	Los Angeles, California	MCI/WorldCom
MAE-Houston	Houston, Texas	MCI/WorldCom
MAE-Dallas	Dallas, Texas	MCI/WorldCom
MAE-New York	New York City, New York	MCI/WorldCom
MAE-Chicago	Chicago, Illinois	MCI/WorldCom
FIX-East	College Park, Maryland	University of Maryland
FIX-West	Moffett Field, California	NASA Ames Research Center
CIX	Santa Clara, California	Wiltel
Digital PAIX	Palo Alto, California	Digital Equipment Corporation

* One of the original four NSF NAPs

Zusätzlich zu den großen NAPs in Tabelle 2.1, wo die NSPs zusammenkommen, gibt es mehrere kleinere NAPs. Diese verbinden normalerweise kleine, regionale Provider miteinander. Beispiele regionaler NAPs sind Seattle Internet eXchange (SIX) und der New Mexico Network Access Point.

Zusammen mit der Formation der NAPs unterstützte die NSF das Routing Arbiter (RA) Project. Eine der Aufgaben des RA ist, die Stabilität und die Umgänglichkeit des Internets zu erhöhen. Aus diesem Grund schlug das RA eine Datenbank vor, (die RADB oder Routing Arbiter Database) die Routen (Topologie) und Richtlinien (bevorzugte Routen) der Service Provider speichert. Die Datenbank wird an NAPs an einem R*oute Server*, an einer UNIX-

Workstation oder an einem Server mit BGP unterhalten. Anstatt mit jedem anderen Router des NAPs benachbart zu sein, ist der Router jedes Providers nur mit dem Route Server benachbart. Routen und Richtlinien werden dem Server mitgeteilt, der eine hochentwickelte Sprache namens RIPE-181 verwendet, um die Informationen zu erhalten und weiterzugeben. Die richtigen Routen werden dann an die anderen Router weitergegeben.

Obwohl der Route Server BGP spricht und die Routen bearbeitet, versendet er keine Pakete. Stattdessen informiert er mit Updates die Router über den besten next Hop Router, der jeweils direkt über den NAP erreichbar ist. Sie kennen diese Art von System schon aus Kapitel 1, wo bei EGP indirekte Nachbarn besprochen wurden. Indem sie one-to-many Peering anstatt many-to-many Peering ermöglichen, erhöhen Route Server die Stabilität, die Fügsamkeit und den Verkehrsdurchsatz des NAPs.

Die NAPs und das RA-Projekt bewiesen, dass die miteinander im Wettbewerb stehenden Service Providers zusammen für die Stabilität und Fügsamkeit des Internets sorgen können. Folglich zog die NSF ihre finanzielle Unterstützung der Route Servers und NAPs am 1. Januar 1997 zurück und überließ das System den kommerziellen Providern. Obwohl die Internetforschung durch öffentlich unterstützte Projekte wie Internet2, GigaPOPs und den very high-speed Backbone Network Service (vBNS) weitergeht, kann das heutige Internet als kommerzielles System angesehen werden.

Auf Grund des Übergangs zum kommerziellen Netz ist die Struktur des Internets heute leider nicht mehr so übersichtlich wie in den letzten Abschnitten beschrieben. Die größten Service Provider peeren zum Beispiel aus verschiedenen finanziellen und wettbewerbsbedingten Gründen lieber direkt als über einen Route Server. Das Peering geschieht außerdem auf mehreren Ebenen und nicht nur in den oberen Regionen wie in Abbildung 2.6.

Wenn zwei oder mehr Service Provider vereinbaren, dass sie Routen über ein NAP teilen, entweder durch Route Server oder direkt, dann gehen sie ein *Peering Agreement* ein. Peering Agreements können direkt zwischen zwei Providern (ein *bilateral* Peering Agreement) oder zwischen einer Gruppe ähnlich großer Provider (ein *multilateral* Peering Agreement oder MLPA) geschlossen werden. Die Verkehrsmuster spielen beim Bestimmen der finanziellen Bedingungen natürlich eine große Rolle. Wenn der Verkehr zwischen den beiden Partnern relativ gleichmäßig in beide Richtungen fließt, dann wechselt meistens kaum Geld die Hände. Das Peering ist dann für beide Partner gerecht. Wenn der Verkehr allerdings viel mehr in eine Richtung fließt als in die andere, was zum Beispiel der Fall ist, wenn ein kleiner Service Provider mit einem größeren peert, dann muss der kleinere Provider norma-

lerweise für diese Unausgewogenheit mit Geld bezahlen. Der Grund hierfür ist, dass der kleine Provider mehr vom Peering hat als der Große.

Ein weiterer Faktor, der das Bild des Internets verwischt, sind die Standorte der Peering Points. NAPs, in denen viele Providers zusammen kommen, wie die in Tabelle 2.1, heißen *public* Peering Sites. Zusätzlich zu diesen Public Sites haben die Service Provider Hunderte von kleineren NAPs erschaffen, die an Orten stehen, an denen zwei Service Provider Strukturen haben. Die Peering Agreements an solchen Orten sind normalerweise *privat*, also zwischen zwei oder einer kleinen Gruppe von Providern. Privates Peering wird gefördert, da es Staus an den nationalen NAPs verhindert, zur Routen-Vielfalt beiträgt und die Verzögerungen für manchen Verkehr verringert.

Des Weiteren wird die Struktur aus Abbildung 2.6 zusätzlich verändert, weil nationale und regionale Service Provider ebenfalls lokalen Internetzugang verkaufen und somit mit den lokalen ISPs Wettbewerb betreiben. Der »Anfangspunkt« der Routen Traces in Beispiel 2.7 ist zum Beispiel ein dial-in POP, der Concentric Network gehört – ein Backbone-Provider. Regionale Service Provider sind ebenfalls oft an den Backbone NAPs vertreten. Sie können sich so mit einem oder mehreren Network Service Providern über den NAP verbinden oder mit anderen regionalen Service Providern Kontakt aufnehmen und dabei Network Service Provider umgehen.

Die Routen Traces in Beispiel 2.7 zeigen einen Teil der Internet Backbone-Struktur. Beide Traces stammen von einem Concentric Network POP in Denver. Im ersten Trace überqueren die Pakete den Backbone von Concentric Networks und gelangen zu MAE-East, wo sie sich mit dem BBN Planet Backbone (Zeilen 3 und 4) verbinden. Die Pakete überqueren BBN Planets Backbone zu einem Tier II NAP, der von BBN und US West in Minneapolis geteilt wird (Zeilen 10 und 11) und kommen dann am Ziel im Westen der USA an.

Beispiel 2.7: Routen Traces von einem Concentric Network POP in Denver

```
--- traceroute to www.uswest.com (205.215.207.54),
 30 hops max, 18 byte packets

  1 (  207.155.168.5)  ts003e01.den-co.concentric.net 174 ms
  2 (  207.155.168.1)  rt001e0102.den-co.concentric.net.168.155.207.IN-ADDR.ARPA 162 ms
  3 (   207.88.24.29)  us-dc-wash-core1-a1-0d12.rtr.concentric.net 385 ms
  4 (   192.41.177.2)  maeeast2.bbnplanet.net 225 ms
  5 (       4.0.1.93)  p2-2.vienna1-nbr2.bbnplanet.net 232 ms
  6 (      4.0.3.130)  p3-1.nyc4-nbr2.bbnplanet.net 222 ms
  7 (       4.0.5.26)  p1-0.nyc4-nbr3.bbnplanet.net 223 ms
  8 (      4.0.3.121)  p2-1.chicago1-nbr1.bbnplanet.net 235 ms
  9 (       4.0.5.89)  p10-0-0.chicago1-br1.bbnplanet.net 239 ms
 10 (       4.0.2.18)  h1-0.minneapol1-cr1.bbnplanet.net 258 ms
```

```
 11 (    4.0.246.254)  h1-0.uswest-mn.bbnplanet.net 260 ms
 12 (207.225.159.221)  207.225.159.221 249 ms
 13 ( 205.215.207.54)  www.uswc.uswest.net 258 ms

--- traceroute to www.rmi.net (166.93.8.30),
    30 hops max, 18 byte packets

  1 (  207.155.168.5)  ts003e01.den-co.concentric.net 152 ms
  2 (  207.155.168.1)  rt001e0102.den-co.concentric.net.168.155.207.IN-ADDR.ARPA 161 ms
  3 (   207.88.24.21)  207.88.24.21 190 ms
  4 (   207.88.0.253)  us-ca-scl-core1-f9-0.rtr.concentric.net 189 ms
  5 (   207.88.0.178)  207.88.0.178 206 ms
  6 ( 144.228.207.73)  sl-gw18-chi-5-1-0-T3.sprintlink.net 210 ms
  7 (  144.232.0.217)  sl-bb11-chi-3-3.sprintlink.net 216 ms
  8 (  144.232.0.174)  sl-bb5-chi-4-0-0.sprintlink.net 211 ms
  9 (   144.232.8.85)  sl-bb7-pen-5-1-0.sprintlink.net 225 ms
 10 (   144.232.5.53)  sl-bb10-pen-1-3.sprintlink.net 236 ms
 11 (   144.232.5.62)  sl-nap1-pen-4-0-0.sprintlink.net 228 ms
 12 (  192.157.69.13)  p219.t3.ans.net 263 ms
 13 ( 140.223.60.209)  f1-1.t60-6.Reston.t3.ans.net 264 ms
 14 (  140.223.65.17)  h12-1.t64-0.Houston.t3.ans.net 286 ms
 15 (  140.223.25.14)  h13-1.t80-1.St-Louis.t3.ans.net 283 ms
 16 (  140.223.25.29)  h14-1.t24-0.Chicago.t3.ans.net 292 ms
 17 (   140.223.9.18)  h14-1.t96-0.Denver.t3.ans.net 309 ms
 18 ( 140.222.96.122)  f1-0.c96-10.Denver.t3.ans.net 313 ms
 19 (  207.25.224.14)  h1-0.enss3191.t3.ans.net 306 ms
 20 (  166.93.46.246)  166.93.46.246 305 ms
 21 (    166.93.8.30)  www.rmi.net 285 ms
```

Die Pakete des zweiten Traces unternehmen eine kleine USA-Tour, bevor sie an ihrem Ziel, nur ein paar Kilometer vom Sende-Ort entfernt, ankommen. Erst folgen sie Concentrics Backbone durch einen Router in California (Zeile 4) und dann zum Chicago NAP, wo sie mit dem Sprint Backbone verbunden werden (Zeile 6). Die Pakete werden an den New York NAP in Pennsauken, New Jersey geroutet, wo sie an den ANS Backbone weitergegeben werden (Zeilen 11 und 12). Dann besuchen sie Router in Reston, Houston, St. Louis und Chicago (noch einmal) und kommen schließlich wieder in Denver an.

Wie die Pakete im letzten Beispiel haben auch wir eine etwas längere Route gewählt, um zum eigentlichen Thema zurückzukehren: CIDR.

2.1.5 CIDR: Verringern von Routing-Tabellen-Explosionen

Auf Grund der hierarchischen Struktur des Internets ist auch die Struktur des Routen-Verdichtungssystems in Schichten aufgeteilt. In den oberen Schichten werden große Blöcke benachbarter Class C-Adressen durch die Internet Assigned Numbers Authority (IANA) weltweit an verschiedene

Adress-Verwalter, so genannte *regionale IP-Registrierer* vergeben. Zurzeit gibt es drei regionale Registrierer. Die regionale Registrierung für Nord- und Südamerika, die Karibik und die Region Schwarzafrika (Sub-Saharan Africa) ist die American Registry for Internet Numbers (ARIN). ARIN ist außerdem dafür zuständig, Adressen an die globalen Service Provider zu verteilen. Die regionale Registrierung für Europa, den Nahen Osten, Nordafrika und Teile Asiens (Gebiete der früheren Sowjetunion) ist die Resèaux IP Europèens (RIPE). Die regionale Registrierung für den Rest Asiens und den Pazifik ist die Asia Pacific Network Information Center (APNIC).

Anmerkung

ARIN wurde aus dem InterNIC (von Network Solutions, Inc.) 1997 ausgegliedert, um das Management von IP-Adressen vom Management von Domänennamen zu trennen.

Tabelle 2.2 zeigt das ursprüngliche Schema für das Verteilen der Class C-Adressen an die Regionen der Registrierer, allerdings sind manche Einträge heute nicht mehr aktuell. Beispiel 2.8 zeigt, dass die mit »Andere « bezeichneten Blöcke im Moment vergeben werden. Die regionalen Registrierer geben Teile dieser Blöcke an die großen Service Provider oder an lokale IP-Registrierer weiter. Normalerweise sind die Blöcke, die in dieser Schicht verteilt werden, nicht kleiner als 32 benachbarte Class C-Adressen (normalerweise größer). Concentric Network hat zum Beispiel den Block 207.155.128.0/17 bekommen, der 128 benachbarten Class C-Adressen entspricht (siehe Beispiel 2.8).

Tabelle 2.2: CIDR-Adressenvergabe nach geografischer Region

Region	Address Bereich
Multiregional	192.0.0.0–193.255.255.255
Europa	194.0.0.0–195.255.255.255
Andere	196.0.0.0–197.255.255.255
Nordamerica	198.0.0.0–199.255.255.255
Zentral-/Südamerika	200.0.0.0–201.255.255.255
Pazifischer Raum	202.0.0.0–203.255.255.255
Andere	204.0.0.0–205.255.255.255
Andere	206.0.0.0–207.255.255.255

Beispiel 2.8: Wenn an Adresse 207.155.128.5 aus Beispiel 2.7 ein WHOIS ausgeführt wird, wird die Adresse als Teil eines /17 CIDR-Blocks von Concentric Network angezeigt.

```
--- looking up 207.155.128.5
--- performing WHOIS on "207.155.128.5", please wait...
--- contacting host whois.arin.net
--- smart query on "207.155.128"

Concentric Research Corp. (NETBLK-CONCENTRIC-CIDR)
   10590 N. Tantau Ave.
   Cupertino, CA 95014

   Netname: CONCENTRIC-CIDR
   Netblock: 207.155.128.0 - 207.155.255.255
   Maintainer: CRC

   Coordinator:
      DNS and IP ADMIN (DIA-ORG-ARIN) hostmaster@CONCENTRIC.NET
     (408) 342-2800
Fax- (408) 342-2810

   Domain System inverse mapping provided by:

   NAMESERVER3.CONCENTRIC.NET 206.173.119.72
   NAMESERVER2.CONCENTRIC.NET 207.155.184.72
   NAMESERVER1.CONCENTRIC.NET 207.155.183.73
   NAMESERVER.CONCENTRIC.NET 207.155.183.72

   Record last updated on 13-Feb-97.
   Database last updated on 29-Jan-99 16:12:40 EDT.
```

Die Service Provider, die diese Blöcke bekommen, geben sie als kleinere Blöcke an ihre Teilnehmer weiter. Wenn diese Teilnehmer selbst ISPs sind, dann werden die Blöcke erneut aufgeteilt. Der Vorteil, die Class C-Adressen als Blöcke namens *CIDR-Blöcke* zu vergeben, ist, dass die Zusammenfassung sehr erleichtert wird. Weitere Informationen über das Verteilen der Adressen finden Sie in RFC 2050 (www.isi.edu/in-notes/rfc2050.txt).

Um dies zu erläutern, nehmen wir den Fall an, dass Concentric Network einem Teilnehmer einen Teil seines Blocks 207.155.128.0/17 zuweist, und zwar 207.155.144.0/20. Wenn dieser Teilnehmer ein ISP ist, wird ein Teil dieses Teils, zum Beispiel 207.155.148.0/22, an einen seiner Teilnehmer vergeben. Dieser Teilnehmer gibt seinen Block /22 (lies »slash twenty-two«) seinem ISP an. Dieser ISP fasst alle seine Teilnehmer für Concentric Network mit der Adresse 207.155.144.0/20 zusammen und Concentric Network fasst wiederum all seine Teilnehmer für die NAPs mit der Adresse 207.155.128.0/17 zusammen.

Die Angabe einer zusammengefassten Adresse an die oberen Schichten ist natürlich wesentlich praktischer als vielleicht Hunderte von einzelnen Adressen zu übermitteln. Ein weiterer Vorteil eines solchen Schemas ist die Stabilität, die das Internet so gewinnt. Wenn sich der Status eines Netzwerks in einer Low-Level-Domäne verändert, wird diese Veränderung nur bis zum nächsten Verdichtungspunkt gespürt.

Tabelle 2.3 zeigt die verschiedenen Größen von CIDR-Blöcken, ihre entsprechende Class C-Netzwerkgröße und die Zahl der Hosts, die jeder Block repräsentieren kann.

Tabelle 2.3: CIDR-Blockgrößen

CIDR-Block-Präfix-Größe	Entsprechende Zahl von Class C-Adressen	Zahl der möglichen Hostadressen
/24	1	254
/23	2	510
/22	4	1022
/21	8	2046
/20	16	4094
/19	32	8190
/18	64	16.382
/17	128	32.766
/16	256	65.534
/15	512	131.070
/14	1024	262.142
/13	2048	524.286

2.1.6 CIDR: Verringert Mangel der Class B-Adressen

Der Mangel an Class B-Adressen ergab sich aus einem grundsätzlichen Fehler im Design der IP-Adressklassen. Eine Class C-Adresse bietet 254 Hostadressen, während eine Class B-Adresse 65.534 Hostadressen ermöglicht. Dies stellt eine große Lücke dar. Vor CIDR hätte einem Unternehmen, das 500 Hostadressen braucht, eine Class C-Adresse nicht genügt. Wahrscheinlich hätte man eine Class B-Adresse angefordert, obwohl dadurch 65.000 Hostadressen verschwendet worden wären. Bei CIDR können die Bedürfnisse durch einen /23-Block erfüllt werden. Die Hostadressen, die sonst verschwendet worden wären, bleiben erhalten.

2.1.7 Schwierigkeiten bei CIDR

Obwohl CIDR sowohl das Wachstum der Internet-Routing-Tabellen als auch das Verschwinden der Class B-Adressen verlangsamt, hat es für die Benutzer auch einige Probleme geschaffen.

Das erste Problem ist das der Tragbarkeit. Wenn Sie zum Beispiel einen CIDR-Block zugewiesen bekommen, sind die Adressen höchstwahrscheinlich Teil eines größeren Blocks, der dem ISP zugewiesen ist. Was, wenn Ihr ISP jetzt aber nicht Ihre Erwartungen oder vertraglichen Bedingungen erfüllt oder Sie ein anderes, weitaus günstigeres Angebot eines anderen ISP bekommen? Ein Wechsel des ISP bedeutet höchstwahrscheinlich, dass sie alles neu adressieren müssen. Es ist unwahrscheinlich, dass ein ISP einem Teilnehmer erlaubt seinen Block zu behalten, wenn er zu einem neuen Provider wechselt. Regionale Registrierer empfehlen außerdem, dass die Adressen zurückgegeben werden, wenn ein Teilnehmer den ISP wechselt.

Für einen Endbenutzer kann das Readressieren manchmal zu größeren Schwierigkeiten führen. Am einfachsten ist es wahrscheinlich für diejenigen, die private Adressen innerhalb ihrer eigenen Routing-Domäne und Network Address Translation (siehe Kapitel 4) zu verwenden. In diesem Fall müssen nur die tatsächlich an die Öffentlichkeit angebundenen Adressen geändert werden, die internen Benutzer bleiben weitestgehend unberührt. Andererseits gibt es auch Benutzer, die all ihren Netzwerkgeräten statische, öffentliche Adressen gegeben haben. Diese Benutzer müssen dann jedes Gerät einzeln umstellen.

Selbst wenn der Endbenutzer den CIDR-Block über die ganze Domäne verteilt hat, kann das Readressieren durch das Verwenden von DHCP (oder BOOTP) erleichtert werden. In diesem Fall müssen nur die DHCP Scopes geändert werden und die Benutzer rebooten, allerdings müssen manche statischen Adressen, zum Beispiel bei Servern und Routern, einzeln umgestellt werden.

Das Problem ist wesentlich größer, wenn Sie ein ISP sind und Ihren upstream Provider wechseln wollen. Sie müssen dann nicht nur Ihr eigenes Netzwerk umstellen, sondern auch die Ihrer Teilnehmer, die einen Teil des CIDR-Blocks zugewiesen bekommen haben.

CIDR ist außerdem ein Problem für diejenigen, die sich mit mehreren Service Providern verbinden wollen. Multihoming (das in diesem Kapitel später noch ausführlicher besprochen wird) wird zur Absicherung verwendet, sodass Endbenutzer nicht von einem einzigen ISP und dessen Zuverlässigkeit abhängig sind. Das Problem ist, dass ihre Adressen natürlich aus nur einem

Block (von einem ISP) stammen und diese nun einem anderen Provider bekannt gegeben werden müssen.

Abbildung 2.7 zeigt, was passieren kann: Hier besitzt ein Teilnehmer den /23 CIDR-Block, der ein Teil des größeren 20er Blöcke des ISP1 ist. Wenn der Teilnehmer sich mit ISP2 verbindet, will er sicherstellen, dass der Verkehr ihn über beide ISPs erreichen kann. Deshalb muss er seinen /23-Block durch ISP2 bekannt geben. Das Problem ergibt sich, wenn der ISP2 den /23-Block dem Rest der Welt angibt, denn nun haben alle Router »dort draußen« eine Route zu 205.113.48.0/20, die von ISP1 veröffentlicht wird, und eine Route zu 205.113.50.0/23, die von ISP2 veröffentlicht wird. An den Teilnehmer adressierte Pakete werden eher entlang der genaueren Route gesendet und so geht der Großteil des Verkehrs aus dem Internet durch ISP2 – auch wenn die Quelle eigentlich viel näher an ISP1 ist.

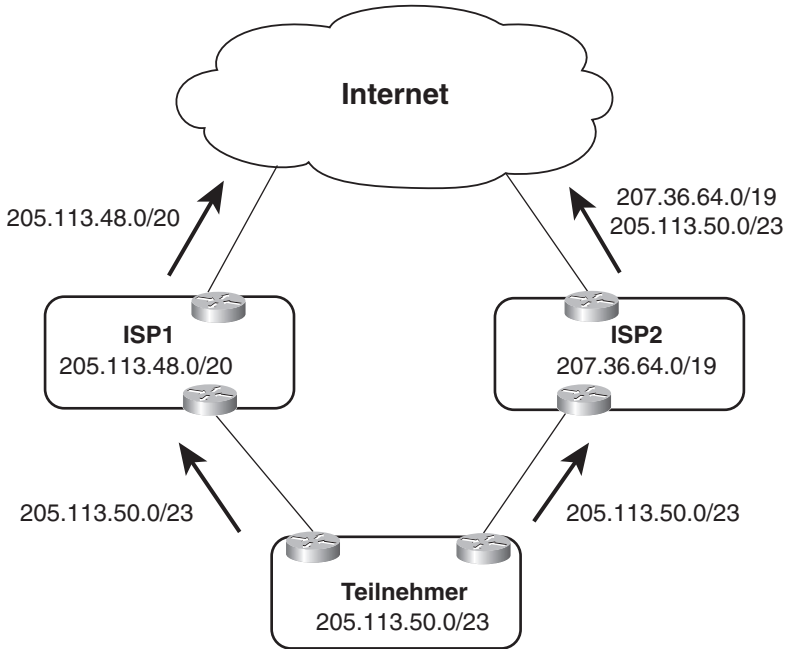

Abb. 2.7: Eingehender Internetverkehr geht über die genauere Route.

In Abbildung 2.7 ist es sogar möglich, dass die Route 205.113.50.0/23 ISP1 über das Internet bekannt gegeben wird. Dies sollte nicht passieren, da die meisten ISPs Route-Filter benutzen, um zu verhindern, dass eigene Routen ein weiteres Mal in ihre Domäne kommen. Allerdings kann niemand garantieren, dass der ISP1 richtig filtert. Wenn die genauere Route aus dem Inter-

net angenommen wird, nimmt selbst Verkehr, der von ISP1 stammt, die Route über ISP2 zu 205.113.50.0/23 anstatt den direkten Weg zu wählen.

Wenn ein Teilnehmer multihomed sein möchte, muss ISP1 die genauere Route zusammen mit seinem CIDR-Block bekannt geben (siehe Abbildung 2.8). Die meisten Provider tun dies nicht, beziehungsweise nur ungern, weil sie so in ihren CIDR-Block »ein Loch schießen« (manchmal auch *address leaking* genannt). Die Angabe einer genaueren Route stellt nicht nur zusätzliche administrative Arbeit für den ISP dar, sondern verringert auch die Effektivität von CIDR.

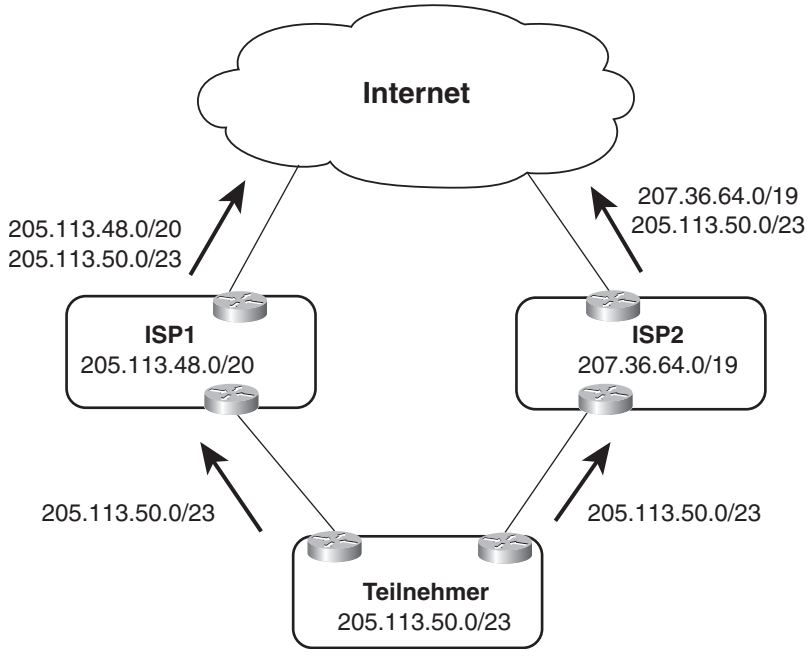

Abb. 2.8: ISP1 »schießt ein Loch« in seinen CIDR-Block.

Obwohl die Abbildungen 2.7 und 2.8 zeigen, dass ISP1 nur eine einzige Verbindung zum Internet hat, ist dies in Wirklichkeit normalerweise anders. ISPs haben meistens viele Verbindungen zu größeren Providern und zu NAPs. Der Provider muss an jeder dieser Verbindungen die genauere Route zusammen mit dem CIDR-Block bekannt geben und er muss außerdem vielleicht noch die Route-Filter für eingehenden Verkehr verändern. Die Administration wird außerdem noch erschwert, weil ISP1 und ISP2 zusammenarbeiten müssen, um sicherzustellen, dass der /23-Block des Teilnehmers richtig veröffentlicht wird. Da ISP1 und ISP2 Wettbewerber sind, möchten sie vielleicht nicht zusammenarbeiten.

Selbst wenn der Teilnehmer in Abbildung 2.8 ISP1 und ISP2 dazu bringen kann, seinen /23-Block anzugeben, gibt es ein weiteres Problem. Manche Tier I-Provider akzeptieren nämlich nur Präfixe von /19 oder kleiner, um die Backbone-Level Routing-Tabellen gut kontrollieren zu können. Wenn ISP1 oder ISP2 durch einen solchen Provider angeschlossen sind, können sie den /23-Block des Teilnehmers nicht bekannt geben. Der Vorgang, durch den alle CIDR-Adressen mit einem höheren Präfix als /19 herausgefiltert werden, ist inzwischen so bekannt, dass das Präfix /19 oft einfach *globally routable* Adresse genannt wird. Das hat zu bedeuten, dass ein längerer CIDR-Präfix, zum Beispiel ein /21 oder /22, nicht in allen Teilen des Internets veröffentlicht wird. Denken Sie daran, dass die Teile des Internets, für die Sie nicht erreichbar sind, auch für Sie eigentlich nicht erreichbar sind.

Anmerkung

Viele Tier I-Provider haben ihre /19-Regeln gelockert, da es immer mehr Beschwerden von Teilnehmern gab.

Eine mögliche Lösung für die multihomed Teilnehmer in Abbildung 2.8 ist *provider-independent* Adressen zu bekommen (auch als *portable* Address Space bekannt). Dies bedeutet, dass der Teilnehmer sich einen Block sichern kann, der nicht ein Teil der Blöcke von ISP1 oder ISP2 ist; beide ISPs können nun den Block des Teilnehmers verwalten, ohne dass dabei der eigene Block eingeschränkt wird. Seit es ARIN gibt, ist es wesentlich einfacher geworden, einen unabhängigen Block zu bekommen, als es zu Zeiten war, als InterNIC noch existierte. Obwohl ARIN empfiehlt, dass Sie entweder Adressen von Ihrem Provider verwenden oder Adressen von einem Provider Ihres Providers, kann eine Provider-unabhängige Adresse als letzte Lösung gesehen werden. Allerdings gibt es immer noch genügend Schwierigkeiten.

Wenn Sie multihomen, ist es erstens wahrscheinlich, dass Ihre ursprünglichen Adressen von einem ISP stammen. Der Wechsel zu unabhängigen Adressen bedeutet also, dass Sie Ihr Netzwerk readressieren müssen, die Schwierigkeiten eines solchen Unterfangens wurden ja schon besprochen (wenn Sie Ihre IP-Adressen bekamen, bevor CIDR eingeführt wurde, dann erledigt sich die Frage natürlich).

Zweitens verteilen die Registrierer die Adressen je nach aktuellem und nicht nach prognostiziertem Bedarf. Diese Richtlinie bedeutet, dass Sie wahrscheinlich gerade genug Platz für Ihre aktuellen Bedürfnisse bekommen. Wenn Sie nun doch noch mehr Platz brauchen, müssen Sie sich hierfür beim Registrierer bewerben und zeigen, dass Sie den bereits zugeteilten Platz auch

gut nutzen. ARIN benötigt zum Beispiel einen Beweis der effizienten Nutzung entweder durch Benutzung des Shared WHOIS Project (SWIP) oder durch die Benutzung des Referral WHOIS Server (RWHOIS). Beim oft verwendeten SWIP werden WHOIS-Informationen einer SWIP-Template hinzugefügt und an ARIN gemailt. Um RWHOIS zu benutzen, benötigen sie einen RWHOIS-Server, den ARIN erreichen kann, um WHOIS-Informationen zu bekommen. In beiden Fällen gibt WHOIS Informationen, die zeigen, ob Sie Ihre Adressen effizient nutzen und ob Sie weitere Adressen benötigen.

Natürlich haben Sie immer noch ein Problem, wenn Sie keine globally routable (/19) Adressen bekommen. Im Endeffekt machen CIDR-Regeln Multihoming zu einem Problem für kleine Teilnehmer und ISPs. Der folgende Abschnitt beschreibt Multihoming etwas genauer und geht auf alternative Topologien ein.

2.2 Wer braucht BGP?

Es benötigen nicht so viele Netzwerke BGP, wie man vielleicht denkt. Oft wird vermutet, dass ein Netzwerk, das aus mehreren Routing-Domänen besteht, immer BGP zwischen den Domänen einsetzen muss. BGP ist sicherlich eine Option, aber wieso soll alles durch das Einführen eines weiteren Protokolls viel komplizierter gemacht werden?

Nehmen wir als Beispiel ein multinationales Unternehmensnetzwerk, das aus 3.000 Routern und vielleicht 150,000 Benutzern besteht. Abbildung 2.9 zeigt, wie ein so großes Netzwerk vielleicht aufgebaut werden kann. Das ganze Netzwerk wird mit OSPF geroutet und in acht geografische OSPF-Routing-Domänen aufgeteilt. Obwohl die Abbildung nur die Backbone-Gegenden jeder OSPF-Domäne zeigt, ist jede der Domänen in viele OSPF-Gegenden aufgeteilt, die den geografischen Regionen entsprechen.

BGP kann verwendet werden, um die OSPF-Domänen zu verbinden, es ist aber nicht nötig. Stattdessen verteilt sich jede OSPF Backbone-Gegend in einen einzigen globalen Backbone. Der globale Backbone ist eine weitere OSPF-Domäne, bestehend aus einer einzigen OSPF-Gegend. Obwohl dieser Kern aus high-end-Routern besteht, die den hohen Verkehr switchen müssen, ist die Belastung der Router mit Routing-Tabellen und OSPF-Verarbeitung recht gering. Auf Grund der Art, in der die Adressen im Netzwerk vergeben sind, gibt jede der acht OSPF-Domänen nur eine einzige zusammengefasste Route zum globalen Backbone an. Die Zusammenfassung ist für dieses System von fundamentaler Bedeutung. Es gibt vermutlich in diesem Netzwerk so viele Subnets, dass OSPF ohne Zusammenfassung an ihnen

»ersticken« würde. Dies würde zu schlechter Leistung und möglichen Router-Problemen führen.

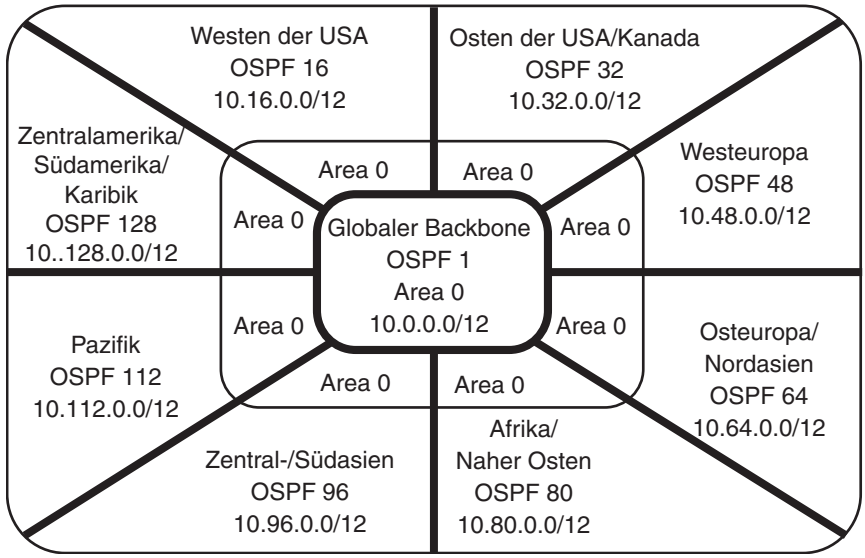

Abb. 2.9: Selbst ein sehr großes Netzwerk kann nur mit Multiple IGP-Domänen gebaut werden.

Die hierarchischen Konstruktionen der physischen Topologie und der Adressen sind zwei der drei Faktoren, die das Netzwerk in Abbildung 2.9 so einfach machen. Der dritte Faktor ist die gemeinsame Administration des gesamten Netzwerks. Eine gemeinsame Administration bedeutet, dass Routing-Richtlinien im gesamten Netzwerk gleichmäßig angewandt werden. In diesem Fall bestimmt die Richtlinie die Adressen, die in jeder OSPF Gegend benutzt werden und bestimmt außerdem, dass alle OSPF-Prozesse nur über OSPF miteinander verbunden sind.

Anmerkung

Eine Routing-Richtlinie (policy) ist ein entworfener und konfigurierter Prozess, der die Verkehrsmuster innerhalb eines Netzwerks kontrolliert, indem Routen und deren Eigenschaften gemanagt werden. Redistribution, Route-Filter und Routen-Karten sind die am häufigsten verwendeten Werkzeuge, um Richtlinien mit Cisco IOS-Software umzusetzen.

Natürlich gibt es im wahren Leben kaum Unternehmen in der Größe dieses Beispiels, die wie in Abbildung 2.9 die Möglichkeit haben, »von Grund auf« in solch einer koordinierten, logischen Weise aufgebaut zu werden. Viele, wenn nicht gar alle Netzwerke sind aus anderen, kleineren Netzwerken entstanden, als Unternehmen sich zusammenschlossen usw. Folglich haben viele verschiedene Administratoren unterschiedliche Entscheidungen für ihre Teile des Netzwerks getroffen, was dazu führen kann, dass es schon ein kleines Wunder ist, wenn die vielen verschiedenen Teile überhaupt miteinander funktionieren.

Hinzu kommt aber noch der Wunsch, Routing-Richtlinien durchführen zu können. Der Verkehr von manchen Domänen des Netzwerks soll vielleicht immer bestimmte Links oder Routen nehmen, vielleicht sollen aber auch nur bestimmte Routen zwischen Domänen veröffentlicht werden. In den meisten Fällen können diese Wünsche immer noch durch Umverteilung zwischen IGPs und durch Tools wie Route-Filter und Routen-Karten erreicht werden. BGP sollte nur eingeführt werden, wenn es dafür einen guten Grund gibt, zum Beispiel wenn IGPs nicht die nötigen Mittel haben, um die Richtlinien umzusetzen oder wenn die Größe der Routing-Tabellen nicht mehr durch Zusammenfassung verringert werden kann. BGP ist zum Beispiel nützlich, wenn viele verschiedene IGPs in der Domäne verwendet werden. Hier ist es vielleicht einfacher BGP zu verwenden als zu versuchen, in den IGPs zu redistributieren.

Wenn Sie versuchen zu entscheiden, ob BGP nötig ist, dann denken Sie an die Gründe, die überhaupt zur Entwicklung externer Routing-Protokolle geführt haben. Externe Routing-Protokolle werden verwendet, um zwischen Autonomen Systemen zu routen – das heißt zwischen Netzwerken, die unter verschiedenen Administratoren stehen. In einem Unternehmensnetzwerk, selbst in einem sehr großen mit vielen Domänen unter verschiedenen Administratoren sollte es normalerweise eine zentrale Autorität geben, die Routing-Richtlinien mithilfe der Mittel der IGPs vorschreiben und durchsetzen kann. Wenn aber Autonome Systeme verbunden werden müssen, dann kann BGP benutzt werden.

Meistens geht es bei Fällen, in denen BGP gebraucht wird, um Internetverbindungen – entweder zwischen einem Teilnehmer und einem ISP oder (wahrscheinlicher) zwischen ISPs. Selbst bei solchen Verbindungen Autonomer Systeme ist BGP vielleicht nicht notwendig. Der Rest dieses Abschnitts bespricht typische inter-AS-Topologien und zeigt, wo BGP gebraucht wird und wo nicht.

2.2.1 Ein Singlehomed Autonomes System

Abbildung 2.10 zeigt einen Teilnehmer, der durch eine einzige Verbindung an einen ISP angeschlossen ist. BGP oder jedes andere Routing-Protokoll ist hier überflüssig. Wenn die einzige Verbindung ausfällt, muss keine Routing-Entscheidung getroffen werden, da es keinen anderen Weg gibt. Ein Routing-Protokoll bringt also nichts. Bei dieser Topologie fügt der Teilnehmer dem Grenz-Router eine statische default Route hinzu und redistributiert diese Route in seinem AS.

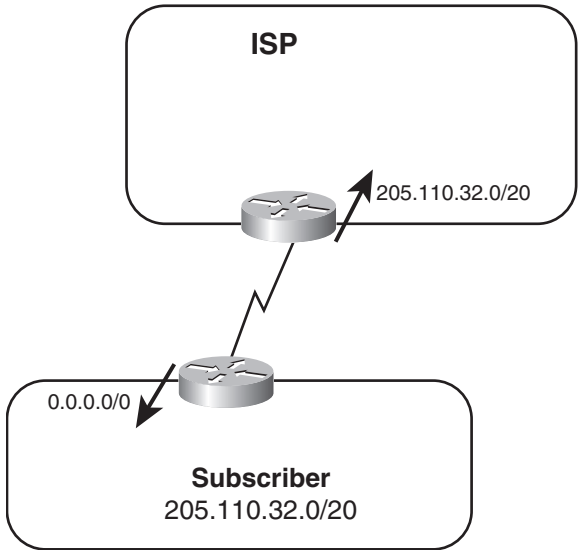

Abb. 2.10: Statische Routen sind alles, was man in dieser singlehomed Topologie braucht.

Der ISP fügt ebenfalls eine statische Route hinzu, die zu den Adressen des Teilnehmers führt, und gibt diese Route in seinem AS an. Wenn die Adressen des Teilnehmers ein Teil des Blocks des ISPs sind, dann geht die Route, die vom ISP-Router veröffentlicht wird, natürlich nicht weiter als bis zum eigenen AS des ISP. »Der Rest der Welt« erreicht den Teilnehmer, indem zum Adressenblock des ISP geroutet wird, die genauere Route wird erst innerhalb des AS des ISP bestimmt.

Es ist wichtig beim Arbeiten mit inter-AS-Verkehr daran zu denken, dass jeder physische Link eigentlich zwei logische Links enthält: einen für ankommenden Verkehr, einen für ausgehenden Verkehr (siehe Abbildung 2.11).

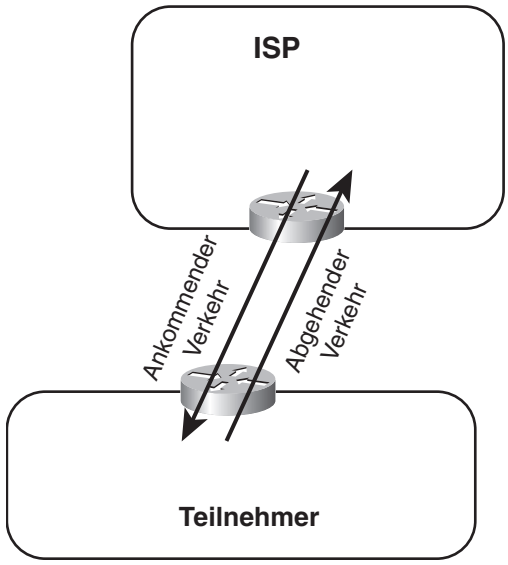

Abb. 2.11: Jeder physische Link zwischen Autonomen Systemen enthält zwei logische Links, einen für ausgehende, einen für ankommende Pakete.

Die Routen, die Sie in jeder Richtung bekannt geben, haben einen Einfluss auf den Verkehr. Avi Freedman, der viele ausgezeichnete Artikel über ISPs geschrieben hat, nennt eine Routen-Angabe ein Versprechen, Pakete zum Zieladressenblock der Route zu bringen. In Abbildung 2.10 gibt der Router des Teilnehmers eine default Route in das local AS an – das ist ein Versprechen, Pakete ohne genauere Zieladresse dorthin zu bringen. Der ISP-Router gibt eine Route zu 205.110.32.0/20 an und verspricht Verkehr zum AS des Teilnehmers zu bringen. Der ausgehende Verkehr aus dem AS des Teilnehmers ist ein Ergebnis der default Route und der ankommende Verkehr ist ein Ergebnis der Route, die vom Router des ISP bekannt gegeben wird. Dieses Konzept mag vielleicht im Moment recht einfach und selbstverständlich erscheinen, es ist aber sehr wichtig, sich bei der Analyse komplexerer Systeme daran zu erinnern.

Die eindeutige Schwäche der Topologie in Abbildung 2.10 ist, dass die ganze Verbindung von verschiedenen Fehlerpunkten unterbrochen werden kann. Wenn der einzige Datentransfer-Link ausfällt, wenn ein Router oder eine der Schnittstellen ausfällt, wenn die Konfiguration eines Routers ausfällt, wenn ein Prozess innerhalb des Routers nicht funktioniert oder wenn einer der Administratoren einen Fehler macht, dann kommt der Teilnehmer nicht mehr an das Internet. Was in diesem Bild fehlt, ist Absicherung oder *Redundancy.*

2.2.2 Multihoming zu einem einzigen Autonomen System

Abbildung 2.12 zeigt eine verbesserte Topologie mit abgesicherten Links zu einem Provider. Wie der eingehende und ausgehende Verkehr über diese Links verteilt wird, hängt davon ab, wie die Links verwendet werden. Es kann zum Beispiel sein, dass ein Link ein primärer Link zum Internet ist – sagen wir T1 – und dass der andere Link nur als Backup verwendet wird. In einem solchen Fall ist der Backup-Link wahrscheinlich langsamer.

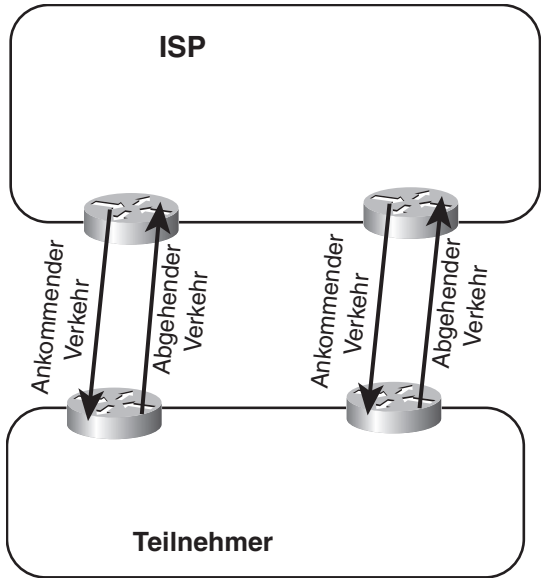

Abb. 2.12: Multihoming zu einem einzigen Autonomen System

Wenn der Absicherungs-Link nur als Backup verwendet wird, ist BGP wieder nicht nötig. Die Routen können genau wie im vorherigen Szenario veröffentlicht werden, nur dass die Routen, die mit dem Backup-Link weite Entfernungen zugewiesen bekommen, weshalb sie nur benutzt werden, wenn der primäre Link ausfällt.

Beispiel 2.9 zeigt, wie die Konfiguration der Router mit dem primären und sekundären Link aussehen könnte.

Beispiel 2.9: Primäre und sekundäre Link-Konfigurationen für Multihoming zu einem einzigen Autonomen System

```
Primary Router
router ospf 100
 network 205.110.32.0 0.0.15.255 area 0
 default-information originate metric 10
```

```
!
ip route 0.0.0.0 0.0.0.0 205.110.168.108
```

```
Backup Router
router ospf 100
 network 205.110.32.0 0.0.15.255 area 0
 default-information originate metric 100
!
ip route 0.0.0.0 0.0.0.0 205.110.168.113 150
```

In dieser Konfiguration hat der Backup-Router eine default Route, bei der die administrative Entfernung auf einen Wert von 150 gestellt ist, sodass sie nur in der Tabelle erscheint, wenn die primäre Route ausfällt. Außerdem bekommt die Backup-Route eine höhere Metrik als die primäre Route, um sicherzustellen, dass andere Router in der OSPF-Domäne die primäre default Route wählen. Die Type der OSPF-Metrik beider Routen ist E2, also bleiben die veröffentlichten Metriken innerhalb der OSPF-Domäne überall gleich. So wird sichergestellt, dass die Metrik der primären Route immer niedriger bleibt als die der sekundären Route, egal wie die Wegkosten ausfallen. Beispiel 2.10 zeigt die default Routen in einem internen Router der OSPF-Domäne.

Beispiel 2.10: Der erste Eintrag zeigt die primäre externe Route; der zweite Eintrag zeigt die Backup-Route, die verwendet wird, nachdem die primäre Route ausfällt.

```
Phoenix#show ip route 0.0.0.0
Routing entry for 0.0.0.0 0.0.0.0, supernet
  Known via "ospf 1", distance 110, metric 10, candidate default Path
  Tag 1, type extern 2, forward metric 64
  Redistributing via ospf 1
  Last update from 205.110.36.1 on Serial0, 00:01:24 ago
  Routing Descriptor Blöcke:
  * 205.110.36.1, from 205.110.36.1, 00:01:24 ago, via Serial0
      Route metric is 10, traffic share count is 1

Phoenix#show ip route 0.0.0.0
Routing entry for 0.0.0.0 0.0.0.0, supernet
  Known via "ospf 1", distance 110, metric 100, candidate default Path
  Tag 1, type extern 2, forward metric 64
  Redistributing via ospf 1
  Last update from 205.110.38.1 on Serial1, 00:00:15 ago
  Routing Descriptor Blöcke:
  * 205.110.38.1, from 205.110.38.1, 00:00:15 ago, via Serial1
      Route metric is 100, traffic share count is 1
```

Obwohl ein System mit primärem Link und Backup eine Absicherung ermöglicht, nutzt es die Bandbreite der Leitungen nicht optimal aus. Es ist bes-

ser, wenn sich beide Links gegenseitig für den Fall eines Link- oder Router-Ausfalls absichern und ansonsten gleichmäßig benutzt werden. Bei einem solchen System läuft die Konfiguration bei beiden Routern wie im Beispiel 2.11 zu sehen ist.

Beispiel 2.11: Konfiguration für Lastenverteilung bei Multihoming desselben AS

```
router ospf 100
 network 205.110.32.0 0.0.15.255 area 0
 default-information originate metric 10 metric-type 1
!
ip route 0.0.0.0 0.0.0.0 205.110.168.108
```

Die statischen Routen beider Router haben dieselbe administrative Entfernung und die default Routen haben gleiche Metriken (10). Die default Routen haben nun eine Type der OSPF-Metrik von E1. Bei dieser Metrik bezieht jeder Router in der OSPF-Domäne die internen Kosten der Route in seine Berechnung der besten Route mit ein. Folglich wählt jeder Router die Route, deren Ausgangspunkt für ihn am nächsten liegt (siehe Abbildung 2.13).

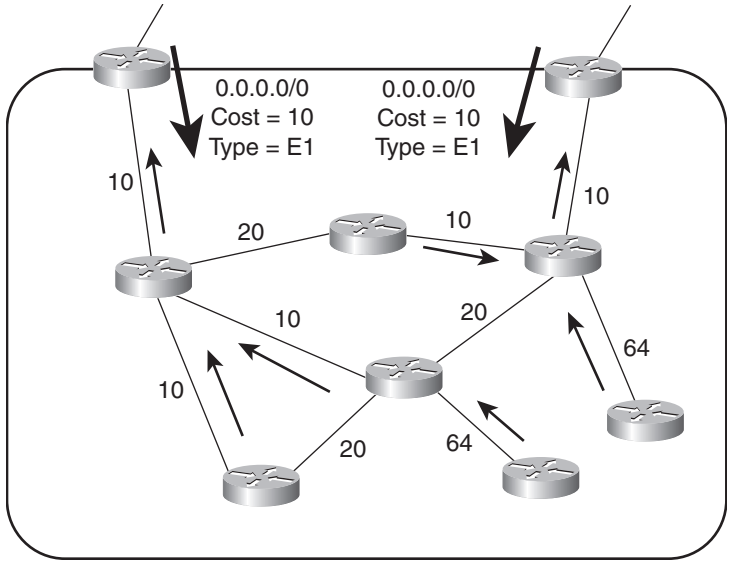

Abb. 2.13: Grenz-Router geben eine default Route mit einer Metrik von 10 und einer Type der OSPF-Metrik von E1 an.

In den meisten Fällen genügt es, die default Routen dem AS von mehreren Ausgangspunkten aus bekannt zu geben und die Adressen des AS an denselben Punkten zusammenzufassen, um ein effizientes System zu erschaffen.

Die einzige Frage ist, ob das asymmetrische Verkehrsmuster zu einem Problem wird. Wenn die geografische Entfernung zwischen den Ausgangspunkten so groß ist, dass Verzögerungen auftreten, denn dann müssen Sie das Routing vielleicht noch genauer kontrollieren. In diesem Fall sollten Sie BGP verwenden.

Nehmen wir zum Beispiel an, dass die zwei Exit-Router in Abbildung 2.12 sich in Los Angeles und London befinden. Vielleicht wollen Sie, dass der Verkehr für die östliche Hälfte des Globus den Router London verwendet und der Verkehr für die westliche Hälfte den Router Los Angeles. Denken sie daran dass die ankommenden Routen-Angaben ihren ausgehenden Verkehr beeinflussen. Wenn der Provider Ihrem AS die Routen durch BGP bekannt gibt, haben die internen Router genauere Informationen über externe Ziele. BGP bietet außerdem die Mittel, Routing-Richtlinien für die externen Ziele festzulegen.

Die ausgehenden Routen-Angaben beeinflussen hingegen den eingehenden Verkehr. Wenn die internen Routen dem Provider über BGP bekannt gegeben werden, haben Sie Einfluss darauf, welche Routen an welchem Ausgangspunkt bekannt gegeben werden, und die Möglichkeit (in gewissem Maße), die Entscheidungen des Providers über den in Ihr AS eingehenden Verkehr zu beeinflussen.

Bei der Überlegung, ob BGP verwendet werden soll, müssen die Vorteile von BGP dem Problem der komplexeren Routing-Situation entgegengestellt werden. Sie sollten BGP nur dann verwenden, wenn Sie sich damit einen Vorteil bei der Kontrolle des Verkehrs schaffen können. Analysieren Sie ankommenden und ausgehenden Verkehr einzeln. Wenn es wichtig erscheint, den ankommenden Verkehr zu kontrollieren, dann verwenden Sie BGP, um Ihre Routen an den Provider zu geben, behalten Sie aber einfach die default Route zu Ihrem AS.

Wenn es aber nur wichtig ist, den ausgehenden Verkehr zu kontrollieren, dann verwenden Sie BGP nur, um Routen von Ihrem Provider zu empfangen. Bedenken Sie, was es bedeutet, die Routen von Ihrem Provider zu empfangen. »Taking full BGP routes« bedeutet, dass Ihnen Ihr Provider die gesamte Internet-Routing-Tabelle angibt. Zur Zeit sind das etwa 88.000 Routen, wie in Beispiel 2.12 zu sehen ist. Um eine solche Tabelle zu speichern und zu bearbeiten, benötigen Sie einen relativ leistungsfähigen Router und mindestens 64 MB Speicher (obwohl 128 MB empfohlen werden). Ein einfaches default Routing-Schema lässt sich hingegen mit einem einfacheren Router und viel weniger Speicherplatz realisieren.

Beispiel 2.12: Die IP-Routing Übersicht zeigt 88.269 BGP Einträge an

```
route-server>show ip route summary
Route Source    Networks    Subnets     Overhead    Memory (bytes)
connected       0           1           56          144
static          2           1           168         432
bgp 65000       76302       11967       4943064     12847416
  External: 88269 Internal: 0 Local: 0
internal        779                                 906756
Total           77083       11969       4943288     13754748
route-server>
```

Anmerkung

Die Zusammenfassung der Routing-Tabellen in Beispiel 2.12 stammt von einem öffentlichen Route Server unter route-server.ip.att.net. Ein weiterer Server, an dem Telnet benutzt werden kann, ist route-server.cerf.net. Die Anzahl der BGP-Einträge verändert sich zwar ein wenig, ist aber bei allen Servern gleich.

»Taking partial BGP routes« ist ein Kompromiss zwischen der Übernahme aller Routen und überhaupt keiner Übernahme. Wie der Name sagt, sind diese Routen ein Teil der vollen Internet-Routing-Tabelle. Ein Provider kann so zum Beispiel nur Routen zu seinen anderen Teilnehmer, und eine default Route für den Rest des Internets bekannt geben. Der folgende Abschnitt stellt ein Szenario vor, bei dem es sinnvoll ist, nur einen Teil der Routen zu nehmen.

Es muss auch beachtet werden, dass bei der Verwendung von BGP die Routing-Domäne des Teilnehmers mit einer Autonomen Systemnummer gekennzeichnet werden muss. Wie IP-Adressen sind auch Autonome Systemnummern nur in begrenzter Zahl verfügbar und werden deswegen von den regionalen Registrierern vergeben. Ebenfalls wie bei IP-Adressen ist ein Teil der Autonomen Systemnummern für den privaten Gebrauch reserviert: die AS-Nummern 64.512 bis 65.535. Meistens benutzen Teilnehmer, die mit einem Server verbunden sind (ob single- oder multihomed), eine Autonome Systemnummer aus dem reservierten Teil. Der Service Provider filtert die private AS-Nummer aus dem veröffentlichten BGP-Pfad heraus.

Obwohl die Topologie in Abbildung 2.12 eine Verbesserung derjenigen in Abbildung 2.10 ist, weil die absichernden Router und Links hinzugefügt wurden, gibt es immer noch einen Punkt, der bei einem Ausfall das ganze System lahm legt: der ISP selbst. Wenn der ISP die Verbindung zum Internet verliert, dann gilt dies auch für den Teilnehmer. Auch wenn der ISP größere interne Probleme hat, leidet der Teilnehmer darunter.

2.2.3 Multihoming zu mehreren Autonomen Systemen

Abbildung 2.14 zeigt eine Topologie, in der ein Teilnehmer an mehr als einen ISP angeschlossen ist. Zusätzlich zu den bereits besprochenen Vorteilen von Multihoming schützt sich dieser Teilnehmer gegen einen möglichen Verlust der Internetverbindung auf Grund eines ISP-Ausfalls.

Für ein kleines Unternehmen oder einen kleinen ISP gibt es mehrere Schwierigkeiten mit Multihoming zu mehreren Service Providern. Die Probleme die auftreten, wenn die Adressen eines Teilnehmer ein Teil des Adressblocks des Providers sind, wurden schon besprochen:

- Der ursprüngliche Provider muss davon überzeugt werden, in seinen CIDR-Block ein »Loch zu schießen«.
- Der zweite Provider muss überzeugt werden, Adressen anzugeben, die nicht aus seinem Block stammen.
- Beide Provider müssen gewillt sein, miteinander zu koordinieren, um die Adressen des Teilnehmers richtig bekannt geben zu können.
- Wenn der Adressblock des Teilnehmers kleiner als ein /19 ist (was bei einem kleinen Teilnehmer wahrscheinlich ist), akzeptieren manche Backbone-Provider die Route vielleicht nicht.

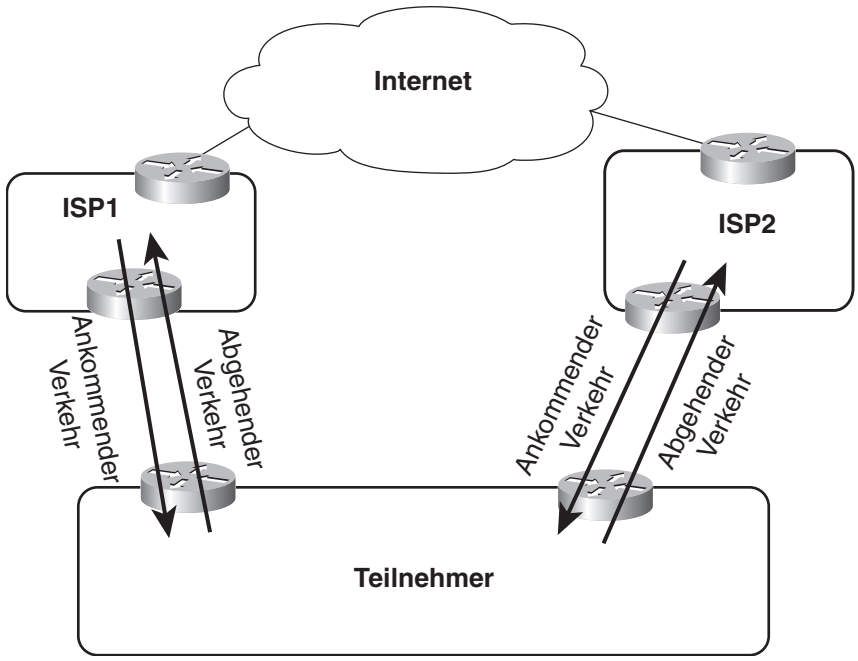

Abb. 2.14: Multihoming zu mehreren Autonomen Systemen

Die besten Kandidaten für Multihoming zu mehreren Providern sind große Unternehmen und ISPs, die groß genug sind, um provider-independent Adressen (oder diejenigen, die bereits solche Adressen besitzen) und eine öffentliche Autonome Systemnummer zu bekommen.

Der Teilnehmer in Abbildung 2.14 könnte immer noch vermeiden, BGP zu benutzen. Eine Option ist, einen ISP als primären Internetzugang und den anderen als Backup zu benutzen; eine weitere Möglichkeit ist eine default Route zu beiden Providern einzurichten und den Rest den Routing Chips zu überlassen. Wenn ein Teilnehmer aber schon das Geld ausgibt, um Multihoming zu ermöglichen und mit zwei ISPs einen Vertrag abschließt, dann ist wohl keine dieser beiden Lösungen akzeptabel. In diesem Fall ist BGP die beste Option.

Wieder sollten der eingehende und der ausgehende Verkehr einzeln analysiert werden. Für ankommenden Verkehr wird die höchste Zuverlässigkeit erreicht, wenn alle internen Routen an beide Provider weitergegeben werden. Diese Konfiguration stellt sicher, dass alle Ziele innerhalb des AS des Teilnehmers durch beide ISPs erreichbar sind. Obwohl beide Provider dieselben Routen bekommen, gibt es Fälle, in denen eingehender Verkehr einen der Wege vorziehen sollte. BGP ermöglicht es Ihnen, diese Einstellungen vorzunehmen.

Bei ausgehendem Verkehr sollte gut überlegt werden, welche Routen vom Provider angenommen werden. Wenn von beiden Providern alle Routen angenommen werden, wird die beste Route für jedes Internetziel gewählt. In manchen Fällen wird aber ein Provider für den generellen Internetzugang vorgezogen, während der andere nur für bestimmte Ziele zu verwenden ist. In diesem Fall können vom bevorzugten Provider alle Routen abgenommen werden und vom anderen nur ein Teil der Routen. Es kann zum Beispiel sein, dass Sie den zweiten Provider nur für den Zugang zu dessen Teilnehmern und als Backup für den Rest des Internets verwenden wollen (siehe Abbildung 2.15). Dieser Router sendet dann nur die Routen seiner Kunden und der Teilnehmer und konfiguriert eine default Route zum sekundären ISP, die verwendet wird, wenn der primäre ISP ausfällt.

Die Full Routes, die ISP1 verschickt, enthalten wahrscheinlich die Routen zu den Kunden von ISP2. Da dieselben Routen aber auch von ISP2 angeboten werden, nehmen die Router des Teilnehmers den Weg durch ISP2, da er kürzer ist. Wenn der Link zu ISP2 ausfällt, benutzt der Teilnehmer die längeren Routen über ISP1 und den Rest des Internets, um an die Kunden von ISP2 zu gelangen.

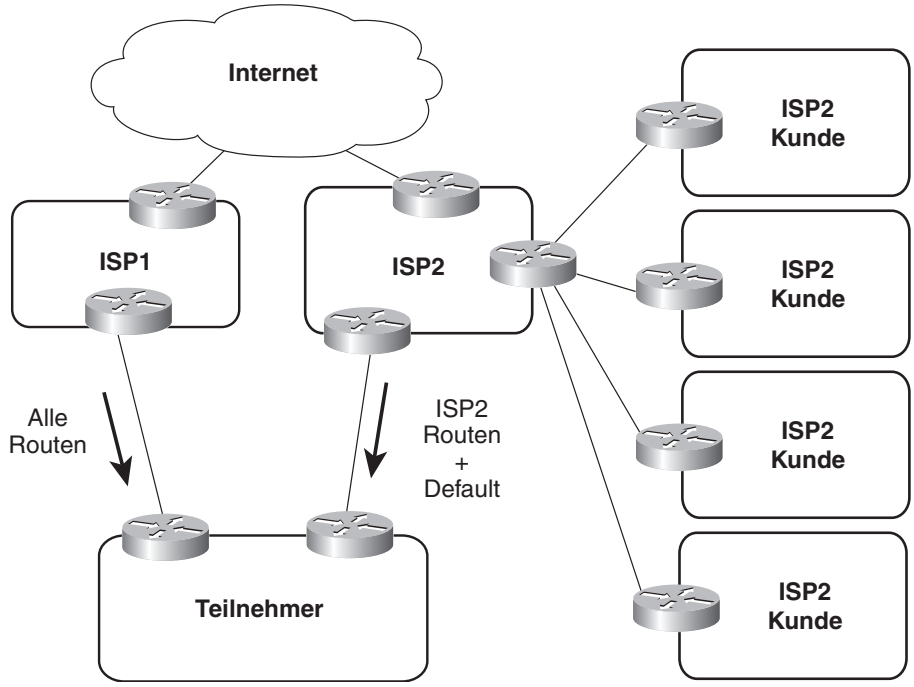

Abb. 2.15: ISP1 ist der bevorzugte Provider für den Großteil des Internets; ISP2 wird nur verwendet, um dessen andere Kunden zu erreichen und dient als Backup- Internetzugang.

Der Teilnehmer benutzt normalerweise ISP1, um Ziele, die nicht im Kundenbereich von ISP2 liegen, zu erreichen. Wenn einige dieser genaueren Routen über ISP1 verloren gehen, dann benutzt der Teilnehmer die default Route durch ISP2.

Wenn Router, CPU und Speicher verhindern, dass alle Routen angenommen werden können, ist die beste Lösung, von beiden ISPs partielle Routen anzunehmen. Jeder Provider sendet die Routen zu den eigenen Kunden und der Teilnehmer programmiert default Routen zu beiden Providern. In diesem Fall wird zwar ein Teil der Routing-Genauigkeit verloren, dafür werden aber Router-Ressourcen gespart.

Eine weitere Möglichkeit wäre, dass jeder ISP die Routen seiner Kunden verschickt und dazu noch die Routen zu den Kunden der upstream Provider. In Abbildung 2.16 ist ISP1 zum Beispiel an Sprint angeschlossen, während ISP2 mit MCI verbunden ist. Die Routen, die ISP1 an den Teilnehmer sendet, bestehen aus allen Routen von ISP1-Kunden und allen Routen zu Sprint Kunden. Die von ISP2 kommenden Routen bestehen aus denen der Kunden von

ISP2 und denen der Kunden von MCI. Der Teilnehmer weist außerdem auf default Routen bei beiden Providern hin. Auf Grund der Größe der beiden Backbone Service Provider hat der Teilnehmer genügend Routen, um für die meisten Ziele effiziente Routing-Entscheidungen zu treffen. Trotzdem sind die Routing-Tabellen wesentlich kleiner, als sie es mit allen Routen wären.

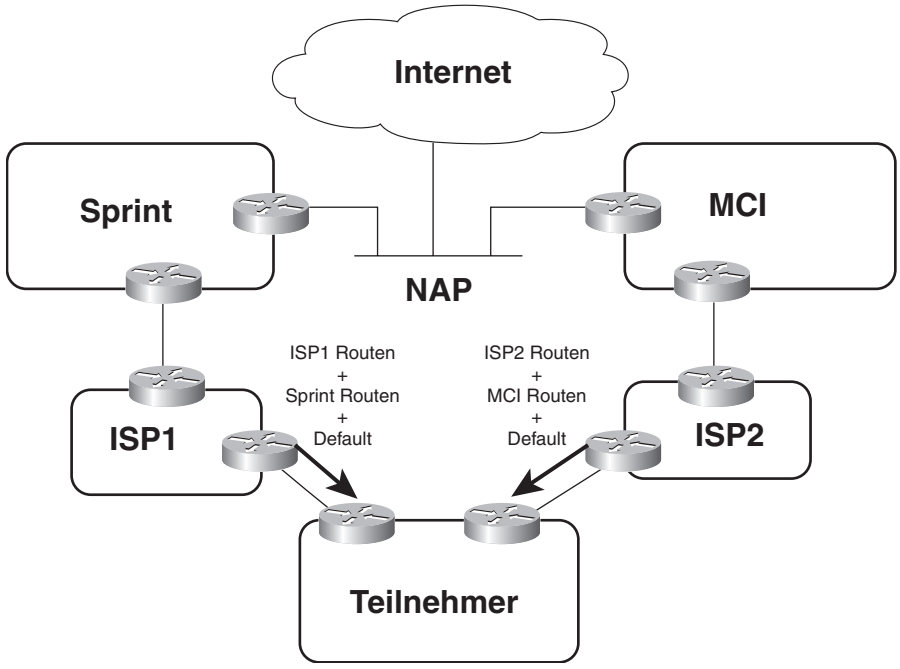

Abb. 2.16: Der Teilnehmer nimmt partielle Routen von beiden ISPs. Diese bestehen aus den Routen der Kunden der ISPs und den Kunden-Routen von deren upstream Provider.

Der Rest dieses Kapitels (nach zwei kurzen, warnenden Abschnitten) beschäftigt sich mit dem Betrieb von BGP und den Möglichkeiten, die es zur Umsetzung von Richtlinien für ein- und ausgehenden Verkehr bietet.

2.2.4 Eine Bemerkung zur »Lastenverteilung«

Die wichtigsten Vorteile von Multihoming sind die Absicherung und – vielleicht nicht ganz so wichtig – die erhöhte Bandbreite. Erhöhte Bandbreite bedeutet jedoch nicht, dass beide Links mit derselben Effizienz benutzt werden. Sie sollten nicht erwarten, dass sich der Verkehr genau 50:50 über die beiden ISPs verteilt, einer der ISPs ist fast immer besser angeschlossen als der andere. Es kann zum Beispiel sein, dass ein ISP oder sein upstream Provider

bessere Router oder bessere physische Links oder mehr NAP-Verbindungen als der andere ISP hat oder dass ein ISP topologisch gesehen an den öfter benutzten Zielen näher dran ist als der andere.

Dies bedeutet jedoch nicht, dass Sie nicht durch das Aufbringen von viel Zeit und Geduld die Routen-Präferenzen so einstellen können, dass der Verkehr ziemlich gleich verteilt ist. Das Problem ist jedoch, dass Sie Ihre Internetleistung wahrscheinlich verringern, wenn Sie den Verkehr dazu zwingen, weniger optimale Routen zu verwenden, nur um die Last zu verteilen. Alles, was Sie durch eine solche Aktion wirklich erreichen ist, dass die Benutzerzahlen bei beiden ISPs recht ähnlich sind. Seien Sie nicht besorgt, wenn 75 Prozent Ihres Verkehrs den einen Link verwenden und nur 25 Prozent den anderen benutzen. Multihoming ist wichtig für die Absicherung und um die Routing-Effizienz zu erhöhen, Lastenverteilung spielt eigentlich keine Rolle.

2.2.5 BGP-Gefahren

Das Erschaffen einer Peering-Beziehung bei BGP beinhaltet eine interessante Kombination von Vertrauen und Misstrauen. Der BGP-Peer ist in einem anderen AS, also müssen Sie dem Administrator dieses Systems vertrauen und darauf bauen, dass er weiß, was er tut. Gleichzeitig unternehmen Sie, wenn Sie schlau sind, jeden Schritt, um sich vor einem möglichen Fehler im andern System zu schützen. Beim Aufbau einer BGP-Peering-Verbindung ist die Paranoia ein guter Weggefährte.

Erinnern Sie sich an die frühere Beschreibung einer Routen-Angabe, in der diese als Versprechen definiert wurde, ein Paket an die veröffentlichte Adresse zu vermitteln. Die Routen, die Sie bekannt geben, beeinflussen die Pakete, die sie empfangen und die Routen, die sie empfangen, beeinflussen die Pakete, die Sie versenden. In einer guten BGP-Peering-Beziehung sollten beide Seiten genau wissen, welche Routen in welcher Richtung veröffentlicht werden müssen. Wieder müssen die beiden Arten von Verkehr, der eingehende und der ausgehende, einzeln behandelt werden. Jeder Peer sollte sich sicher sein, dass er nur die richtigen Routen angibt und sollte Routen-Filter oder andere Mittel wie AS_PATH-Filter, die in Kapitel 3 beschrieben werden, verwenden, um auch sicherzustellen, dass nur die richtigen Routen empfangen werden.

Ihr ISP zeigt vielleicht ein wenig Geduld, wenn Sie bei Ihrer BGP-Konfiguration Fehler machen, die schlimmsten Probleme entstehen aber, wenn Fehler auf beiden Seiten der Peering-Beziehung gemacht werden. Nehmen wir zum Beispiel an, dass Sie aus irgendeinem Grund fälschlicherweise 207.46.0.0/16 an Ihren ISP bekannt geben. Der ISP filtert diese falsche Route fahrlässiger-

weise nicht heraus, also wird sie an den Rest des Internets weitergegeben. Dieser CIDR-Block gehört allerdings Microsoft und Sie haben durch Ihre Angabe gerade behauptet, dass Sie eine Route zu diesem Ziel haben. Ein beträchtlicher Teil der Internetbenutzer könnte nun Ihre Route über Ihre Domäne als die beste ansehen und Ihnen Pakete senden. So werden Sie von ungewollten Paketen überflutet und zerstören außerdem den Verkehr, der eigentlich zu Microsoft gehen sollte. Microsoft wird davon nicht begeistert sein und wenig Verständnis haben.

Abbildung 2.17 zeigt ein weiteres Beispiel eines BGP-Routing-Fehlers. Dasselbe Netzwerk wurde bereits in Abbildung 2.15 gezeigt, hier sind aber die Routen der Kunden von ISP2, die Sie von diesem erlernt haben, fälschlicherweise an ISP1 veröffentlicht worden.

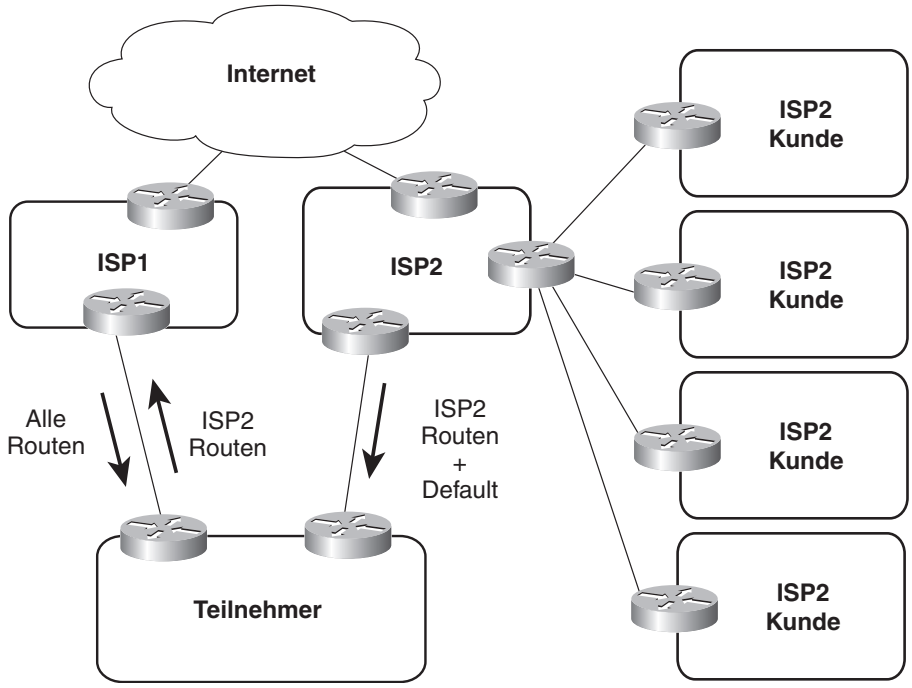

Abb. 2.17: Dieser Teilnehmer gibt Routen, die er von ISP2 erlernt hat, an ISP1 bekannt und lädt so Pakete ein, die für ISP2 und dessen Kunden gedacht sind, seine Domäne zu durchqueren.

Es ist sehr wahrscheinlich, dass ISP1 und seine Kunden die Domäne des Teilnehmers als besten Pfad zu ISP2 und dessen Kunden sehen. In diesem Fall wird kein Verkehr vernichtet, da der Teilnehmer ja wirklich eine Route zu

ISP2 hat. Der Teilnehmer wird also zur Transitdomäne für Pakete von ISP1 zu ISP2, was natürlich den eigenen Verkehr verlangsamt. Und da die Routen von ISP2 zu ISP1 immer noch durch das Internet führen, hat der Teilnehmer bei ISP2 asymmetrisches Routen verursacht.

Dieser Abschnitt soll klarstellen, dass BGP von Grund auf für die Kommunikation zwischen Autonomen Systemen entwickelt wurde. Eine erfolgreiche und zuverlässige BGP-Peering-Beziehung benötigt ein gutes Wissen sowohl über die anzugebenden Routen, als auch über die Routing-Richtlinien der beiden Partner.

2.3 BGP-Grundlagen

BGP baut, ähnlich wie EGP, eine auf Unicast basierende Verbindung zu jedem seiner BGP sprechenden Peers auf. Um die Zuverlässigkeit der Peer-Verbindung zu erhöhen, benutzt BGP TCP (Port 179) als grundlegendes Übertragungsmittel. Die Update-Mechanismen von BGP werden ebenfalls vereinfacht, indem die TCP Schicht die Anerkennungen, die Weitermeldung und das Sequencing übernimmt. Da BGP auf TCP aufbaut, muss eine eigene Verbindung zu jedem Peer geschaffen werden.

BGP ist ein Distance Vector-Protokoll, weil jede BGP Node von downstream Nachbar-Routen aus deren Routing-Tabellen benötigt; die Node berechnet ihre Routen basierend auf diesen veröffentlichten Routen und gibt die Ergebnisse an upstream Nachbarn weiter. Andere Distance Vector-Protokolle beschreiben Entfernungen jedoch mit einer einzigen Zahl, die für den Hop-Count steht oder bei IGRP und EIGRP eine Summe der gesamten Schnittstellenverzögerungen und der niedrigsten Bandbreite darstellt. BGP benutzt stattdessen eine Liste von AS-Nummern, durch die ein Paket reisen muss, um am Ziel anzukommen (siehe Abbildung 2.18). Da diese Liste die Route beschreibt, die ein Paket nehmen muss, ist BGP auch ein *Path vector* Routing-Protokoll, anders als die traditionellen Distance Vector-Protokolle. Die Liste der AS-Nummern, die mit einer BGP-Route zusammenhängen, wird *AS_PATH* genannt und ist eine von vielen *Pfadeigenschaften,* die mit jeder Route verbunden werden. Pfadeigenschaften (engl. Path Attributes) werden in einem späteren Abschnitt genauer beschrieben.

In Kapitel 1 wurde bereits vermittelt, dass EGP kein wahres Routing-Protokoll ist, weil es keinen voll entwickelten Algorithmus für die Berechnung der kürzesten Route hat, zudem kann es keine Route Loops entdecken. Die Eigenschaft AS_PATH qualifiziert BGP dagegen als ein Routing-Protokoll,

da beide Eigenschaften erfüllt werden. Die kürzeste inter-AS-Route wird einfach durch die niedrigste Zahl von AS-Nummern bestimmt. In Abbildung 2.18, empfängt AS7 zwei Routen zu 207.126.0.0/16. Eine der Routen hat vier AS-Hops, die andere hat drei. AS7 wählt den kürzesten Weg, (4,2,1).

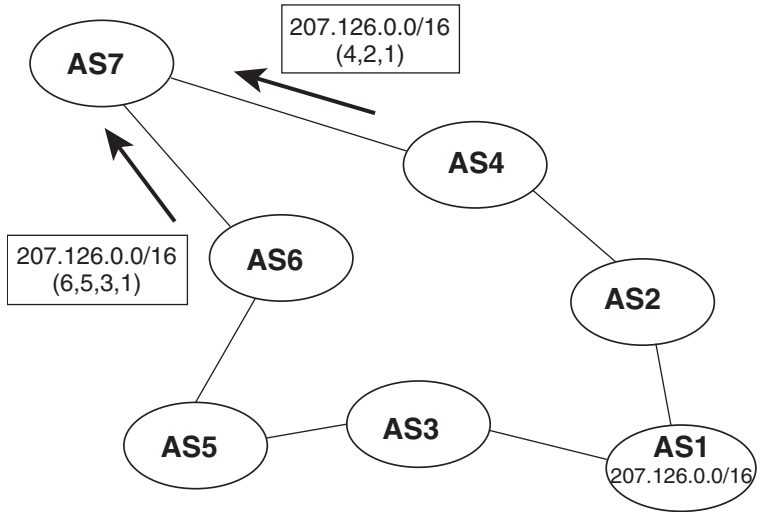

Abb. 2.18: BGP bestimmt den kürzesten Loop-freien inter-AS-Pfad durch eine Liste von AS-Nummern, die AS_PATH-Attribute genannt werden.

Route Loops können durch die AS_PATH-Eigenschaft auch sehr leicht entdeckt werden. Wenn ein Router ein Update empfängt, das seine lokale AS-Nummer in der AS_PATH enthält, weiß er, dass es einen Routing Loop gibt. In Abbildung 2.19 hat AS7 eine Route an AS8 veröffentlicht. AS8 gibt die Route AS9 an, dieses sendet sie zurück an AS7. AS7 sieht die eigene AS-Nummer in der AS_PATH und nimmt das Update nicht an, so wird ein möglicher Routing Loop verhindert.

BGP gibt keine Auskunft über Details der Topologien innerhalb der AS. Da BGP nur einen Baum von Autonomen Systemen sieht, kann man sagen, dass BGP eine andere Übersicht über das Internet hat als IGP, welches nur Topologien innerhalb eines AS sieht. Da diese andere Übersicht nicht wirklich mit der Übersicht von IGPs kompatibel ist, verwenden Cisco-Router eine eigene Routing-Tabelle für BGP-Routen. Beispiel 2.13 zeigt eine typische BGP-Routing-Tabelle, die mit dem Befehl **show ip bgp** angezeigt wird.

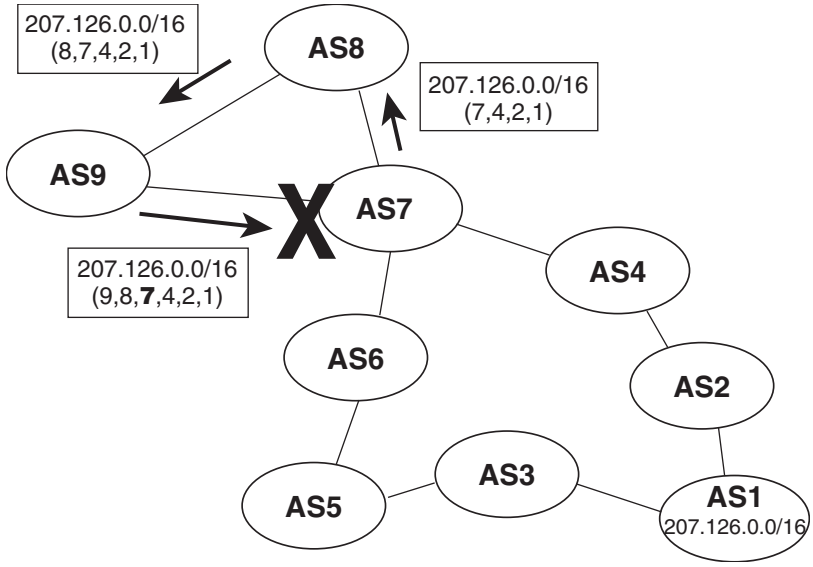

Abb. 2.19: Wenn ein BGP-Router seine eigene AS-Nummer in der AS_PATH einer Route von einem anderen AS sieht, dann wird das Update abgewiesen.

*Beispiel 2.13: Der Befehl **show ip bgp** zeigt die BGP-Routing-Tabelle.*

```
route-server>show ip bgp
BGP table version is 4639209, local router ID is 12.0.1.28
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
*  3.0.0.0     192.205.31.225                         0 7018 701 80 i
*              192.205.31.161                         0 7018 701 80 i
*>             192.205.31.33                          0 7018 701 80 i
*              192.205.31.97                          0 7018 701 80 i
* 4.0.0.0      192.205.31.225                         0 7018 1 i
*              192.205.31.161                         0 7018 1 i
*>             192.205.31.33                          0 7018 1 i
*              192.205.31.97                          0 7018 1 i
* 6.0.0.0      192.205.31.226                         0 7018 568 721 1455 i
*              192.205.31.225                         0 7018 568 721 1455 i
*              192.205.31.161                         0 7018 701 6113 568 721 1455 i
*>             192.205.31.34                          0 7018 568 721 1455 i
*              192.205.31.33                          0 7018 568 721 1455 i
*              192.205.31.97                          0 7018 1239 568 721 1455 i
* 9.2.0.0/16   192.205.31.225                         0 7018 1 1673 1675 i
*              192.205.31.161                         0 7018 701 1673 1675 i
 --More--
```

Obwohl die BGP-Routing-Tabelle in Beispiel 2.13 anders aussieht als die AS-interne Routing-Tabelle, die mit dem Befehl**show ip route** abgerufen wird, enthält sie dieselben Elemente. Die Tabelle enthält Zielnetzwerke, next Hop Router, und ein Maß, durch das der kürzeste Pfad ausgewählt werden kann. Die Spalten **Metric**, **LocPrf**, und **Weight** werden später noch besprochen, im Moment interessiert die Spalte **Path** am meisten. Diese Spalte enthält die AS_PATH-Eigenschaften eines jeden Netzwerks. Jede AS_PATH endet mit einem **i**, was nach der Legende **ORIGIN codes** bedeutet, dass der Pfad bei einem IGP endet.

Für jedes Zielnetzwerk sind außerdem mehrere next Hops aufgelistet. Die BGP-Tabelle enthält, anders als die AS-interne Routing-Tabelle, die nur die Routen enthält, die gerade benutzt werden, alle bekannten Routen. Ein **>** nach dem * (valid) in der am weitesten links gelegenen Spalte zeigt, welche Route der Router momentan benutzt. Die beste Route ist die mit der kürzesten AS_PATH. Wenn mehrere Routen gleichrangige Pfade haben wie in der Tabelle von Beispiel 2.13, dann muss der Router andere Kriterien benutzen, um die Route zu wählen. Dies wird später besprochen.

Wenn es parallele, gleichwertige Routen zu einem Ziel gibt wie in Beispiel 2.13, dann wählt Ciscos EBGP-Ausführung im Gegensatz zu anderen IP Routing-Protokollen, automatisch eine Route aus und verteilt die Belastung nicht auf bis zu vier Routen. Wie bei anderen IP Routing-Protokollen wird der Befehl **maximum-Paths** benutzt, um die Grundeinstellung der maximalen Zahl von parallelen Routen zwischen 1und 6 einzugeben. Der Lastenausgleich ist nur mit EBGP möglich. IBGP kann nur einen Link verwenden.

Der Nachbar, mit dem ein BGP sprechender Router gepeert wird, kann entweder im selben AS oder in einem andern AS sein. Wenn der Nachbar in einem anderen AS ist, ist er ein *externer Peer* und das BGP wird *external BGP* (EBGP) genannt. Wenn der Nachbar im selben AS ist, ist der Nachbar ein *interner Peer* und das BGP wird *internal BGP* (IBGP) genannt. Bei der Konfiguration von IBGP gibt es mehrere besondere Themen, die unter dem Abschnitt »IBGP- und IGP-Synchronisation« besprochen werden.

Wenn zwei Nachbarn eine BGP-Peer-Verbindung aufbauen, teilen sie zuerst die Informationen ihrer BGP-Routing-Tabellen miteinander. Nach diesem Austausch gibt es weitere Teil-Updates – das heißt sie tauschen nur Routing-Informationen aus, wenn sich etwas ändert und geben selbst dann nur die Informationen über die Veränderung weiter. Da BGP keine regelmäßigen Routing-Updates verwendet, müssen die Peers Keepalive-Nachrichten austauschen, um sicherzustellen, dass die Verbindung weiterbesteht. Die Grundeinstellung für Keepalive-Nachrichten ist bei Cisco 60 Sekunden

(RFC 1771 gibt keine genaue Keepalive-Zeit an); wenn drei Zeitabstände vergehen (180 Sekunden), ohne dass ein Peer eine Keepalive-Nachricht empfängt, dann stuft dieser Peer den anderen als down ein. Diese Abstände können Sie mit dem Befehl **timers bgp** verändern.

2.3.1 BGP-Nachrichtenarten

Bevor eine BGP-Peer-Verbindung hergestellt werden kann, müssen die Nachbarn den Standard TCP three-way Handshake durchführen und eine TCP-Verbindung zu Port 179 herstellen. TCP bietet die Zerteilungs-, Weitermeldungs-, Anerkennungs- und Sequencing-Funktionen, die für eine stabile Verbindung notwendig sind, und erspart somit BGP diese Arbeit. Alle BGP-Nachrichten sind Unicasts, die an andere Nachbarn über eine TCP-Verbindung gesendet werden.

BGP verwendet vier Nachrichtenarten:

- Open
- Keepalive
- Update
- Notification

Der folgende Abschnitt beschreibt, wie diese Nachrichten benutzt werden; eine vollständige Übersicht der Nachrichtenformate und der Variablen jedes Feldes finden Sie im Abschnitt »BGP-Nachrichtenformate.«

Open-Nachricht

Nachdem die TCP-Session eröffnet ist, senden beide Nachbarn Open-Nachrichten. Jeder Nachbar verwendet diese, um sich zu identifizieren und seine BGP-Parameter für den Betrieb von BGP anzugeben. Die Open-Nachricht enthält folgende Informationen:

- **BGP-Versionsnummer –** Hier wird die BGP-Version veröffentlicht, die beim Sender der Nachricht läuft (2, 3, oder 4). Die Grundeinstellung der Router ist Version 4, sie kann aber mit dem Befehl **neighbor version** geändert werden. Wenn ein Nachbar eine frühere Version von BGP benutzt, wird die Open-Nachricht mit Version 4 zurückgewiesen; der BGP-4-Router wechselt dann auf BGP-3 und sendet eine weitere Open-Nachricht mit dieser Version. Diese Verhandlung geht weiter, bis sich beide Nachbarn auf eine Version einigen.

– **Autonome Systemnummer –** Dies ist die AS-Nummer des sendenden Routers. Sie bestimmt, ob EBGP (wenn die AS-Nummer des Nachbarn anders ist) oder IBGP (wenn die AS-Nummern gleich sind) verwendet wird.

– **Hold Time –** Dies ist die maximale Sekundenzahl, die ein Router auf eine weitere Keepalive- oder eine Update-Nachricht warten darf. Die Hold Time muss entweder 0 Sekunden (dann werden keine Keepalives verschickt) oder mindestens drei Sekunden dauern; die Cisco Hold Time-Grundeinstellung ist 180 Sekunden. Wenn die Hold Times der Nachbarn sich unterscheiden, wird die kleinere der beiden Zeiten akzeptiert.

– **BGP Identifier –** Dies ist eine IP-Adresse, die den Nachbarn identifiziert. Cisco IOS bestimmt den BGP Identifier genau wie die OSPF-Router ID: Die numerisch höchste Loopback-Adresse wird verwendet; wenn keine Loopback-Schnittstelle mit einer IP-Adresse konfiguriert wurde, wird die numerisch höchste IP-Adresse an einer physischen Schnittstelle ausgewählt.

– **Freiwillige Parameter –** Dieses Feld wird für freiwillige Mittel wie Authentisierung, Multi-Protokoll-Unterstützung und Route Refresh verwendet.

Keepalive-Nachricht

Wenn ein Router die Parameter in einer Open-Nachricht akzeptiert, antwortet er mit einem Keepalive. Weitere Keepalives werden alle 60 Sekunden in der Cisco-Grundeinstellung oder in Abständen von einem Drittel der akzeptierten Hold Time gesendet.

Update-Nachricht

Updates geben neue Routen, nicht mehr gültige Routen oder beides an. Updates enthalten folgende Informationen:

– **Network Layer Reachability Information (NLRI) –** Dies sind (Länge, Präfix) Tupel, die IP-Adress-Präfixe und deren Länge bekannt geben. Wenn 206.193.160.0/19 zum Beispiel bekannt gegeben würde, würde der Länge-Teil das /19 spezifizieren und das Präfix-Teil 206.193.160.

– **Pfadeigenschaften –** Die Pfadeigenschaften, die später in einem gleichnamigen Abschnitt beschrieben werden, sind Charakteristika der veröffentlichten NLRI. Die Eigenschaften bieten Informationen, die BGP ermöglichen, die kürzeste Route zu wählen, Routing Loops zu entdecken, und Routing-Richtlinien bestimmen.

– **Zurückgenommene Routen –** Dies sind (Länge, Präfix) Tupel, die Ziele beschreiben, die unerreichbar geworden sind und zurückgezogen wurden.

Obwohl mehrere Präfixe im NLRI-Feld enthalten sein können, beschreibt jedes Update nur eine BGP-Route (da die Pfadeigenschaften nur einen einzigen Pfad beschreiben, dieser Pfad kann allerdings zu mehreren Zielen führen). Dies zeigt erneut, dass BGP eine ganz andere Übersicht über ein Netzwerk hat als IGP, dessen Routen immer zu einer einzigen Zieladresse führen.

Notification-Nachricht (Benachrichtigung)

Die Notification-Nachricht wird verschickt, wann immer ein Fehler gefunden wird, und führt immer dazu, dass die BGP-Verbindung geschlossen wird. Der Abschnitt »BGP-Nachrichtenformate« enthält eine Liste möglicher Fehler, die eine Benachrichtigung erzeugen.

Ein Beispiel ist die Verhandlung der BGP-Version zwischen den Nachbarn. Wenn nach dem Erstellen der TCP-Verbindung ein BGP-3-Sprecher eine Open-Nachricht mit Version 4 bekommt, dann antwortet der Router mit einer Notification-Nachricht, in der steht, dass die Version nicht unterstütz wird. Die Verbindung wird geschlossen und der Nachbar versucht mit BGP-3 eine neue Verbindung aufzubauen.

2.3.2 Die BGP Finite State Machine

Die drei Stufen beim Aufbau und Betrieb einer BGP-Verbindung können mit einer Finite State Machine beschrieben werden. Abbildung 2.20 und Tabelle 2.4 zeigen die komplette BGP Finite State Machine und die Input-Ereignisse, die zu Statusänderungen führen können.

Tabelle 2.4 Die Input-Events (IE) von Abb. 2.20

IE	Beschreibung
1	BGP Start
2	BGP Stop
3	BGP Transportverbindung offen
4	BGP Transportverbindung zu
5	BGP Transportverbindung offen gescheitert
6	BGP Transport fatal error
7	ConnectRetry Timer abgelaufen

Tabelle 2.4 Die Input-Events (IE) von Abb. 2.20 (Forts.)

IE	Beschreibung
8	Hold Timer abgelaufen
9	Keepalive Timer abgelaufen
10	Empfang Open-Nachricht
11	Empfang Keepalive-Nachricht
12	Empfang Update-Nachricht
13	Empfang Notification-Nachricht

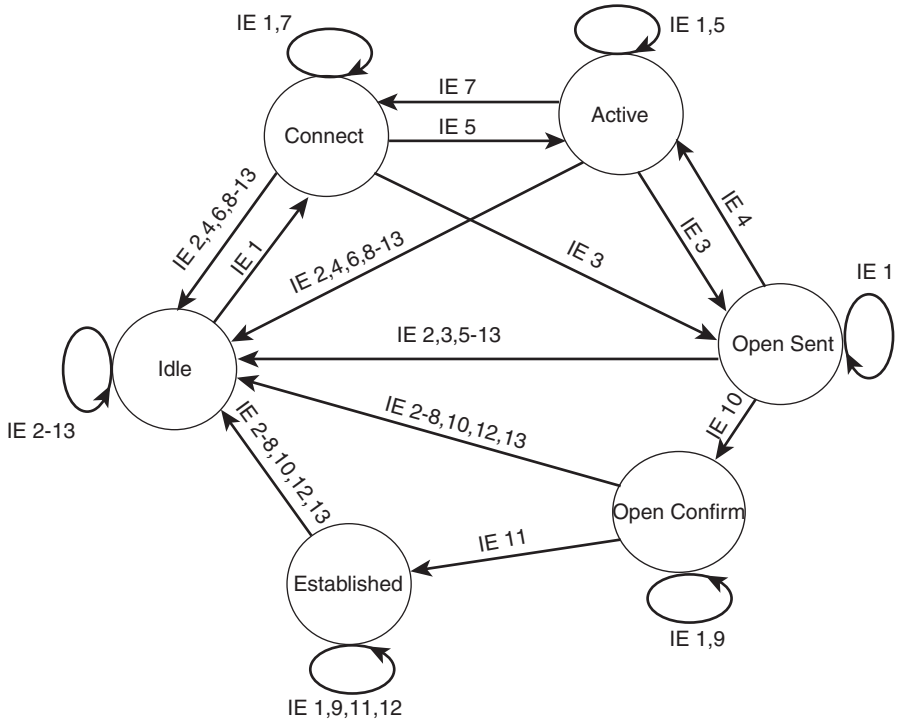

Abb. 2.20: Die BGP Finite State Machine

Die folgenden Abschnitte geben eine kurze Beschreibung der sechs Status, die in Abbildung 2.20 gezeigt werden.

Idle-Status (Leerlauf)

BGP fängt immer im Idle-Status (Leerlauf-Status) an, in dem alle eingehenden Verbindungen abgelehnt werden. Im Falle eines Starts (IE 1) initialisiert

der BGP-Prozess alle BGP-Ressourcen, startet den ConnectRetry Timer, initialisiert eine TCP-Verbindung zum Nachbarn, hört auf die TCP-Initialisierung des Nachbarn und ändert den Status auf Connect um. Der Start Event wird durch eine Konfiguration des BGP-Prozesses oder durch das Zurücksetzen eines existierenden Prozesses oder des BGP-Prozesses durch Router-Software hervorgerufen.

Ein Fehler führt dazu, dass ein BGP-Prozess in den Idle-Status kommt. Hier kann der Router automatisch versuchen, einen Start Event zu beginnen. Allerdings sollte es Einschränkungen für den Router geben – ständiges Neustarten bei einem dauerhaften Fehler führt zu Flapping. Deshalb stellt der Router bei der ersten Umstellung in den Idle-Status den ConnectRetry Timer und kann dann nicht versuchen, BGP neu zu starten, bis der Timer abgelaufen ist. Ciscos anfängliche ConnectRetry-Zeit beträgt 60 Sekunden. Diese Zeit verdoppelt sich nach jedem fehlgeschlagenen Versuch, steigt also exponentiell, wenn der Fehler dauerhaft ist.

Connect-Status

In diesem Status wartet der BGP-Prozess darauf, dass die TCP-Verbindung hergestellt wird. Wenn diese erfolgreich ist, dann stellt das BGP den ConnectRetry Timer zurück, beendet die Initialisierung, sendet eine Open-Nachricht zum Nachbarn und geht in den OpenSent-Status über. Wenn die TCP-Verbindung nicht erfolgreich ist, dann wartet der BGP-Prozess weiter darauf, dass eine Verbindung von einem Nachbarn initiiert wird, stellt den ConnectRetry Timer zurück und geht in den Active-Status über.

Wenn der ConnectRetry Timer im Connect-Status ausläuft, wird der Timer zurückgestellt, ein weiterer Versuch gemacht, um eine TCP-Verbindung herzustellen, und der Prozess wird im Connect-Status gehalten. Jeder andere Input Event führt zum Leerlauf.

Active-Status

In diesem Status versucht BGP eine TCP-Verbindung zum Nachbarn herzustellen. Wenn dies erfolgreich ist, stellt BGP den ConnectRetry Timer zurück, beendet die Initialisierung, sendet eine Open-Nachricht an den Nachbarn und geht zu OpenSent über. Der Hold Timer wird auf vier Minuten gestellt.

Wenn der ConnectRetry Timer ausläuft, während BGP im Active-Status ist, geht der Prozess zurück in den Connect-Status und der ConnectRetry Timer wird zurückgestellt. Außerdem wird eine TCP-Verbindung zum Peer hergestellt, sodass weiter auf Nachrichten vom Peer gehört werden kann. Wenn

ein Nachbar versucht, eine TCP-Verbindung mit einer unerwarteten IP-Adresse herzustellen, dann wird der ConnectRetry Timer zurückgestellt, die Verbindung wird abgelehnt und der lokale Prozess bleibt im Active-Status. Jeder andere Input Event (außer einem Start Event, der im Active-Status ignoriert wird) bedeutet, dass in den Idle-Status geschaltet wird.

OpenSent-Status

In diesem Status wurde eine Open-Nachricht verschickt und BGP wartet auf ein Open von seinem Nachbarn. Wenn eine Open-Nachricht empfangen wird, werden alle Felder der Nachricht überprüft. Wenn irgendwelche Fehler gefunden werden, wird eine Notification-Nachricht verschickt und der Status schaltet wieder auf Leerlauf.

Wenn in der Open-Nachricht keine Fehler gefunden werden, wird eine Keepalive-Nachricht versendet und der Keepalive Timer gestartet. Die Hold Time wird ausgehandelt und der kleinere Wert wird angenommen. Wenn die Hold Time null ist, werden der Hold und der Keepalive Timer nicht gestartet. Die Peer-Verbindung wird entweder als extern oder intern eingestuft, je nach der AS-Nummer des Nachbarn, der Status wird auf OpenConfirm geschaltet.

Wenn die TCP-Verbindung ausfällt, schließt der lokale BGP-Prozess die BGP-Verbindung, setzt den ConnectRetry Timer zurück, wartet auf eine neue Verbindung zum Nachbarn und geht in den Active-Status über. Jeder andere Input Event (außer einem Start Event, der ignoriert wird) führt zum Idle-Status.

OpenConfirm-Status

In diesem Status wartet der BGP-Prozess auf eine Keepalive- oder eine Notification-Nachricht. Wenn ein Keepalive empfangen wird, geht der Status auf Established (verbunden) über. Wenn eine Notification (Benachrichtigung) empfangen oder die TCP-Verbindung unterbrochen wird, dann wird der Idle-Status wiederhergestellt.

Wenn der Hold Timer ausläuft, ein Fehler gefunden wird oder es einen Stop Event gibt, wird eine Notification-Nachricht an den Nachbarn verschickt und die BGP-Verbindung getrennt. Der Status ist dann Idle.

Established (Verbindungs-) Status

In diesem Status ist die BGP-Peer-Verbindung voll aufgebaut und die Peers können Update-, Keepalive- und Notification-Nachrichten austauschen. Wenn ein Update oder eine Keepalive-Nachricht empfangen wird, wird der

Hold Timer neu gestartet (wenn die Hold Time nicht sowieso auf Null gestellt ist). Wenn eine Notification-Nachricht empfangen wird, dann geht der Status auf Idle über. Jeder andere Event (wieder mit Ausnahme eines Start Event, der ignoriert wird) führt dazu, dass eine Notification-Nachricht gesendet wird und dass der Status auf Idle geschaltet wird.

2.3.3 Pfadeigenschaften (Path Attributes)

Eine *Pfadeigenschaft* ist eine Charakteristik einer veröffentlichten BGP-Route. Manche Pfadeigenschaften sind bereits bekannt, wie die IP-Zieladresse und der next Hop Router, da sie bei allen Routen gleich sind. Andere wie ATOMIC_AGGREGATE beziehen sich nur auf BGP und sind deshalb vielleicht nicht bekannt. Die Pfadeigenschaften sorgen nicht nur für die Grundinformationen, die für Routing notwendig sind, sondern ermöglichen zudem, dass BGP-Routing-Richtlinien mitgeteilt und durchgeführt werden können.

Jede Pfadeigenschaft fällt unter eine der folgenden vier Kategorien:

- Well-known mandatory
- Well-known discretionary
- Optional transitive
- Optional nontransitive

Die Namen dieser vier Kategorien zeigen, dass es zwei Subklassen gibt, die wiederum Subklassen haben. Eine Eigenschaft ist erstens entweder *well-known*, was bedeutet, dass sie von allen BGP-Ausführungen erkannt werden muss, oder *Optional*, was bedeutet, dass nicht alle BGP-Ausführungen sie unterstützen.

Well-known-Eigenschaften sind entweder *mandatory*, was bedeutet, dass sie in allen BGP-Update-Nachrichten enthalten sein müssen, oder sie sind *discretionary*, was bedeutet, dass sie nicht unbedingt in jeder Update-Nachricht enthalten sind.

Wenn eine optionale Eigenschaft *transitive* ist, soll der BGP-Prozess den Pfad, in dem sie enthalten ist, akzeptieren, auch wenn die Eigenschaft nicht unterstützt wird, und sollte den Pfad an seine Peers weitergeben. Wenn eine optionale Eigenschaft *nontransitive* ist, dann kann ein BGP-Prozess, der die Eigenschaft nicht kennt, das Update, in dem sie enthalten war, einfach ignorieren, der Pfad wird nicht an andere Nachbarn weitergegeben.

Tabelle 2.5 zeigt die Pfadeigenschaften und die folgenden Abschnitte erklären die verschiedenen Eigenschaften. Kapitel 3, »Konfiguration und Fehler-

beseitigung von Border Gateway-Protokoll 4«, zeigt die Konfiguration, das Filtern und die Manipulation der Pfadeigenschaften.

*Tabelle 2.5 Pfadeigenschaften**

Eigenschaft	Klasse
ORIGIN	Well-known mandatory
AS_PATH	Well-known mandatory
NEXT_HOP	Well-known mandatory
LOCAL_PREF	Well-known discretionary
ATOMIC_AGGREGATE	Well-known discretionary
AGGREGATOR	Optional transitive
COMMUNITY	Optional transitive
MULTI_EXIT_DISC (MED)	Optional nontransitive
ORIGINATOR_ID	Optional nontransitive
CLUSTER_LIST	Optional nontransitive

* Es gibt eigentlich noch mehr Eigenschaften, als hier in Tabelle 2.5 zu sehen sind; diese sind aber weder in RFC 1771 enthalten, noch werden sie von Cisco unterstützt, deshalb werden sie in diesem Buch nicht behandelt.

Die ORIGIN-Eigenschaft Attribute

ORIGIN ist eine well-known mandatory Eigenschaft, die die Quelle des Routing-Updates angibt. Wenn BGP mehrere Routen hat, verwendet es ORIGIN als einen der Faktoren, der die beste Route bestimmt. Es gibt folgende ORIGINs:

– **IGP –** Die Network Layer Reachability Information (NLRI) wurde von einem-Protokoll erlernt, das intern im Quell-AS verwendet wird. Ein IGP ORIGIN erhält die höchste Präferenz der ORIGIN-Werte. BGP-Routen bekommen einen IGP ORIGIN, wenn sie von einer IGP-Routing-Tabelle durch das **network** Statusment erlernt werden (vgl. Kapitel 3).

– **EGP –** Die NLRI wurde vom Externen Gateway-Protokoll erlernt. EGP erhält nach IGP die zweithöchste Präferenz.

– **Incomplete –** Die NLRI wurde irgendwie erlernt. Incomplete ist der ORIGIN Wert der am niedrigsten eingestuft wird. Incomplete bedeutet nicht, dass die Route irgendwie fehlerhaft ist, es bedeutet nur, dass die Informationen über die Herkunft der Route nicht komplett sind. Routen, die BGP

durch Redistribution lernt, erhalten die incomplete ORIGIN Eigenschaft, da es nicht möglich ist die ursprüngliche Herkunft der Route festzustellen.

Die AS_PATH-Eigenschaft

AS_PATH ist eine well-known mandatory Eigenschaft, die eine Sequenz von AS-Nummern verwendet, um die von der NLIR bestimmte inter-AS-Route zum Ziel zu beschreiben. Wenn ein BGP-Sprecher eine Route erzeugt – das heißt wenn er NLRI über ein neues Ziel innerhalb seines AS informiert – fügt er dem AS_PATH seine AS-Nummer hinzu. Wenn weitere BGP-Sprecher die Route dann extern weitergeben, fügen sie dem AS_PATH ihre eigenen AS-Nummern hinzu (siehe Abbildung 2.21). Folglich beschreibt der AS_PATH-Pfad alle Autonomen Systeme, durch die die Nachricht gesendet wurde, zu Anfang das letzte AS und am Ende der Liste dann das Quell-AS.

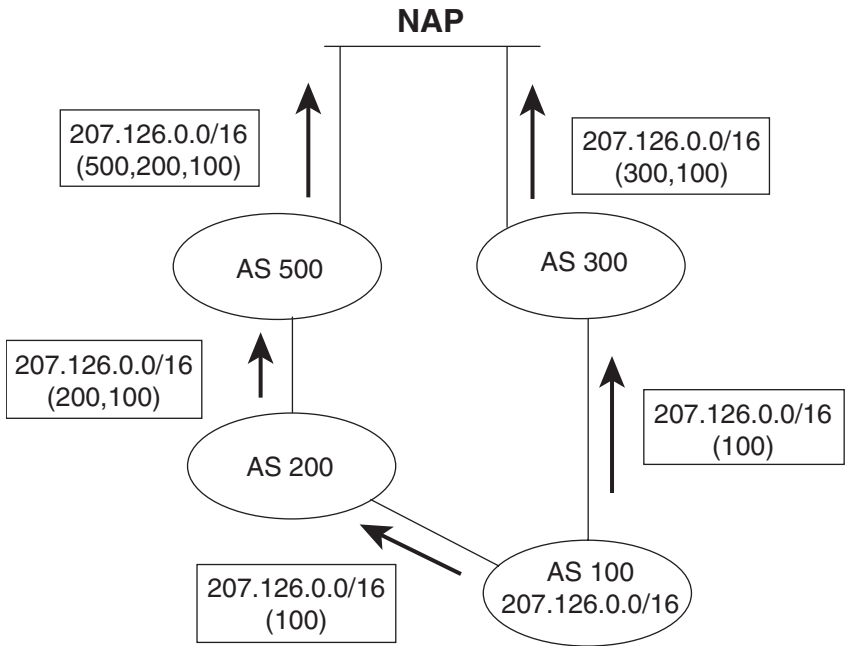

Abb. 2.21: AS-Nummern werden am Anfang der AS_PATH-Liste hinzugefügt.

Ein BGP-Router fügt seine AS-Nummern dem AS_PATH nur dann hinzu, wenn ein Update an einen Nachbarn in einem anderen AS gesendet wird. Die AS-Nummer wird dem AS_PATH nur hinzugefügt, wenn die Route an einen EBGPNachbarn veröffentlicht wird. Wenn die Route zwischen IBGP-Peers – also Peers mit derselben AS-Nummer – veröffentlicht wird, dann wird keine Nummer hinzugefügt.

Normalerweise ist es sinnlos, eine AS-Nummer mehrmals auf der Liste einzutragen, da dies für die Funktion der AS_PATH-Eigenschaft keinen Nutzen hat. Es gibt allerdings einen Fall, in dem es doch Sinn macht: Sie erinnern sich, dass die ausgehenden Routen-Angaben den eingehenden Verkehr direkt beeinflussen. Die Route vom NAP zu AS 100 in Abbildung 2.21 geht durch AS 300, da der AS_PATH Wert dieser Route geringer ist. Was aber passiert, wenn der Link zu AS 200 der von AS 100 bevorzugte Pfad für ankommenden Verkehr ist? Die Links entlang der (500,200,100) Routen könnten zum Beispiel DS3 sein, während die Links über (300,100) nur DS1 sind. Oder vielleicht ist AS 200 der primäre Provider und AS 300 nur ein absichernder Provider. Der ausgehende Verkehr wird an AS 200 gesendet, also sollte der ankommende Verkehr den gleichen Pfad nehmen.

AS 100 kann den ankommenden Verkehr beeinflussen indem es die AS_PATH-Eigenschaft seiner veröffentlichten Route verändert (siehe Abbildung 2.22). Wenn AS 100 in die Liste, die an AS 300 gesendet wird, seine eigene AS-Nummer mehrmals einfügt, dann denken die Router am NAP, dass der (500,200,100) Pfad kürzer ist. Dieser Vorgang, das zusätzliche Hinzufügen der AS-Nummern zur AS_PATH-Eigenschaft, wird *AS Path Prepending* genannt.

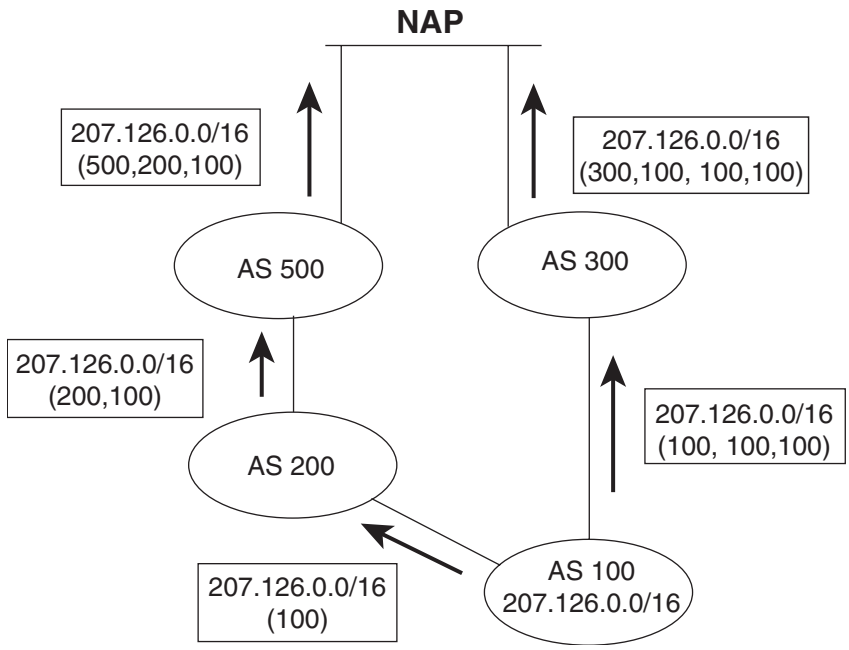

Abb. 2.22: AS 100 hat am Anfang der AS_PATH-Eigenschaft, die an AS 300 veröffentlicht wird, mehrmals seine eigene AS-Nummer eingefügt.

Die andere Funktion der AS_PATH-Eigenschaft ist wie gesagt das Verhindern von Loops. Der Vorgang ist sehr einfach: Wenn ein BGP-Router von einem externen Peer eine Route empfängt, dessen AS_PATH seine eigene AS-Nummer enthält, dann weiß der Router, dass die Route sich in einem Loop befindet. Solche Routen scheiden aus.

Die NEXT_HOP-Eigenschaft

Wie der Name schon sagt, beschreibt diese well-known mandatory Eigenschaft die IP-Adresse des next Hop Routers, der sich auf der Route zum jeweiligen Ziel befindet. Die IP-Adresse die von der BGP NEXT_HOP-Eigenschaft beschrieben wird, ist nicht immer die eines benachbarten Routers. Es bestehen folgende Regeln:

– Wenn der sendende Router, der die Route angibt, und der empfangende Router in verschiedenen Autonomen Systemen sind (externe Peers), dann ist die NEXT_HOP-Eigenschaft die IP-Adresse der Schnittstelle des sendenden Routers.

– Wenn der sendende Router und der empfangende Router im selben AS sind (interne Peers) und die NLRI des Updates von einem Ziel innerhalb desselben AS handelt, dann ist NEXT_HOP die IP-Adresse des Nachbarn, der die Route angibt.

– Wenn der sendende Router und der empfangende Router interne Peers sind und die NLRI des Updates auf ein Ziel in einem anderen AS hinweist, dann ist NEXT_HOP die IP-Adresse des externen Peers, von dem die Route erlernt wurde.

Abbildung 2.23 erläutert die erste Regel. Hier sind der veröffentlichende Router und der empfangende Router in verschiedenen Autonomen Systemen. Die NEXT_HOP-Eigenschaft ist die Schnittstellen-Adresse des externen Peers. Bis jetzt verhält sich alles wie bei jedem anderen Routing-Protokoll.

Abbildung 2.24 erläutert die zweite Regel. Diesmal sind der sendende Router und der empfangende Router im selben AS und das veröffentlichte Ziel ist ebenfalls im selben AS. Die NEXT_HOP-Eigenschaft ist die IP-Adresse des Routers, der die Route veröffentlicht hat.

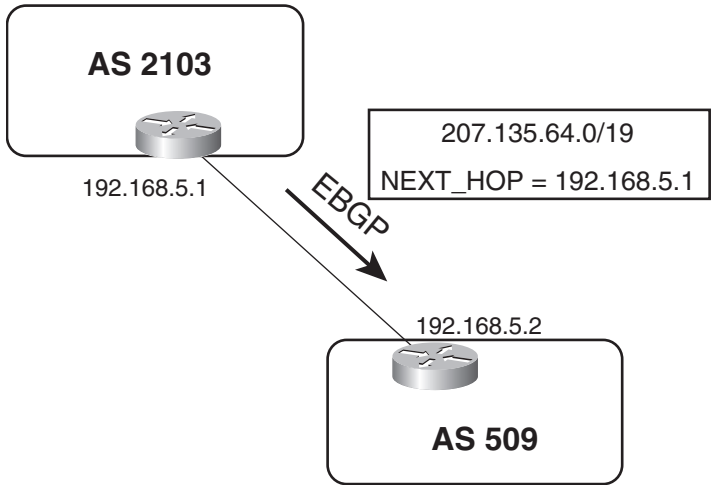

Abb. 2.23: Wenn ein BGP-Update über EBGP veröffentlicht wird, ist die NEXT_HOP-Eigenschaft die IP-Adresse des externen Peers.

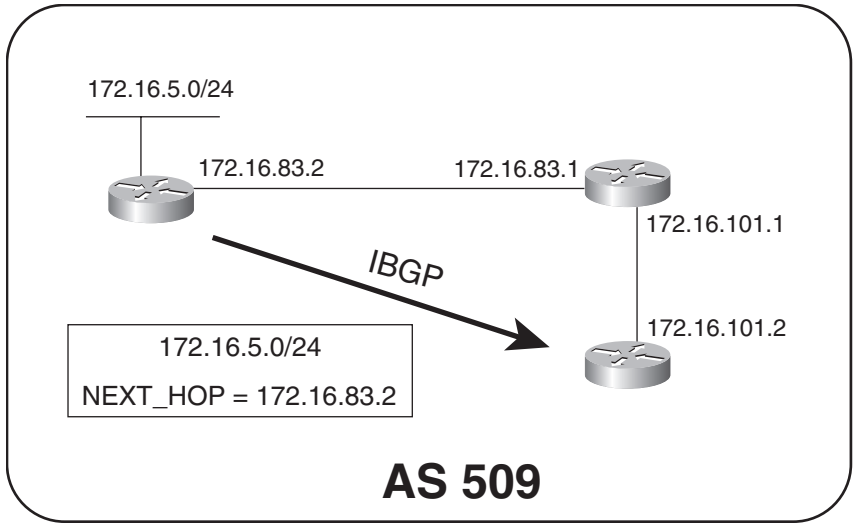

Abb. 2.24: Wenn ein BGP-Update über IBGP veröffentlicht wird und das Ziel im selben AS liegt, dann ist die NEXT_HOP-Eigenschaft die IP-Adresse des sendenden Routers.

Der bekannt gebende und der empfangende Router liegen nicht an einem gemeinsamen Link, die IBGP TCP-Verbindung geht durch einen IGP sprechenden Router. Dies wird im Abschnitt »Internes BGP« näher besprochen; im Moment ist nur wichtig, dass der empfangende Router einen rekursiven Routen-Lookup durchführen muss (rekursive Lookups werden in *Routing TCP/IP, Band I* besprochen), um ein Paket an das veröffentlichte Ziel zu senden. Zuerst wird nach Ziel 172.16.5.30 gesucht; diese Route hat einen next Hop von 172.16.83.2. Da diese IP-Adresse nicht zu einem der direkt an den Router angeschlossenen Subnets gehört, muss der Router die Route zu 172.16.83.2 nachschlagen. Diese Route, die über IGP erlernt wird, hat einen next Hop von 172.16.101.1. Das Paket kann nun losgeschickt werden. Dieses Beispiel ist wichtig, da es zeigt, wie sehr IBGP von IGP abhängig ist.

Abbildung 2.25 erläutert die dritte Regel. Hier wurde eine Route durch EBGP erlernt und dann an einen internen Peer weitergegeben. Da das Ziel in einem anderen AS liegt, ist der next Hop der Router, durch den die IBGP-Verbindung die Schnittstelle des externen Routers, von dem die Route erlernt wurde, verläuft.

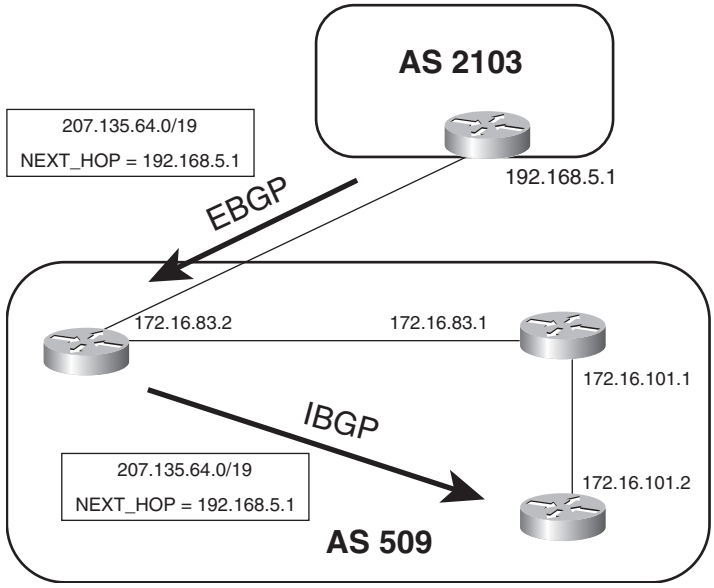

Abb. 2.25: Wenn ein BGP-Update über IBGP veröffentlicht wird und das Ziel in einem anderen AS liegt, dann ist die NEXT_HOP-Eigenschaft die IP-Adresse des externen Peers, von dem die Route erlernt wurde.

In Abbildung 2.25 muss der IBGP-Peer einen rekursiven Routen-Lookup durchführen, bevor ein Paket an 207.135.64.0/19 versendet werden kann. Es kann jedoch ein Problem entstehen. Das Netzwerk 192.168.5.0, zu dem die next Hop-Adresse gehört, gehört nicht zu AS 509. Die internen Nachbarn wissen über dieses Netzwerk nur dann Bescheid, wenn der AS-Grenz-Router dieses Netzwerk angibt. Wenn das Netzwerk nicht in den Routing-Tabellen steht, dann ist die next Hop-Adresse für 207.135.64.0/19 unerreichbar, die Pakete für dieses Ziel gehen verloren. Obwohl die Route zu 207.135.64.0/19 in der BGP-Tabelle des internen Peers steht, ist sie nicht in der IGP-Routing-Tabelle, weil die next Hop Adresse für diesen Router ungültig ist.

Bei der ersten Möglichkeit der Lösung des Problems muss sichergestellt werden, dass das externe Netzwerk, das die beiden ASs verbindet, den internen Routern bekannt ist. Obwohl statische Routen verwendet werden könnten, ist die praktische Lösungzu bevorzugen, IGP an den externen Schnittstellen im passiven Modus laufen zu lassen. Dies kann in manchen Fällen unerwünscht sein. Als zweite Lösungsmöglichkeit ist eine Konfigurations-Option zu verwenden, die dazu führt, dass der AS-Grenz-Router in AS 509 statt der IP-Adresse des externen Peers seine eigene IP-Adresse als NEXT_HOP angibt. Die internen Peers hätten dann eine next Hop Router-Adresse von 172.16.83.2, die auch IGP bekannt ist. Diese Lösung, die **next Hop-self** genannt wird, wird in Kapitel 3 genauer behandelt.

Die LOCAL_PREF-Eigenschaft

LOCAL_PREF ist eine Abkürzung von »local preference«. Diese well-known discretionary Eigenschaft wird nur in Updates zwischen internen BGP-Peers verwendet; sie wird nicht an andere Autonome Systeme weitergegeben. Die Eigenschaft wird verwendet, um anzugeben, wie sehr ein BGP-Router eine veröffentlichte Route vorzieht. Wenn ein interner BGP-Sprecher mehrere Routen zum selben Ziel bekommt, werden die LOCAL_PREF-Eigenschaften der Routen verglichen. Verwendet wird dann die Route mit der höchsten LOCAL_PREF.

Abbildung 2.26 erläutert, wie die LOCAL_PREF-Eigenschaft verwendet wird. AS 2101 bekommt Routen von zwei ISPs, ISP1 ist aber der bevorzugte Service Provider. Der Router, der mit ISP1 verbunden ist, gibt die Routen von diesem Provider mit einer LOCAL_PREF von 200 an, der mit ISP2 verbundene Router gibt die Routen seines ISP hingegen mit einer LOCAL_PREF von 100 (Grundeinstellung) an. Alle internen Peers, auch des Routers, der an ISP2 angeschlossen ist, bevorzugen nun die Routen, die von ISP1 erlernt wurden, gegenüber den Routen zu denselben Zielen über ISP2.

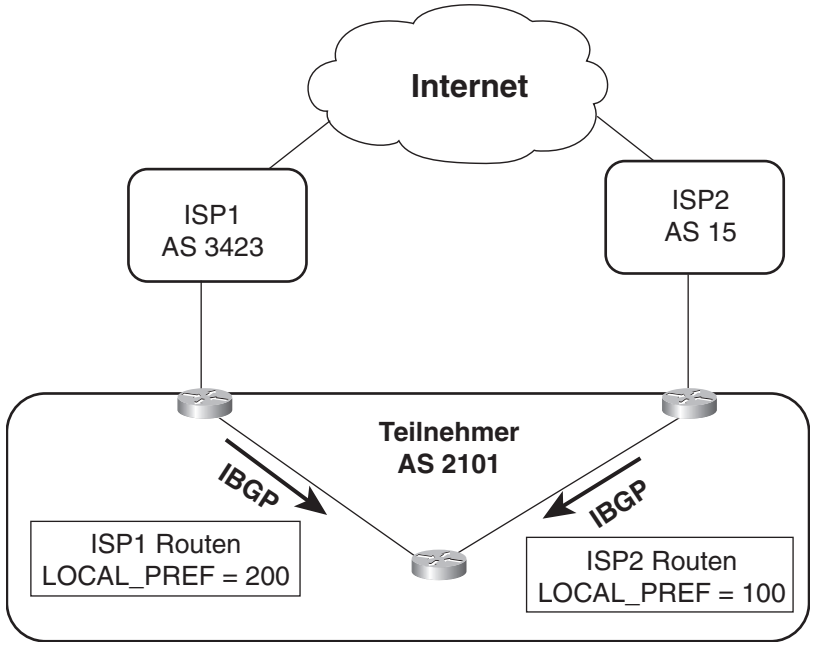

Abb. 2.26: Die LOCAL_PREF-Eigenschaft teilt internen Peers mit, wie sehr eine Route bevorzugt werden soll. Je höher der Wert, desto mehr wird eine Route bevorzugt.

Die MULTI_EXIT_DISC-Eigenschaft

Die LOCAL_PREF-Eigenschaft beeinflusst nur den Verkehr, der das AS verlässt. Um den ankommenden Verkehr zu beeinflussen, wird die Eigenschaft MULTI_EXIT_DISC, kurz MED, verwendet. Diese optional nontransitive Eigenschaft wird in EBGP-Updates getragen und ermöglicht einem AS, ein anderes AS über den bevorzugten Zugangspunkt zu informieren. Wenn alle anderen Faktoren gleich sind, vergleicht ein AS die MEDs mehrerer Routen zum gleichen Ziel. Anders als bei LOCAL_PREF, wo der höchste Wert bevorzugt wird, wird hier der niedrigste MED-Wert bevorzugt. Dies ist so, weil MED eine Metrik ist und bei einer Metrik der niedrigste Wert – die niedrigste Entfernung – bevorzugt wird.

Anmerkung

Bei BGP-2 und BGP-3 heißt die MULTI_EXIT_DISC-Eigenschaft INTER_AS-Metrik.

Abbildung 2.27 zeigt, wie MED verwendet werden kann. Hier ist ein Teilnehmer doppelt mit einem ISP verbunden. AS 525 möchte, dass der Verkehr eher den DS-3-Link benutzt, der DS-1-Link wird nur als Backup verwendet. MED wird in den Updates, die durch den DS-3-Link gehen, auf 0 gestellt (Grundeinstellung), in den Updates, die durch den DS-1-Link gehen, hat MED einen Wert von 100. Wenn die beiden Routen keine sonstigen Unterschiede aufweisen, verwendet der ISP den DS-3-Link wegen der niedrigeren MED.

Innerhalb des ISP wird zwischen den Routern IBGP verwendet. Die MEDs von AS 525 werden zwischen diesen internen Nachbarn ausgetauscht, sodass sie beide wissen, welche Route bevorzugt wird. MEDs werden jedoch vom empfangenden AS nicht weitergegeben. Wenn der ISP zum Beispiel einem anderen AS die Adresse 206.25.160.0/19 angibt, werden die MEDs des ursprünglichen AS nicht weitergegeben. Das bedeutet, dass MEDs nur verwendet werden, um Verkehr zwischen zwei direkt verbundenen Autonomen Systemen zu beeinflussen; das Bevorzugen von Routen kann über dieses benachbarte AS nur dann hinausgehen, wenn die AS_PATH-Eigenschaften verändert werden. Dies wurde bereits besprochen.

MEDs werden zudem nicht verglichen, wenn zwei Routen zum selben Ziel von zwei verschiedenen Autonomen Systemen veröffentlicht werden. Wenn der ISP in Abbildung 2.27 Angaben über 206.25.160.0/19 sowohl von AS 525 als auch von einem anderen AS empfängt, werden die MEDs von diesen Autonomen Systemen nicht verglichen. MEDs sind nur dazu gedacht, in einem AS bei mehreren Eingangspunkten die Prioritäten festzulegen.

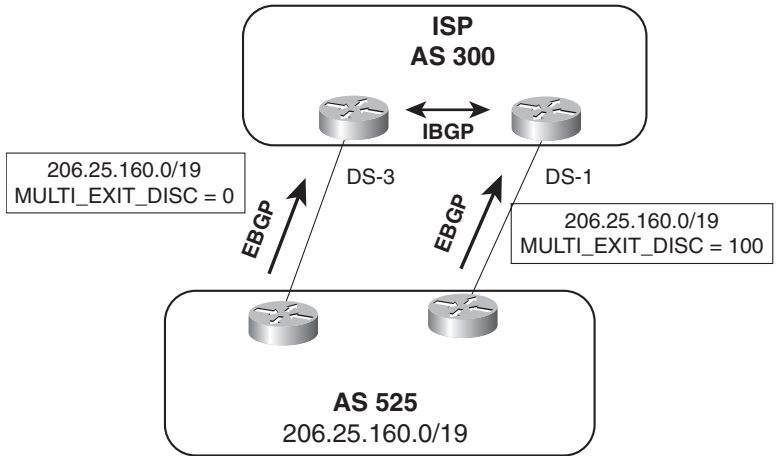

Abb. 2.27: Niedrigere MED-Eigenschaften bei Routen über den DS-3-Link führen dazu, dass der ISP diesen Link bevorzugt.

Die ATOMIC_AGGREGATE- und AGGREGATOR-Eigenschaften

Ein BGP sprechender Router kann einem anderen BGP-Sprecher überschneidende Routen bekannt geben. Überschneidende Routen sind ungleiche Routen, die zum selben Ziel führen. Die Routen 206.25.192.0/19 und 206.25.128.0/17 sind zum Beispiel überschneidend. Die erste Route ist in der zweiten enthalten, obwohl die zweite Route auch zu anderen genaueren Routen außer 206.25.192.0/19 führt.

Wenn ein Router entscheiden muss, welche Route am besten ist, wählt er immer die genauere Route aus. Bei der Angabe von Routen hat der BGP-Sprecher hingegen mehrere Möglichkeiten mit überschneidenden Routen umzugehen:

- Angabe von beiden, der genaueren und der ungenaueren Route
- Angabe der genaueren Route
- Angabe des nicht überschneidenden Teils der Route
- Zusammenfassen der Routen und Angabe der Zusammenfassung
- Angabe der ungenaueren Route
- Keine Angabe der Routen

Es wurde bereits besprochen, dass bei der Zusammenfassung (Routen-Verdichtung) einige Routen-Informationen verloren gehen und dass das Routing so weniger genau wird. Wenn in einem BGP sprechenden Router eine Zusammenfassung durchgeführt wird, gehen Details des Pfades verloren. Abbildung 2.28 zeigt diesen Verlust von Details des Pfades.

AS 3113 gibt eine zusammengefasste Adresse an die Adressen, die aus verschiedenen Autonomen Systemen stammen. Da die Zusammenfassung aus diesem AS stammt, wird dessen AS-Nummer der AS_PATH-Liste hinzugefügt. Die Pfad-Informationen zu manchen der genaueren Präfixe, die in der Zusammenfassung enthalten sind, gehen verloren.

ATOMIC_AGGREGATE ist eine well-known discretionary Eigenschaft, die benutzt wird, um downstream Router zu warnen, dass Pfad-Informationen verlorengegangen sind.

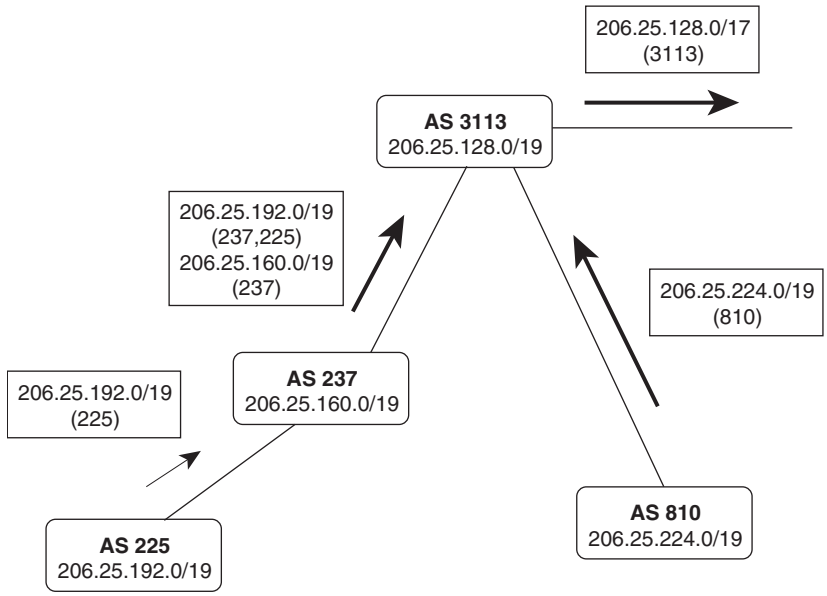

Abb. 2.28: Die Zusammenfassung von BGP-Routen führt zum Verlust von Pfad-Informationen.

Immer wenn ein BGP-Sprecher genauere Routen in eine ungenauere Verdichtung zusammenfasst (die fünfte Option auf der Liste) und somit Informationen verloren gehen, muss der BGP-Sprecher der Zusammenfassung die ATOMIC_AGGREGATE-Eigenschaft hinzufügen. Jeder downstream BGP-Sprecher, der eine Route mit der ATOMIC_AGGREGATE-Eigenschaft empfängt, kann die NLRI-Information dieser Route nicht genauer wiedergeben, beim Weitergeben der Route an andere Peers muss außerdem die ATOMIC_AGGREGATE-Eigenschaft erhalten bleiben.

Wenn die ATOMIC_AGGREGATE-Eigenschaft eingestellt ist, hat der BGP-Sprecher die Möglichkeit, die AGGREGATOR-Eigenschaft einzustellen. Diese optional transitive Eigenschaft gibt Informationen über den Ort an, an dem die Zusammenfassung durchgeführt wurde, sie enthält zum Beispiel die AS-Nummer und die IP-Adresse des Routers, von dem die zusammengefasste Route stammt (siehe Abbildung 2.29). Ciscos Ausführung von BGP verwendet die BGP-Router ID als die IP-Adresse in der entsprechenden Eigenschaft.

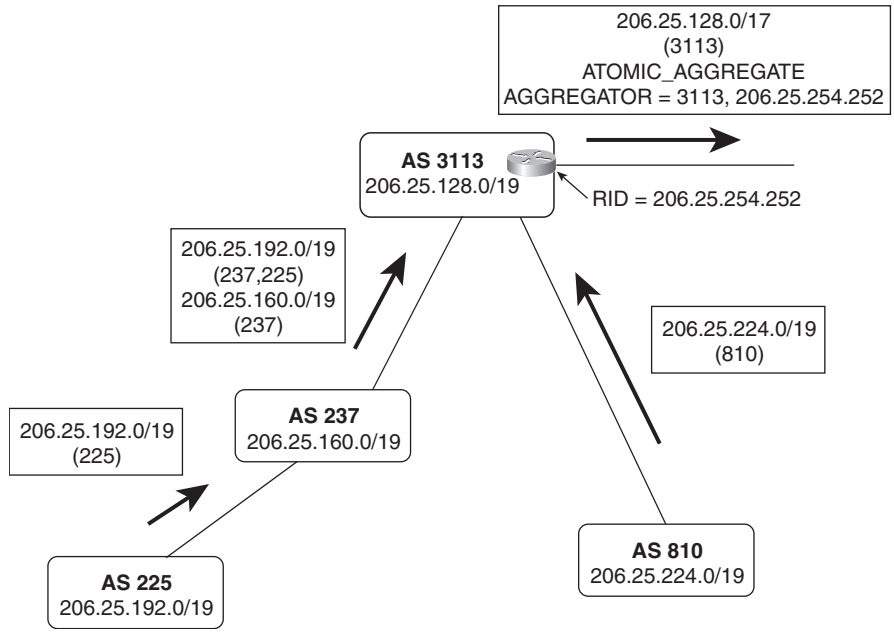

Abb. 2.29: Die ATOMIC_AGGREGATE-Eigenschaft zeigt, dass Pfad-Informationen verloren wurden, die AGGREGATOR-Eigenschaft zeigt, wo die Zusammenfassung stattfand.

Die COMMUNITY-Eigenschaft

COMMUNITY ist eine optional transitive Eigenschaft, die entwickelt wurde, um das Durchsetzen von Richtlinien zu erleichtern. Früher war es eine Cisco-eigene Eigenschaft, heute wird es in RFC 1997 veröffentlicht[8]. Die COMMUNITY-Eigenschaft dient dazu, ein Ziel als Mitglied einer Gruppe von Zielen mit einer oder mehreren gemeinsamen Eigenschaften zu identifizieren. Ein ISP kann zum Beispiel allen seinen Kunden-Routen eine COMMUNITY-Eigenschaft zuweisen. Der ISP kann dann die Eigenschaften LOCAL_PREF und MED- mithilfe des COMMUNITY-Wertes einstellen und muss dies nicht für jede Route einzeln vornehmen.

Die COMMUNITY-Eigenschaft ist ein Set von vier Oktettwerten. RFC 1997 bestimmt, dass die ersten beiden Oktette das Autonome System sind und die letzten beiden Oktette ein administrativ bestimmtes Kennzeichen, was zu einem Format von AA:NN führt. Das default Cisco-Format ist allerdings NN:AA. Diese Grundeinstellung kann durch den Befehl **ip bgp-community new-format** auf das Format RFC 1997 geändert werden.

Nehmen wir zum Beispiel an, eine Route von AS 625 hat ein COMMUNITY-Kennzeichen von 70. Die COMMUNITY-Eigenschaft ist im AA:NN-Format 625:70 und wird in hex als Verkettung der beiden Nummern dargestellt: 0x02710046, wo 625 = 0x0271 und 70 = 0x0046. Die RFCs benutzen die hex-Darstellung, COMMUNITY-Eigenschaften werden an Cisco-Routern aber dezimal verwendet. 625:70 ist zum Beispiel 40960070 (der dezimale Gegenwert von 0x2710046).

Die COMMUNITY-Werte von 0 (0x00000000) bis 65535 (0x0000FFFF) und von 4294901760 (0xFFFF0000) bis 4294967295 (0xFFFFFFFF) sind reserviert. In diesem reservierten Bereich sind mehrere bekannte Communities definiert:

– **INTERNET –** Die Internet-Community hat keinen besonderen Wert; alle Routen gehören automatisch zu ihr. Empfangene Routen dieser Community werden frei weitergegeben.

– **NO_EXPORT (4294967041 oder 0xFFFFFF01) –** Routen mit diesem Wert können nicht an EBGP-Peers weitergegeben und, falls eine Confederation programmiert ist, nicht an Ziele außerhalb der Confederation veröffentlicht werden (Confederations werden in einem späteren Abschnitt, »Management von BGP-Peering in großem Umfang« genauer besprochen).

– **NO_ADVERTISE (4294967042 oder 0xFFFFFF02) –** Routen mit diesem Wert können nicht weitergegeben werden, weder an EBGP- noch an IBGP-Peers.

– **LOCAL_AS (4294967043 oder 0xFFFFFF03) –** RFC 1997 nennt diese Eigenschaft NO_EXPORT_SUBCONFED. Routen mit diesem Wert können nicht an EBGP-Peers weitergegeben werden, auch nicht an Peers in anderen Autonomen Systemen, die in derselben Confederation sind.

Kapitel 3 enthält Beispiele über den Gebrauch von Communities beim Umsetzen von Routing-Richtlinien.

Die ORIGINATOR_ID- und CLUSTER_LIST-Eigenschaften

ORIGINATOR_ID und CLUSTER_LIST sind optionale, nontransitive Eigenschaften, die von Route Reflectors benutzt werden, welche im Abschnitt »Management von BGP-Peering in großem Umfang« beschrieben werden. Beide Eigenschaften werden verwendet, um Routing-Loops zu verhindern. Die ORIGINATOR_ID ist ein 32-Bit-Wert, der von einem Route Reflector erschaffen wird. Der Wert ist die Router ID des Ursprungs-Routers der Rou-

te im lokalen AS. Wenn der Ursprung seine RID in der ORIGINATOR_ID einer empfangenen Route sieht, weiß er, dass es einen Loop gibt, also wird die Route ignoriert.

CLUSTER_LIST ist eine Sequenz von Route-Reflection Cluster Ids, durch die die Route verläuft. Wenn ein Route Reflector seine lokale Cluster ID in der CLUSTER_LIST einer empfangenen Route sieht, weiß er, dass es einen Loop gegeben hat, also wird die Route ignoriert.

2.3.4 Administrative Weight (Gewichtung)

Administrative Weight ist ein Cisco-eigener BGP-Parameter, der sich nur auf Routen innerhalb eines Routers bezieht. Er wird nicht an andere Router weitergegeben. Das Weight ist eine Zahl zwischen 0 und 65,535, die einer Route zugewiesen werden kann; je höher das Weight, desto mehr wird die Route bevorzugt. Bei der Auswahl der besten Route verwendet BGP Weight vor allen anderen Routen-Charakteristika außer Specificity. In der Grundeinstellung haben alle Routen, die von einem Peer erlernt werden, ein Weight von 0, während alle vom lokalen Router generierten Routen ein Weight von 32.768 haben.

Administrative Weights können bei einzelnen Routen eingestellt werden oder auch für Routen, die von einem bestimmten Nachbarn erlernt werden. Peer A und Peer B könnten zum Beispiel dieselben Routen an einen BGP-Sprecher veröffentlichen. Indem den Routen von Peer A ein höheres Weight gegeben wird, bevorzugt der BGP-Sprecher die Routen dieses Peers. Diese Präferenz gilt nur bei diesem einen Router; Weights sind nicht in BGP-Updates oder anderen Nachrichten an die Nachbarn enthalten.

2.3.5 AS_SET

Die AS_PATH-Eigenschaft wurde bis jetzt als eine geordnete Sequenz von AS-Nummern besprochen, die den Pfad zu einem bestimmten Ziel vorgeben. Es gibt eigentlich zwei Arten von AS_PATH:

- **AS_SEQUENZ –** Dies ist die geordnete Liste von AS-Nummern, die bereits besprochen wurde.
- **AS_SET –** Dies ist eine ungeordnete Liste von AS-Nummern auf einem Pfad zum Ziel.

Die beiden Arten werden in der AS_PATH-Eigenschaft durch einen Type-Code unterschieden, wie im Abschnitt »BGP-Nachrichtenformate« bereits beschrieben wurde.

Anmerkung

Es gibt in Wirklichkeit vier Arten von AS_PATH. Im Abschnitt »Confederations« finden Sie mehr Informationen über die anderen Arten: AS_CONFED_SEQUENCE und AS_CONFED_SET.

Eine der wichtigsten Aufgaben von AS_PATH ist das Verhindern von Loops. Wenn ein BGP-Sprecher seine eigene AS-Nummer in einer von einem externen Router empfangenen Route erkennt, weiß er, dass es einen Loop gibt, und so wird die Route ignoriert. Bei Zusammenfassungen wie bei der in Abbildung 2.28 gehen allerdings manche AS_PATH-Details verloren. Folglich wird es wahrscheinlicher, dass es doch einen Loop gibt.

Nehmen wir zum Beispiel an, dass AS 810 in Abbildung 2.28 eine alternative Verbindung zu einem anderen AS hat (siehe Abbildung 2.30). Die Zusammenfassung von AS 3113 wird an AS 6571 veröffentlicht, von dort zu AS 810.

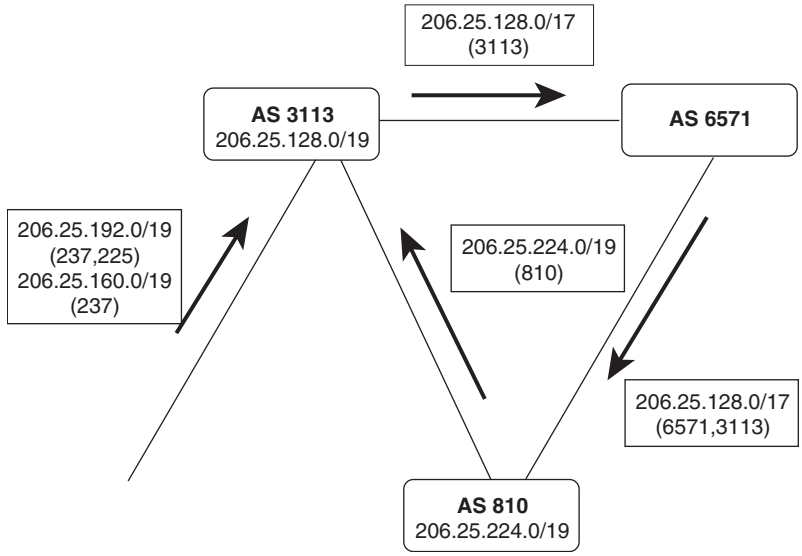

Abb. 2.30: Der Verlust von Pfad-Details bei der Zusammenfassung kann zu inter-AS-Routing-Loops führen.

Da die AS-Nummern »hinter« dem Verdichtungspunkt nicht in der AS_PATH-Eigenschaft stehen, bemerkt AS 810 nicht, dass es hier einen Loop geben kann. Es könnte auch sein, dass ein Netzwerk innerhalb AS 810, wie zum Beispiel 206.25.225.0/24, ausfällt. Die Router innerhalb dieses AS

haben dann auch die zusammengefasste Route von AS 6571 und so gibt es einen Loop.

Wenn man es sich genau überlegt, muss für die Loop-Verhinderungsfunktion des AS_PATH eigentlich die Liste der AS-Nummern nicht geordnet sein. Der empfangende Router muss nur feststellen können, ob die eigene AS-Nummer ein Teil des AS_PATH ist. Hier hilft AS_SET.

Wenn ein BGP-Sprecher eine Zusammenfassung von NLRI-Informationen aus anderen Autonomen Systemen erstellt, können alle AS in die AS_PATH-Eigenschaft als AS_SET eingetragen werden. Abbildung 2.31 zeigt zum Beispiel das Netzwerk aus Abbildung 2.28 mit einem AS_SET an der zusammengefassten Route.

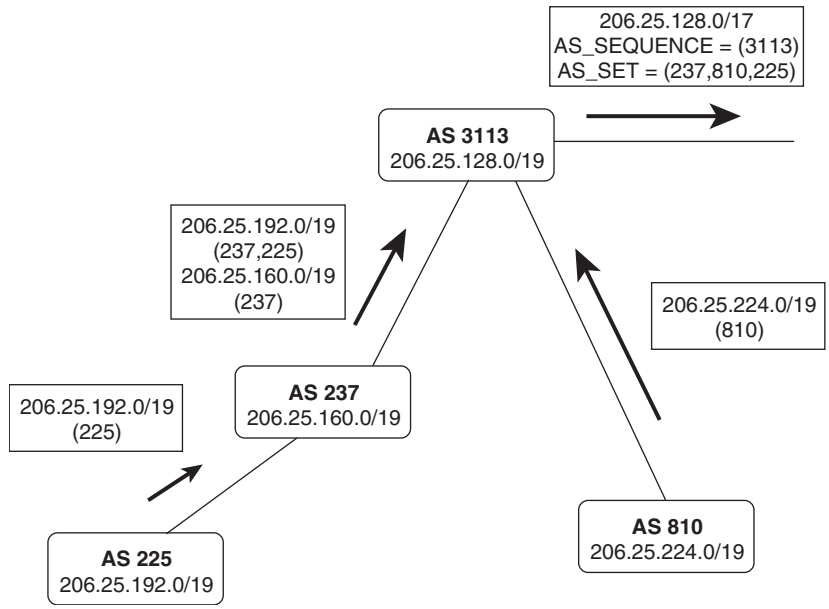

Abb. 2.31: Ein AS_SET in der AS_PATH-Eigenschaft einer zusammengefassten Route stellt die durch die Verdichtung verlorene Loop-Verhinderung wieder her.

Der zusammenfassende Router beginnt trotzdem eine AS_SEQUENZ, sodass die Router die Route zur Zusammenfassung zurückverfolgen können, ein AS_SET wird beigefügt, um Routing-Loops zu verhindern. An diesem Beispiel können Sie auch sehen, warum AS_SET eine ungeordnete Liste ist. Hinter dem zusammenfassenden Router in AS 3113 gibt es verzweigte Wege zu den Autonomen Systemen, aus denen die zusammengefassten Routen

stammen. Eine geordnete Liste dieser Routen ist wegen der vielen verschiedenen Wege unmöglich.

Wenn dem AS_PATH ein AS_SET hinzugefügt wird, dann muss das ATOMIC_AGGREGATE bei der Zusammenfassung nicht dabei sein. Das AS_SET gibt den downstream Routern über die Verdichtung Bescheid und enthält mehr Informationen als das ATOMIC_AGGREGATE.

AS_SET hat aber auch einen kleinen Nachteil. Sie verstehen bereits, dass einer der Vorteile der Zusammenfassung von Routen die Erhöhung der Stabilität der Routen ist. Wenn ein Netzwerk ausfällt, das zur Verdichtung gehört, dann wird der Fehler nicht weiter als bis zum Verdichtungspunkt bekannt gegeben. Wenn der Zusammenfassung aber ein AS_SET hinzugefügt wird, dann wird diese Stabilität verringert. Wenn zum Beispiel der Link zu AS 225 in Abbildung 2.31 ausfällt, verändert sich das AS_SET; diese Veränderung wird dann weitläufig bekannt gegeben, nicht nur bis zum Verdichtungspunkt.

2.3.6 Der BGP-Entscheidungsprozess

Die BGP-Routing-Information Database (RIB) besteht aus drei Teilen:

- **Adj-RIBs-In –** Speichert unverarbeitete Routing-Informationen, die durch Updates von Peers erhalten wurden. Die Routen in Adj-RIBs-In werden als mögliche Routen gesehen.
- **Loc-RIB –** Enthält die Routen, die der BGP-Sprecher ausgewählt hat, nachdem lokale Routing-Richtlinien auf die Routen in Adj-RIBs-In angewandt wurden
- **Adj-RIBs-Out –** Enthält die Routen, die der BGP-Sprecher seinen Nachbarn angibt

Diese drei Teile der Routing-Information Database können als drei verschiedene Datenbanken aufgeteilt werden, die RIB kann aber auch eine einzige Datenbank sein, in der die drei Teile enthalten sind und nur einzeln gekennzeichnet sind.

Der BGP-Entscheidungsprozess wählt die Routen mithilfe der Routing-Richtlinien aus der Adj-RIBs-In-Datenbank aus und gibt sie in die Loc-RIB- und Adj-RIBs-Out-Datenbanken ein. Der Entscheidungsprozess enthält drei Phasen:

- Phase 1 berechnet, wie stark die möglichen Routen bevorzugt werden. Diese Phase wird in Gang gesetzt wann immer der Router ein BGP-Update von einem Peer aus einem benachbarten AS empfängt, welches eine

neue Route, eine veränderte Route oder eine zurückgenommene Route enthält. Jede Route wird einzeln behandelt und bekommt eine positive, ganze Zahl zugewiesen, die die Bevorzugung dieser Route ausdrückt.

– Phase 2 wählt die beste Route aus allen möglichen Routen zu einem Ziel aus und installiert sie im Loc-RIB. Diese Phase wird in Gang gesetzt, wenn Phase 1 beendet ist.

– Phase 3 bringt die entsprechenden Routen zu Adj-RIBs-Out, sodass sie an Peers weitergegeben werden können. Sie beginnt wenn die Loc-RIB-Datenbank verändert wurde und erst nachdem Phase 2 zu Ende ist. Routen-Verdichtungen werden während dieser Phase durchgeführt, sofern sie notwendig sind.

Phase 2 benutzt immer die genaueste aller Routen zu einem Ziel, es sei denn die Routing-Richtlinie schreibt etwas anderes vor. Wenn die Adresse, die in der NEXT_HOP-Eigenschaft steht, unerreichbar ist, wird die Route nicht ausgewählt. Dies hat besondere Bedeutung für internes BGP und wird im Abschnitt »IBGP- und IGP-Synchronisation« näher beschrieben.

Sie sollten nun bereits eine Übersicht über die verschiedenen Eigenschaften haben, die einer BGP-Route zugewiesen werden können, um Routing-Richtlinien an einem Router bei internen Peers, bei benachbarten Autonomen Systemen und generell durchsetzen zu können. Eine Sequenz und Regeln sind nötig, um diese Eigenschaften zu verwenden, besonders wenn ein Router zwischen mehreren, genau gleichen Routen zum selben Ziel auswählen muss. Folgende Kriterien werden benutzt, um in diesen Situationen Entscheidungen zu treffen:

1. Bevorzuge die Route mit dem höchsten administrative Weight. Dies ist eine Cisco-eigene Funktion, da BGP administrative Weight ein Cisco-Parameter ist.

2. Wenn die Weights gleich sind, bevorzuge die Route mit dem höchsten LOCAL_PREF-Wert.

3. Wenn die LOCAL_PREF-Werte gleich sind, bevorzuge die Route, die vom eigenen Router stammt. Das bedeutet, bevorzuge eine Route, die durch ein IGP am selben Router erlernt wurde.

4. Wenn die LOCAL_PREF-Werte gleich sind und es keine lokal abstammende Route gibt, bevorzuge die Route mit dem kürzesten AS_PATH.

5. Wenn die AS_PATH-Längen gleich sind, dann bevorzuge die Route mit dem niedrigsten ORIGIN-Code. IGP ist niedriger als EGP, welches wiederum niedriger ist als Incomplete (unvollständig).

6. Wenn die ORIGIN-Codes gleich sind, dann bevorzuge die Route mit dem niedrigsten MULTI_EXIT_DISC-Wert. Dieser Vergleich wird nur durchgeführt, wenn die AS-Nummer bei allen entsprechenden Routen gleich ist.
7. Wenn die MED gleich sind, bevorzuge EBGP-Routen gegenüber Confederation EBGP-Routen und bevorzuge Confederation EBGP-Routen gegenüber IBGP-Routen.
8. Wenn die Routen immer noch gleich sind, bevorzuge die Route mit dem kürzesten Pfad zum BGP NEXT_HOP. Dies ist die Route mit der niedrigsten IGP-Metrik zum next Hop Router.
9. Wenn die Routen immer noch gleich sind, sind sie aus dem selben benachbarten AS und BGP ist mit dem **maximum-paths** Befehl aktiviert. Installieren Sie in diesem Fall alle Gleichkostenrouten im Loc-RIB.
10. Wenn Multipath nicht aktiviert ist, bevorzuge die Route mit der niedrigeren BGP-Router ID.

2.3.7 Route Dampening

Route Flaps (Routenflattern) tragen sehr zur Instabilität des Internets bei – genau wie bei anderen Netzwerken auch. Flaps entstehen, wenn eine gültige Route zuerst als ungültig und dann wieder als gültig veröffentlicht wird. Das Problem ist eindeutig: jedes Mal wenn der Status einer Route sich verändert, muss diese Veränderung im gesamten Netzwerk bekannt gegeben werden. Jeder Router, der eine Veränderung erfährt, muss dann seine Tabellen neu berechnen. Bandbreite sowie CPU-Ressourcen werden also benötigt.

Anmerkung

Sie hören vielleicht manchmal den Ausdruck *Route Oscillation,* der von manchen anstatt *Route Flapping* verwendet wird, allerdings gibt es einen Unterschied: Oscillations sind regelmäßig, Flaps sind es nicht.

Die meisten Leute machen unbeständige physische Links oder fehlerhafte Router, Schnittstellen für Route Flapping verantwortlich und dies ist auch richtig. Ein weiterer sehr verbreiteter Grund für Route Flaps, der vielleicht sogar noch bedeutender ist, sind Menschen. Techniker, die in der Telco-Zentrale oder in ihrem Schaltschrank herumwerkeln, können natürlich Flaps herbeiführen, vergessen Sie aber nicht den unerfahrenen Netzwerk-Administrator, der einfach nur versucht, einen Router zu konfigurieren oder einen

Fehler zu beseitigen. Vielleicht fügt er immer wieder eine Route hinzu, um sie wenige Sekunden später zu entfernen, vielleicht verändert er den Status einer Schnittstelle oder beendet eine BGP Session. Wenn seine Routen-Veränderungen an den ISP weitergegeben werden, kann seine unvorsichtige Vorgehensweise das ganze Internet belasten.

Wie schlimm können die Auswirkungen einer Instabilität sein? Befassen wir uns mit einem einzelnen überforderten BGP-Router. Eine upstream Verbindung wird instabil, was dazu führt, dass mehrere Routen zu flappen anfangen. Der Router kann die vielen Veränderungen nicht verkraften und fällt aus. Die downstream Router müssen jetzt nicht nur die flappenden Routen verarbeiten, sondern auch die nun unerreichbaren Routen des ausfallenden Routers. Dies kann zu einer Lawine führen, die vielleicht sogar Ihr gesamtes Netzwerk erfasst. Keine angenehme Angelegenheit.

Sie haben bereits gesehen, wie eine Zusammenfassung dabei hilft, solche Unsicherheiten zu vermeiden. Wenn eine der zusammengefassten Routen ausfällt, verändert sich die Zusammenfassung nicht. Die Pakete für die ausgefallene Route werden weiterhin an die zusammengefasste Adresse gesendet; der Zusammenfasser der Adressen hat schließlich die Informationen und ignoriert die Pakete.

Die Zusammenfassung ist allerdings nicht immer möglich. Ein ISP-Teilnehmer kann zum Beispiel eine provider-independent IP-Adresse haben. Da die Adresse außerhalb des Adressblocks des Providers ist, muss sie einzeln veröffentlicht werden. In der Diskussion über Multihoming haben Sie außerdem erlernt, dass Zusammenfassung bei Multihoming zu mehreren Providern nicht möglich ist.

Selbst wenn ein ISP eine stabile Route für den Rest des Internets schaffen kann, indem die Routen der Teilnehmer zusammengefasst werden, führt diese Zusammenfassung nicht zu mehr Stabilität beim ISP selbst. Ein Route Flap beeinflusst immer noch alle Router hinter dem Verdichtungspunkt.

Route Dampening ist eine Methode, die verhindert, dass instabile Router in einem Netzwerk veröffentlicht werden. Es verhindert zwar nicht, dass ein Router instabile Routen annimmt, es verhindert aber, dass sie weitergegeben werden. Obwohl Route Dampening schon seit längerer Zeit existiert, wurde es erst vor kurzem in RFC 2439 zusammengefasst (www.isi.edu/in-notes/tr.rfc2439.txt).

Ein Router der Route Dampening verwendet, gibt jeder Route eine dynamische Wertung, die sich nach der Stabilität der Route richtet. Wenn eine Route flappt, bekommt sie einen *Penalty (Strafe)*; je mehr sie flappt, desto mehr

Strafen sammelt sie. Es gibt außerdem eine Zeitspanne namens *Half-Life (Halbwertszeit)*. Die Strafe wird innerhalb der Halbwertszeit auf die Hälfte reduziert. Wenn der Wert der Penalties eine bestimmte Schwelle namens *suppress Limit* überschreitet, wird die Route abgestellt – das heißt sie wird nicht mehr veröffentlicht. Die Route bleibt abgestellt, bis die Halbwertszeit die Strafe unter eine weitere Schwelle senkt, die *reuse Limit* genannt wird. Wenn dies passiert, wird die Route wieder veröffentlicht. Die Penalties der Route können aber auch manuell entfernt werden; was sehr nützlich ist, wenn ein Fehler behoben wurde und die Route wieder verwendet werden soll.

Wenn das suppress Limit nicht besonders niedrig eingestellt ist, führt eine einziger Flap nicht dazu, dass eine Route abgestellt wird. Die Halbwertszeit bringt die Strafe schließlich irgendwann auf Null. Wenn eine Route schneller flappt als die Halbwertszeit die Strafen reduziert, dann wird die Schwelle überschritten und die Route abgestellt. Obwohl die Route weiter Strafen sammelt, nachdem sie abgestellt wurde, kann die Route nicht mehr als einen bestimmten Strafenwert erreichen. Dieser Wert heißt *maximum suppress limit*. Dies sorgt dafür, dass eine Route, die vielleicht innerhalb einer Sekunde mehrere dutzend Male geflappt hat, nicht für immer abgestellt bleibt, obwohl wieder alles funktionstüchtig ist.

Die Cisco-Grundeinstellungen für die verschiedenen Route Dampening-Variablen sind:

- **Penalty –** 1000 pro flap
- **Suppress Limit –** 2000
- **Reuse Limit –** 750
- **Half-Life –** 15 Minuten
- **Maximum suppress time –** 60 Minuten, oder viermal Half-Life

Beispiele für die Konfiguration und Benutzung von Route Dampening an Cisco-Routern finden Sie im Fallbeispiel »Route Dampening« in Kapitel 3.

2.4 IBGP- und IGP-Synchronisation

Mit einigen Ausnahmen wird internes BGP – BGP zwischen Peers im selben AS – nur bei Multihoming verwendet. IBGP ermöglicht Grenz-Routern, NLRI und andere Eigenschaften auszutauschen, um eine systemweite Routing-Richtlinie zu ermöglichen. IBGP wird außerdem verwendet, wenn ein Grenz-Router in einem Transit-AS Routen, die von einem externen Peer er-

lernt wurden, an andere Grenz-Router weitergibt, sodass diese die Routen anderer externer Peers veröffentlichen können.

Vielleicht denken Sie jetzt, dass in manchen Fällen IBGP als IGP verwendet werden kann. Ein ISP AS ist zum Beispiel hauptsächlich durch EBGP mit anderen Autonomen Systemen verbunden und trägt größtenteils Transitverkehr. Wieso sollte man nicht IBGP innerhalb des AS laufen lassen, damit man ein einziges Routing-Protokoll hat? Das Problem ist, dass dafür jeder IBGP-Router mit jedem andern seiner Art gepeert werden muss – das heißt das IBGP-Netzwerk muss *fully meshed,* also voll vernetzt, sein. Dieser Abschnitt erklärt warum ein IGP notwendig ist, um IBGP zu unterstützen und warum Synchronisation zwischen IGP und IBGP wichtig ist. Fully meshed IBGP wird aus zwei Gründen verwendet:

– um BGP-Routing-Loops innerhalb eines AS zu verhindern,

– um sicherzustellen, dass alle Router auf einer BGP-Route wissen, wie sie Pakete an das Ziel schicken müssen.

Routen, die über IBGP veröffentlicht werden, werden natürlich nur innerhalb eines AS ausgetauscht. Folglich verändert sich die AS_PATH-Liste nicht. Die lokale AS-Nummer wird dem AS_PATH erst hinzugefügt, wenn die Route einem EBGP-Peer bekannt gegeben wird. Aus diesem Grund haben IBGP-Routen nicht die Möglichkeit sich durch die Liste gegen Loops zu schützen wie es die EBGP-Routen tun. Um Loops zu verhindern, gibt BGP keine Routen an, die von einem anderen IBGP-Peer erlernt wurden.

Abbildung 2.32 zeigt was passiert, wenn die IBGP-Peers nicht voll vernetzt sind. Hier wurde IBGP-Peering zwischen Seattle und Tacoma und zwischen Tacoma und Spokane konfiguriert. Seattle und Tacoma tauschen NLRI über ihre lokalen Netzwerke aus, Spokane und Tacoma tun dies ebenfalls. Seattle und Spokane tauschen jedoch keine NLRI aus.

Abbildung 2.33 zeigt, wie durch komplettes Vernetzen erreicht werden kann, dass alle IBGP-Peers Informationen austauschen können. Seattle und Spokane sind Peers, obwohl es keinen direkten Datenlink zwischen den beiden gibt. Die von BGP benutzte TCP-Verbindung verläuft durch Tacoma, ist aber logisch gesehen eine direkte Verbindung zwischen Seattle und Spokane. Es ist wichtig dies zu erkennen, da Seattle und Spokane die Adressen der Links, die sie verbinden, wissen müssen, um eine TCP-Verbindung aufzubauen.

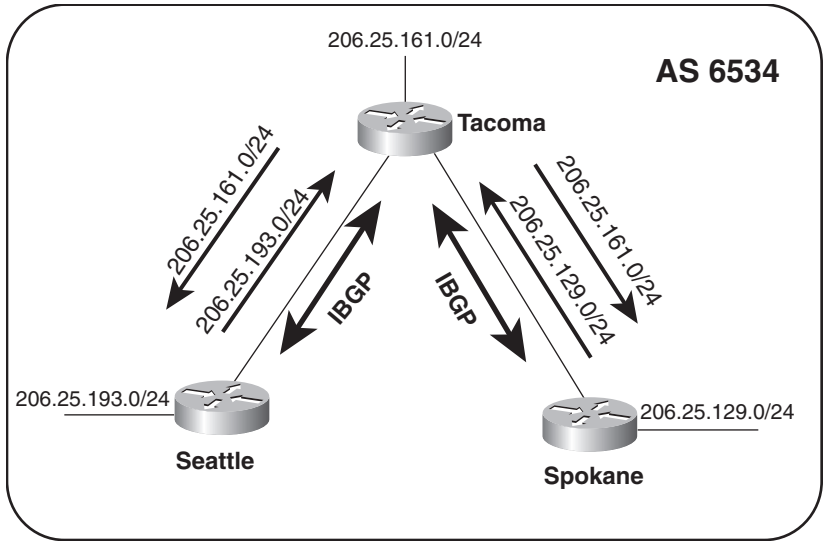

Abb. 2.32: In einer teilweise vernetzten IBGP-Umgebung werden NLRI nicht vollständig ausgetauscht, weil Routen, die von einem IBGP-Peer erlernt werden, nicht an andere IBGP-Peers weitergeleitet werden.

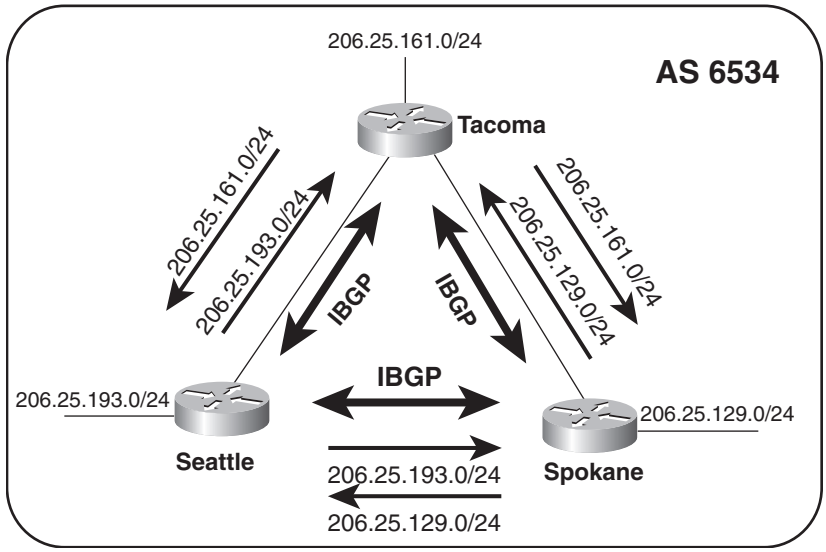

Abb. 2.33: In einer voll vernetzten IBGP-Umgebung ist jeder IBGP-Router mit jedem anderen IBGP-Router verbunden, also werden alle NLRI ausgetauscht.

Auf den ersten Blick ist einfach sicherzustellen, dass alle Adressen bekannt sind – die Adressen müssen an jedem Router in die BGP-Befehle **network** eingegeben werden (in Kapitel 3 beschrieben). Leider ist es nicht immer so einfach.

Beispiel 2.14 zeigt Seattles BGP-Routing-Tabelle und die IGP-Routing-Tabelle. Wenn der Router Pakete verschicken soll, muss sich die Zieladresse in der IGP-Routing-Tabelle befinden.

Beispiel 2.14: Obwohl es in der BGP-Routing-Tabelle mehrere Routen gibt, werden diese nicht automatisch in die IGP-Routing-Tabelle eingetragen.

```
Seattle#show ip bgp
BGP table version is 7, local router ID is 206.25.193.1
Status codes: s suppressed, * valid, > best, i - internal
ORIGIN codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric LocPrf Weight Path
*> 192.168.1.0      0.0.0.0                0         32768 i
* i                 192.168.1.1            0    100      0 i
*>i192.168.2.0      192.168.1.1            0    100      0 i
*>i206.25.161.0     192.168.1.1            0    100      0 i
*> 206.25.193.0     0.0.0.0                0         32768 i

Seattle#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

C    206.25.193.0 is directly connected, Loopback0
C    192.168.1.0 is directly connected, Serial0
Seattle#
```

Wie die Bildschirmausgabe in Beispiel 2.14 zeigt, enthält die BGP-Tabelle mehrere Routen, auch die Adressen der Datenlinks Seattle-Tacoma und Spokane-Tacoma (192.168.1.0/24 und 192.168.2.0/24). Nur Seattles direkt verbundene Links stehen aber in der IGP-Routing-Tabelle. Spokanes Netzwerk 206.25.129.0/24 ist nicht einmal in der BGP-Tabelle, was bedeutet, dass Seattle und Spokane nicht richtig peeren.

ANMERKUNG

Vergleichen Sie die Weights der direkten Links in der BGP-Tabelle mit den Weights der Routen, die von Tacoma erlernt wurden.

Beispiel 2.14 zeigt das Problem der *Synchronisation*. Die Regel der Synchronisation besagt:

Bevor eine Route, die von einem IBGP-Nachbarn erlernt wurde, in die IGP-Routing-Tabelle kommt oder an einen BGP-Peer veröffentlicht wird, muss die Route über IGP bekannt gegeben worden sein.

Beim Netzwerk in Abbildung 2.33 können die BGP-Routen nicht in die IGP-Routing-Tabelle eingetragen werden, weil an den Routern kein IGP läuft und die Synchronisation vorschreibt, dass die Route über IGP bekannt sein muss, bevor sie eingetragen werden kann.

Um zu verstehen, warum die Regel der Synchronisation existiert, beschäftigen wir uns nun mit dem Netzwerk in Abbildung 2.34. Hier wird IBGP nicht als internes Gateway-Protokoll verwendet. Stattdessen wird ein echtes IGP (OSPF) benutzt. Salt Lake und Provo sind mit zwei verschiedenen Autonomen Systemen verbunden und tauschen die von EBGP erlernten Routen über eine IBGP-Verbindung aus. Die TCP-Verbindung für diesen IBGP-Austausch verläuft durch Orem und Ogden.

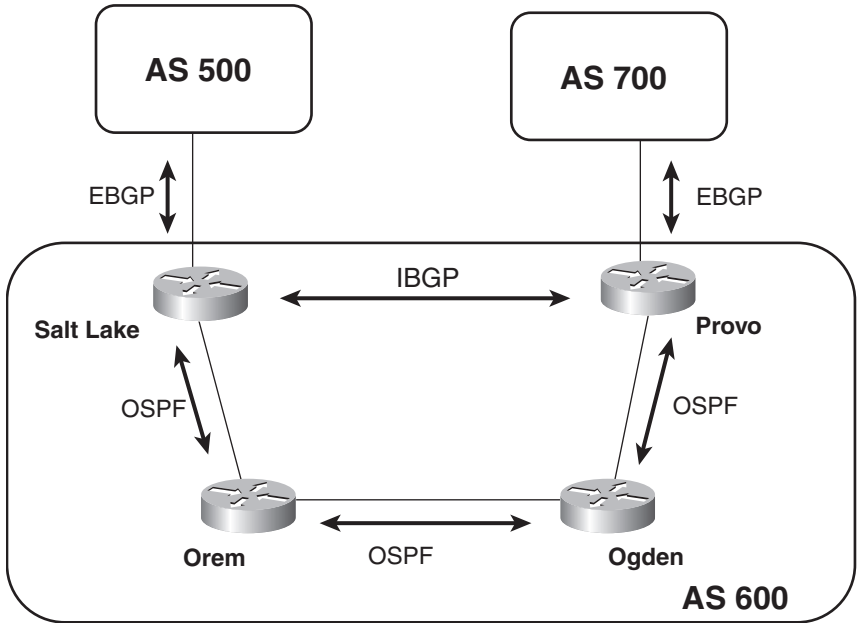

Abb. 2.34: Dieses Netzwerk hat teilweise vernetztes IBGP zwischen Salt Lake und Provo und verwendet OSPF als IGP.

Nehmen wir als Nächstes an, Salt Lake lernt eine Route zu 196.223.18.0/24 von AS 500, gibt diese Route über IBGP an Provo weiter und verwendet eine next Hop-self Richtlinie, um die NEXT_HOP-Eigenschaft auf die eigene Router ID umzustellen. Provo gibt diese Route dann an AS 700 weiter. Die Router in AS 700 beginnen nun Pakete, die für 196.223.18.0/24 bestimmt sind, an Provo zu senden (Denken Sie daran, dass eine Routen-Angabe ein Versprechen ist, die Pakete zur Zieladresse zu bringen). Nun beginnen sich Fehler einzuschleichen. Provo schlägt Netzwerk 196.223.18.0/24 nach und sieht, dass es über Salt Lake zu erreichen ist. Dann sucht er nach Salt Lakes IP-Adresse und sieht, dass sie über den next Hop Router Ogden erreichbar ist. Also wird das Paket für 196.223.18.0/24 an Ogden gesendet. Die externen Routen werden zwischen Salt Lake und Provo jedoch über IBGP ausgetauscht; die OSPF-Router haben kein Wissen über die externen Routen. Darum findet Ogden, nachdem er das Paket bekommt, keinen Eintrag für 196.223.18.0/24. Der Router ignoriert das Paket und alle weiteren Pakete für diese Adresse. Der Verkehr für Netzwerk 196.223.18.0/24 verschwindet.

Wenn die OSPF-Router in Abbildung 2.34 über die externen Routen Bescheid wissen, dann verläuft die eben beschriebene Situation natürlich anders. Ogden weiß dann, dass 196.223.18.0/24 über Salt Lake zu erreichen ist und versendet das Paket richtig. Synchronisation verhindert, dass Pakete in einem Transit-AS durch ein IGP mit unzureichenden Informationen geschluckt werden.

Wenn Provo die Angabe über 196.223.18.0/24 von Salt Lake erhält, wird die Route der BGP-Tabelle hinzugefügt. Danach wird die IGP-Routing-Tabelle überprüft, um festzustellen, ob es für die Route einen Eintrag gibt. Wenn nicht, dann weiß Provo, dass die Route dem IGP unbekannt ist und dass die Route nicht weitergegeben werden kann. Wenn das IGP für 196.223.18.0/24 einen Eintrag in der Routing-Tabelle macht (das heißt wenn IGP über die Route Bescheid weiß), wird Provos BGP-Route mit der IGP-Route synchronisiert. Nun kann der Router anfangen, die Route seinen BGP-Peers mitzuteilen.

Wenn wir uns nun Abbildung 2.33 und Beispiel 2.14 wieder zuwenden, können wir erkennen, dass die Synchronisation verhindert, dass das voll vernetzte IBGP funktioniert. Tacoma ist nun in einer paradoxen Situation: Es werden Routen von Seattle und Spokane empfangen, sie können aber nicht in die IGP-Routing-Tabelle eingetragen oder weiterverbreitet werden. Es gibt kein IGP, das sie dort eintragen kann.

Synchronisation ist eine etwas veraltete Eigenschaft von BGP, die annimmt, dass Routen im IGP verteilt werden. Wie dieses Beispiel zeigt, können alle

Router in einem voll vernetzten IBGP jedoch alle wichtigen BGP-Routen einfach durch BGP erfahren. Synchronisation verhindert in diesem Fall nur, dass BGP-Routen innerhalb des BGP bleiben und IGP nur für IBGP-Verbindungen verwendet wird.

Glücklicherweise kann bei Cisco-Routern die Synchronisation abgeschaltet werden. Beispiel 2.15 zeigt Seattles BGP und IGP-Routing-Tabellen, nachdem die Synchronisation abgeschaltet wurde. Tacoma hat die Routen von Spokane weitergegeben und die Pakete werden nun richtig verschickt.

Beispiel 2.15: Seattle hat volle NLRI in den BGP und IGP-Routing-Tabellen, nachdem die Synchronisation auf den drei Routern in Abb. 2.33 abgeschaltet wurde.

```
Seattle#show ip bgp
BGP table version is 11, local router ID is 206.25.193.1
Status codes: s suppressed, * valid, > best, i - internal
ORIGIN codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric LocPrf Weight Path
*> 192.168.1.0      0.0.0.0                0         32768 i
* i                 192.168.1.1            0    100      0 i
*>i192.168.2.0      192.168.1.1            0    100      0 i
* i                 192.168.2.1            0    100      0 i
*>i206.25.129.0     192.168.2.1            0    100      0 i
*>i206.25.161.0     192.168.1.1            0    100      0 i
*> 206.25.193.0     0.0.0.0                0         32768 i

Seattle#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

C   206.25.193.0 is directly connected, Loopback0
B   206.25.129.0 [200/0] via 192.168.2.1, 00:07:34
C   192.168.1.0 is directly connected, Serial0
B   192.168.2.0 [200/0] via 192.168.1.1, 00:07:42
B   206.25.161.0 [200/0] via 192.168.1.1, 00:07:43

Seattle#ping 206.25.129.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 206.25.129.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/8 ms
Seattle#
```

Die Moral der Geschichte ist, dass eine von zwei Konfigurations-Optionen durchgeführt werden muss, wenn IBGP funktionieren soll:

– Die externen Routen müssen dem IGP mitgeteilt werden, sodass IGP mit BGP synchronisiert werden kann. Der Nachteil dieser Vorgehensweise ist, dass bei einer großen Anzahl von BGP-Routen, wie einer ganzen Internet-Routing-Tabelle, die IGP-Router-Prozessoren und der Speicher doch sehr beansprucht werden. In den meisten Fällen könne Router mit dieser Last nicht umgehen und fallen aus. Es gab schon mehrere große Ausfälle, die zustande kamen, weil BGP-Routen aus Versehen an OSPF oder IS-IS weitergegeben wurden. Bei einem dieser Fälle war ein großer Provider 19 Stunden lang nicht erreichbar.

– Die IBGP-Router müssen voll vernetzt sein und die Synchronisation muss abgeschaltet sein. Jeder Router kennt die externen Routen über BGP und das Abschalten der Synchronisation ermöglicht es, die Routen in die Routing-Tabelle einzutragen, ohne zuerst das IGP zu informieren. Der Nachteil dieser Option ist, dass es ein sehr aufwändiges Unterfangen ist, jeden Router in einem relativ großen AS mit allen anderen IBGP-Routern zu peeren. Diese Option wird trotzdem fast immer verwendet, wenn es um Internet-Routen geht. Im nächsten Abschnitt werden zwei Werkzeuge vorgestellt, die helfen, das voll vernetzte IBGP-Netzwerk, die Route Reflectors und die Confederations zu kontrollieren.

Kapitel 3 zeigt mehrere Beispiele von IBGP-Konfigurationen. Außerdem werden die Nachteile der beiden Konfigurationen noch einmal besprochen und einige Lösungsversuche vorgestellt.

2.5 Management von BGP-Peering in großem Umfang

Im vorigen Abschnitt wurde erwähnt, dass es sehr schwierig werden kann, bei einem großen AS die IBGP-Peers voll zu vernetzen. Dies ist nur eines der Probleme die auftreten, wenn man versucht mit BGP in einem großen Umfang zu arbeiten. BGP hat vier Werkzeuge, die das Management extrem vieler BGP-Peers erleichtern:

– Peer-Gruppen

– Communities

– Route Reflectors

– Confederations

Die ersten beiden Werkzeuge erleichtern das Management von Routing-Richtlinien zwischen mehreren Peers, intern oder extern. Die beiden letzten Werkzeuge erleichtern das Management von IBGP bei einer großen Anzahl von Peers.

2.5.1 Peer-Gruppen

In großen BGP-Netzwerken, gelten die Richtlinien an einem Router oft auch für viele der anderen. Es kann zum Beispiel sein, dass dieselben Eigenschaften in den Updates zu mehreren Peers sein sollen oder dass derselbe Filter für viele Routen von bestimmten Peers benutzt wird. In solchen Fällen kann die Konfiguration erleichtert werden, indem die Peers mit denselben Richtlinien zusammen in eine *Peer-Gruppe* kommen.

Eine Peer-Gruppe wird an einem Cisco-Router durch einen Namen und ein Set von Routing-Richtlinien bestimmt. Die Peers werden dann der Peer-Gruppe hinzugefügt. Veränderungen der Richtlinie können dann an der Gruppe vorgenommen werden und müssen nicht mehr bei jedem Peer einzeln gemacht werden. Peer-Gruppen sind auch sehr nützlich, um die Leistung eines Routers zu erhöhen. Anstatt immer die Datenbank nach der Richtlinie für jedes Update zu fragen, kann der Router ein einziges Mal nachschlagen, ein einziges Update erzeugen und dies dann an alle Peers der Gruppe senden.

Manchmal kann es sein, dass ein Mitglied oder mehrere Mitglieder einer Peer-Gruppe zusätzliche Eigenschaften besitzen. In diesem Fall können die zusätzlichen Richtlinien bei den entsprechenden Nachbarn einfach zu den normalen Richtlinien der Gruppe hinzugefügt werden.

2.5.2 Communities

Peer-Gruppen bestimmen die Richtlinien einer Gruppe von Routern, Communities hingegen bestimmen die Richtlinien für eine Gruppe von Routen. Ein Router fügt eine Route einer bereits konfigurierten Community hinzu indem die COMMUNITY-Eigenschaft auf einen Wert gestellt wird, der sie als Mitglied einer existierenden Community ausweist. Benachbarte Router können dann ihre Richtlinien wie das Filtern und die Redistribution je nach COMMUNITY der Route durchsetzen. Die COMMUNITY-Eigenschaft, die einen bekannten Wert annehmen kann oder einen vom Netzwerkadministrator zugewiesen bekommen kann, wurde im Abschnitt »Die COMMUNITY-Eigenschaft« genauer behandelt.

Es können pro Route mehr als eine COMMUNITY-Eigenschaft eingestellt werden. Ein Router, der eine Route mit mehreren COMMUNITY-Eigenschaften empfängt, kann die Richtlinie entweder auf alle diese Eigenschaften ausrichten oder ein Subset der Eigenschaften beachten. Wenn Routen mit COMMUNITY-Eigenschaften zusammengefasst werden, erbt die Verdichtung die COMMUNITY-Eigenschaften aller Routen.

2.5.3 Route Reflectors

Route Reflectors sind nützlich, wenn ein AS eine große Anzahl von IBGP-Peers enthält (mehr Informationen finden Sie in RFC 1966 unter www.isu-edu/in-notes/rfc1771.txt). Wenn die EBGP-Routen nicht in das IGP des AS gegeben werden, müssen alle IBGP-Peers voll vernetzt sein. Wenn es *n* Router gibt, dann gibt es $n(n-1)/2$ IBGP-Verbindungen im AS. Abbildung 2.35 zeigt zum Beispiel sechs voll vernetzte IBGP-Router, wahrlich keine zu hohe Anzahl; selbst hier werden jedoch 15 IBGP-Verbindungen benötigt.

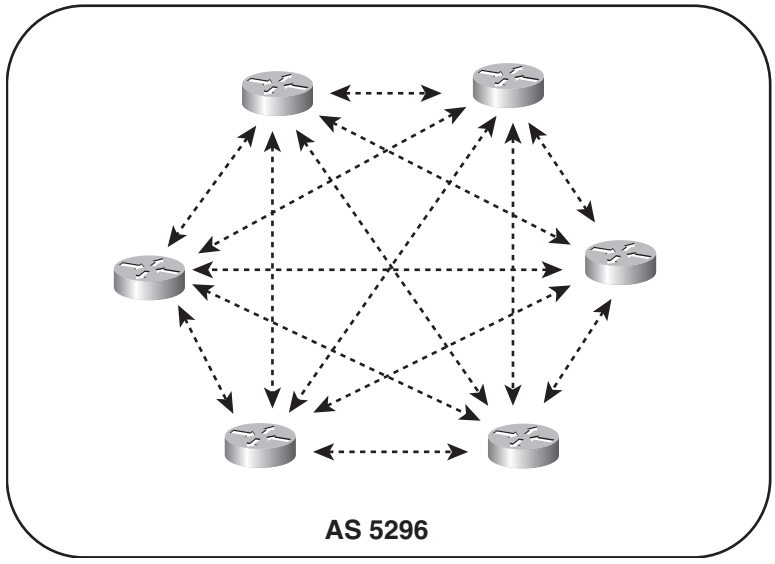

Abb. 2.35: Voll vernetzte IBGP-Peers

Route Reflectors bieten eine Alternative zu voll vernetzten IBGP-Peers. Ein Router wird als Route Reflector (RR) konfiguriert, die anderen IBGP-Routers, die *Clients*, peeren nur mit dem RR anstatt mit allen anderen IBGP-Routern (siehe Abbildung 2.36). Folglich geht die Zahl der Peering-Verbin-

dungen von $n(n-1)/2$ auf $n-1$ zurück. Ein Route Reflector und seine Clients werden zusammen *Cluster* genannt.

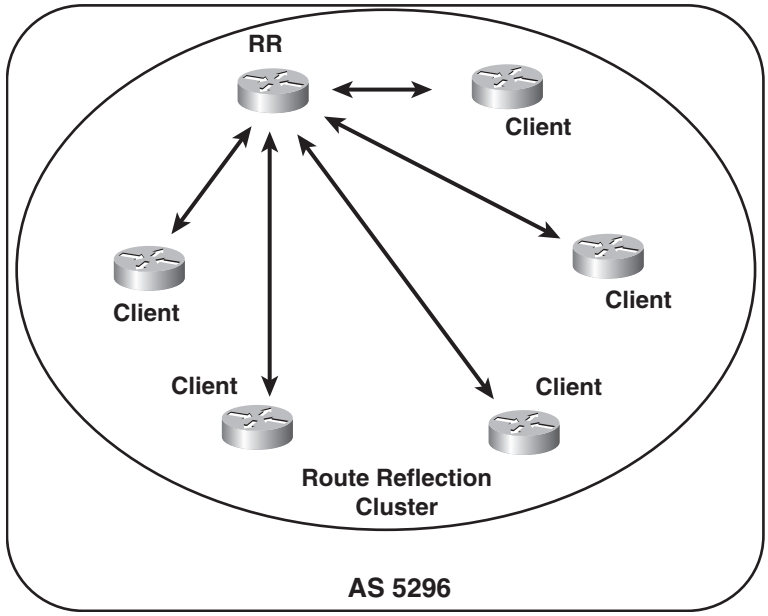

Abb. 2.36: IBGP-Clients in einem Route Reflection Cluster peeren nur mit dem Route Reflector, was die Zahl der nötigen IBGP-Verbindungen verringert.

Route Reflectors funktionieren, indem sie die Regel umgehen, wonach IBGP-Peers keine Routen veröffentlichen dürfen, die sie von anderen IBGP-Peers erlernen. Im Netzwerk in Abbildung 2.36 lernt der Route Reflector zum Beispiel Routen von seinen Clients. Anders als bei normalen IBGP-Routern kann der RR diese Routen an alle seine Peers weitergeben, egal ob Clients oder nicht. Die Routen von einem IBGP-Client werden also vom RR an die anderen Clients »reflektiert«. Um Routing-Loops oder andere Routing-Fehler zu vermeiden, kann der RR die Eigenschaften der empfangenen Routen nicht verändern.

Ein Client-Router in einem Route Reflection Cluster kann mit externen Nachbarn gepeert sein, kann intern aber nur mit dem RR oder anderen Clients seines Clusters peeren. Der RR kann jedoch mit internen und externen Nachbarn außerhalb des eigenen Clusters peeren und so deren Routen zu den Clients leiten (siehe Abbildung 2.37).

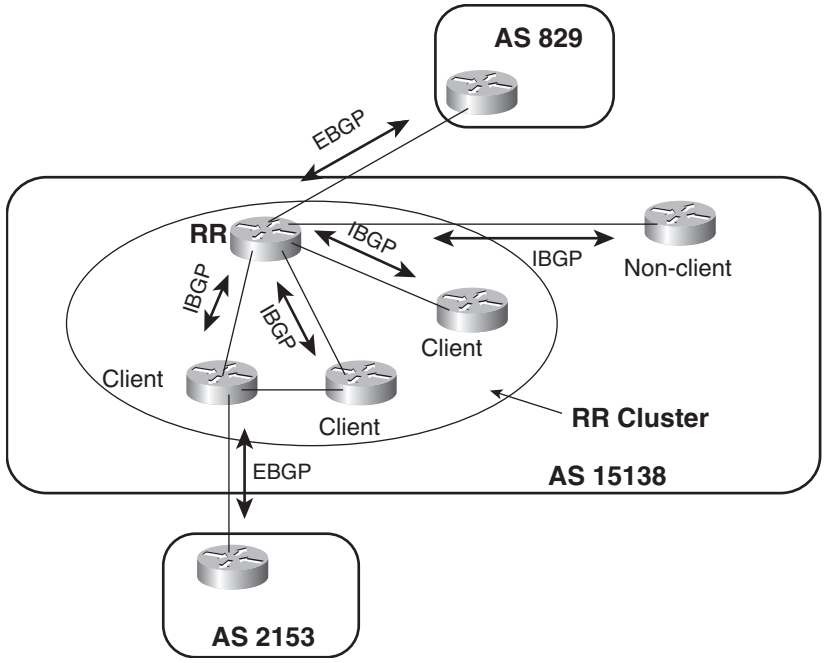

Abb. 2.37: Route Reflection Cluster Peering-Beziehungen

Wenn ein RR mehrere Routen zum selben Ziel empfängt, benutzt er die normalen BGP-Kriterien, um sich zu entscheiden. RFC 1966 definiert drei Regeln, die der RR benutzt, um zu entscheiden, an wen die Route veröffentlicht wird, abhängig vom Ursprung der Information:

– Wenn die Route von einem IBGP-Peer erlernt wurde, der kein Client ist, wird sie nur an Clients weitergegeben.

– Wenn die Route von einem Client erlernt wurde, wird sie an alle Peers, ob Clients oder nicht Clients, bis auf den Ursprungs-Client weitergegeben.

– Wenn die Route von einem EBGP-Peer erlernt wurde, wird sie an alle Clients und Non-Clients weitergegeben.

Die Route Reflector-Funktion muss auch vom Route Reflector selbst unterstützt werden. Aus Sicht der Clients peeren sie nur mit einem internen Nachbarn. Dies ist eine gute Eigenschaft von Route Reflectors, da selbst Router mit relativ einfachen BGP-Ausführungen Clients in einem Route Reflection Cluster sein können.

Das Konzept eines Route Reflectors ist dem Konzept eines Route Servers, das bereits besprochen wurde, sehr ähnlich. Das Hauptziel beider Geräte ist

die Zahl der nötigen Peering-Verbindungen zu reduzieren, indem ein Peering-Punkt für die Nachbarn geschaffen wird. Die Nachbarn sind dann beim Lernen von Routen von diesem Gerät abhängig. Der Unterschied zwischen Route Reflectors und Route Servern ist, dass die Route Reflectors auch Router sind, während die Route Server kein Routen übernehmen.

Ein einziger RR bringt wie ein einziger Route Server einen Schwachpunkt in das System. Wenn der RR ausfällt, verlieren die Clients ihre einzige NLRI-Quelle. Aus diesem Grund kann ein Cluster zur Absicherung mehr als einen RR haben (siehe Abbildung 2.38). Die Clients haben physische Verbindungen zu beiden Route Reflectors, die auch miteinander gepeert sind. Wenn ein RR ausfällt, haben die Clients noch immer einen Zugang zum anderen RR und bekommen somit weiterhin Erreichbarkeitsinformationen.

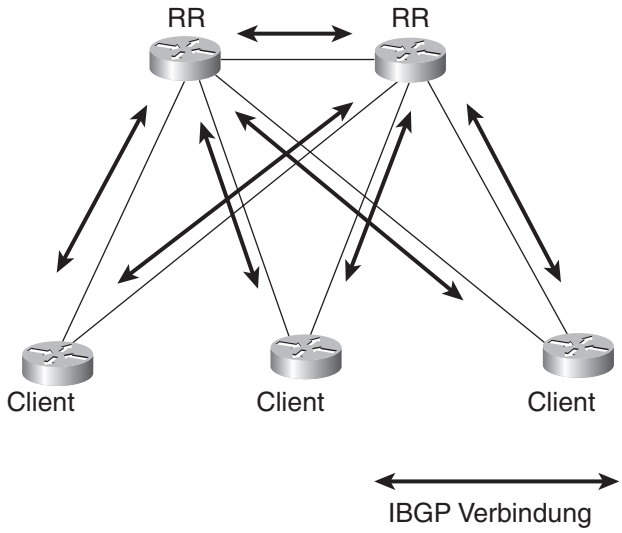

Abb. 2.38: Ein Cluster kann zur Absicherung mehrere Route Reflectors haben.

Anmerkung

Obwohl es möglich ist, dass ein Client nur mit einem RR verbunden ist und trotzdem mit beiden RRs gepeert ist, sollte dieser Aufbau vermieden werden, da er den Vorteil des Systems zunichte macht. Der Client ist immer noch von einem einzigen RR abhängig und bekommt keine Informationen, wenn der Router ausfällt, zu dem er die Verbindung hat.

Ein AS kann auch mehrere Cluster haben. Abbildung 2.39 zeigt ein AS mit zwei Clustern. jeder Cluster hat einen absichernden Route Reflector und die Cluster selbst sichern sich ebenfalls gegenseitig ab.

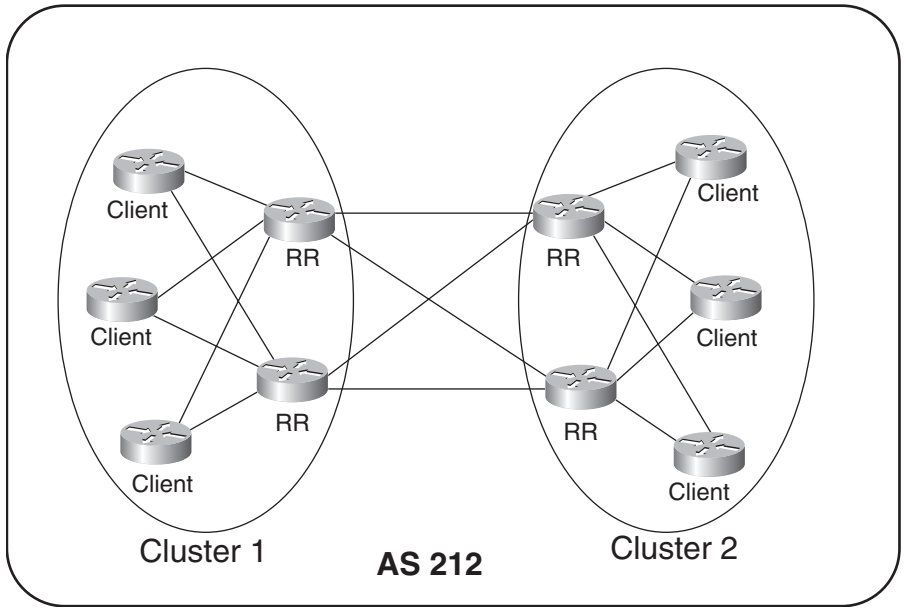

Abb. 2.39: Mehrere Route Reflection Clusters innerhalb eines Autonomen Systems

Da Clients nicht wissen, dass sie Clients sind, kann ein Route Reflector selbst ein Client eines anderen Route Reflectors sein. Folglich können »nestartige« Route Reflection Cluster gebaut werden (siehe Abbildung 2.40).

Obwohl Clients nicht mit Routern außerhalb ihres eigenen Clusters peeren, können sie miteinander gepeert sein. Ein Route Reflection Cluster kann also voll vernetzt sein (siehe Abbildung 2.41). Wenn die Clients voll vernetzt sind, wird der Route Reflector so konfiguriert, dass er keine Routen von einem Client zum anderen leitet. Es werden nur Routen von Clients zu Non-Clients und von Non-Clients zu Clients geroutet.

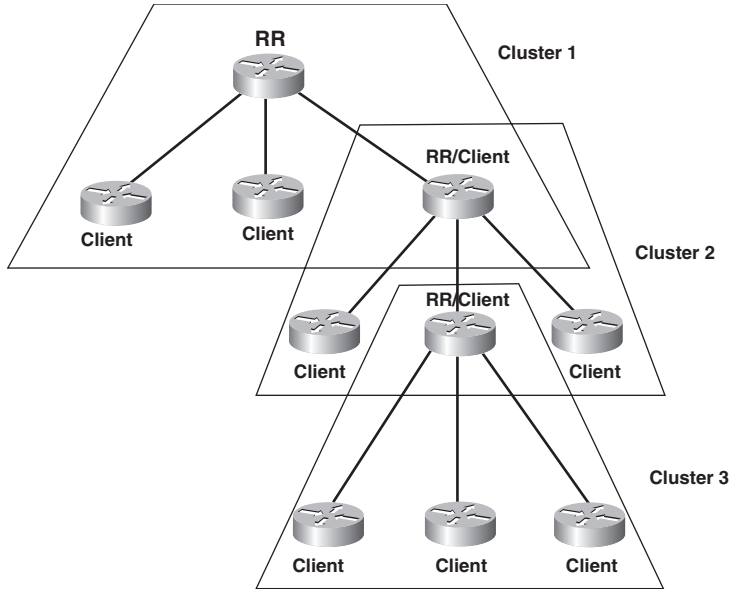

Abb. 2.40: Ein Route Reflector kann der Client eines anderen Route Reflectors sein.

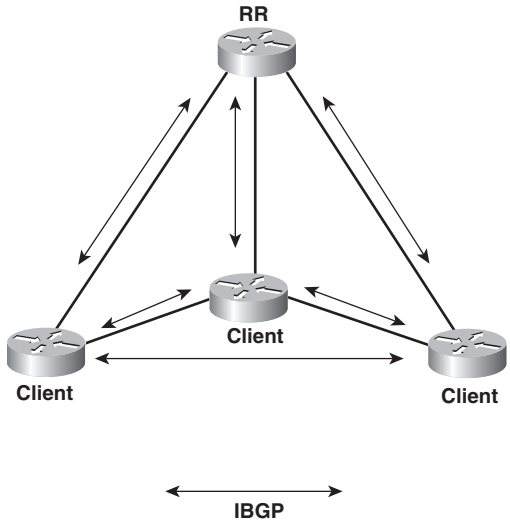

Abb. 2.41: Ein Route Reflection Cluster kann voll vernetzt sein.

Im Abschnitt »IBGP- und IGP-Synchronisation« wurde besprochen, dass BGP keine Routen weitergeben kann, die ein interner Peer von einem anderen internen Peer erlernt hat, weil sich die AS_PATH-Eigenschaft dabei nicht verändert und so Routing-Loops entstehen könnten. Bei einem Route Reflector kann diese Regel jedoch umgangen werden. Routing-Loops werden von Route Reflectors durch zwei BGP-Pfadeigenschaften verhindert: ORIGINATOR_ID und CLUSTER_LIST.

ORIGINATOR_ID ist eine optionale nontransitive Eigenschaft, die vom Route Reflector erschaffen wird. Die ORIGINATOR_ID ist die Router ID des Ursprungs der Route innerhalb des lokalen AS. Ein Route Reflector gibt eine Route nie zurück zum Ursprung an; sollte der Ursprungs-Router doch ein Update mit seiner eigenen RID erhalten, wird dieses ignoriert.

Jeder Cluster innerhalb eines AS muss durch eine einzigartige 4-Oktett *Cluster ID* identifiziert werden. Wenn der Cluster einen einzigen Route Reflector enthält, ist die Cluster ID die Router ID des Route Reflectors. Wenn sich in einem Cluster mehrere Route Reflectors befinden, muss jeder RR manuell mit einer Cluster ID konfiguriert werden.

CLUSTER_LIST ist eine optionale nontransitive Eigenschaft, die die Cluster IDs festhält, wie es die AS_PATH-Eigenschaft bei AS-Nummern macht. Wenn ein RR eine Route von einem Client zu einem Non-Client leitet, wird die Cluster ID der CLUSTER_LIST hinzugefügt. Wenn die CLUSTER_LIST leer ist, wird vom RR eine kreiert. Wenn ein RR ein Update empfängt, wird die CLUSTER_LIST überprüft. Wenn die eigene Cluster ID sich in der Liste befindet, bedeutet dies, dass es einen Routing-Loop gibt und das Update wird ignoriert.

2.5.4 Confederations

Confederations sind eine weitere Möglichkeit, um große Zahlen von IBGP-Peers zu verwalten. Eine Confederation ist ein AS, das in eine Gruppe von subAutonomen Systemen aufgeteilt ist, die *Member autonomous systems* genannt werden (siehe Abbildung 2.42). Die BGP-Sprecher sprechen mit Peers im selben Member AS IBGP und mit anderen Peers EBGP. Die Confederation bekommt eine *Confederation ID*, die den Peers außerhalb der Confederation als AS-Nummer der gesamten Confederation veröffentlicht wird. Externe Peers sehen nicht die interne Struktur der Confederation; sie sehen nur ein normales AS. In Abbildung 2.42 ist AS 9184 die Confederation ID.

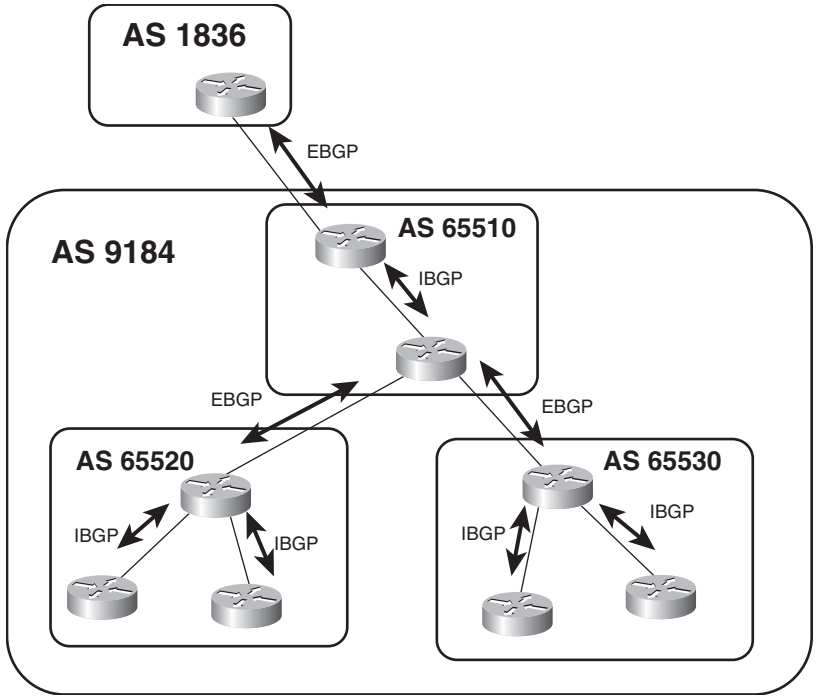

Abb. 2.42: Eine typische Confederation

Sie kennen die Taktik der Aufteilung großer Systeme, um sie übersichtlicher zu machen. IP-Subnets sind Teile von IP-Netzwerken und VLSM teilt wiederum diese Subnets auf. Ähnlich sind auch Autonome Systeme Teile von großen Netzwerken (wie dem Internet). Confederations sind Teile eines Autonomen Systems.

Der Abschnitt »AS_SET« beschreibt zwei Arten von AS_PATH-Eigenschaft: AS_SEQUENCE und AS_SET. Confederations fügen AS_PATH zwei weitere Arten hinzu:

– **AS_CONFED_SEQUENCE –** Dies ist eine geordnete Liste von AS-Nummern entlang einer Route zu einem Ziel. Wird sie genau wie die AS_SEQUENCE verwendet, gehören die AS-Nummern der Liste zu Autonomen Systemen innerhalb der lokalen Confederation.

– **AS_CONFED_SET –** Dies ist eine ungeordnete Liste von AS entlang einer Route zu einem Ziel. Sie wird genau verwendet wie AS_SET, nur gehören die AS-Nummern in der Liste zu Autonomen Systemen innerhalb der lokalen Confederation.

Durch das Verwenden der AS_PATH-Eigenschaft in den Updates zwischen Mitgliedern Autonomer Systeme, gibt es weiterhin keine Loops. Aus der Perspektive eines BGP-Routers innerhalb eines Member AS sind alle Peers in einem anderen Member AS externe Nachbarn.

Wenn ein Update an einen Peer außerhalb von Confederation gesendet wird, werden die Informationen AS_CONFED_SEQUENCE und AS_CONFED_SET von der AS_PATH-Eigenschaft entfernt. Die Confederation ID wird dafür dem AS_PATH hinzugefügt, wie immer am Anfang der Liste. Aus diesem Grund sehen die externen Peers die Confederation als ein einziges AS und nicht als eine Gruppe Autonomer Systeme. Wie Abbildung 2.42 zeigt, ist es oft so, dass AS-Nummern aus dem reservierten Bereich 64512 bis 65535 verwendet werden, um die Member AS innerhalb einer Confederation zu benennen.

Bei der Wahl der Routen bleibt der BGP-Entscheidungsprozess gleich, es gibt jedoch eine Extraregel: EBGP-Routen außerhalb der Confederation werden gegenüber EBGP-Routen zu Member AS bevorzugt, diese wiederum vor IBGP-Routen. Ein weiterer Unterschied zwischen Confederations und normalen Autonomen Systemen ist die Art und Weise wie manche Eigenschaften behandelt werden. Eigenschaften wie NEXT_HOP und MED können unverändert an EBGP-Peers in anderen Member AS derselben Confederation gesendet werden und die LOCAL_PREF-Eigenschaft kann auch verschickt werden.

Anders als bei Umgebungen von Route Reflectors, in denen nur der Route Reflector selbst Route Reflection unterstützen muss, müssen alle Router einer Confederation die Confederations-Funktionalität unterstützen. Diese Unterstützung ist deswegen notwendig, weil alle Router die Arten AS_CONFED_SEQUENCE und AS_CONFED_SET der AS_PATH-Eigenschaft erkennen müssen. Da diese AS_PATH-Arten jedoch entfernt werden, wenn eine Nachricht aus der Confederation heraus gesendet wird, müssen Router in anderen Autonomen Systemen die Confederation nicht unterstützen.

In besonders großen Autonomen Systemen können Confederations und Route Reflectors zusammen benutzt werden. Sie können einen oder mehrere RR Cluster mit einem oder mehreren Member AS konfigurieren, wenn sie die IBGP-Peers besonders gut kontrollieren wollen.

2.6 BGP-Nachrichtenformate

BGP-Nachrichten werden in TCP-Segmenten durch TCP Port 179 getragen. Die maximale Nachrichtengröße ist 4.096 Oktette, die minimale Größe ist 19 Oktette. Alle BGP-Nachrichten haben einen gemeinsamen Header (siehe Abbildung 2.43). Diesem Header kann bei manchen Nachrichten ein Datenteil folgen.

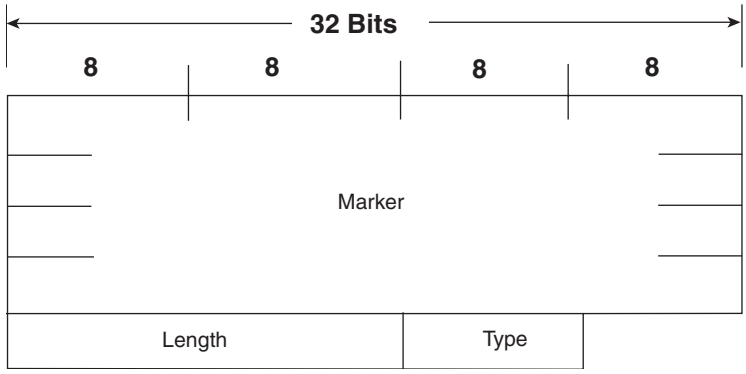

Abb. 2.43: Der BGP-Nachricht-Header

Marker ist ein 16-Oktett-Feld, das verwendet wird, um einen Verlust der Synchronisation zwischen BGP-Peers aufzuspüren und das auch dafür zuständig ist, Nachrichten zu beglaubigen, wenn diese Option eingeschaltet ist. Wenn es sich um eine Open-Nachricht handelt oder diese keine Authentisierungs-Informationen enthält, wird das Marker-Feld nur mit Einsen ausgefüllt. Ansonsten wird der Wert das Marker-Feldes während des Authentisierungs-Prozesses berechnet.

Length ist ein 0-Oktett-Feld, das die gesamte Länge der Nachricht mit Header in Oktetten bestimmt.

Type ist ein 0-Oktett-Feld, das die Nachrichtenart angibt. Tabelle 2.6 zeigt die verschiedenen Type-Codes.

Tabelle 2.6: BGP Type-Codes

Code	Type
1	Open
2	Update
3	Notification
4	Keepalive

2.6.1 Die Open-Nachricht

Die Open-Nachricht, deren Format in Abbildung 2.44 zu sehen ist, ist die erste Nachricht, die verschickt wird, nachdem eine TCP-Verbindung aufgebaut wurde. Wenn eine Open-Nachricht akzeptiert wird, wird eine Keepalive-Nachricht geschickt, um die Open-Nachricht zu bestätigen. Nachdem die Open-Nachricht bestätigt wurde, ist die BGP-Verbindung im Established Status und Update-, Keepalive- und Notification-Nachrichten können verschickt werden.

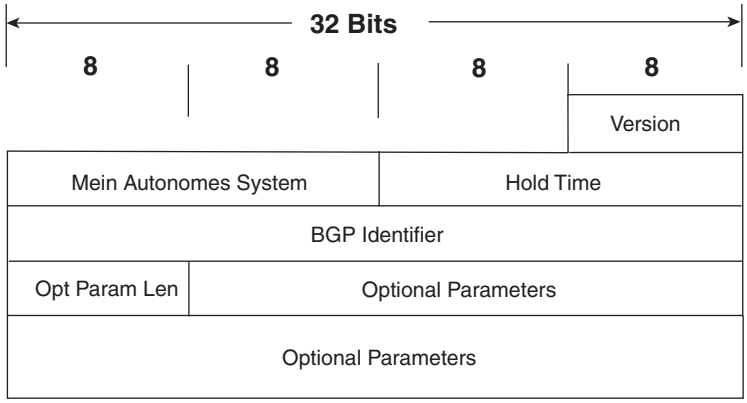

Abb. 2.44: Das BGP Open-Nachrichtenformat

Die BGP Open-Nachricht enthält folgende Felder:

- **Version –** Ein 1-Oktett-Feld, das die BGP-Version, die beim Sender läuft, angibt
- **Mein Autonomes System –** Ein 2-Oktett-Feld, in dem die AS-Nummer des Senders steht
- **Hold Time –** Eine 2-Oktett-Zahl, die für die Zahl der vom Sender vorgeschlagenen Sekunden für die Hold Time steht. Ein Empfänger vergleicht die Zahl im Hold Time-Feld mit der eigenen Hold Time und akzeptiert den kleineren Wert oder lehnt die Verbindung ab. Die hold Time muss entweder null oder mindestens drei Sekunden betragen.
- **BGP Identifier –** Die Router ID des Ursprungs-Routers. Ein Cisco-Router verwendet für die Router ID entweder die höchste IP-Adresse seiner Loopback-Schnittstellen, oder wenn es keine Loopback-Schnittstellen gibt, die höchste IP-Adresse seiner physischen Schnittstellen.

– **Optional Parameters Length –** Ein 1-Oktett-Feld, das die Länge des folgenden Optional Parameters-Feldes in Oktetten angibt. Wenn der Wert des Feldes Null ist, hat die Nachricht kein Feld Optional Parameters.

– **Optional Parameters –** Ein Feld mit variabler Länge, welches eine Liste von freiwilligen Parametern enthält. Jeder Parameter wird durch ein 1-Oktett-Type-Feld, einem 1-Oktett-Length-Feld und einem unterschiedlich langem Feld mit dem Parameter selbst beschrieben.

2.6.2 Die Update-Nachricht

Die Update-Nachricht, deren Format in Abbildung 2.45 gezeigt ist, wird verwendet, um eine einzige mögliche Route zu einem Peer anzugeben oder um mehrere mögliche Routen zurückzuziehen oder beides.

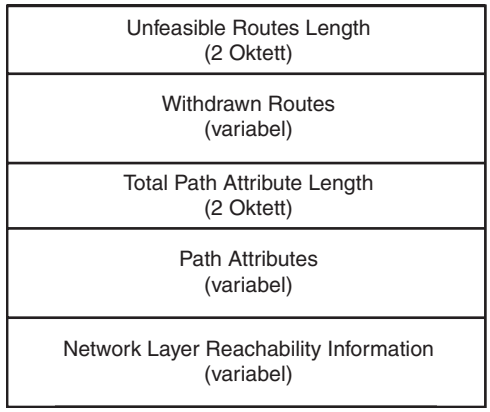

Abb. 2.45: Das BGP-Update-Nachrichtenformat

Die BGP-Update-Nachricht enthält folgende Felder:

– **Unfeasible Routes Length –** Ein 2-Oktett-Feld, das die Länge des folgenden Feldes Withdrawn Route angibt. Ein Wert von Null bedeutet, dass keine Routen zurückgenommen werden und die Nachricht also kein Withdrawn Route-Feld enthält.

– **Withdrawn Routes –** Ein Feld mit variabler Länge, das eine Liste von Routen enthält, die nicht mehr verwendet werden. Jede Route wird mit einem (Length, Präfix) Tupel beschrieben, bei dem Length die Länge des Präfixes ist und Präfix das IP-Adress-Präfix der zurückgenommenen Route ist. Wenn der Teil Length des Tupels null ist, passt das Präfix zu allen Routen.

– **Total Path Attribute Length –** Ein 2-Oktett-Feld, das die Länge des folgenden Path Attribute-Feldes in Oktetten angibt. Ein Wert von Null bedeutet, dass die Nachricht keine Eigenschaften oder NLRI enthält.

– **Path Attributes –** Ein Feld mit variabler Länge, welches die Eigenschaften angibt, die mit dem folgenden NLRI-Feld verbunden werden. Jede Pfadeigenschaft (Path Attribute) besteht aus Attribute Type, Attribute Length, Attribute Value. Die Attribute Type ist ein 2-Oktett-Feld, welches aus vier Flag-Bits, vier unbenutzten Bits und einem Attribute Type-Code besteht (siehe Abbildung 2.46).

– **Network Layer Reachability Information –** Ein Feld mit variabler Länge, welches eine Liste von (Length, Präfix) Tupeln enthält. Die Length gibt die Länge, (in Bits) des folgenden Präfixes an, das Präfix ist das IP-Adress-Präfix der NLRI. Ein Length-Wert von Null bedeutet, dass das Präfix mit allen IP-Adressen übereinstimmt.

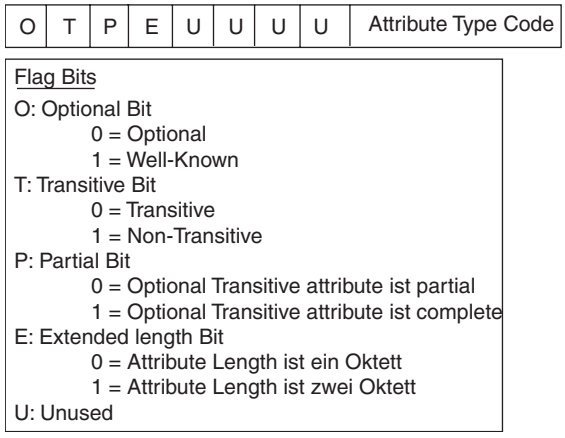

Abb. 2.46: Der Attribute Type-Teil des Path Attributes-Feldes

Abbildung 2.46 zeigt die häufigsten Attribute Type-Codes und die möglichen Eigenschaftswerte jeder Attribute Type.

*Tabelle 2.7: Attribute Types und Eigenschaftswerte (Attribute Value)**

Attribute Type-Code	Attribute Type	Attribute Value-Code	Attribute Value
1	ORIGIN	0	IGP
		1	EGP
		2	Unvollständig

Tabelle 2.7: Attribute Types und Eigenschaftswerte (Attribute Value) (Forts.)*

Attribute Type-Code	Attribute Type	Attribute Value-Code	Attribute Value
2	AS_PATH	1	AS_SET
		2	AS_SEQUENCE
		3	AS_CONFED_SET
		4	AS_CONFED_SEQUENCE
3	NEXT_HOP	0	Next-hop IP Adresse
4	MULTI_EXIT_DISC	0	4-Bytes MED
5	LOCAL_PREF	0	4-Bytes LOCAL_PREF
6	ATOMIC_AGGREGATE	0	Nichts
7	AGGREGATOR	0	AS Nummer und IP Addresse des Aggregators
8	COMMUNITY	0	4-Bytes »community identifier«
9	ORIGINATOR_ID	0	4-Bytes Router ID des Urhebers
10	CLUSTER_LIST	0	Variabel lange Liste der Cluster IDs

* Es gibt auch andere Eigenschafts-Typen, die aber nicht zu Cisco-Produkten gehören und in diesem Buch deswegen nicht behandelt werden.

2.6.3 Die Keepalive-Nachricht

Keepalive-Nachrichten werden in einem Abstand ausgetauscht, der ein Drittel der Hold Time beträgt, aber nie niedriger als eine Sekunde ist. Wenn die Hold Time 0 ist, werden keine Keepalives verschickt.

Die Keepalive besteht nur aus dem 19-Oktett-BGP-Nachrichten-Header ohne weitere Daten.

2.6.4 Die Notification-Nachricht

Notification-Nachrichten, deren Format in Abbildung 2.47 zu sehen ist, werden verschickt, wenn ein Fehler auftritt. Die BGP-Verbindung wird sofort beendet, wenn eine solche Nachricht verschickt wird.

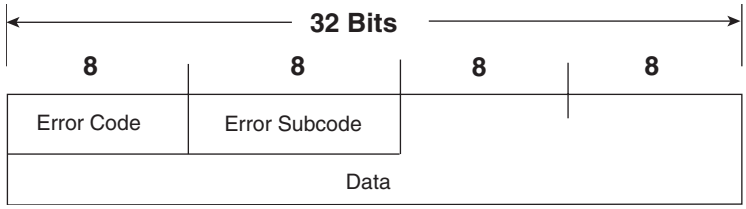

Abb. 2.47: Das BGP Notification-Nachrichtenformat

Die BGP Notification-Nachricht enthält folgende Felder:

– **Error Code –** Ein 1-Oktett-Feld, das die Art des Fehlers beschreibt

– **Error Subcode –** Ein 1-Oktett-Feld, das genauere Informationen über den Fehler gibt. Tabelle 2.8 zeigt die möglichen Fehler-Codes und deren Subcodes.

– **Data –** Ein Feld mit variabler Länge, das verwendet wird, um den Grund für den Fehler herauszufinden. Der Inhalt des Data-Feldes ist vom Fehler-Code und Subcode abhängig.

Tabelle 2.8 BGP Notification-Nachricht – Fehler-Codes und Fehler-Subcodes

Fehler-Code	Fehler	Fehler-Subcode	Subcode-Detail
1	Message Header Error	1	Verbindung nicht synchronisiert
		2	Fehlerhafte Nachrichtenlänge
		3	Fehlerhafte Nachrichtenart
2	Open Message Error	1	Nicht unterstützte Versionsnummer
		2	Fehlerhftes Peer AS
		3	Fehlerhafter BGP Identifier
		4	Nicht unterstützter Optional Parameter
		5	Authentisierungsfehler
		6	Inakzeptable Hold Time
3	Update Message Error	1	missgebildete Eigenschaften-Liste
		2	Unbekannte well-known Eigenschaft
		3	Fehlende well-known Eigenschaft
		4	Attribute flags Error

Tabelle 2.8 BGP Notification-Nachricht – Fehler-Codes und Fehler-Subcodes

Fehler-Code	Fehler	Fehler-Subcode	Subcode-Detail
		5	Attribute length Error
		6	Ungültige ORIGIN- Eigenschaft
		7	AS Routing-Loop
		8	Ungültige NEXT_HOP-Eigenschaft
		9	Freiwillige Eigenschaft Error
		10	Ungültiges Network-Feld
		11	missgebildeter AS_PATH
4	Hold Timer abgelaufen	0	–
5	Finite State Machine Error	0	–
6	Cease	0	–

2.7 Endnoten

1 K. Lougheed and Y. Rekhter, »RFC 1105: A Border Gateway-Protocol (BGP)« (in Arbeit)

2 K. Lougheed and Y. Rekhter, »RFC 1163: A Border Gateway-Protocol (BGP)« (in Arbeit)

3 K. Lougheed and Y. Rekhter, »RFC 1267: A Border Gateway-Protocol 3 (BGP-3)« (in Arbeit)

4 Y. Rekhter and T. Li, »RFC 1771: A Border Gateway-Protocol 4 (BGP-4)« (in Arbeit)

5 Internet Engineering Steering Group, R. Hinden, Editor, »RFC 1517: Applicability Statement for the Implementation of Classless Inter-Domain Routing (CIDR)« (in Arbeit)

6 V. Fuller et al., »RFC 1519: Classless Inter-Domain Routing (CIDR): An Address Assignment and Aggregation Strategy« (in Arbeit)

7 Y. Rekhter and C. Topolcic, »RFC 1520: Exchanging Routing-Information Across Provider Boundaries in the CIDR Environment« (in Arbeit)

8 R. Chandra and P. Traina, »RFC 1997: BGP-Communities Attribute« (in Arbeit)

2.8 Ausblick

Da Sie nun die Grundlagen von BGP und einigen verwandten Themen kennen, zeigt Ihnen Kapitel 3, wie Sie BGP auf Cisco-Routern konfigurieren und Fehler beseitigen. Zusätzlich werden Sie lernen, wie man Routing-Richtlinien umsetzt und wie BGP und IGPs redistributiert werden können.

2.9 Empfohlene Literatur

Halabi B. und D. McPherson, *Internet-Routing Architectures, Second Edition*. Indianapolis, Indiana: Cisco Press; 2000.

Diese Buch wird von vielen als das Standardwerk über BGP-4 gesehen.

Stewart J.W. III. *BGP4: Inter-Domain Routing in the Internet*. Reading, Massachusetts: Addison Wesley Longman; 1999.

Obwohl dieses Buch sich nicht nur auf Cisco bezieht, ist es ein nützlicher und präziser BGP-Überblick.

2.10 Wiederholungsfragen

1. Was ist der wichtigste Unterschied zwischen BGP-4 und früheren BGP-Versionen?

2. Welche beiden Probleme sollte CIDR verringern?

3. Was ist der Unterschied zwischen classfull und classless IP-Routern?

4. Was ist der Unterschied zwischen classfull und classless IP Routing-Protokollen?

5. Fassen Sie die Adressen 172.17.208.0/23, 172.17.210.0/23, 172.17.212.0/23 und 172.17.214.0/23 zu einer einzigen Verdichtung zusammen und benutzen Sie dabei eine möglichst lange Adressmaske.

6. Was ist ein Adresspräfix?

7. Die Routing-Tabelle aus Beispiel 2.16 stammt von einem classless Router. Zu welcher next Hop Adresse versendet ein Router Pakete mit folgenden Zieladressen?

   172.20.3.5

   172.20.1.67

   172.21.255.254

   172.16.50.50

   172.16.0.224

   172.16.51.50

   172.17.40.1

   172.17.41.1

   172.30.1.1

Beispiel 2.16: Die Routing-Tabelle für Wiederholungsfrage 7

```
Stratford#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

     172.20.0.0 is variably subnetted, 6 subnets, 2 masks
D       172.20.0.0 255.255.0.0 [90/409600] via 172.20.5.2, 00:01:50, Ethernet0
D       172.20.2.0 255.255.255.0
           [90/409600] via 172.20.6.2, 00:01:50, Ethernet1
D       172.20.3.0 255.255.255.0
           [90/5401600] via 172.20.6.2, 00:01:50, Ethernet1
C       172.20.5.0 255.255.255.0 is directly connected, Ethernet0
C       172.20.6.0 255.255.255.0 is directly connected, Ethernet1
C       172.20.7.0 255.255.255.0 is directly connected, Ethernet2
     172.16.0.0 is variably subnetted, 3 subnets, 2 masks
D       172.16.50.0 255.255.255.0
           [90/409600] via 172.20.6.2, 00:01:50, Ethernet1
D       172.16.0.0 255.255.255.0
           [90/460800] via 172.20.6.2, 00:01:51, Ethernet1
D       172.16.0.0 255.255.0.0 [90/409600] via 172.20.7.2, 00:01:51, Ethernet2
     172.17.0.0 is subnetted (mask is 255.255.255.0), 1 subnets
D       172.17.40.0 [90/2841600] via 172.20.7.2, 00:01:52, Ethernet2
D   172.16.0.0 (mask is 255.240.0.0) [90/409600] via 172.20.5.2, 00:01:52, Ethernet0
Stratford#
```

8. Erklären Sie, wie eine Zusammenfassung dabei hilft, Instabilitäten im Netzwerk zu verstecken.

9. Erklären Sie, wie eine Zusammenfassung asymmetrische Verkehrsmuster hervorrufen kann.

10. Ist asymmetrischer Verkehr nicht wünschenswert?

11. Was ist ein NAP?

12. Was ist ein Route Server?

13. Was sind provider-independent Adressen und warum kann es von Vorteil sein, eine zu besitzen?

14. Wieso kann es ein Problem sein, einen /21 provider-independent Adressblock zu haben?

15. Was ist eine Routing-Richtlinie?

16. Welches unterliegende Protokoll verwendet BGP, um sich mit Nachbarn zu verständigen?

17. Was sind die vier BGP-Nachrichtenarten, wie werden sie jeweils benutzt?

18. In welchem oder in welchen Status können BGP-Peers Updates austauschen?

19. Was ist NLRI?

20. Was ist eine Pfadeigenschaft?

21. Was sind die vier Kategorien von BGP-Pfadeigenschaften?

22. Was ist die Funktion der AS_PATH-Eigenschaft?

23. Was sind die verschiedenen Arten von AS_PATH?

24. Was ist die Funktion der NEXT_HOP-Eigenschaft?

25. Was ist die Funktion der LOCAL_PREF-Eigenschaft?

26. Was ist die Funktion der MULTI_EXIT_DISC-Eigenschaft?

27. Welche Eigenschaft ist oder welche Eigenschaften sind nützlich, wenn ein BGP-Sprecher eine Route zusammenfasst?

28. Was ist ein BGP Administrative Weight?

29. Welche Route wählt BGP, wenn es eine EBGP-Route und eine IBGP-Route zum selben Ziel gibt?

30. Router A hat zwei IBGP-Routen zum selben Ziel. Pfad A hat eine LOCAL_PREF von 300 und drei AS-Nummern in der AS_PATH. Pfad B hat eine LOCAL_PREF von 200 und zwei AS-Nummern in der AS_PATH. Welchen Pfad wählt der Router aus, wenn es keinen anderen Unterschied zwischen den Routen gibt?

31. Was ist Route Dampening?

32. Definieren Sie Penalty, suppress Limit, Reuse Limit und Half-Life im Zusammenhang mit Route Dampening.

33. Was ist IGP-Synchronisation, warum ist sie wichtig?

34. Unter welchen Umständen kann IGP-Synchronisation sicher abgeschaltet werden?

35. Was ist eine BGP-Peer-Gruppe?

36. Was ist eine BGP-Community?

37. Was ist ein Route Reflector? Was ist ein Route Reflection Client? Was ist ein Route Reflection Cluster?

38. Was ist die Funktion der Pfadeigenschaften ORIGINATOR_ID und CLUSTER_LIST?

39. Was ist eine BGP-Confederation?

40. Können Route Reflectors innerhalb einer Confederation verwendet werden?

41. Welche Bedeutung hat die Funktion **next Hop-self**? Gibt es für diese Funktion gleichwertige Alternativen?