Last significant update: January 2009; last update to simplify and clarify confusing points: September 2024.

CHAPTER 3
Interdomain Internet Routing

Our goal is to explain how routing between different administrative domains works in the Internet. We discuss how Internet Service Providers (ISPs) exchange routing information, packets, and (above all) money between each other, and how the way in which they buy service from and sell service to each other and their customers influences routing. We discuss the salient features of the Border Gateway Protocol, Version 4 (BGP4, which we will refer to simply as BGP), the current *interdomain routing protocol* in the Internet. Finally, we discuss a few interesting failures and shortcomings of the routing system. These notes focus only on the essential elements of interdomain routing, sometimes sacrificing implementation or operational detail for clarity and generality.

■ 3.1 Autonomous Systems

Network attachment points for Internet hosts are identified using IP addresses, which are 32-bit (in IPv4) or 128-bit (in IPv6) numbers. The fundamental reason for the scalability of the Internet's network layer is its use of *topological addressing*. Unlike an Ethernet or Wi-Fi MAC address that is location-independent and always identifies a host's network interface independent of where it is connected in the network, an IP address of a connected network interface depends on its location in the network topology. Topological addressing allows routes to Internet destinations to be *aggregated* in the forwarding tables of the routers (as a simple example of aggregation, any IP address of the form 18.* in the Internet is in MIT's network, so external routers need only maintain a routing entry for 18.* to correctly forward packets to MIT's network), and allows routes to be *summarized* and exchanged by the routers participating in the Internet's routing protocols. In the absence of aggressive aggregation, there would be no hope for scaling the routing system to hundreds of millions of hosts connected over tens of millions of networks located in tens of thousands of ISPs and organizations.

Routers use *address prefixes* to identify a contiguous range of IP addresses in their routing messages. For example, an address prefix of the form 18.31.* identifies the IP address range 18.31.0.0 to 18.31.255.255, a range whose size is 2^{16}. Each routing message involves a router telling a neighboring router information of the form "To get to address prefix P, you

can use the link on which you heard me tell you this,"[1] together with information about the route (the information depends on the routing protocol and could include the number of hops, cost of the route, other ISPs on the path, etc.).

A highly idealized view of the Internet is shown in Figure 3-1, where end-hosts connect to routers, which connect to other routers to form a nice connected graph of "peer" routers that cooperate nicely using routing protocols that exchange "minimium-cost" or equivalent information to provide global connectivity. The same view posits that the graph induced by the routers and their links has a large amount of redundancy and the Internet's routing algorithms are designed to rapidly detect faults and problems in the routing substrate and find ways (paths) to avoid them. In addition to routing around failures, we might expect clever routing protocols to perform load-sensitive routing to move traffic away from congested paths on to less-loaded paths.

Unfortunately, this ideal is quite misleading as far as the wide-area Internet is concerned. The real story of the Internet routing infrastructure is that Internet service is provided by a large number of commercial enterprises, generally in competition with each other. Cooperation, required for global connectivity, is generally at odds with the need to be a profitable commercial enterprise, which often occurs at the expense of one's competitors—the same people with whom one needs to cooperate. How this *competitive cooperation* ("co-opetition") is achieved in practice provides an interesting study of how technical research can be shaped and challenged by commercial realities.

Figure 3-2 depicts a more realistic view of the Internet infrastructure. Here, ISPs cooperate to provide global connectivity for their respective customer networks, enabling customer networks to be reached from elsewhere on the Internet and providing a way for customer networks to deliver packets to other networks. An important point is that ISPs aren't all equal; they come in a variety of sizes and internal structure. Some are bigger and more connected to other networks than others. A simplified, but useful, model is that they come in three varieties, "medium," "large," and "huge", and these colloquial descriptions have names given to them. *Tier-3 ISPs* typically have dozens of localized (in geography) customers (typically, companies), *Tier-2 ISPs* generally have regional scope (e.g., state-wide, region-wide, or small-country-wide), while *Tier-1 ISPs* have global scope in the sense that their routing tables actually have explicit routes to all currently reachable Internet prefixes (i.e., they have *no default routes*). There are 14 Tier-1 ISPs as of 2024[2].

The previous paragraph used the term "route", which we define as a mapping from an IP prefix (a range of addresses) to a link, with the property that packets for any destination within that prefix may be sent along the corresponding link. A router adds such entries to its routing table after selecting the best way to reach each prefix from among route advertisements sent by its neighboring routers.

Routers exchange route advertisements with each other using a routing protocol. The current wide-area routing protocol in the Internet, which operates between routers at the boundary between ISPs, is *BGP* (Border Gateway Protocol, Version 4) [20, 19]. More precisely, the wide-area routing architecture is divided into *autonomous systems* (ASes) that exchange reachability information. An AS is owned and administered by a single com-

[1] It turns out that this simple description is not always true, notably in the iBGP scheme discussed later, but it's true most of the time.

[2] `https://en.wikipedia.org/wiki/Tier_1_network`

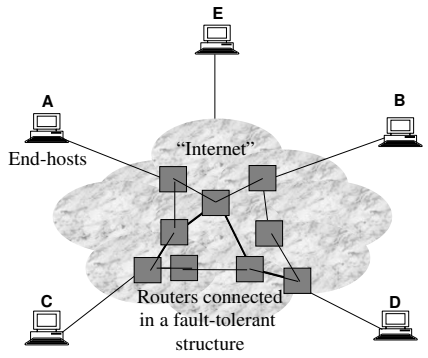


Figure 3-1: A misleading view of the Internet routing system.

mercial entity. It implements policies to decide how to send packets from it to the rest of the Internet and how to export its routes to other ASes to control, to some degree, how packets should arrive into it. The routes it exports are, in general, its own, those of its customers, and other routes it may have learned from other ASes, but it has complete control over what routes it exports to any given other AS. Until 2007, each AS in the Internet was identified by a unique 16-bit number, but since 2007 (due to potential exhaustion of the bit-space), the Internet Assigned Numbers Authority (IANA) has been allocating 32-bit AS identifiers to entitites.

A different routing protocol operates within each AS. These routing protocols are called *Interior Gateway Protocols* (IGPs), and include protocols like the Routing Information Protocol (RIP) [10], Open Shortest Paths First (OSPF) [16], Intermediate System-Intermediate System (IS-IS) [17], and Enhanced Interior Gateway Routing Protocol (EIGRP). By contrast, BGP is an interdomain routing protocol. Operationally, a key difference between BGP and IGPs is that the former is concerned with exchanging reachability information between ASes in a scalable manner while allowing each AS to implement autonomous *routing policies*, whereas the latter are used within an AS to exchange routes for address prefixes of that AS and optimize path cost metrics such as load, latency, or the number of hops. In general, IGPs don't scale as well as BGP does with respect to the number of participants involved.

The rest of this chapter is in two parts: first, we will look at inter-AS relationships (transit and peering); then, we will study some salient features of BGP. We won't talk about IGPs like RIP and OSPF; to learn more about them, read a standard networking textbook (e.g., Peterson & Davie or Kurose & Ross).

■ 3.2 Inter-AS Relationships: Transit and Peering

The Internet is made up of many different types of ASes, from universities and corporations to regional ISPs to nation-wide ISPs. Smaller ASes (e.g., universities, corporations,

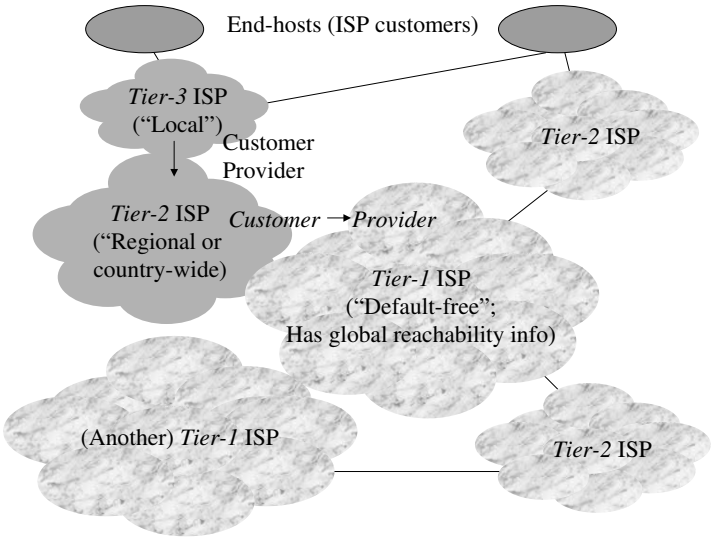


Figure 3-2: A more accurate picture of the wide-area Internet routing system, with various types of ISPs defined by their respective reach. *Tier-1* ISPs have "default-free" routing tables (i.e., they don't have any default routes), and have global reachability information. There are 14 of these as of September 2024).

etc.) typically purchase Internet connectivity from ISPs. Smaller regional ISPs, in turn, purchase connectivity from larger ISPs with large "backbone" networks. Most companies and universities in the world, except for a few that were early to the Internet, don't have their own AS identifiers. Their IP addresses are assigned by their ISPs, who perform routing on their behalf.

Consider the picture shown in Figure 3-3. It shows an ISP, X, directly connected to a *provider* from whom it buys Internet service and a few *customers* to whom it sells Internet service. In addition, the figure shows two other ISPs with whom it is directly connected, exchanging routing information with them via BGP.

The different types of ASes lead to different types of business relationships between them, which in turn translate to different policies for exchanging and selecting routes. There are two prevalent forms of AS-AS interconnection today. The first form is *provider-customer transit* (aka "transit"), wherein one ISP (the "provider" P in Figure 3-3) provides access to all (or most) destinations in its routing tables. Transit is meaningful in an inter-AS relationship where financial settlement is involved; the provider charges its customer for Internet access in return for forwarding packets on behalf of the customer to destinations elsewhere and delivering packets from elsewhere to the customer. Another example of a transit relationship in Figure 3-3 is between X and its customers (the C_is).

The second prevalent form of inter-AS interconnection is called *peering*. Here, two ASes (typically ISPs) provide mutual access to a subset of each other's routing tables. The subset of interest here is their own transit customers (and the ISPs own internal addresses). Like transit, peering is a business deal, but it may not involve financial settlement. While paid peering is common in some parts of the world, in many cases they are reciprocal agreements without a financial settlement. As long as the traffic ratio between the concerned

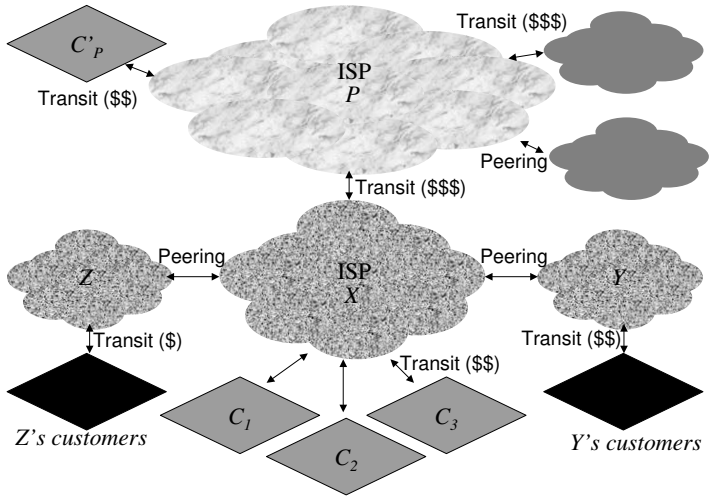


Figure 3-3: Common inter-AS relationships: transit and peering.

ASs is not highly asymmetric (e.g., less than 4:1 is a commonly believed and quoted ratio), there's usually no financial settlement. Peering deals are usually confidential and protected with non-disclosure agreements.

■ 3.2.1 Peering v. Transit

A key point to note about peering relationships is that they are often between business competitors. There are two main reasons for peering relationships:

1. **Tier-1 peering:** An Internet that had only provider-customer transit relationships would require a single (large) Tier-1 at the root, because cycles in the directed graph of ASes don't make sense (a cycle would require that the money paid by each AS to its provider would flow from an AS back to itself). That single Tier-1 would in effect be a monopoly because it would control all Internet traffic (before the commercialization of the Internet, the NSFNET was in fact such a backbone). Commercial competition has led to the creation of a handful of large Tier-1 ISPs (nine today, up from five a few years ago). In effect, these ISPs act as a sort of cartel because an ISP has to be of a certain size and control a reasonable chunk of traffic to be able to enter into a peering relationship with any other Tier-1 provider. Peering between Tier-1 ISPs ensures that they have explicit default-free routes to all Internet destinations (prefixes).

2. **Saving money:** Peering isn't restricted to large Tier-1's, though. If a non-trivial fraction of the packets emanating from a an AS (regardless of its size) or its customers is destined for another AS or its customers, then each AS has an incentive to avoid paying transit costs to its respective providers. Of course, the best thing for each AS to do would be to wean away the other's customers, but that may be hard to achieve. The next best thing, which would be in their mutual interest, would be to avoid paying transit costs to *their* respective providers, but instead set up a transit-free link

between each other to forward packets for their direct customers. In addition, this approach has the advantage that this more direct path would lead to better end-to-end performance (in terms of latency, packet loss rate, and throughput) for their customers.

Balancing these potential benefits are some forces against peering. Transit relationships generate revenue; settlement-free peering contracts don't. Peering relationships typically need to be renegotiated from time to time and asymmetric traffic ratios must be handled to mutual satisfaction. Above all, these relationships are often between competitors vying for the same customer base.

In the discussion so far, we have implicitly used an important property of current inter-domain routing: *A route advertisement from B to A for a destination prefix is an agreement by B that it will forward packets sent via A destined for any destination in the prefix.* This implicit agreement implies that one way to think about Internet economics is to *view ISPs as charging customers for entries in their routing tables*. Of course, the data rate of the interconnection is also crucial and is the major determinant of an ISP's pricing policy.

A word on pricing is in order. ISPs charge for Internet access in one of two principal ways. The first is a fixed price for a certain speed of access (fixed pricing is a common way to charge for home or small business access in the US). The second measures traffic and charges according to the amount of bandwidth used. A common approach is to measure usage in 5-minute windows over the duration of the pricing period (typically one month). These 5-minute averages are samples that form a distrubution. The ISP then calculates the 95th (or 90th) percentile of this distribution and charges an amount that depends on this value, using the terms set in the contract. ISPs also offer discounts for outages experienced; some ISPs now specify delay guarantees (through their network) in the contract for certain kinds of traffic. It's unclear to what extent customer networks can and do go about verifying that providers comply with their contractual terms.

The take-away lesson here is that providers make more money from a customer if the amount of traffic sent on behalf of the customer increases. This simple observation is the basis for a large number of complex routing policies that one sees in practice. In the rest of this section, we describe some common examples of routing policy.

■ 3.2.2 Exporting Routes to Make or Save Money

Each AS (ISP) needs to make decisions on which routes to export to its neighboring ASes using BGP. The reason why export policies are important is that no ISP wants to act as transit for packets that it isn't somehow making money on. Because packets flow in the opposite direction to the best route advertisement for any destination, an AS should advertise routes to neighboring ASes with care.

Transit customer routes. To an ISP, its customer routes are the most important because that helps all potential senders in the Internet reach its customers. Thus it's in the ISP's best interest to advertise routes to its customers to all neighboring ASes. The more traffic that an ISP carries on behalf of a customer, the "fatter" the pipe that the customer would need, implying higher revenue for the ISP. In addition, if a destination prefix were advertised from multiple neighboring ASes, an AS should prefer the advertisement made from a customer over all other choices (in particular, over peering or transit provider links).

Transit provider routes. Does an ISP want to provide *transit* to the routes exported by its provider to it? Most likely not, because the ISP isn't making any money on providing such transit facilities. An example of this situation is shown in Figure 3-3, where C'_P is a customer of P, and P has exported a route to C'_P to X. It isn't in X's interest to advertise this route to everyone, e.g., to other ISPs with whom X has a peering relationship. An important exception to this, of course, is X's transit customers who are paying X for service—the service X provides its customers C_i's is that they can reach any location on the Internet via X, so it makes sense for X to export as many routes to X as possible.

Peer routes. It usually makes sense for an AS to export only selected routes from its routing tables to other peering ASes. It makes sense to export routes to all its transit customers. It also makes sense to export routes to addresses within an AS. It does not make sense, however, to export an AS's transit provider routes to other peering ASes, because that may cause a peering AS to use the advertising AS to reach a destination advertised via a transit provider. Doing so would expend the AS's resources but not lead to revenue.

The same situation applies to routes learned from other peering relationships. Consider AS Z in Figure 3-3, with its own transit customers. It doesn't make sense for X to advertise routes to Z's customers to another peering AS (Y), because X doesn't make any money on Y using X to get packets to Z's customers!

These arguments show that most ISPs end up providing *selective transit*: typically, full transit capabilities for their own transit customers in both directions, some transit (between mutual customers) in a peering relationship, and transit only for one's transit customers (and ISP-internal addresses) to one's providers.

The discussion so far may have given the impression that BGP is the only way in which to exchange reachability information between an ISP and its customers or between two ASes. That is not true in practice, though; a large fraction of end-customers (typically customers who don't provide large amounts of further transit and/or aren't ISPs) don't run BGP sessions with their providers. The reason is that BGP is complicated to configure, administer, and manage, and isn't useful if the set of addresses in the customer's network doesn't often change. These customers interact with their providers via *static routes*. These routes are usually manually configured. Of course, information about customer address blocks will in general be exchanged by a provider using BGP to other ASes (ISPs) to achieve global reachability to the customer's network.

For the several thousands of networks that run BGP, deciding which routes to export is an important policy decision. Such decisions are expressed using *route filters*, which are rules that decide which routes an AS's routers should filter to routers of neighboring ASes.

■ 3.2.3 Importing Routes to Make or Save Money

In addition to deciding how to filter routes while exporting them, when a router hears many possible routes to a destination network, it needs to decide which route to import into its forwarding tables. The problem boils down to *ranking* routes to each destination prefix.

Deciding which BGP routes to import is an involved process and requires a consideration of several attributes of the advertised routes. For the time being, we consider only one of the many things that an AS needs to consider, but it's the most important concern: *who advertised the route*?

Typically, when an AS's border router[3] (e.g., X in Figure 3-3) hears advertisements about its transit customers from other ASes (e.g., because the customer is multi-homed, i.e., has multiple distinct ISPs), it needs to ensure that packets to the customer do not traverse additional ASes unnecessarily. The main reason is that an AS doesn't want to spend money paying its providers for traffic destined to its direct customer, and wants to increase its value as perceived by the customer. This requirement means that customer routes are prioritized over routes to the same network advertised by providers or peers. Second, peer routes are likely more preferable to provider routes, because the purpose of peering was to exchange reachability information about mutual transit customers. These two observations imply that typically routes are imported in the following priority order:

customer > **peer** > **provider**

This rule (and many others like it) can be implemented in BGP using a special attribute that's locally maintained by routers in an AS, called the LOCAL PREF attribute. The first rule in route selection with BGP is to rank routes according to the LOCAL PREF attribute and pick the one with the highest value. Only when this attribute is *not* set for a route that other attributes of a route are even considered in the ranking procedure.

That said, in practice most routes are not selected using the LOCAL PREF attribute; other attributes like the length of the AS path tend to be quite common. We discuss these other route attributes and the details of the BGP route selection process, also called the *decision process*, when we discuss the mechanics of BGP in Section 3.3.

■ 3.2.4 Routing Policy = Ranking + Filtering

Network operators express a wide range of routing policies, but to first approximation, most of them can be boiled down to *ranking decisions* and *export filters*.

■ 3.3 BGP

We now turn to how reachability information is exchanged using BGP, and see how routing policies like the ones explained in the previous section can be expressed and realized. We start with a discussion of the main design goals in BGP and summarize the protocol. Most of the complexity in wide-area routing is not in the protocol, but in how BGP routers are configured to implement policy, and in how routes learned from other ASes are disseminated within an AS.

■ 3.3.1 Design Goals

In the old NSFNET, the backbone routers exchanged routing information over a tree topology, using a routing protocol called the Exterior Gateway Protocol (EGP). Because the backbone routing information was exchanged over a tree, the routing protocol was simple. The evolution of the Internet from a singly administered backbone to its current commercial structure made the NSFNET EGP obsolete and required a more sophisticated protocol.

The design of BGP was motivated by three important needs:

[3] A router running BGP connected directly to another AS.

1. **Scalability.** The division of the Internet into ASes under independent administration was done while the backbone of the then Internet was under the administration of the NSFNet. When the NSFNet was "turned off" in the early 1990s and the Internet routing infrastructure opened up to competition in the US, a number of ISPs providing different sizes sprang up. The growth in the number of networks (and hosts) has continued to date. To support this growth, routers must be able to handle an increasing number of prefixes, BGP must ensure that the amount of advertisement traffic scales well with "churn" in the network (parts of the network going down and coming up), and BGP must converge to correct loop-free paths within a reasonable amount of time after any change. Achieving these goals is not easy.

2. **Policy.** The ability for each AS to implement and enforce various forms of routing policy was an important design goal. One of the consequences of this was the development of the BGP attribute structure for route announcements, allowing route filtering, and allowing each AS to rank its available routes arbitrarily.

3. **Cooperation under competitive circumstances.** BGP was designed in large part to handle the transition from the NSFNet to a situation where the "backbone" Internet infrastructure would no longer be run by a single administrative entity. This structure implies that the routing protocol should allow ASes to make purely local decisions on how to route packets, from among any set of choices. Moreover, BGP was designed to allow each AS to keep its ranking and filtering policies confidential from other ASes.

But what about security? Ensuring the authenticity and integrity of messages was understood to be a good goal, but there was a pressing need to get a reasonable routing system in place before the security story was fully worked out. Efforts to secure BGP, notably S-BGP [14], have been worked out and involve external registries and infrastructure to maintain mappings between prefixes and the ASes that own them, as well as the public keys for ASes. These approaches have not been deployed in the Internet for a variety of reasons, including the fact that existing routing registries tend to have a number of errors and omissions (so people can't trust them a great deal).

Misconfigurations and malice cause connectivity outages from time to time because routing to various destinations gets fouled up. The next section gives some examples of past routing problems that have made the news; those examples illustrate the adage that complex systems fail for complex reasons.

■ 3.3.2 Protocol Details

As protocols go, BGP is not an overly complicated protocol (as we'll see later, what makes its operation complicated is the variety and complexity of BGP router configurations). The basic operation of BGP—the protocol state machine, the format of routing messages, and the propagation of routing updates—are all defined in the protocol standard (RFC 4271, which obsoletes RFC 1771) [20]. BGP runs over TCP on a well-known port (179). To start participating in a *BGP session* with another router, a router sends an OPEN message after establishing a TCP connection to it on the BGP port. After the OPEN is completed, both routers exchange their tables of all active routes (of course, applying all applicable route

filtering rules). Each router then integrates the information obtained from its neighbor into its routing table. The entire process may take many seconds to a few minutes to complete, especially on sessions that have a large number of active routes.

After this initialization, there are two main types of messages on the BGP session. First, BGP routers send route UPDATE messages sent on the session. These updates only send any routing entries that have changed since the last update (or transmission of all active routes). There are two kinds of updates: *announcements*, which are changes to existing routes or new routes, and *withdrawals*, which are messages that inform the receiver that the named routes no longer exist. A withdrawal usually happens when some previously announced route can no longer be used (e.g., because of a failure or a change in policy). Because BGP uses TCP, which provides reliable and in-order delivery, routes do not need to be periodically announced, unless they change.

But, in the absence of periodic routing updates, how does a router know whether the neighbor at the other end of a session is still functioning properly? One possible solution might be for BGP to run over a transport protocol that implements its own "is the peer alive" message protocol. Such messages are also called "keepalive" messages. TCP, however, does not implement a transport-layer "keepalive" (with good reason), so BGP uses its own. Each BGP session has a configurable keepalive timer, and the router guarantees that it will attempt to send at least one BGP message during that time. If there are no UPDATE messages, then the router sends the second type of message on the session: KEEPALIVE messages. The absence of a certain number BGP KEEPALIVE messages on a session causes the router to terminate that session. The number of missing messages depends on a configurable times called the *hold timer*; the specification recommends that the hold timer be at least as long as the keepalive timer duration negotiated on the session.

More details about the BGP state machine may be found in [2, 20].

Unlike many IGP's, BGP does not simply optimize any metrics like shortest-paths or delays. Because its goals are to provide reachability information and enable routing policies, its announcements do not simply announce some metric like hop-count. Rather, they have the following format:

$$IP\ prefix : Attributes$$

where for each announced IP prefix (in the "A/m" format), one or more attributes are also announced. There are a number of standardized attributes in BGP, and we'll look at some of them in more detail below.

We already talked about one BGP attribute, LOCAL PREF. This attribute isn't disseminated with route announcements, but is an important attribute used locally while selecting a route for a destination. When a route is advertised from a neighboring AS, the receiving BGP router consults its configuration and may set a LOCAL PREF for this route.

■ 3.3.3 eBGP and iBGP

There are two types of BGP sessions: *eBGP* sessions are between BGP-speaking routers in different ASes, while *iBGP* sessions are between BGP routers in the same AS. They serve different purposes, but use the same protocol.

eBGP is the "standard" mode in which BGP is used; after all, BGP was designed to exchange network routing information between different ASes in the Internet. eBGP ses-

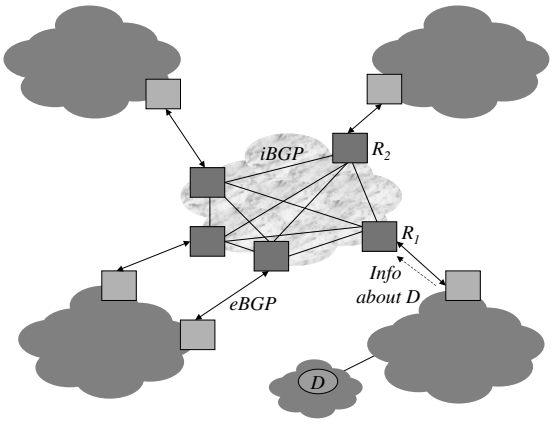


Figure 3-4: eBGP and iBGP.

sions are shown in Figure 3-4, where the BGP routers implement route filtering rules and exchange a subset of their routes with routers in other ASes. These sessions generally operate over a one-hop IP path (i.e., over directly connected IP links).

In general, each AS will have more than one router that participates in eBGP sessions with neighboring ASes. During this process, each router will obtain information about some subset of all the prefixes that the entire AS knows about. Each such eBGP router must disseminate routes to the external prefix to all the other routers in the AS. This dissemination must be done with care to meet two important goals:

1. *Loop-free forwarding.* After the dissemination of eBGP learned routes, the resulting routes (and the subsequent forwarding paths of packets sent along those routes) picked by all routers should be free of deflections and forwarding loops [6, 9].

2. *Complete visibility.* One of the goals of BGP is to allow each AS to be treated as a single monolithic entity. This means that the several eBGP-speaking routes in the AS must exchange external route information so that they have a complete view of all external routes. For instance, consider Figure 3-4, and prefix D. Router R_2 needs to know how to forward packets destined for D, but R_2 hasn't heard a direct announcement on any of its eBGP sessions for D.[4] By "complete visibility", we mean the following: *for every external destination, each router picks the same route that it would have picked had it seen the best routes from each eBGP router in the AS.*

   The dissemination of externally learned routes to routers inside an AS is done over *internal BGP* (iBGP) sessions running in each AS.

An important question concerns the topology over which iBGP sessions should be run. One possibility is to use an arbitrary connected graph and "flood" updates of external

[4]It turns out that each router inside doesn't (need to) know about all the external routes to a destination. Rather, the goal is for each router to be able to discover the best routes of the border routers in the AS for a destination.

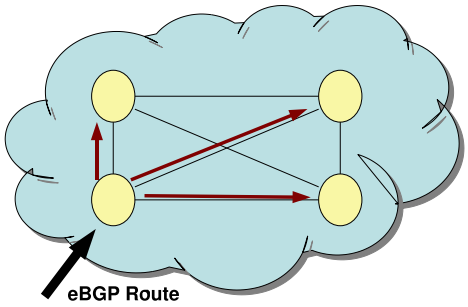


Figure 3-5: Small ASes establish a "full mesh" of iBGP sessions. Each circle represents a router within an AS. Only eBGP-learned routes are re-advertised over iBGP sessions.

routes to all BGP routers in an AS. Of cours, an approach based on flooding would require additional techniques to avoid routing loops. The original BGP specification solved this problem by simply setting up a *full mesh* of iBGP sessions (see Figure 3-5, where every eBGP router maintains an iBGP session with every other BGP router in the AS. Flooding updates is now straightforward; an eBGP router simply sends UPDATE messages to its iBGP neighbors. An iBGP router does not have to send any UPDATE messages because it does not have any eBGP sessions with a router in another AS.

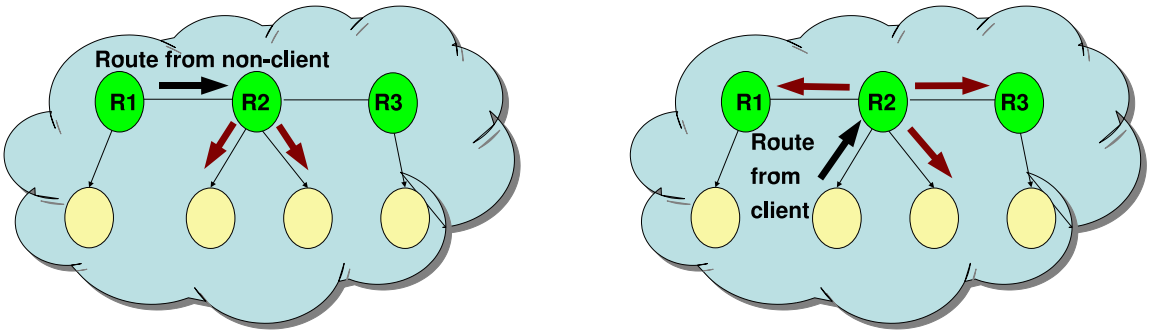
It is important to note that *iBGP is not an IGP* like RIP or OSPF, and it cannot be used to set up routing state that allows packets to be forwarded correctly between internal nodes in an AS. Rather, iBGP sessions, running over TCP, provide a way by which routers inside an AS can use BGP to exchange information about external routes. In fact, iBGP sessions and messages are themselves routed between the BGP routers in the AS via whatever IGP is being used in the AS!

One might wonder why iBGP is needed, and why one can't simply use whatever IGP is being used in the AS to also send BGP updates. There are several reasons why introducing eBGP routes into an IGP is inconvenient, though it's possible to use that method. The first reason is that most IGPs don't scale as well as BGP does with respect to the number of routes being sent, and often rely on periodic routing announcements rather than incremental updates (i.e., their state machines are different). Second, IGPs usually don't implement the rich set of attributes present in BGP. To preserve all the information about routes gleaned from eBGP sessions, it is best to run BGP sessions inside an AS as well.

The requirement that the iBGP routers be connected via a complete mesh limits scalability: a network with e eBGP routers and i other interior routers requires $e(e-1)/2+ei$ iBGP sessions in a full-mesh configuration. While this quadratic scaling is not a problem for a small AS with only a handful of routers, large backbone networks typically have several hundred (or more) routers, requiring tens of thousands of iBGP sessions. This quadratic scaling does not work well in those cases. The following subsection discusses how to improve the scalability of iBGP sessions.

■ 3.3.4 iBGP Scalability

Two methods to improve iBGP scalability are currently popular. Both require the manual configuration of routers into some kind of hierarchy. The first method is to use *route reflec-*



(a) Routes learned from non-clients are re-advertised to clients only.

(b) Routes learned from clients are re-advertised over all iBGP sessions.

Figure 3-6: Larger ASes commonly use route reflectors, which advertise some iBGP-learned routes, as described above. Directed edges between routers represent iBGP sessions from route reflectors to clients (e.g., router $R2$ is a route reflector with two clients). As in Figure 3-5, all routers re-advertise eBGP-learned routes over all iBGP sessions.

tors [1], while the second sets up *confederations* of BGP routers [22]. We briefly summarize the main ideas in route reflection here, and refer the interested reader to RFC 3065 [22] for a discussion of BGP confederations.

A route reflector is a BGP router that can be configured to have *client* BGP routers. A route reflector selects a single best route to each destination prefix and announces that route to all of its clients. An AS with a route reflector configuration follows the following rules in its route updates:

1. If a route reflector learns a route via eBGP or via iBGP from one of its clients, the it re-advertises that route over all of its sessions to its clients.

2. If a route reflector learns a route via iBGP from a router that is not one of its clients, then it re-advertises the route to its client routers, *but not over any other iBGP sessions*.

Having only one route reflector in an AS causes a different scaling problem, because it may have to support a large number of client sessions. More importantly, if there are multiple egress links from the AS to a destination prefix, a single route-reflector configuration may not use them all well, because all the clients would inherit the single choice made by the route reflector. To solve this problem, many networks deploy multiple route reflectors, organizing them hierarchically. Figure 3-6 shows an example route reflector hierarchy and how routes propagate from various iBGP sessions.

BGP route updates propagate differently depending on whether the update is propagating over an eBGP session or an iBGP session. An eBGP session is typically a *point-to-point* session: that is, the IP addresses of the routers on either end of the session are directly connected with one another and are typically on the same local area network. There are some exceptions to this practice (i.e., "multi-hop eBGP"), but directly connected eBGP sessions is normal operating procedure. In the case where an eBGP session is point-to-point, the next-hop attribute for the BGP route is guaranteed to be reachable, as is the other end of the point-to-point connection. A router will advertise a route over an eBGP session regardless of whether that route was originally learned via eBGP or iBGP.

Route Attribute	Description
Next Hop	IP Address of the next-hop router along the path to the destination. On eBGP sessions, the next hop is set to the IP address of the border router. On iBGP sessions, the next hop is not modified.
AS path	Sequence of AS identifiers that the route advertisement has traversed.
Local Preference	This attribute is the first criteria used to select routes. It is not attached on routes learned via eBGP sessions, but typically assigned by the import policy of these sessions; preserved on iBGP sessions.
Multiple-Exit Discriminator (MED)	Used for comparing two or more routes from the same neighboring AS. That neighboring AS can set the MED values to indicate which router it prefers to receive traffic for that destination. *By default, not comparable among routes from different ASes.*

Table 3-1: Important BGP route attributes.

On the other hand, an iBGP session may exist between two routers that are *not* directly connected, and it may be the case that the next-hop IP address for a route learned via iBGP is more than one IP-level hop away. In fact, as the next-hop IP address of the route is typically one of the border routers for the AS, this next hop may not even correspond to the router on the other end of the iBGP session, but may be several *iBGP* hops away. In iBGP, the routers rely on the AS's internal routing protocol (i.e., its IGP) to both (1) establish connectivity between the two endpoints of the BGP session and (2) establish the route to the next-hop IP address named in the route attribute.

Configuring an iBGP topology to correctly achieve loop-free forwarding and complete visibility is non-trivial. Incorrect iBGP topology configuration can create many types of incorrect behavior, including persistent forwarding loops and oscillations [9]. Route reflection causes problems with correctness because not all route reflector topologies satisfy visibility (see [8] and references therein).

■ 3.3.5 Key BGP Attributes

We're now in a position to understand what the anatomy of a BGP route looks like and how route announcements and withdrawals allow a router to compute a forwarding table from all the routing information. This forwarding table typically has one chosen path in the form of the egress interface (port) on the router, corresponding to the next neighboring IP address, to send a packet destined for a prefix. Recall that each router implements the longest prefix match on each packet's destination IP address.

Exchanging Reachability: NEXT HOP Attribute

A BGP route announcement has a set of attributes associated with each announced prefix. One of them is the NEXT HOP attribute, which gives the IP address of the router to send the packet to. As the announcement propagates across an AS boundary, the NEXT HOP field is changed; typically, it gets changed to the IP address of the border router of the AS the announcement came from.

The above behavior is for eBGP speakers. For iBGP speakers, the first router that introduces the route into iBGP sets the NEXT HOP attribute to its so-called loopback address (the address that all other routers within the AS can use to reach the first router). All the other iBGP routers within the AS *preserve* this setting, and use the ASes IGP to route any packets destined for the route (in the reverse direction of the announcement) toward the NEXT HOP IP address. In general, packets destined for a prefix flow in the opposite direction to the route announcements for the prefix.

Length of AS Paths: ASPATH Attribute

Another attribute that changes as a route annoucement traverses different ASes is the ASPATH attribute, which is a *vector* that lists all the ASes (in reverse order) that this route announcement has been through. Upon crossing an AS boundary, the first router prepends the unique identifier of its own AS and propagates the announcement on (subject to its route filtering rules). This use of a "path vector"—a list of ASes per route—is the reason BGP is classified as a *path vector protocol*.

A path vector serves two purposes. The first is *loop avoidance*. Upon crossing an AS boundary, the router checks to see if its own AS identifier is already in the vector. If it is, then it discards the route announcement, since importing this route would simply cause a routing loop when packets are forwarded.

The second purpose of the path vector is to help pick a suitable path from among multiple choices. If no LOCAL PREF is present for a route, then the ASPATH length is used to decide on the route. Shorter ASPATH lengths are preferred to longer ones. However, it is important to remember that BGP isn't a strict shortest-ASPATH protocol (classical path vector protocols would pick shortest vectors), since it pays attention to routing policies. The LOCAL PREF attribute is always given priority over ASPATH. Many routes in practice, though, end up being picked according to shortest-ASPATH.

■ 3.3.6 Multi-Exit Discriminator (MED)

So far, we have seen the two most important BGP attributes: LOCAL PREF and ASPATH. There are other attributes like the multi-exit discriminator (MED) that are used in practice. The MED attribute is sometimes used to express route preferences between two ASes that are connected at multiple locations.[5]

When two ASes are linked at multiple locations, and one of them prefers a particular transit point over another for some (or all) prefixes, the LOCAL PREF attribute isn't useful. MEDs were invented to solve this problem.

It may be best to understand MED using an example. Consider Figure 3-7 which shows

[5]The MED attribute is quite subtle, and tends to create about as many problems as it solves in practice, as multiple papers over the past few years have shown.

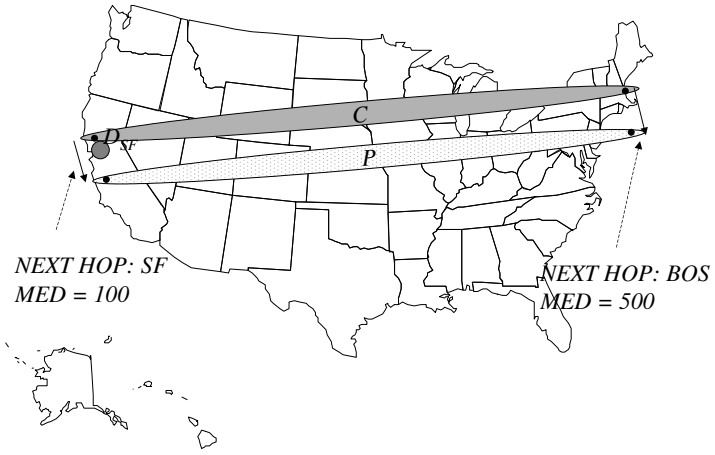


Figure 3-7: MED's are useful in many situations, e.g., if C is a transit customer of P, to ensure that cross-country packets to C traverse P's (rather than C's wide-area network). However, if C and P are in a peering relationship, MED may (and often will) be ignored. In this example, the MED for D_SF is set to 100 at the SF exchange point, and 500 in Boston, so P can do the right thing if it wants to.

a provider-customer relationship where both the provider P and customer C have national footprints. Cross-country bandwidth is a much more expensive resource than local bandwidth, and the customer would like the provider to incur the cost of cross-country transit for the customer's packets. Suppose we want to route packets from the east coast (Boston) destined for D_{SF} to traverse P's network and not C's. We want to prevent P from transiting the packet to C in Boston, which would force C to use its own resources and defeat the purpose of having P as its Internet provider.

A MED attribute allows an AS, in this case C, to tell another (P) how to choose between multiple NEXT HOP's for a prefix D_{SF}. Each router will pick the smallest MED from among multiple choices coming from the same neighbor AS. No semantics are associated with how MED values are picked, but they must obviously be picked and announced consistently amongst the eBGP routers in an AS. In our example, a MED of 100 for the SF NEXT HOP for prefix D_{SF} and a MED of 500 for the BOS NEXT HOP for the same prefix accomplishes the desired goal.

An important point to realize about MED's is that they are usually ignored in AS-AS relationships that don't have some form of financial settlement (or explicit arrangement, in the absence of money). In particular, most peering arrangements ignore MED. This leads to a substantial amount of *asymmetric routing* in the wide-area Internet. For instance, if P and C were in a peering relationship in Figure 3-7, cross-country packets going from C to P would traverse P's wide-area network, while cross-country packets from P to C would traverse C's wide-area network. Both P and C would be in a hurry to get rid of the packet from their own network, a form of routing sometimes called *hot-potato routing*. In contrast, a financial arrangement would provide an incentive to honor MED's and allow

"cold-potato routing" to be enforced.

The case of large content hosts peering with tier-1 ISPs is an excellent real-world example of cold-potato routing. For instance, an ISP might peer with a content-hosting provider to obtain direct access to the latter's customers (popular content-hosting sites), but does not want the hosting provider to free-load on its backbone. To meet this requirement, the ISP might insist that its MEDs be honored.

3.3.7 Putting It All Together: BGP Path Selection

We are now in a position to discuss the set of rules that BGP routers in an AS use to select a route from among multiple choices.

These rules are shown in Table 3-2, in priority order. These rules are actually slightly vendor-specific; for instance, the Router ID tie-break is not the default on Cisco routers, which select the "oldest" route in the hope that this route would be the most "stable."

Priority	Rule	Remarks
1	LOCAL PREF	Highest LOCAL PREF (§3.2.3). E.g., Prefer transit customer routes over peer and provider routes.
2	ASPATH	Shortest ASPATH length (§3.3.5) *Not* shortest number of Internet hops or delay.
3	MED	Lowest MED preferred (§**??**). May be ignored, esp. if no financial incentive involved.
4	eBGP > iBGP	Did AS learn route via eBGP (preferred) or iBGP?
5	IGP path	Lowest IGP path cost to next hop (egress router). If all else equal so far, pick shortest internal path.
6	Router ID	Smallest router ID (IP address). A random (but unchanging) choice; some implementations use a different tie-break such as the oldest route.

Table 3-2: How a BGP-speaking router selects routes. There used to be another step between steps 2 and 3 in this table, but it's not included in this table because it is now obsolete.

3.4 BGP in the Wild

From time to time, the fragility of the interdomain routing system manifests itself by disrupting connectivity or causing other anomalies. These problems are usually caused by misconfigurations, malice, or slow convergence. BGP is also increasingly used to allow customer networks to connect to multiple different providers for better load balance and fault-tolerance. Unfortunately, as we will see, BGP doesn't support this goal properly.

3.4.1 Hijacking Routes by Mistake or for Profit

One set of problems stems from the lack of *origin authentication*, i.e., the lack of a reliable and secure way to tell which AS owns any given prefix. The result is that it's possible for any AS (or BGP-speaking node) to originate a route for any prefix and possibly cause

traffic from other networks sent to any destination in the prefix to come to the AS. Here are two interesting examples of this behavior:

1. YouTube diverted to Pakistan: On February 24 2008, YouTube, the popular video sharing web site, became unreachable to most people on the Internet. To understand what happened, a few facts are useful:

1. Internet routers use the route corresponding to the *longest prefix match* (LPM) to send packets; i.e., if the destination IP address matches more than one prefix entry in the routing table, use the entry with the longest match between the destination address and the prefix.

2. `www.youtube.com` resolves to multiple IP addresses, all in the same "/24"; i.e., the first 24 bits of their IP addresses are all the same.

3. The Pakistani government had ordered all its ISPs to block access to YouTube. In general, there are two ways to block traffic from an IP address; the first is to simply drop all packets to and from that address, while the second is to *divert* all traffic going *to* the address to a different location, where one might present the user with a web page that says something like "We're sorry, but your friendly government has decided that YouTube isn't good for your mental well-being."[6] The latter may provide a better customer experience because it tells users what's going on, so they aren't in the dark, don't make unnecessary calls to customer support, and don't send a whole lot of traffic by repeatedly making requests to the web site. (Such diversion is quite common; it's used in many public Wi-Fi spots that require a sign-on, for example.)

Pakistan Telecom introduced a /24 routing table entry for the range of addresses to which `www.youtube.com` resolves. So far, so good. Unfortunately, because of a misconfiguration (presumably caused by human error by a stressed or careless engineer) rather than malice, routers from Pakistan Telecom leaked this /24 routing advertisement to one of its ISPs (PCCW in Hong Kong). Normally, this leak should not have caused a problem *if* PCCW knew (as it should have) the valid set of IP prefixes that Pakistan Telecom owned. Unfortunately, perhaps because of another error or oversight, PCCW didn't ignore this route, and in fact presumably prioritized this route over all the other routes it already knew to the relevant addresses (recall that the typical rule is for customer routes to be prioritized over peer and provider routes). At this stage, computers inside PCCW and its customer's networks would've been "blackholed" from YouTube, and all traffic destined there would've been sent to Pakistan Telecom.

The problem was much worse because essentially the entire Internet was unable to get to YouTube. That's because of the LPM method used to find the best route to a destination. Under normal circumtances, there is no /24 advertised on behalf of YouTube by its ISPs; those routes are contained in advertisements that cover a (much) wider range of IP addresses. So, when PCCW's neighbors saw PCCW advertising a more-specific route, they followed the rules and imported those routes into their routing tables, readvertising them to their respective neighbors, until the entire Internet (except, presumably, a few places

[6] I don't know what the page actually said.

such as YouTube's internal network itself) had this poisoned routing table entry for the IP addresses in question.

This discussion highlights a key theme about large systems: when they fail, the reasons are complicated. In this case, the following events all occurred:

1. The Pakistani government decided to censor a particular site because it was afraid that access would create unrest.

2. Pakistan Telecom decided to divert traffic using a /24 and leaked it in error.

3. PCCW, which ought to have ignored the leaked advertisement, didn't.

4. The original correct advertisements involved less-specific prefixes; in this case, had they been /24s as well, the problem may not have been as widespread.

5. Pakistan Telecom inadvertently created a massive traffic attack on itself (and on PCCW) because YouTube is a very popular site getting lots of requests. Presumably the amount of traffic made diagnosis difficult because packets from tools like `traceroute` might not have progressed beyond points of congestion, which might have been upstream of Pakistan Telecom.

   It appears that the first indication of what might have really happened came from an investigation of the logs of routing announcements that are available to various ISPs and also publicly. They showed that a new AS had started originating a route to a prefix that was previously always been originated by others.

   This observation suggests that a combination of public "warning systems" that maintain such information and flag anomalies might be useful (though there are maany legitimate reasons why routes often change origin ASes too). It also calls for the maintenance of a correct registry containing prefix to owner AS mappings; studies have shown that current registries unfortunately have a number of errors and omissions.

Far from being an isolated incident, such problems (black holes and hijacks) arise from time to time.[7] There are usually a few serious incidents of this kind every year, though selectively taking down a popular site tends to make the headlines more easily. There are also several smaller-scale incidents and anomalies that show up on a weekly basis.

2. Spam from hijacked prefixes: An interesting "application" of routing hijacks using principles similar to the one discussed above is in transmitting hard-to-trace email spam. On the Internet, it is easy for a source to spoof a source IP address and pretend to send packets from an IP address that it hasn't legitimately been assigned. Email, however, runs atop TCP, which uses a feedback channel for acknowledgments, and email is a bi-directional protocol. So, untraceable source spoofing is a bit trickier because the spoofer must also be able to *receive* packets at the spoofed IP address.

An ingenious solution to this problem is for the bad guy to convince one or more upstream ISPs (which might themselves be a bit sketchy or just look the other way because

[7] The "AS 7007" incident in 1997 was perhaps the first massive outage that partitioned most of Sprint's large network and customer base from the rest of the Internet. In that incident, a small ISP in Florida (AS number 7007) was the party that originated the misconfigured route leak, but other ISPs were also culpable in heeding those unrealistic route advertisements.

Finding	Time-frame
Serious routing pathology rate of 3.3%	Paxson 1995
10% of routes available less than 95% of the time	Labovitz et al. 1997
Less than 35% of routes available 99.99% of the time	Labovitz et al. 1997
40% of path outages take 30+ minutes to repair	Labovitz et al. 2000
5% of faults last more than 2 hours, 45 minutes	Chandra et al. 2001
Between 0.23% and 7.7% of "path-hours" experienced serious 30-minute problems in 16-node overlay	Andersen et al. 2001
Networks dual-homed to Tier-1 ISPs see many loss bursts on route change	Wang et al. 2006
50% of VoIP disruptions are highly correlated with BGP updates	Kushman 2006

Table 3-3: Internet path failure observations, as reported by several studies.

they're getting paid to ignore questionable behavior) to pay attention to BGP routing announcements. The bad guy temporarily hijacks a portion of the IP address space, typically an unassigned one (though it doesn't have to be), by sending announcements about that prefix. When that announcement propagates, routers in the Internet know how to reach the corresponding addresses. The bad guy initiates a large number of email connections, dumps a whole lot of spam, and then after 45 minutes or an hour simply withdraws the advertised route and disappears. Later, when one tries to trace a path to the offending source IP addresses of the spam, there is absolutely no trace! A recent study found that 10% of spam received at a "spam trap" (a domain with a number of fake email receiver addresses set up to receive spam to analyze its statistics) came from IP addresses that corresponded to such route hijacks [18].

Of course, if all BGP announcements were logged and carefully maintained, one can trace where messages sent from hijacked routes came from, but oftentimes these logs aren't maintained correctly at a fine-enough time granularity. In time, such logs will probably be maintained at multiple BGP vantage points in the Internet, and at that time those wishing to send untraceable garbage may have to invent some other way of achieving their goals.

■ 3.4.2 Convergence Problems

With BGP, faults take many seconds to detect and it may take several minutes for routes to con- verge to a consistent state afterwards. Upon the detection of a fault, a router sends a withdrawal message to its neighbors. To prevent routing advertisements from propagating through the entire network and causing routing table calculations for failures or routing changes that might only be transient, each router implements a route flap damping scheme by not paying attention to frequently changing advertisements from a router for a prefix. Damping is believed by many to improve scalability, but it also increases convergence time.

In practice, researchers have found that wide-area routes are often unavailable. The empirical observations summarized in Table 3-3 are worth noting.

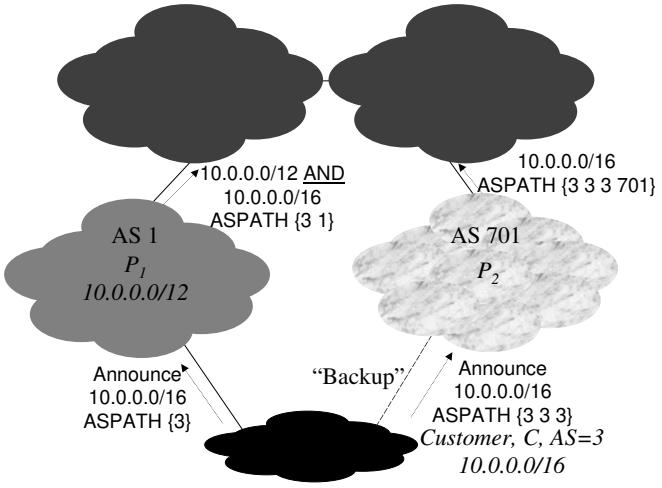


Figure 3-8: Customer C is multi-homed with providers P_1 and P_2 and uses provider-based addressing from P_1. C announces routes to itself on both P_1 and P_2, but to ensure that P_2 is only a backup, it might use a hack that pads the ASPATH attribute as shown above. However, notice that P_1 must announce (to its providers and peers) *explicit* routes on both its regular address block *and* on the customer block, for otherwise the path through P_2 would match based on longest prefix in the upstream ASes.

■ 3.4.3 Multi-homing

BGP allows an AS to be multi-homed, supporting multiple links (and eBGP sessions) between two ASes, as well an AS connecting to multiple providers. In fact, there is no restriction on the AS topology that BGP itself imposes, though prevalent routing policies restrict the topologies one observes in practice. Multi-homing is used to tolerate faults and balance load. An example is shown in Figure 3-8, which shows the topology and address blocks of the concerned parties. This example uses *provider-based addressing* for the customer, which allows the routing state in the Internet backbones to scale better because transit providers can aggregate address blocks across several customers into one or a small number of route announcements to their respective providers.

Achieving scalable and efficient multi-homing with BGP is still an open research question. As the number of multi-homed customer networks grows, the stress (in terms of routing churn, convergence time, and routing table state) on the interdomain routing system will increase. In addition, the interaction between LPM, failover and load balance goals, and hacks like the AS path padding trick are complex and often cause unintended consequences.

■ 3.5 Summary

The Internet routing system operates in an environment of "competitive cooperation", in which different independently operating networks must cooperate to provide connectivity

while competing with each other. It must also support a range of routing policies, some of which we discussed in detail (transit and peering), and must scale well to handle a large and increasing number of constituent networks.

BGP, the interdomain routing protocol, is actually rather simple, but its operation in practice is extremely complex. Its complexity stems from configuration flexibility, which allows for a rich set of attributes to be exchanged in route announcements. There are a number of open and interesting research problems in the area of wide-area routing, relating to failover, scalability, configuration, correctness, load balance (traffic engineering), security, and policy specification. Despite much activity and impressive progress over the past few years, interdomain routing remains hard to understand, model, and make resilient.

■ Acknowledgments

These notes have evolved over the past few years. I thank Nick Feamster for a productive collaboration on various Internet routing problems that we had during his PhD research with me between 2001-06. Figures 3-5 and 3-6 are taken from Nick's dissertation. Thanks also to Jennifer Rexford and Mythili Vutukuru for several discussions on interdomain routing.