BGP

Border Gateway Protocol

Class I - Introduction

The Rosetta Stone – discovered in 1799, it was the key that finally allowed scholars to understand Egyptian Hieroglyphs. It contained writing in three languages: Hieroglyphs, Demotic and Greek. The Rosetta Stone enabled the translation from a known language to the then indecipherable hieroglyphs.

Gary A. Donahue

October 17, 2001

gad@gad.net

Comments welcome

Table of Contents

Introduction

Intended Audience

This document is intended for those people wishing to learn what BGP is and what it can be used for. It is not intended to be a complete BGP course, and will not delve into such topics as "how to configure a router for BGP" etc.

Anyone who has an interest in BGP from a basic standpoint will be well served by this document.

This document is the first in a series of three.

Routing Protocols in General

What is a routing protocol? Well if you're interested in BGP, then you probably already know this, but lets recap:

routing protocol: In an internet, a service protocol that is used (by routers, but not by hosts) to maintain routing tables; routing protocols are classified as either (a) interior gateway protocols, or (b) exterior gateway protocols.

An interior gateway protocol (IGP) is used to share routing information within an autonomous system, while an exterior gateway protocol (EGP) is used to exchange routing information *between* autonomous systems.

An *autonomous system* is simply a group of devices under a single entity's control. For example all of AT&T could be considered an autonomous system, but since they are so large, they actually have many autonomous systems within the company (MIS, CERFnet etc.). An autonomous system is like a country. Each country has its own rules, and is responsible for what goes on within its own borders.

Commonly used IGPs include RIP, IGRP, EIGRP and OSPF. Each of these have their pros and cons, but they are all designed for networks under the control of a single entity. The difficulty comes in when we need to share routing information with someone outside of our own control. For example if I have my own ISP called GAD.net, and wanted to peer with AT&T and Quest, then how do I share routing information with them? IGPs are designed with the idea that I basically have a finite network within an autonomous system. AT&T & Quest are clearly not under my control.

EGPs are designed to connect autonomous systems together. Think of an EGP as a common language which different countries (who may all have individual languages) may share information. While each country may continue to run by its own rules and customs, certain standards must be adhered to in order to communicate effectively *between* those countries.

What is BGP?

Border Gateway Protocol (BGP) is the method of choice for exchanging routing information between autonomous systems. In LAN/WAN design we often speak of an OSPF system or an EIGRP autonomous system, but this may not be valid when speaking of autonomous system in relation to BGP.

An autonomous system may contain multiple IGPs, or in fact, no IGPs. BGP really doesn't care, because its job is to exchange information between autonomous systems.

BGP is a *path vector protocol*, which differs a bit from the normal *link state protocols* and *distance vector protocols* we are all used to dealing with.

BGP (specifically BGPv4) is a classless protocol. Since we don't talk about networks per se in classless routing, we define what we are advertising as a *prefix*. Simply put 10.0.0.0/16 is a prefix (the first 16 bits). 192.168.0.0/16 is a prefix as well. If we don't use the term prefix, we could say that 10.0.0.0/16 (smaller than the classful network it divides) is a subnet and 192.168.0.0/16 is a supernet (larger than the classful network it describes). BGP is classless, so it doesn't care about Class limitations at all. Instead we care only about the number of bits used to differentiate the networks from the hosts. BGP calls the network portion we are dealing with the prefix.

BGP advertises, learns and determines the *path* to a *prefix*. That path is comprised of a list of *Autonomous Systems* called the *AS-Path*. Since BGP concerns it self with paths, and not next hops, BGP is considered to be a path vector protocol.

How BGP differs from IGPs

BGP is very different from IGPs you may be used to dealing with. For example while BGP does deal with routes, in reality it really works with *paths*. Since an autonomous system may contain a variety of information that BGP doesn't care about, we essentially figure out how to get to a network by following a *path of autonomous systems.* Whats the difference?

Look at it this way –

Doing a trace route from Exodus to www.goober.net, we see the following:

```
route-server.exodus.net>traceroute www.goober.net
Translating "www.goober.net"...domain server (209.1.221.10) [OK]
Translating "www.goober.net"...domain server (209.1.221.10) [OK]

Type escape sequence to abort.
Tracing the route to www.connectright.com (209.150.146.104)

  1 dcr03-p0-2.sntc02.exodus.net (209.1.169.182) 0 msec 4 msec 0 msec
  2 bbr02-g4-0.sntc02.exodus.net (216.33.154.132) 0 msec 0 msec 0 msec
  3 ibr01-p1-0.paix01.exodus.net (209.185.249.26) 0 msec 4 msec 4 msec
  4 ibr02-g1-1.paix01.exodus.net (206.79.9.246) 4 msec 0 msec 4 msec
  5 globalcrossing-px.exodus.net (206.79.9.2) 0 msec 4 msec 0 msec
  6 so6-0-0-2488M.cr2.PAO2.gblx.net (207.136.163.125) [AS 3549] 4 msec 0 msec 4 msec
  7 pos0-0-2488m.cr1.CHI1.gblx.net (208.49.59.242) [AS 3549] 52 msec 56 msec 52 msec
  8 so0-0-0-622M.ar3.CHI1.gblx.net (208.49.59.214) [AS 3549] 56 msec 56 msec 52 msec
  9 OLM.t3-2-2-3.ar3.CHI1.gblx.net (208.49.33.46) [AS 3549] 56 msec 56 msec 56 msec
 10 www.connectright.com (209.150.146.104) [AS 11443] 56 msec 56 msec 56 msec
```

Note the fact that this traceroute includes something you may not have seen before: an AS number. This is the Autonomous System Number (more on this later).
Looking at the above traceroute, we see that it takes *ten hops* to get to our location. If we look at the AS numbers however, we see that the destination is *two autonomous systems* away. Exodus is an AS, followed by AS# 3549, then AS# 11443.

"So what" you ask? Well one of the interesting features of BGP is that a BGP router may have paths for *every destination network on the Internet*. Remember, BGP is for exchanging routing information between autonomous systems. Because of this, it is used in a rather important role: BGP is the primary routing protocol for the core routers of the Internet.

The Internet is comprised of many, many entities, all of which need to be able to connect to any other connected entity at any time (this is the essence of the Internet). Most entities eventually connect to massive peering points on the Internet called NAPs and MAEs. In order for the paths to all these entities to be knows to all other entities on the Internet, an enormous amount of routing information must be exchanged. Think of every router connected to the Internet needing to know how to get to every other network in the world. Now imagine that this information is constantly changing – the resulting CPU load and network traffic would be unbearable.

Autonomous Systems

As stated earlier, autonomous systems are groups of equipment and networks, which are under the control of a single entity. In order to identify these autonomous systems, they are assigned a unique identifier called the Autonomous System Number (ASN). The ASN is assigned to an entity by ARIN, just like legitimate public IP networks.

Lets look at our traceroute again:

```
route-server.exodus.net>traceroute www.goober.net
Translating "www.goober.net"...domain server (209.1.221.10) [OK]
Translating "www.goober.net"...domain server (209.1.221.10) [OK]

Type escape sequence to abort.
Tracing the route to www.connectright.com (209.150.146.104)

  1 dcr03-p0-2.sntc02.exodus.net (209.1.169.182) 0 msec 4 msec 0 msec
  2 bbr02-g4-0.sntc02.exodus.net (216.33.154.132) 0 msec 0 msec 0 msec
  3 ibr01-p1-0.paix01.exodus.net (209.185.249.26) 0 msec 4 msec 4 msec
  4 ibr02-g1-1.paix01.exodus.net (206.79.9.246) 4 msec 0 msec 4 msec
  5 globalcrossing-px.exodus.net (206.79.9.2) 0 msec 4 msec 0 msec
  6 so6-0-0-2488M.cr2.PAO2.gblx.net (207.136.163.125) [AS 3549] 4 msec 0 msec 4 msec
  7 pos0-0-2488m.cr1.CHI1.gblx.net (208.49.59.242) [AS 3549] 52 msec 56 msec 52 msec
  8 so0-0-0-622M.ar3.CHI1.gblx.net (208.49.59.214) [AS 3549] 56 msec 56 msec 52 msec
  9 OLM.t3-2-2-3.ar3.CHI1.gblx.net (208.49.33.46) [AS 3549] 56 msec 56 msec 56 msec
 10 www.connectright.com (209.150.146.104) [AS 11443] 56 msec 56 msec 56 msec
```

Again there are 10 hops, but only two ASNs between exodus and www.goober.net. When routers need to know the path to every network in the world, it makes more sense to store two four-bit entries (ASNs) vs. ten 32-bit (IP Network) entries. This may seem trivial in this case, but imagine the difference when talking about hundreds of thousands of networks.

Here's a pared down version of the BGP table from the same router:

```
route-server.exodus.net>sho ip bgp 209.150.146.104
BGP routing table entry for 209.150.128.0/19, version 13862665
  701 11443
    209.1.40.47 from 209.1.40.47 (209.1.40.47)
      Origin IGP, localpref 1000, valid, internal, best
```

The entries highlighted in bold show the AS-Path. Only two entries as opposed to the 10 entries needed to store the path based on IP hops.

Full tables (all ASN entries for all known networks in BGP) are often in excess of 80M in size. Imagine how big they'd be if they stored actual network information!

Why is BGP needed or used?

Remember how we related Autonomous Systems to countries? Well politically speaking, companies interacting with each other are a lot like countries communicating – especially where routing is concerned. Different languages, security concerns, fear – these are all obstacles to communication between entities.

BGP is the common method for communicating networking information between Autonomous Systems. Since it is an open protocol (One which is not owned by any one group), everyone can speak it freely.

BGP IS the Internet

Since BGP is the standard which all providers use to exchange routing information on the Internet. In essence, if BGP were to go away tomorrow, the Internet would cease to function. It is the language used in the Internet's core, and is essential for proper flow of data every second of every day.

Terms used in BGP

There are some terms we need to define when talking about BGP.

Speaker – In BGP parlance, a speaker is any router that is running a BGP process.

Peer – Often called a neighbor in other routing protocols, a peer is a BGP speaker with whom we have established communication. Peers must be statically configured in BGP. Compare this with other routing protocols that will often discover neighbors dynamically.

Autonomous System – listed in detail above, an Autonomous System is a group of devices under control of a single entity.

Edge Router – a router (BGP Speaker) who is on the edge of an Autonomous System. Edge routers speak eBGP to other edge routers in other autonomous systems.

Prerequisites for running BGP

In order to use BGP in most circumstances, the following items are required. Note that not all items are required in all situations, but usually they are.

Portable IP Network

If the customer's IP addresses were given to them from one of their providers (usually the case), then there is the issue of whether or not those IP addresses can be seen from another provider. *Portability* is the term used to signify if an IP network may be advertised through a provider other than the owner. The IP network to be advertised must be *portable*.

IP addresses are doled out in large chunks called *ARIN Blocks* from the agency that assigns them (American Registry for Internet Numbers – ARIN). When a provider gives an IP network to a client, of the network is part of a larger block, then that network is likely NOT portable. If the block is not part of a larger block, then it is most likely portable.

Autonomous System Number

If the customer wishes to advertise their portable IP network to the rest of the world, (Our provider would do it for us normally), we must obtain an Autonomous System Number (ASN) from ARIN as well.

The Autonomous system number is a unique label, which identifies the customer's Autonomous System to the rest of the Internet. Autonomous System Numbers are sixteen-bit numbers ranging from 1- 65535 (far too small a range given the Internet's growth!). Furthermore, ASN 64512 through 65535 are reserved for private use (like networks 192.168.0.0 and 10.0.0.0 are for IP).

In order to get an ASN from ARIN, you must prove that you are *dual homed* (connected to more than one provider). There is a small fee to obtain an ASN.

For detailed information on Autonomous System Numbers, see RFC 1930.

Provider willing to peer with you

A company cannot just get an ARIN block and turn up BGP. With whom would they advertise their network? BGP is designed to *peer* with routers of another ASN (in the case of eBGP.

Since BGP is so powerful, and as a result dangerous, providers are very strict about peering. In order for a customer's provider to accept them as a BGP peer, they not only agree to the peering, they often wish to do complete audits of the customer's BGP routers as well.

Decide what size routing tables are needed

There are basically three different ways to receive routing information from your provider.

Default only – The only routing updates we get from our providers are default networks (0.0.0.0). This is often used for failover configurations

Full Tables – The provider sends us every routing table entry they know about. This includes all networks and all AS paths from everywhere in the world. Full tables, as of October 2001 are about 85 Megabytes in size. Because of this serious limitation (how much memory does *your* router have?), partial tables may be a better option for many customers. Full tables are often used when connecting to multiple providers.

Partial Tables – The provider sends us a subset of the full table. The subset of routes can be comprised of networks directly connected to the provider, or perhaps networks that are local to the customers geographic location (connected to the same Point of Presence). Partial tables are often used when connecting to different locations of the same provider at the same time.

What can be done with BGP?

So BGP runs the Internet. That's great for tier-I providers like AT&T and Quest, but how does BGP help customers that just have say a corporate office or a website?

BGP is unique among routing protocols in that it is used to alter someone else's routing tables. More specifically, BGP is used to advertise networks and how to reach them. This information is used to update the rest of the world, and lets everyone else make decisions on how to get to that network.

The real benefit of BGP to the average customer is that of multiple connections to the Internet. With the ability to advertise networks to the Internet at large, if we advertise those networks to multiple providers, then we have redundancy. Moreover, if we have multiple paths to our networks, and we can alter other company's routing tables, then we can use these features to assign a preference of one link over the other. How we advertise these networks affects how other company's routing tables look. Let's look at some examples of how we might configure BGP.

Failover with one provider

Redundancy is a good thing. Customers like the idea that should a primary *anything* fail, there is a secondary *thing* to back it up. If that secondary *thing* comes up to replace the primary *thing* automatically, all the better.

Let's look at one of the simplest BGP setups we can design – that of a single customer, connected to a single provider, with a primary and a failover link to that provider.

In this design we are simply using the secondary link in the event of a failure with the first link. The links are not used at the same time, but rather the secondary is *only* used should the primary link fail.

Most service providers will charge a smaller fee for this second link, often called a *shadow* link. Since the secondary link never has any use, there is no real reason to charge a bandwidth fee on it. Many providers will charge a percentage fee (say 10%) of the primary's bandwidth. This design can be put in place using one router or two (more routers equals more redundancy).

In reality BGP is not needed for this deployment, as the paths to the customer's networks and ASNs do not change – you still get to the customer through their single provider. The global routing tables do not change in the event of a failure; hence BGP is, in effect, superfluous. This can more easily be accomplished with floating static routes or even Cisco Express Forwarding (CEF) given the proper router configuration.

Figure 1 shows a network layout reflecting this design.

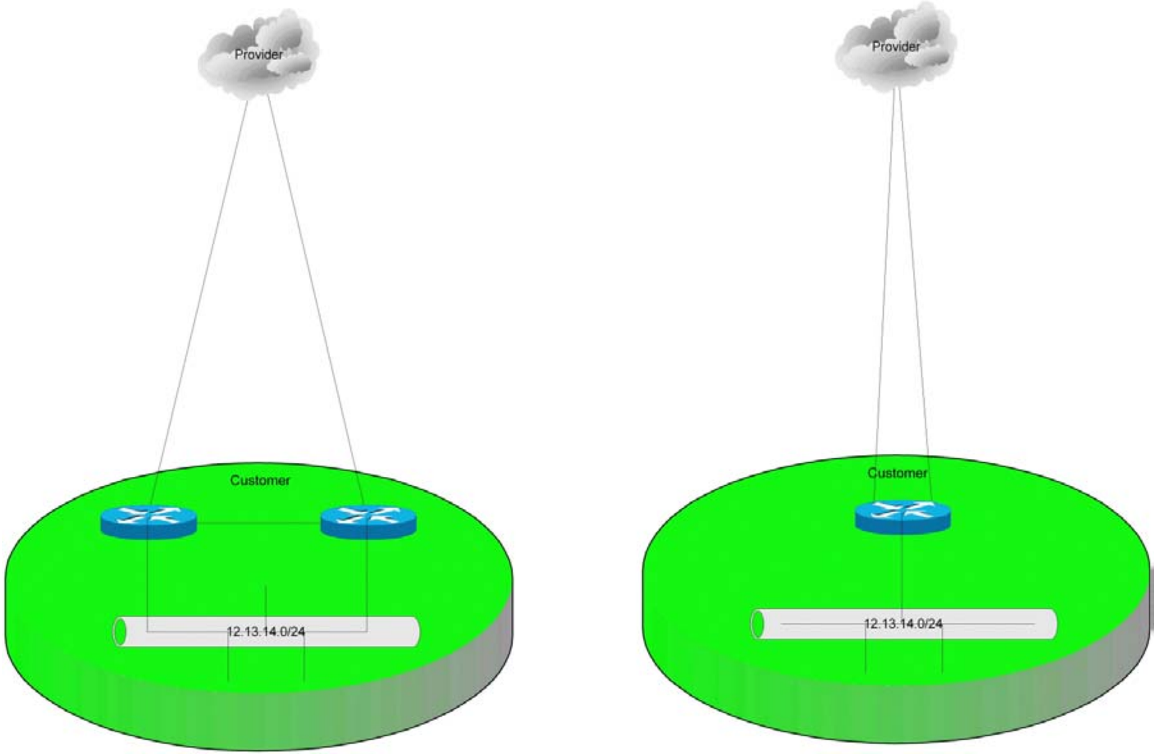


Figure 1

We can easily complicate this "simple" design however, by having our secondary link terminate in another location or even state!

On September 11, 2001, terrorists destroyed the World Trade Towers. What many people may not have known is that the World Trade Center was a major hub for telecommunications in lower Manhattan! Many companies who weren't directly affected by the tragedy itself suffered great telecom outages as a result of complete loss of the World Trade Center.

AT&T's data center on Broad Street in Manhattan was fed by two OC-48 links, one of which came from the World Trade Center. When the Trade Center was destroyed one of their OC-48 was destroyed with it. Since they also had a link that took a completely different path, the data center stayed up, even with the complete and utter destruction of a good portion of lower Manhattan!

While the redundancy of AT&T's data center was probably not accomplished with BGP (though it may have been), the point is that real redundancy and failover is accomplished with thorough planning, including thinking about disasters of such a large scale.

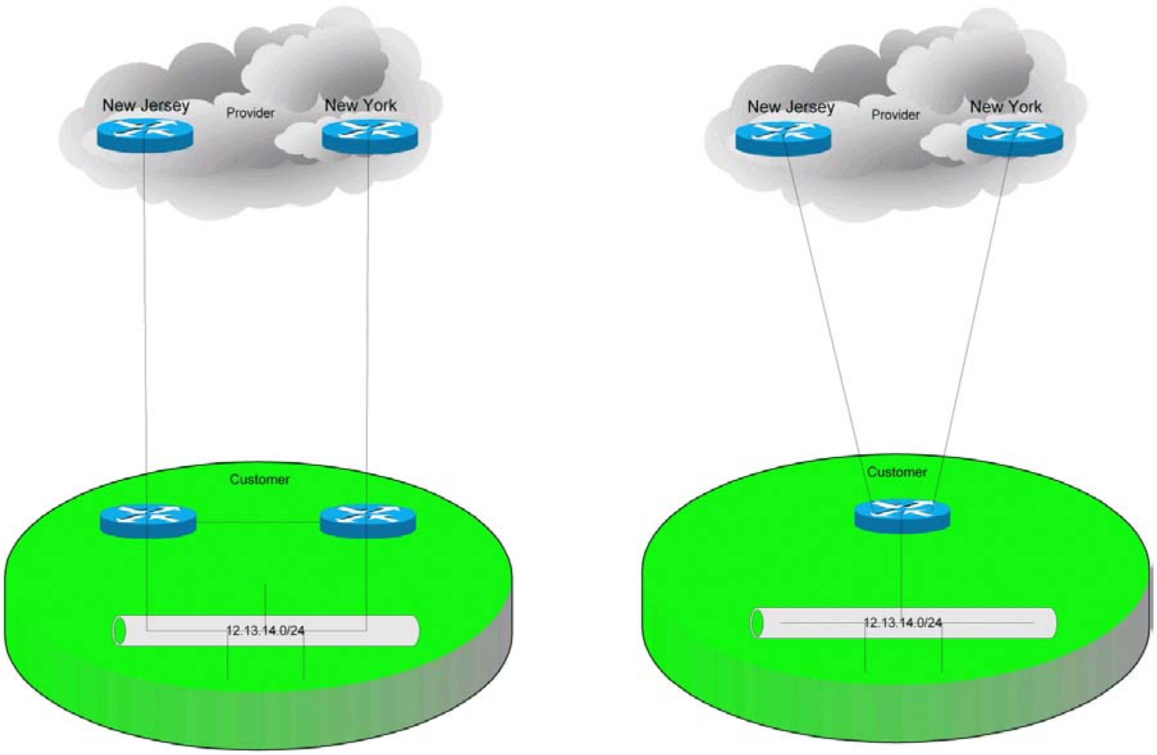


Figure 2

By having a two links that terminated in vastly different locations, AT&T was able to keep their Broad Street data center online even through such a massive outage.

Customers can accomplish the same thing by terminating multiple links with a single provider. Figure 2 shows examples.

In this case BGP would be a good solution, as we need to advertise our network and ASN to different locations within the same provider.

In this example we would not need an ASN from ARIN, as our ASN would not be advertised to the rest of the world, but rather be kept within our provider's AS. We would use a private ASN (given us by our provider) in the range of 64512 through 65535.

We could probably get away with only receiving defaults from the provider, seeing as how the goal is failover from one link to the other.

This design could be accomplished without BGP, but the client would have no control over the routing in the providers cloud. Using BGP, the client can change certain aspects of his network's path should he wish to. For example using BGP the customer can switch which link is the primary at their whim. Using static routes, the provider must do this for them.

Failover with multiple providers

When we talk about redundancy, we like to eliminate single point of failure. On a grand scale, any ISP is a single point of failure. In the early 1980s, MCI had a catastrophic failure during routing maintenance, which only lasted for about six hours. This outage "stopped hundreds of trains on the tracks and disrupted rail traffic throughout Jacksonville, Fla.-based CSX Transportation Inc.'s (CSXT) system" (Sour*ce: http://www.nwfusion.com/news/2000/0508railtrouble.html)*. Another MCI related problem (which was caused by a software upgrade to their frame relay switches) caused many smaller ISP startups to completely fail (Source: http://news.cnet.com/news/0-1004-200-346059.html?tag=mainstry and http://news.cnet.com/news/0,10000,0-1004-200-345841,00.html).

Though these sorts of problems are admittedly rare, the smart customer will not sit idly by waiting for their provider to put them out of business. Customers whose sole source of revenue is their Website should not rely on a single provider, just as they would not rely on a single switch or router in their network infrastructure.

Figure 3 shows the basic idea of two providers with failover links.

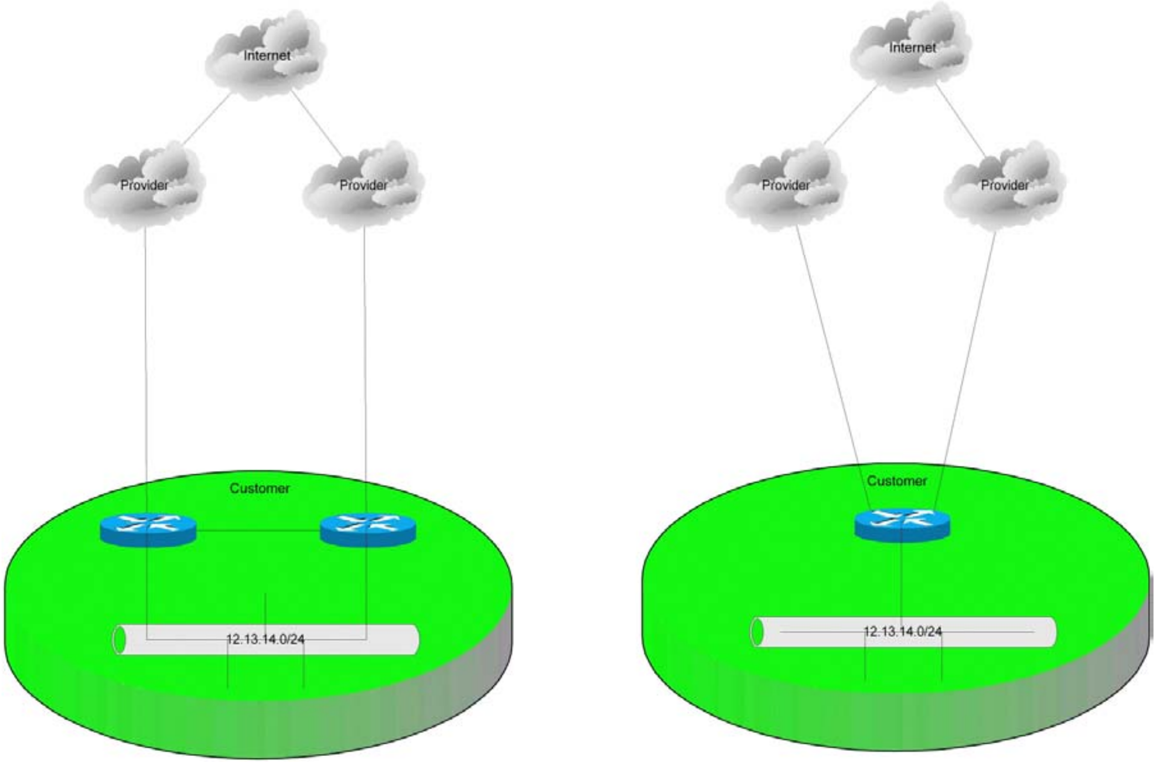


Figure 3

As with failover links using only one provider, only one of the links is "live" and passing data at any given time. The secondary link will only be used in the event of a failure with the first link. Which provider's link is primary is irrelevant to the principle of the design, but in reality one provider will probably be better in some way making them a likely choice. Reasons for choosing one provider over the other may include things like availability, cost, reliability, and even politics.

Even with the addition of another provider, the BGP configuration may be quite simple. Since we are not load balancing (see next section), but are only using one link at a time, we really only need default routes from the providers instead of the normally huge tables associated with BGP. Similarly we will only need to advertise our single IP network. Simply put each provider will get one route (the customer's network) and send one route (a default route).

Load balancing with one provider

Load Balancing, as it relates to BGP is not necessarily what you may think it is.

Traditionally we talk about load balancing in terms of sharing the bandwidth of two links. Bonding two ISDN 64k B channels into one 128k link is a form of load balancing. Bonding two T1s together to get 3Mbps is also a form of load balancing.

We can accomplish the same things with multiple links using BGP. Figure 4 shows the design.

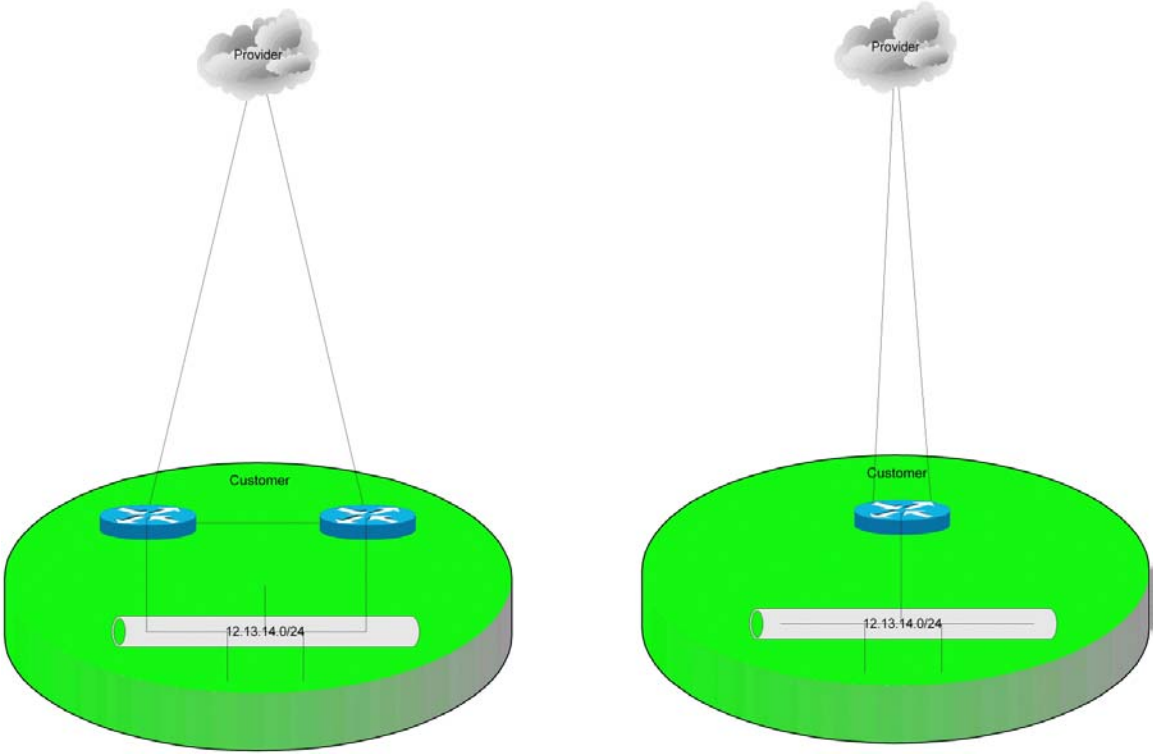


Figure 4

With BGP however, we can do a whole lot more. If we connect to our upstream provider as shown in figure 4, we are pretty limited. If we connect as in figure 5 however, some exciting possibilities arise.

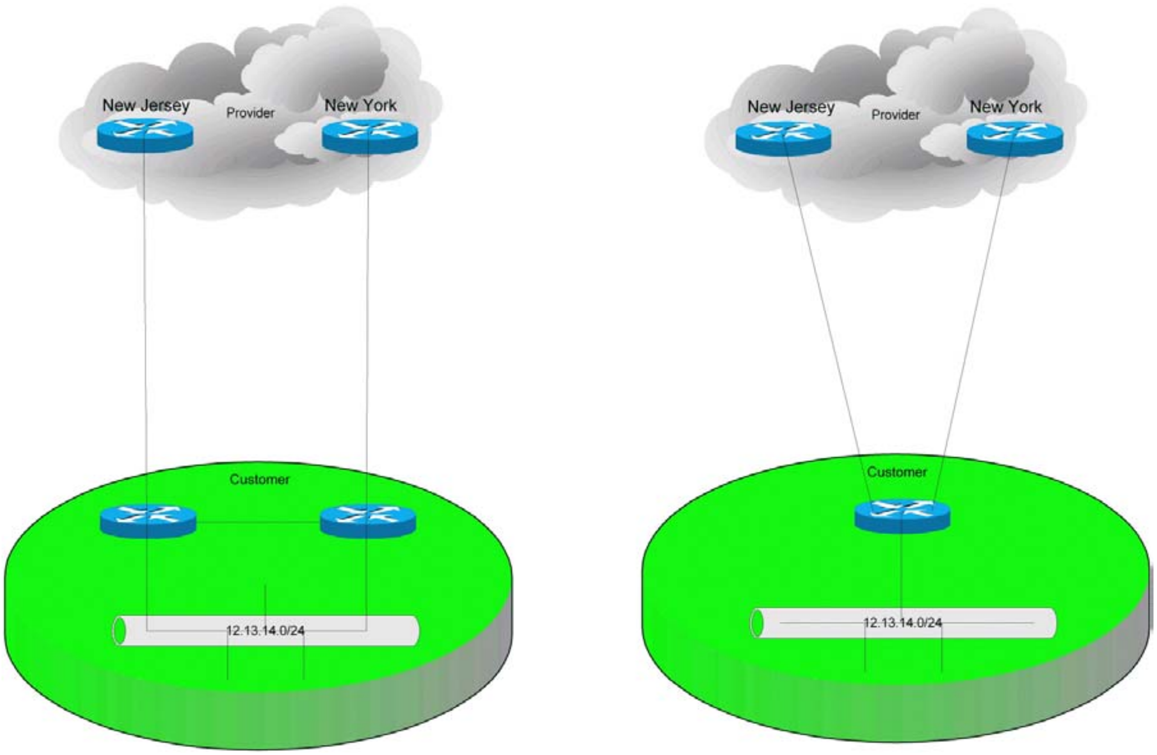


Figure 5

The difference between figures 4 and 5 of course, is that we are connecting to our providers in two geographically distinct locations. These are the same layouts as seen in the “Failover with one provider” section, but in this case we are not doing failover, we are doing load balancing.

The reason figure 5’s design becomes so powerful is that the customer can request *partial tables* from the provider.

Given a large enough provider, if the customer connects to say, New York and New Jersey, the provider can supply partial tables to the customer. Those tables could contain all the information for networks and ASNs that are directly connected to each of those locations. New York’s link would provide only New York related information, and New Jersey would provide only New Jersey information. The router(s) at the customer’s site would then be able to choose the link to New Jersey, should the destination be near New Jersey and so forth. Should the destination not be included in any of the routing updates (California for example), the default route would be used.

Similarly, the provider could supply all information relative to just that provider, and the customer’s router would choose the shortest path (and therefore link) based on this information. Any destination outside of the provider’s realm would follow the default gateway. This would be of benefit if the provider were a tier I provider like AT&T or WorldCom, but would be of little value for a smaller provider like a local ISP.

We could also request full tables in this case, which would allow us to choose the best path out of the provider as well. This might be a good choice with large providers like AT&T who have many connections out of their own AS. Of course full tables requires a lot of memory and horsepower in the edge routers.

Load balancing with multiple providers

Load balancing with multiple providers is just like load balancing with one provider, except there is more redundancy and a better breadth of information in the routing tables.

In figure 6 we see the layout for load balancing to multiple providers.

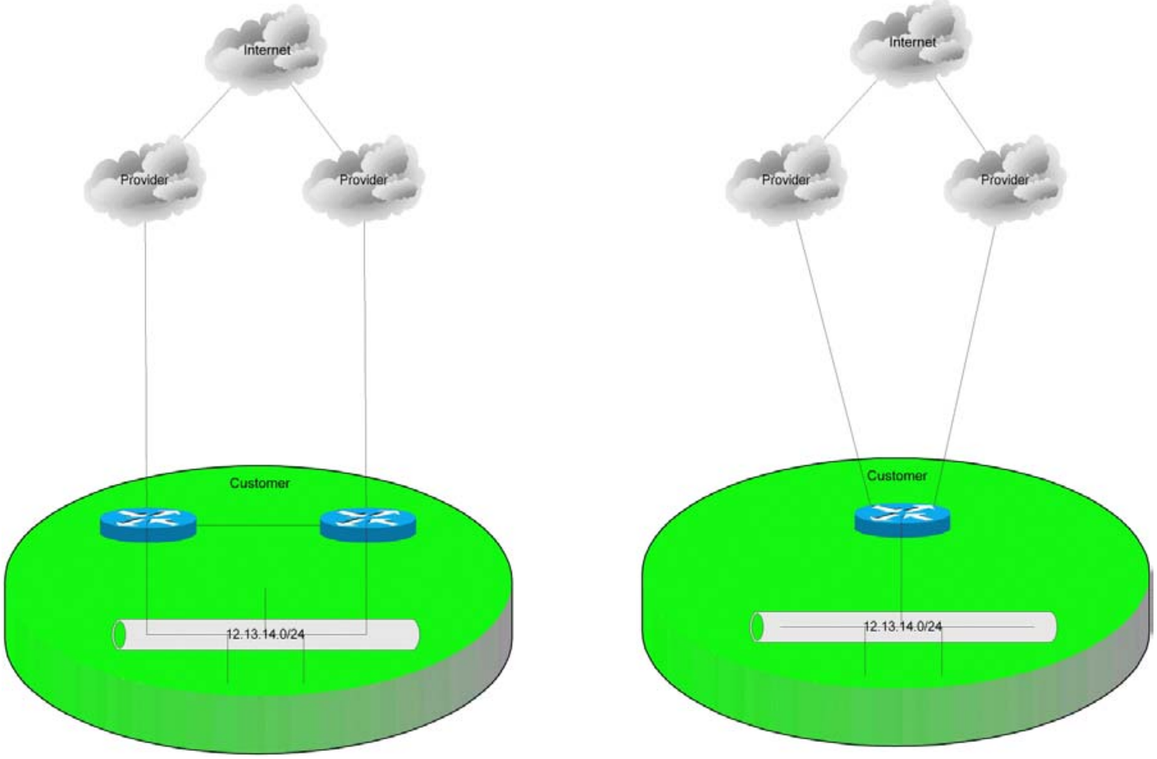


Figure 6

Just as in load balancing with one provider, we have the option of default only, partial tables or full tables. The real benefit to load balancing with multiple providers however, is when we run full tables. Since each provider should know how to get to every ASN in the world, letting the customer's routers choose from each provider's pool of knowledge (choosing the best from each) makes for a very powerful setup.

Having multiple paths in and out, and not relying solely on one provider makes this design very fault tolerant. Using a separate router for every provider on the customer's end makes it even more so. This is reflected in the left image, versus only one router for both providers on the right. The single router on the right side is a single point of failure.

Global load balancing with multiple providers

While BGP can be used for relatively simple designs as seen above, it is also very configurable. We can configure ASNs within ASNs (confederations), we can configure failover between providers at one location while failing the entire location over to another location elsewhere in the world and so on (Figure 7 shows a sample of both).

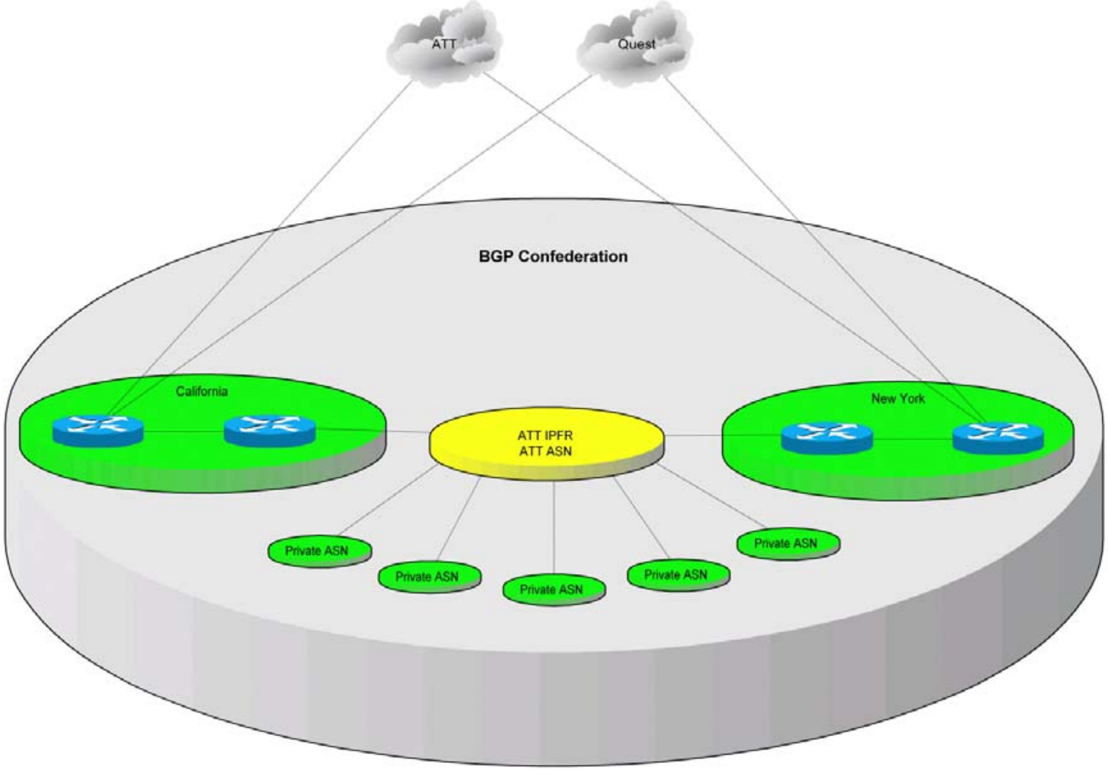


Figure 7

In short BGP can do a whole lot more that I can cover in this document. Suffice to say, if you need to do something along the lines of figure 7, it's either time to find a high-level BGP guy, or time to break out the Halabi book (see the Bibliography).

In short, Figure 7 shows multiple BGP ASN within another AS using a BGP feature called a *confederation*. Each location (California and New York) will get full tables from both AT&T and Quest. They will share those tables with each other through an iBGP link, while also communicating with the ASNs within the confederation. Each location connected to the IPFR cloud will learn a default from the master ASN. Needless to say, this is not an "Introduction to BGP" scenario! It does show the power of BGP, so that's why it's included.

Transit AS

A transit Autonomous system, or Transit AS, is an Autonomous system where information from one connected AS is advertised *through* the AS to another connected AS.

Normally when a customer connects to two providers, they only advertise their own routes to each provider. The routes learned from one provider are not forwarded to the other. As a result, packets sources in one of the providers may not use the customers AS (and therefore their networks) to reach a destination in the other providers AS.

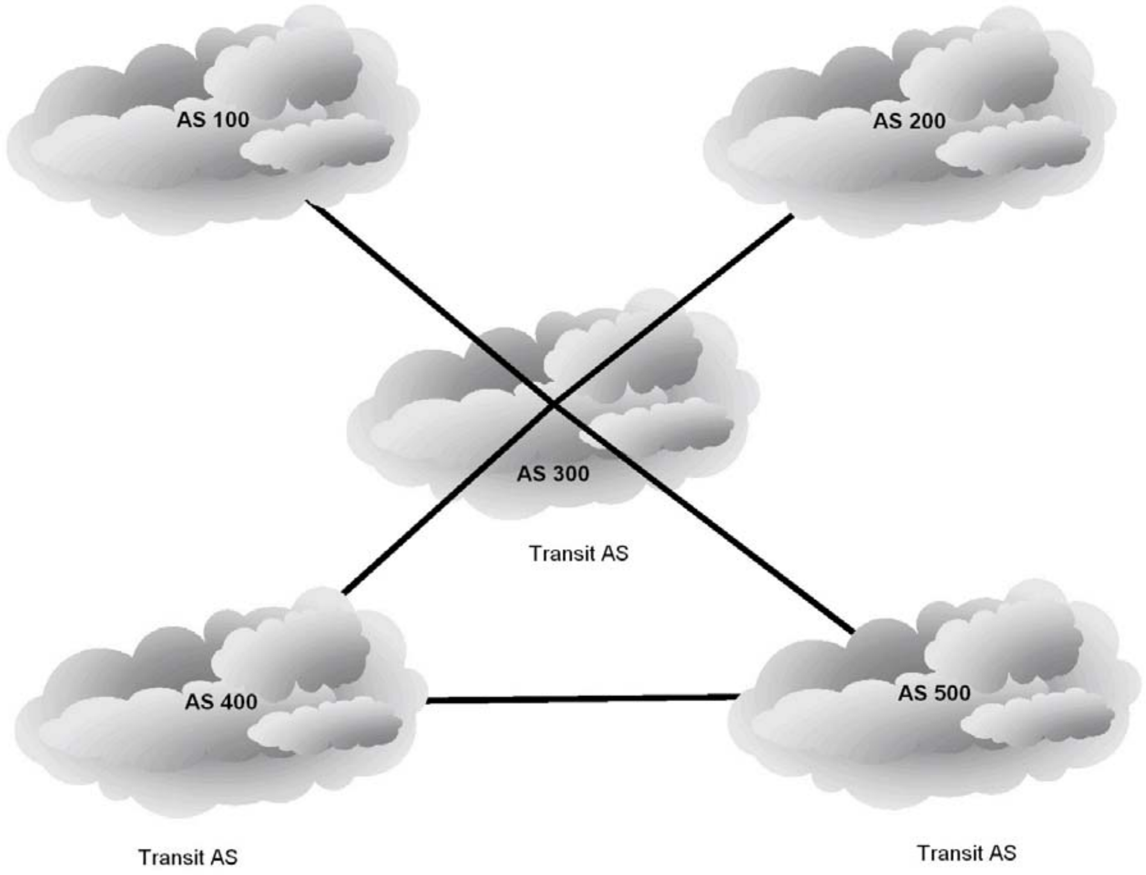


Figure 8

In figure 8, only ASNs 300, 400 & 500 can be transit autonomous systems. Be careful to note that they do not *have* to be transit autonomous systems, but rather they *can* be.

Figure 9 shows how advertisements flow between a customer and his two providers. The example on the left shows how advertisements (and therefore traffic) may only pass between the customer and each individual provider. This is an example of a non-transitive AS.

The example on the right of Figure 9 shows how each provider's advertisements are not only accepted by the customer, they are also forwarded to the other provider. This enables traffic from one provider to cross the customer's network, making them become a *transit AS*.

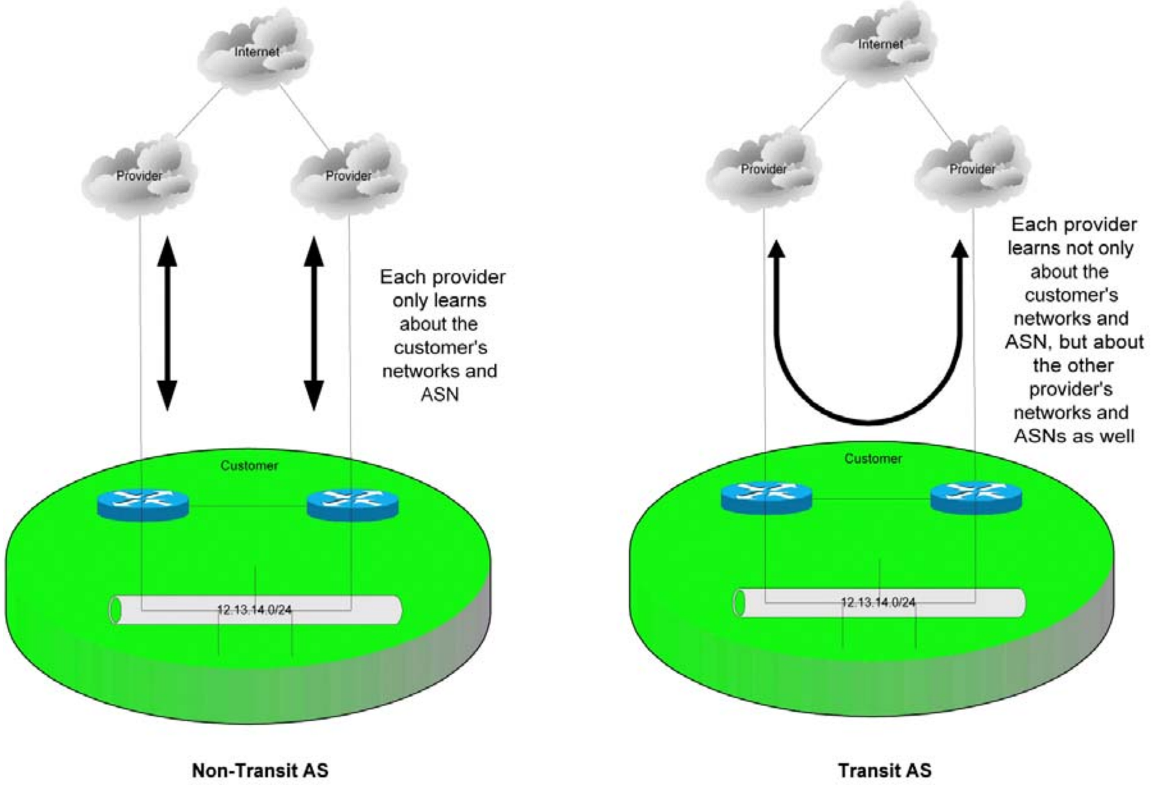

Internet
Provider
Provider
Each provider only learns about the customer's networks and ASN
Customer
12.13.14.0/24
Non-Transit AS
Internet
Provider
Provider
Each provider learns not only about the customer's networks and ASN, but about the other provider's networks and ASNs as well
Customer
12.13.14.0/24
Transit AS

Figure 9

Generally customers in this type of design do not want to become a transit AS. Think of the load on their network should this be allowed.

Generally being a transit AS is only desirable for an ISP, or if you want to peer with other customers of the same provider for backup purposes.

It's not just for IP anymore!

One of the more powerful features of BGP is the fact that it can be used for protocols other than IP. While the average customer may never see this, it is a very valuable feature.

The internal workings of AT&T's IP Enabled Frame Relay rely on this feature, known as Multi-Protocol BGP. Simply put, 64 bit addresses are used (IP only uses 32 bit), and paths to them are advertised using Multi-protocol BGP.

How to destroy the world

OK maybe not the physical world, but lets look at the dark side for a minute. Since an Internet connected BGP speaker advertises its networks and all other BGP speakers on the Internet figure out paths to that network, what would happen if we advertised the *wrong* network?

Say I'm a customer and I'm connected to AT&T's backbone. AT&T has given me a /24 block – 12.10.10.0/24. I'm also dual homed, connecting with UUNET. I, of course, advertise my 12.10.10.0/24 to both providers, so that everyone in the world can find the shortest path to me using either provider. Great!

Now you come in a little hung over one day and need to make a change to my BGP config for whatever reason, only through your Guinness-induced haze you fat-finger the config. Instead of advertising 12.10.10.0/24, you advertise 12.0.0.0/8 (hey – there was a lot of Guinness involved). Now instead of telling the rest of the world that I've got 256 addresses, I'm telling the world that I have 16 *million* addresses! Not only that, but these 16 million addresses happen to be owned by AT&T. Not only have we advertised 16 million addresses, we've told the world that to get to AT&T, come see us! Needless to say, AT&T would be more than a little miffed.

Even more likely is the possibility that you let BGP do auto-summarization, which will, in effect, advertise the classful network of 12.0.0.0/8 even if you only have 12.10.10.0/24. Be very careful an plan ahead when doing any configurations with BGP!

Why you probably can't destroy the world

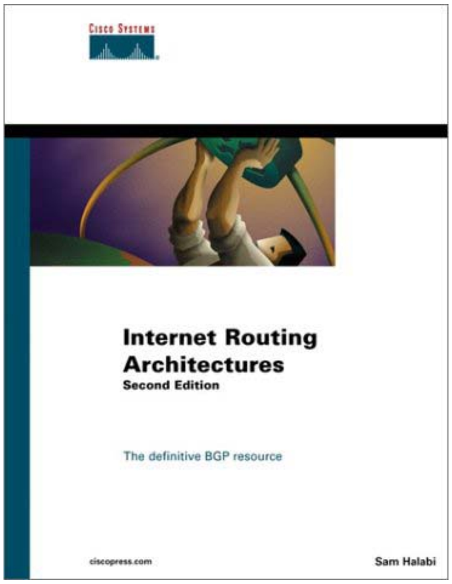
The above example is admittedly a little far-fetched, however mistakes DO happen. What prevents those mistakes from "destroying the world", is the fact that most, if not all providers will not allow you to advertise prefixes that you don't own. This is generally done through filtering of one sort or another. In fact when configuring BGP, it is a good practice to put these filters in your own routers, just to protect yourself from your own stupidity (hey – we've all been there!).

It's also worth noting that in the above example, AT&T would already be advertising their own network, so your advertisement to them would not have any effect on them. UUNET however would now have their normal path to AT&T, and the new one provided by you. You would probably not get the entire world coming to you for AT&T, but you would, no doubt, get a good chunk of it.

There have been instances of mammoth BGP mistakes on the Internet. I've seen entire pacific islands redirected to a small company in California for example. Mistakes are more dangerous when they happen at first tier providers like AT&T and UUNET. Any provider which has direct connections to the MAEs and NAPs on the 'net are capable of causing massive problems with a few keystrokes. This is one of the main reasons that it can be so difficult to get one of these providers to agree to peering with you in the first place.

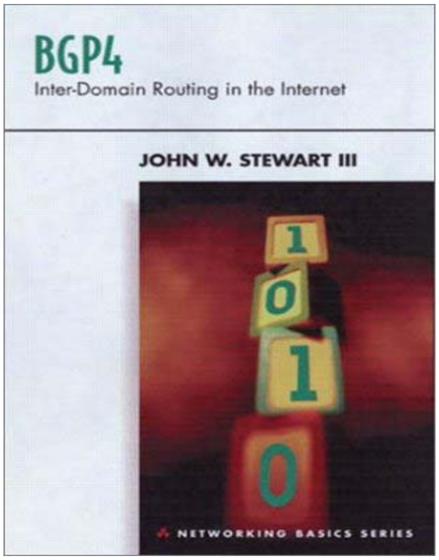
Bibliography

Internet Routing Architectures
Second Edition
Sam Halabi with Danny McPherson
Copyright 2000
Cisco Press
201 West 103rd Street
Indianapolis, IN 46290 USA
www.ciscopress.com
ISBN 1-57870-233-X



This book, often referred to as "The Halabi Book" is the definitive BGP resource for anyone wishing to learn the nitty-gritty details of BGP. The book is geared towards Cisco routers which, is a major plus if that's the environment you need to configure. If you want to know about BGP and how to design BGP networks, then buy this book.

BGP4
Inter-Domain Routing in the Internet
John W. Stewart III
Copyright 1999
Addison-Wesley
Boston USA
ISBN 0-201-37951-1



This book is non-Cisco-centric, and covers the workings of BGP as the open protocol it is. An excellent resource for learning how BGP works, it is less useful in the field than the Halabi book.

Index