

# Border Gateway Protocol v 4

Marcel Noe

November 2004

## 1 Motivation

Das Internet ist ein sehr komplexer Verbund aus vielen unterschiedlichen Netzen. Aufgrund dieser Komplexität und der Tatsache, dass das Internet einem ständigen Strukturwandel unterliegt, ist an eine manuelle Pflege der Routen zwischen diesen unterschiedlichen Netzen nicht mehr zu denken. Aus diesen Gegebenheiten erwächst das Bedürfnis nach einer weitestgehend automatischen Lösung zur Pflege der Routen zwischen den einzelnen Netzen. Diese Lösung muss darüber hinaus noch ein hohes Maß an Robustheit und Ausfallsicherheit mitbringen. Im allgemeinen verwendet man deshalb sogenannte dynamische Routing-Protokolle, die automatisch die Routen zwischen einzelnen Teilnetzen aushandeln. Im Internet kommt hierfür das Border Gateway Protokoll (BGP) zum Einsatz.

## 2 Autonome System

Zunächst definieren wir den Begriff des Autonomen Systems. Ein Autonomes System ist ein Netzwerk bzw. eine Gruppe von Netzwerken, die eine gemeinsame Administration sowie eine gemeinsame Routing Policy (siehe unten) besitzen. Jedes Autonome System ist durch eine eindeutige Nummer – die AS-Nummer – gekennzeichnet, die wie die IP-Adressen von einer zentralen Stelle vergeben wird. [1] Einige Beispiele für Autonome Systeme sind z.B. das BelWue (553), KPN Eurorings (286) sowie das DTAG Netzwerk (Deutsche Telekom AG) (3320).

## 3 Einordnung des BGP

Routing-Protokolle kann man in zwei Bereiche einteilen:[1]

Interne Routing-Protokolle (Interior Gateway Protocols):

Diese Protokolle verbreiten Routing-Informationen innerhalb eines Autonomen Systems.

Beispiele: RIP, OSPF

Externe Routing-Protokolle (Exterior Gateway Protocols):

Diese Protokolle verbreiten Routing-Informationen zwischen Autonomen Systemen.

BGP ist momentan das einzige verwendete externe Routing-Protokoll.

## 4 Routing Algorithmus und grundlegendes Design

Das BGP basiert auf einem sogenannten Distanzvektor Algorithmus. Bei einem Distanzvektor basierten Routing-Protokoll speichert jeder Router einen Distanz-Vektor mit Einträgen zu jedem Zielrouter. Jeder dieser Einträge besteht unter anderem aus den Kosten, um das Ziel zu erreichen, sowie dem nächsten Hop zum Ziel. Falls mehrere Routen zu einem Ziel existieren, wird diejenige mit den geringsten Kosten verwendet.[5]

Das BGP erweitert dieses Verhalten nun um einige Aspekte. Die Basisinformation im BGP ist der BGP Pfad (BGP Path), also die Route zu einem bestimmten Set von CIDR Präfixen. Diese Pfade werden mit einigen Attributen versehen, wovon die wichtigsten AS\_PATH und NEXT\_HOP sind. Der AS\_PATH ist prinzipiell eine Liste von allen autonomen Systemen, die zum Erreichen eines Ziels durchquert werden müssen. Diese Liste dient primär dazu, Schleifen zu verhindern. Ein Router ignoriert alle BGP-Pfade, in denen die eigene AS-Nummer im AS\_PATH-Attribut enthalten ist.

Im NEXT\_HOP-Attribut ist die IP-Adresse des Border Routers des nächstgelegenen AS gespeichert, das zum Erreichen des Ziels durchquert werden muss. Hierin liegt ein großer Unterschied zu anderen Distanzvektorbasierten Protokollen, die den nächsten Hop zu einem Ziel speichern. Der Router, der im BGP-NEXT\_HOP-Attribut steht, kann im Gegensatz hierzu jedoch durchaus

einige Hops entfernt sein.[6]

Hieraus ergeben sich in erster Linie folgende beide Konsequenzen: Zum einen ist BGP für sich alleine nicht ausreichend, um ein bestimmtes Ziel zu erreichen. Hierzu sind entweder statische Routen zu den Border-Routern eines AS oder besser ein internes Routing-Protokoll erforderlich. Zum anderen müssen alle BGP-Teilnehmer innerhalb eines AS vollvermascht sein, d.h. von jedem Router zu jedem anderen Router muss eine BGP-Verbindung existieren. Dies kann sehr schnell zu einem Problem werden, weil die Anzahl der Verbindungen quadratisch mit der Anzahl der Router wächst. Diesem Effekt kann man prinzipiell entweder durch Zusammenschluss mehrerer Router zu einem Sub-AS – einer sogenannten Konföderation – oder durch den Einsatz von Reflektoren entgegenwirken[1]. An dieser Stelle sei lediglich auf diese Möglichkeiten hingewiesen, eine genauere Diskussion würde den Rahmen dieses Dokumentes bei weitem sprengen.

## 5 Auswahl der besten Route

Aufgrund der Struktur des Internets ist es nicht unüblich, dass sehr viele Routen zu einem bestimmten Ziel führen. Man muss nun entscheiden, welche Route die beste ist, um über diese Route den Traffic zu einem Ziel zu senden. Das Standardverhalten von BGP ist, die Route zu einem Ziel, die die größte Übereinstimmung aufweist, zu verwenden, also die mit der längsten CIDR Maske. Gibt es mehrere Routen mit der selben Prefix-Länge, wird diejenige mit dem kürzesten AS\_PATH verwendet. Leider ist die Route mit dem kürzesten AS\_PATH nicht immer auch die beste. Daher gibt es die Möglichkeit, lokale Präferenzen für bestimmte Ziele zu setzen. Hierzu verwendet man sogenannte Route-Maps, um festzulegen, welche Routen bevorzugt werden sollen[1][7].

## 6 Ablauf einer BGP Session

Im Gegensatz zu den meisten anderen Routing-Protokollen, die ihre Informationen per Broadcast über das lokale Netz verteilen, arbeitet BGP mit sogenannten Sessions. Eine Session findet immer zwischen zwei Routern statt. Die beiden an einer Session beteiligten Router nennt man BGP-Nachbarn oder BGP-Peers. BGP verwendet TCP als Transport-Protokoll, unterscheidet sich also auch hier von anderen Routing-Protokollen, die meist direkt auf der IP-Schicht aufsetzen. Die Verbindung findet auf TCP-Port 179 statt und ist während der gesamten Session aktiv. Eine Session beginnt also mit dem Aufbau der Verbindung und gilt als etabliert sobald das erste Keep-Alive Paket gesendet wurde. Eine Session endet mit dem Abbau derselben TCP-Verbindung. Innerhalb einer Session gibt es folgende Steuernachrichten[1][7]:

**Open:** Eine BGP-Session beginnt immer mit einer Open-Message, die sich die Nachbarn direkt nach dem Aufbau der Verbindung gegenseitig zusenden. Diese Message beinhaltet unter anderem die Protokollversion, die von dem jeweiligen Nachbarn verwendet wird, die AS-Nummer, die maximale Idle-Time einer Verbindung sowie weitere optionale Felder wie z.B. Authentifizierungsinformationen oder zusätzliche Features. Außerdem beinhaltet die Message den sogenannten Identifier eines Nachbarn, welcher einfach eine der IP-Adressen eines Interfaces des ist.

**Keep-Alive:** Wenn über einen gewissen Zeitraum (der mit der Open-Message festgelegt wird) keine Steuernachrichten zwischen den Nachbarn gesendet werden, geht der eine Nachbar davon aus, dass die Gegenstelle nichtmehr erreichbar ist und beendet die Session mit einem Timeout. Um dies zu verhindern werden in regelmäßigen Abständen Keep-Alive-Message übertragen.

**Update:** Mithilfe von Update-Message teilen sich die Nachbarn gegenseitig ihre Routing-Tabellen mit. Hierbei wird allerdings nur direkt nach dem Aufbau der Verbindung die gesamte Routing-Tabelle übertragen. Später teilen sich die Nachbarn nur noch Änderungen mit, um den durch BGP verursachten Datenverkehr möglichst gering zu halten.

**Notification:** Notification-Message beenden eine BGP-Session. Sie werden entweder bei einem fatalen Fehler oder beim Terminieren einer Session generiert. In der Nachricht sind die Art des Fehlers, eine genauere Information und optional Diagnosedaten enthalten.

## 7 Routing-Policies

Das Policy-Routing Konzept (auch Routing-Policies genannt) ist ein Feature von BGP, dessen Ziel es ist, den Datenfluss eines Autonomen Systems zu kontrollieren. Zur Implementierung dieses Konzeptes bietet BGP verschiedene Hilfsmittel wie Routen-Filter und Communities (s.u.).

Ziel von Policies ist es, das Standardverhalten des Routing-Protokolls aufzuheben, und das Verhalten an die Wünsche des Administrators anzupassen. Mit Policies versucht man vor allem folgende Ziele zu erreichen:[8]

Kosten sparen: Dies kann erreicht werden, indem man z.B. weniger wichtigen Traffic über weniger zuverlässige, günstigere Routen und wichtigen Traffic über zuverlässigere, teure Routen lenkt.

Last verteilen: Wenn mehrere Routen zu einem Ziel bestehen, die in etwa gleichwertig sind, kann man den Traffic zu diesem Ziel über diese Routen verteilen, um vorhandene Leitungen so besser auszunutzen.

Verteidigung gegen DoS: Indem man die zur Verfügung stehende Bandbreite für typischen Denial-of-Service-Traffic limitiert oder versucht, den DoS-Traffic ins Leere zu lenken (siehe Communities) kann man die Folgen des Angriffs abschwächen.

Beschränkung des Routingtabellen-Umfangs: Um Ressourcen des Routers zu schonen, kann es sinnvoll sein, nicht jede Route, die per BGP verbreitet wird, in die Routing-Tabelle zu übernehmen (siehe Filter). Dies war vor allem früher sehr wichtig, da vielerorts noch Router mit zu wenig Speicher, um alle per BGP verbreiteten Routen zu verwalten, im Einsatz waren.

Quellensensitives Routing: Es ist möglich, Traffic von unterschiedlichen Quellen über unterschiedliche Routen zu lenken.

## 8 Filter

Filter werden verwendet, um eine Vorauswahl zu treffen, welche mittels BGP verbreiteten Routen man in die eigene Routing-Tabelle übernehmen möchte. Je größer die Routing-Tabelle eines Routers wird, desto mehr CPU-Leistung und vor allem Speicher benötigt er, um diese zu verarbeiten. Natürlich kann man einfach leistungsfähigere Hardware einsetzen, aber diese ist teuer und es ist auch nicht unbedingt sinnvoll, jedes Announcement anzunehmen. Welche Announcements man annimmt und welche nicht, hängt von der jeweiligen Routing Policy eines Autonomen-Systems ab. Prinzipiell kann man entweder nach AS-Pfaden oder nach den verbreiteten CIDR Präfixen filtern (Ein CIDR-Präfix ist einfach die Kombination aus einem Teil einer IP Adresse sowie der Angabe der Präfix-Länge — z.B. 192.168.255.0/24, 10.1.0.0/16, 128.0.0.0/2 usw.). Gängig ist es zum einen, Routen auszufiltern, zu deren Erreichen man mehr als eine bestimmte Anzahl verschiedener AS durchqueren müsste, und zum anderen nur Announcements bis zu einer bestimmten Länge zuzulassen.

Es ist allerdings auch möglich, Filter auf ausgehende Announcements zu definieren. So ist es z.B. sinnvoll, nur Announcements der eigenen IP-Bereiche zu erlauben, um zu verhindern, dass Transit-Traffic von Peering-Partnern durch das eigene Netz fließt und so nutzlos Leitungen verstopft bzw. Traffickosten verursacht[1][4].

## 9 Schwarze Löcher

Falsche, vergessene oder unzureichende Filter können dazu führen, dass fremde Systeme Routen empfangen, zu denen sie keinen Traffic senden können. Solche Routen, die zwar verbreitet werden, aber bei denen nie Traffic ankommt, nennt man schwarze Löcher. Ein schwarzes Loch ist einer der schlimmsten Effekte, die beim Einsatz von BGP auftreten können. Das Problem ist, dass die Ursachen und vor allem die Stelle des Verschwindens des Traffics meist nur schwer lokalisiert werden können. Neben dem Ausfall eines Links können Paketfilter oder schlichtweg Fehlkonfiguration die Ursache für das Verschwinden des Traffics sein. Ein typisches Szenario ist, wenn Kunden mit mehreren Netzwerkverbindungen unzureichende Filter gesetzt haben und so versehentlich die Routen von fremden Netzwerken zu einem ihrer Provider verbreiten, und der Provider versäumt hat, Filter zu konfigurieren, die von einem Kunden lediglich Announcements über dessen eigenen IP-Bereich akzeptieren. Da ein Provider Routen über Kunden als besonders günstig konfiguriert hat, werden die Router versuchen, ausgehenden Traffic gesamt oder teilweise über den Kunden zu routen, der in aller Regel dieser Datenflut nicht gewachsen ist. Dass man fälschlicherweise Routen verbreitet, zu denen man eigentlich gar keinen Fremddtraffic transportieren möchte, entsteht durch das fälschliche Injizieren von Routen, die mittels BGP gelernt wurden, in das interne Routing Protokoll. Dies wird genau dann zum Problem, wenn man die Informationen des internen Routing-Protokolls wieder an das BGP exportiert. Dieses Setup ist zwar möglich, setzt aber sehr genaue Pflege der Filter voraus, und ist auf Grund der wenigen Vorteile, die man sich durch eine sehr hohe Fehleranfälligkeit erkaufte, nicht empfehlenswert. Man muss allerdings an dieser Stelle noch anmerken, dass ein Schwarzes Loch manchmal bewußt erzeugt wird, um z.B. DoS Angriffe ins Leere zu leiten[1].

## 10 Communities

Mit Communities besteht die Möglichkeit, Routen zu markieren. Damit kann man Routern, die diese Routen empfangen, signalisieren, wie sie damit umzugehen haben. Communities sind ein sehr nützliches Werkzeug. Man kann es z.B. einsetzen, um eine Route gegenüber anderen zu bevorzugen, oder aber um DoS-Attacken ins Leere zu leiten. Communities werden oft in einer Kombination mit einer Technik, die sich Path-Prepending nennt, eingesetzt. Hierbei wird beim verbreiten an bestimmte Nachbarn die Nummer des eigenen AS öfter als einmal in das AS\_PATH-Attribut eingefügt, womit man eine gewisse Kontrolle hat, über welchen Nachbarn eingehender Traffic ankommt. Darüber hinaus kann man eine Route mit der Quelle, die sie announced hat, markieren. Anhand dieser Markierung kann man dann entscheiden, welche Route man für ausgehenden Traffic zu einem bestimmten Ziel bevorzugt. Auch dies wird dann wieder über Path-Prepending realisiert[1][9].

## 11 Verhalten beim Ausfall eines Pfades

Um das Verhalten beim Ausfall eines Pfades zu verdeutlichen, stellen wir uns folgendes Szenario vor: Provider A und B tauschen Datenpakete über einen Peering-Punkt aus. Dazu betreibt jeder der Provider einen Router (die wir zum besseren Verständnis Router A und Router B nennen), die über eine Fast-Ethernet-Verbindung verbunden sind. Zwischen beiden Routern besteht eine BGP-Session. Auf Grund eines Hardwarefehlers kommt die Betriebssoftware von Router B in einen indeterministischen Zustand, was dazu führt, dass über die Verbindung zwar keine Pakete mehr ankommen, das Interface allerdings nicht heruntergefahren wird und so Router A nicht sofort merkt, dass Router B nicht mehr verfügbar ist. Da Router A nun keine Keep-Alive-Pakete mehr von Router B bekommt, beendet er die BGP-Session nach dem in der Open-Message vereinbarten Timeout. So lange der Timeout allerdings nicht erreicht ist, versucht er weiterhin über diese Verbindung Pakete zu senden, die allerdings ihr Ziel niemals erreichen. Sobald die BGP-Session beendet ist, entfernt Router A die Route über Router B und sucht sich die nächstbeste Route. Diese übernimmt er in seine Routingtabelle. Danach sendet er eine Update-Nachricht an seine anderen BGP-Nachbarn, in der er diese über die geänderte Route informiert. Diese verbreiten diese Information ihrerseits an ihre Nachbarn weiter, was dazu führt, dass sich die neuen Routing-Informationen schneeballartig verbreiten. Alle Router fangen nach dem empfangen der Update Meldung an, ihre Routing Tabellen neu zu berechnen. Da der Weg über Provider A zu Provider B nun evtl. länger ist als der über andere Routen, kann es passieren, dass die Router ganz andere Wege wählen[1][9].

## 12 Schwachstellen von BGP

Beim Einsatz von BGP hat man primär mit zwei Problemen zu kämpfen. Das eine Problem ist die Anzahl der Routen, die über BGP verbreitet werden. Jeder Eintrag in der Routingtabelle belegt Systemressourcen. Von daher kann es sinnvoll sein, die Anzahl der Routen durch Filter einzuschränken. Ein viel größeres Problem von BGP ist allerdings die Tatsache, dass jede Änderung einer Route das ganze Internet betrifft. Eine Update-Message verbreitet sich immer über alle BGP sprechenden Router, die darauf hin ihre Routing-Tabelle neu berechnen müssen. Dies macht BGP sehr gefährlich. Eine kleine Unachtsamkeit in der Konfiguration kann globale Ausmaße annehmen und schlimmsten Falls dazu führen, dass große Teile des Internets sich gegenseitig nicht mehr erreichen können. Diese Effekte kann man zwar durch Filter und andere Sicherheitsmaßnahmen wie Damping reduzieren, aber nie ganz verhindern. BGP ist sozusagen der Single Point of Failure des gesamten Internets.[1][9]

## Literatur

- [1] Iljitsch van Beijnum *The Border Gateway Protocol* O'Reilly Verlag - ISBN: 0-596-00254-8
- [2] *RFC 1771: A Border Gateway Protocol 4 (BGP-4)* <ftp://ftp.rfc-editor.org/in-notes/rfc1771.txt>
- [3] *Wikipedia: Border Gateway Protocol* <http://en.wikipedia.org/wiki/Bgp>
- [4] *Riverstonenet: Routing Policies* <http://www.riverstonenet.com/support/bgp/policies/>
- [5] *Brief explanation of Routing algorithms* <http://mia.ece.uic.edu/~papers/Networking/msg00004.html>
- [6] *Freesort: BGP-4 Protocol Overview* <http://www.freesoft.org/CIE/Topics/88.htm>
- [7] *Signaltonoise: Border Gateway Protocol (BGP)* <http://www.signaltonoise.net/library/bgp.html>
- [8] *Cisco: Policy based Routing* [http://www.cisco.com/warp/public/cc/techno/protocol/tech/policy\\_wp.htm](http://www.cisco.com/warp/public/cc/techno/protocol/tech/policy_wp.htm)
- [9] *BGP ROUTING PART I* <http://www.supersparrow.org/ss-0.0.0/reference/avi/bgp.html>