CHAPTER

# Border Gateway Protocol 15

## What You Will Learn

In this chapter, you will learn about the BGP and the essential role it plays on the Internet. With BGP, routing information is circulated outside the AS and to all routing domains. We'll see how a simple routing policy change can make a destination unreachable.

You will learn about the differences between the Internet BGP (IBGP) and the Exterior Gateway Protocol (EBGP), and why both are needed. We'll also look at BGP attributes and message formats.

The EGP used on the Internet is the Border Gateway Protocol (BGP). IGPs run between the routers inside a routing domain (single AS). BGP runs between different autonomous services (ASs). BGP runs on links between the border routers of these routing domains and shares information about the routes within the AS or learned by the AS with the AS on the other side of the "border."

BGP makes sure that every network and interface in any AS located anywhere on the Internet is reachable from every other place. BGP does not generate any routing information on its own, unlike the IGPs, which essentially "bootstrap" themselves into existence. BGP relies on an underlying IGP (or static routes) as the source of the BGP-distributed information.

BGP runs on the border routers of Ace ISP's AS 65459 (routers P9 and P4) and Best ISP's AS 65127 (routers P7 and P2). These are highlighted in Figure 15.1. An IGP such as OSPF or IS–IS runs on the direct links between routers P9 and P4 and routers P7 and P2, but these are interior links. BGP runs on the other links between the backbone routers.

## BGP AS A ROUTING PROTOCOL

There *are* EGPs defined other than BGP. The Inter-Domain Routing Protocol (IDRP) from ISO is the EGP that was to be used with IS–IS as an IGP. IDRP is also sometimes promoted as the successor to BGP, or the best way to carry IPv6 routing information

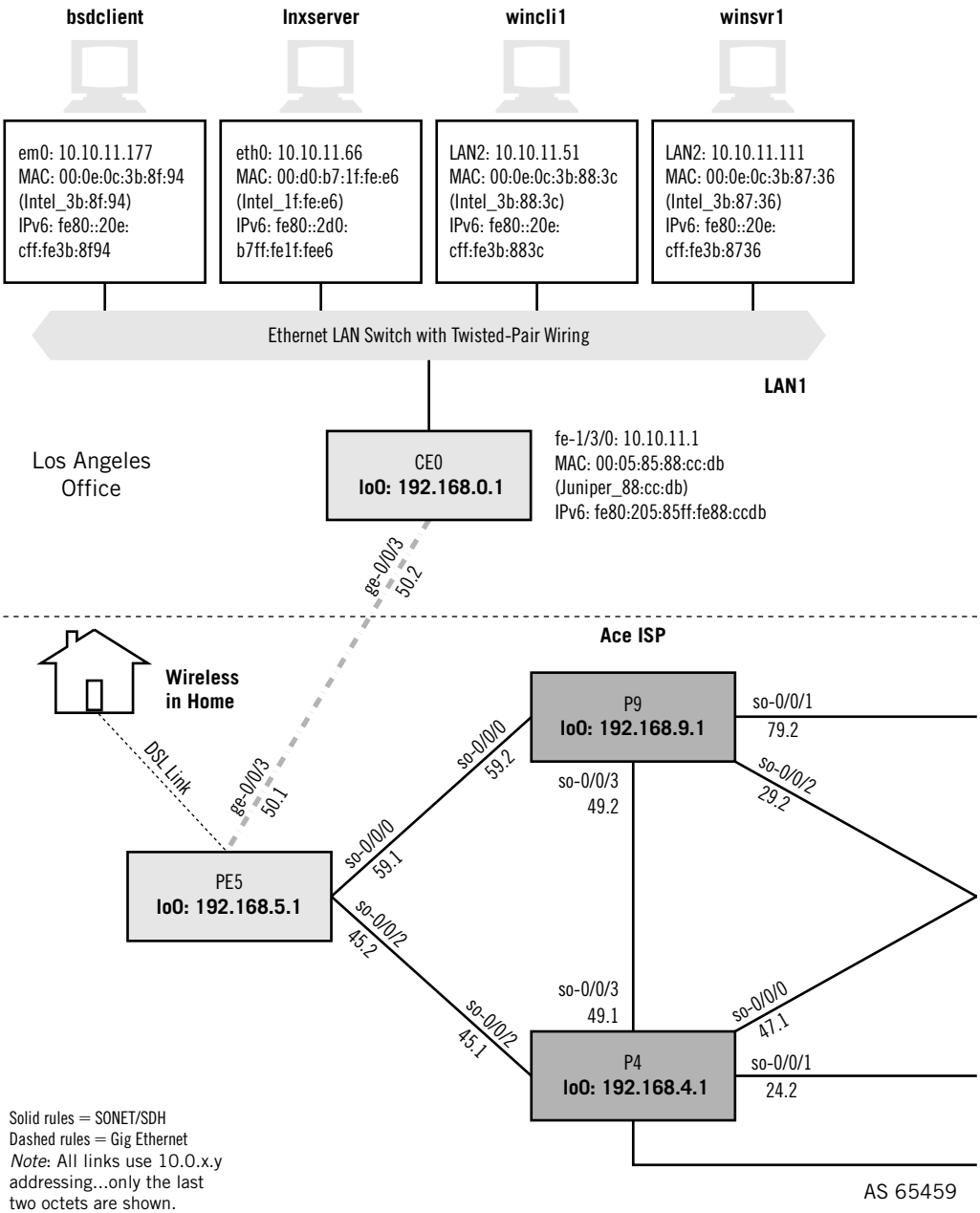
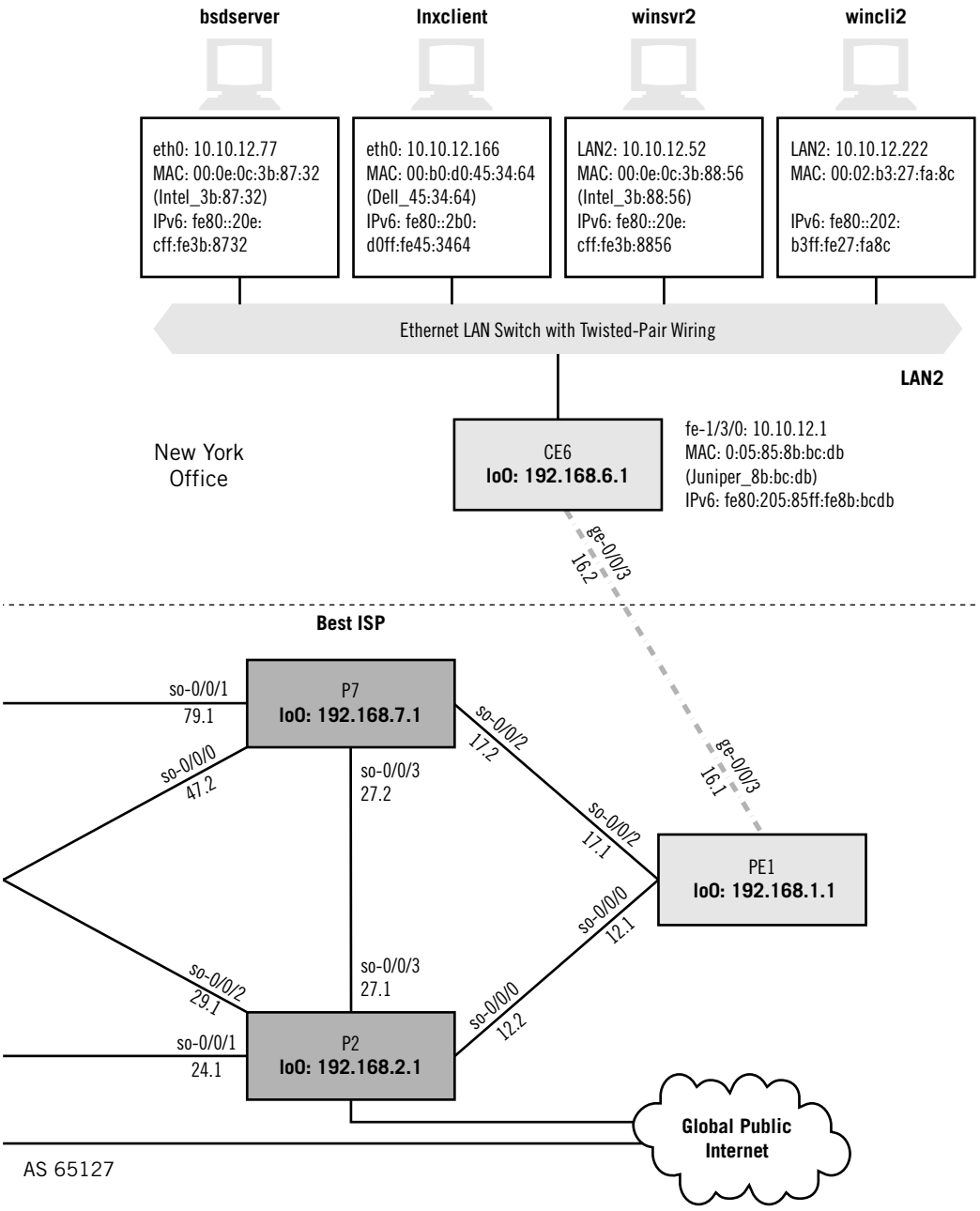



**FIGURE 15.1**

BGP on the Illustrated Network.


bsdserver
lnxclient
winsvr2
wincli2
eth0: 10.10.12.77
MAC: 00:0e:0c:3b:87:32
(Intel_3b:87:32)
IPv6: fe80::20e:
cff:fe3b:8732
eth0: 10.10.12.166
MAC: 00:b0:d0:45:34:64
(Dell_45:34:64)
IPv6: fe80::2b0:
d0ff:fe45:3464
LAN2: 10.10.12.52
MAC: 00:0e:0c:3b:88:56
(Intel_3b:88:56)
IPv6: fe80::20e:
cff:fe3b:8856
LAN2: 10.10.12.222
MAC: 00:02:b3:27:fa:8c
IPv6: fe80::202:
b3ff:fe27:fa8c
Ethernet LAN Switch with Twisted-Pair Wiring
LAN2
New York
Office
CE6
lo0: 192.168.6.1
fe-1/3/0: 10.10.12.1
MAC: 0:05:85:8b:bc:db
(Juniper_8b:bc:db)
IPv6: fe80:205:85ff:fe8b:bcdb
ge-0/0/3
16.2
Best ISP
so-0/0/1
79.1
P7
lo0: 192.168.7.1
so-0/0/2
17.2
so-0/0/0
47.2
so-0/0/3
27.2
ge-0/0/3
16.1
so-0/0/2
17.1
PE1
lo0: 192.168.1.1
so-0/0/0
12.1
so-0/0/2
29.1
so-0/0/3
27.1
so-0/0/0
12.2
so-0/0/1
24.1
P2
lo0: 192.168.2.1
Global Public
Internet
AS 65127

between ISP ASs. However, when it comes to the Internet today, the only EGP worth considering is BGP.

In a very real sense, BGP is not a routing protocol at all. BGP does not really carry routing information from AS to AS, but information *about* routes from AS to AS. Generally, a route that passes through fewer ASs (ISPs) than another is considered more attractive, although there are many other factors (BGP attributes) to consider. BGP is a routing protocol without real routes or metrics, and both of those derive from the IGP. BGP is not a link-state protocol, because the state of links in many AS clouds would be difficult to convey and maintain across the entire network (and links would tend to "average out" to a sort of least common denominator anyway). But it's not a distance-vector protocol either, because more attributes than just AS path length determine active routes. BGP is called a "path-vector" protocol (a vector has a direction as well as value), but mainly because a new term was needed to describe its operation.

BGP information is not even described as a "route." BGP carries network layer reachability information (NLRI). BGP "routes" do not have metrics, like IGP routes, but *attributes*. Together, the BGP NLRI and their attributes allow other ASs to make decisions about the best way to reach a route (network) in another AS. Once a packet is routed to the correct AS through BGP information, the packet is delivered locally using the IGP information.

The differences between BGP and IGPs should always be remembered. Some new to BGP struggle with BGP terminology and concepts because they attempt to interpret BGP features in terms of more familiar IGP features. BGP does not work like an IGP because BGP is *not* an IGP and should not work like an IGP. When BGP passes information from one AS border router to another AS border router inside an AS, a form known as interior BGP (IBGP) is used. When BGP passes information from one AS to another AS, the form of BGP used is called exterior BGP (EBGP).

This chapter does not deal much with routing policies for BGP based on multiple attributes, which determine how the routers use BGP to route packets. Complex routing policies are beyond the scope of this book.

## Configuring BGP

It's important to keep in mind exactly what is meant by a routing domain and routing policy. For example, is CE0 part of AS 65459 or not? This is not as simple a question as it sounds, because there might be a dozen routers behind CE0 that the Ace ISP knows nothing about. But the interface to PE5 is firmly under the control of Ace, and generally all customer site routers are considered part of the ISP's routing domain in the sense that a routing policy on PE5 can always control the routing behavior of CE0.

This does not mean something like preventing the users on LAN1 from running Internet Chat or something. This type of application-level detailing is not what a routing policy is for. Corporate policies of this type (application policing) are best handled by an appliance on site. ISP routing policies determine things like where the

10.10.11.0/24 route to LAN1 is advertised or held back, and which routes are accepted from other sources.

Let's see how easy it is to configure BGP on the border routers. Each of them is essentially identical in basic configuration, so let's use P9 as an example.

```
set protocols bgp group ebgp-to-as65127 type external;
set protocols bgp group ebgp-to-as65127 peer-as 65127;
set protocols bgp group ebgp-to-as65127 neighbor 10.0.79.1;
set protocols bgp group ebgp-to-as65127 neighbor 10.0.29.1;

set protocols bgp group ibgp-mesh type internal;
set protocols bgp group ibgp-mesh local-address 192.168.9.1;
set protocols bgp group ibgp-mesh neighbor 192.168.4.1;
set protocols bgp group ibgp-mesh neighbor 192.168.5.1;
```

BGP configurations are organized into groups that have user-defined names (ebgp-to-as65127 and ibgp-mesh) Note that there are two types of BGP running on the border routers: EBGP and IBGP. EBGP must know the other AS number and IBGP must know the local address to use as a source address (routers typically have many IP addresses). Note that EBGP uses link addresses and IBGP uses the router's "loopback" address, in this case the address assigned to the routing engine. We'll see why this is usually done when we discuss EBGP and IBGP later in this chapter.

We showed at the end of the previous chapter that we could ping IPv6 addresses from the Windows XP client on LAN1 to the Windows XP client on LAN2. Let's see if the same works for the IPv4 addresses on the Unix hosts. All is well between bsdclient and bsdserver.

```
bsdclient# ping 10.10.12.77
PING 10.10.12.1 (10.10.12.77): 56 data bytes
64 bytes from 10.10.12.77: icmp_seq=0 ttl=255 time=0.600 ms
64 bytes from 10.10.12.77: icmp_seq=1 ttl=255 time=0.477 ms
64 bytes from 10.10.12.77: icmp_seq=2 ttl=255 time=0.441 ms
64 bytes from 10.10.12.77: icmp_seq=3 ttl=255 time=0.409 ms
^C
--- 10.10.12.77 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.409/0.482/0.600/0.072 ms
```

The default behavior for BGP is to advertise all active routes that it learns by its own operation, so no special advertising policies are needed on the backbone routers. Because there are direct links in place between the two ISPs to connect the Los Angeles office (LAN1) with the New York office (LAN2), each ISP relies on the routing protocol metrics to make sure traffic flowing between LAN1 (10.10.11/24) and LAN2 (10.10.12/24) is not forwarded onto the Internet. That is, the cost of forwarding a LAN1-LAN2 packet between the provider backbone routers will always be less than using the Internet at large.

However, one day the users on LAN1 and LAN2 discover a curious thing: no one can reach servers on the other LAN. Pings to the local router work fine, but pings to remote hosts on the other LAN produce no results at all.

```
bsdserver# ping 10.10.12.1
PING 10.10.12.1 (10.10.12.1): 56 data bytes
64 bytes from 10.10.12.1: icmp_seq=0 ttl=255 time=0.599 ms
64 bytes from 10.10.12.1: icmp_seq=1 ttl=255 time=0.476 ms
64 bytes from 10.10.12.1: icmp_seq=2 ttl=255 time=0.401 ms
64 bytes from 10.10.12.1: icmp_seq=3 ttl=255 time=0.443 ms
^C
--- 10.10.12.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.401/0.480/0.599/0.071 ms
bsdserver# ping 10.10.11.177
PING 10.10.11.177 (10.10.11.177): 56 data bytes
^C
--- 10.10.11.177 ping statistics ---
5 packets transmitted, 0 packets received, 100% packet loss
```

The remote router cannot be pinged either (presumably, no security prevents them from pinging to another site router's port).

```
bsdserver# ping 10.10.11.1
PING 10.10.11.1 (10.10.11.1): 56 data bytes
^C
--- 10.10.11.1 ping statistics ---
7 packets transmitted, 0 packets received, 100% packet loss
```

## The Power of Routing Policy

There are many things that could be wrong in this situation. In this case, the cause of the problem is ultimately determined to be a feud between the Ace ISP and Best ISPs running the service provider routers. The issue (greatly exaggerated here) is a server located on LAN2 in New York. This essential server provides full-motion video, huge database files, and all types of other information to the clients in Los Angeles on LAN1. Naturally, a lot more packets flow from Best ISP's AS to Ace ISP's AS than the other way around. So, the Ace ISP (AS 65459) controlling border routers P9 and P4 decided that Best ISP (AS 65127) should pay for all these "extra" packets they were delivering from the New York server. Shortly before the LANs stopped communicating, they sent a bill to Best ISP—turning AS 65127 from a peer into a customer.

Naturally, Best ISP was not happy about this new arrangement and refused to pay. So, Ace ISP decided to do a simple thing: they applied a routing policy and did not send any information about the LAN1 network (`10.10.11/24`) to AS 65127's border routers (P7 and P2). If the border routers don't know how to send packets back to LAN1 from the servers on LAN2, Ace ISP will be getting what they paid Best ISP for—which is nothing. (In the real world, the customer paying for LAN1 and LAN2 connectivity would be asked to pay for the asymmetrical traffic load.)

Without the correct routing information available on the routers on both ASs, no one on LAN2 can find a route to LAN1. Even if there were still some connectivity between the sites through Ace and Best ISPs' links to the Internet, this means that the symptom would show up as a sharply increased network delay (and related application timeouts), as packets now wander through many more hops than before. Something would still clearly be wrong.

This large effect comes from a very simple cause. Let's look at the routing tables and policies on P2 and P7 (and P9 and P4) and see what has happened. Best ISP has applied a very specific routing policy to their external BGP session with Ace ISP's border routers. Here's what it looks like on P7.

```
set policy-statement no-10-10-11 term1 from route-filter 10.10.11.0/24 exact;
set policy-statement no-10-10-11 term1 then reject;
```

This basically says,"Out of all the routing protocol information, find (filter) the information matching the network `10.10.11.0/24` exactly and nothing else; then discard (reject) this information and do not use it in the routing or forwarding tables."

This *import policy* on P7 and P2 (Best ISP's routers) is applied on links from neighbor border routers P4 and P9 (Ace ISP's routers). The effect is to block BGP in AS 65127 from learning anything at all about network `10.10.11/24` from P4 and P9. Normally, Best ISP's backbone routers would pass the information about the route to LAN1 through P7 and P2 to all other routers in the AS, including CE6 (LAN2's site router). Without this information, no forwarding table can be built on CE6 to allow packets to reach LAN1. Problem solved: no packets for LAN1 can flow through Best ISP's router network.

Note that Best ISP (AS 65127) still advertises its own LAN2 network (`10.10.12/24`) to Ace ISP, and Ace ISP's routers accept and distribute the information. So, on LAN1 the site router CE0 still knows about both LANs.

```
admin@CE0# show route 10.10/16
inet.0: 38 destinations, 38 routes (38 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.10.11.0/24 *[Direct/0] 00:03:31
> via fe-1/3/0.0
10.10.11.1/32 *[Local/0] 00:03:31
Local via fe-1/3/0.0
10.10.12.0/24 *[BGP/170] 00:00:09
> via ge-0/0/3.0
```

But this makes no difference: Packets can get to LAN2 through CE6 (and from anywhere else in Best ISP's AS), but they have no way to get back if they have a source address of `10.10.12.x`. Let's verify this on CE6.

```
admin@CE6# show route 10.10/16
inet.0: 38 destinations, 38 routes (37 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.10.12.0/24 *[Direct/0] 00:25:42
> via fe-1/3/0.0
10.10.12.1/32 *[Local/0] 00:25:42
Local via fe-1/3/0.0
```

How are packets to get back to `10.10.11/24`? They can't. (The former route to LAN1 is now *hidden* because the network is no longer reachable.) This simple example shows the incredible power of BGP and routing policies on the Internet.

## BGP AND THE INTERNET

BGP is the glue of the Internet. Generally, an ISP cannot link to another ISP unless both run BGP. Contrary to some claims, customer networks (even large customer networks with many routers and multiple ASs) do not have to run BGP between their own networks and to their ISP (or ISPs). Smaller customers especially can define a limited number of static routes provided by the ISP, and larger customers might be able run IGP passively (no adjacency formed) on the border router's ISP interface. It depends on the complexity of the customer and ISP network. A customer with only one link to a single ISP generally does not need BGP at all. But if a routing protocol is needed, it will be BGP.

When a customer network links to two ISPs and runs BGP, routing policies are immediately needed to prevent the large ISPs from seeing the smaller network as a transit AS to each other. This actually happened a number of times in the early days of BGP, when small corporate networks new to BGP suddenly found themselves passing traffic between two huge national ISPs whose links to each other had failed. Why pass traffic through two or three other ISPs when "Small Company, Inc." has a BGP path a single AS long? BGP routing policies are immediately put in place to not advertise routes learned for one national ISP to the other. As long as "you can't get there from here," all will be fine at the little network in the middle.

BGP *summarizes* all that is known about the IP address space inside the local AS and *advertises* this information to other ASs. The other ASs pass this information along, until all ASs running BGP know exactly what is where on the Internet. Without BGP, a single default route must handle all destinations outside the AS. This is okay when a single router leads to the Internet, but inadequate for networks with numerous connections to other ASs and ISPs.

BGP was not the original EGP used on the Internet. The first exterior gateway protocol was Exterior Gateway Protocol (EGP). EGP is still around, but only on isolated portions of the original Internet—such as for the U.S. military. An appreciation of EGP's limitations helps to understand why BGP works the way it does.

### EGP and the Early Internet

In the early 1980s, the Internet had grown to include almost 1000 computers. Several noted that distance-vector routing protocols such as the original Gateway-to-Gateway Protocol (GGP), an IGP, would not scale to a large network environment. If every router

needed to know everything about every route, convergence times when links failed would be very high. GGP routing changes had to happen globally and in a coordinated fashion. But the Internet, even in the 1980s, was a huge network with many different types of computers and routers run by many different organizations.

The answer divided the emerging Internet into independent but interconnected ASs. As seen in Chapter 14, the AS is identified by a 4-byte (32-bit) number assigned by the same authorities that assign IP addresses. We'll use a shorthand such as `65127` instead of the full (and proper) `0.65127` to indicate legacy 2-byte AS numbers. The AS range `64512` through `65535` is reserved for private AS numbers. Inside the AS, the network was assumed to be under the control of a single network administrator. Within the AS, local network matters (addressing, links, new routers, and so on) could be addressed locally with GGP. But GGP ran only within the AS. Between ASs, some way had to be found to communicate what networks were reachable within and through one AS to the other AS.

EGP was the solution. EGP ran on the border routers (gateways), with links to other ASs. EGP routers just sent a list of other routers and the classful major networks that the router could reach. This cut down on the amount of information that needed to be sent between ASs. Today, aggregation should be used as often as possible with BGP instead of classful major network routes, but the intent and result are the same. So, if a BGP router knows about networks `10.10.1.0/24` through `10.10.127.0/24` it can aggregate the route as `10.10.0.0/17` and advertise that one route (NRLI) instead of 128 separate routing updates. Even if a network such as `10.10.11.0/24` is not included in the range, the more specific advertisement of `10.10.11.0/24` and the longest match rule will make sure traffic finds its way to the right place—as long as the route is advertised properly. Nevertheless, there are many reasons people do not aggregate as much as they should, and many of their reasons are flawed. For example, trying to protect a network against "prefix hijacking" is a bad reason not to aggregate.

There is no need for an EGP to reproduce the features of an IGP. An IGP needs to tell every router in the AS which router has which interfaces and what IP addresses are attached to these interfaces or reachable through that router (such as static routes). All that other ASs need to know is which IP addresses are reachable in a particular AS and how to get to a border router on, or nearer to, the target AS.

## The Birth of BGP

EGP suffered from a number of limitations, too technical to recount. After some initial attempts to upgrade EGP, it was decided to create a better EGP (as a class of routing protocol, contrasted with IGPs) than EGP: BGP. BGP was defined in 1989 with RFC 1105 (BGP1 or BGP-1 or BGPv1), revised in 1990 as RFC 1163 (BGP2), and revised again in 1991 as RFC 1267 (BGP3). The version of BGP used today on the Internet, BGP4, emerged in 1994 as RFC 1654 and was extended for classless operation in 1995 as RFC 1771. The baseline BGP specification today is RFC 4271. This chapter describes BGP4.

BGP has been extended for new roles on the Internet. BGP *extended communities* are used with virtual private networks (VPNs). Communities are simply labeled that so they can be used to associate NLRIs that do not share other traits. For example, a community value can be assigned to small customers and another community value used to identify a small customer with multiple sites. There are few limits to the community "tags'" usage. And BGP routes are often the only ones that can use multiprotocol label switching (MPLS) label-switched paths (LSPs). BGP is as easily extensible as IS–IS and OSPF to support new functions and add routing information that needs to be circulated between ASs.

Many organizations find themselves suddenly forced to adapt BGP in a hurry, for instance, when they have to multihome their networks. Also, when they deploy VPNs or MPLS or any one of the many newer technologies used to potentially span ISPs and ASs, BGP is needed. The problem with IGPs is that they cannot easily share information across routing domain boundaries.

## BGP AS A PATH-VECTOR PROTOCOL

One of the problems with EGP was that the metrics looked very much like RIP hop counts. Simple distance vectors were not helpful at the AS level, because hop counts did not distinguish the fast links that began appearing in major ISP network backbones. Destinations that were "close" over two or three 56- or 64-kbps links actually took much longer to reach than through four or five hops over 45-Mbps links, and distance vectors had no protection against routing loops.

Link-state protocols could have dealt with the problem by implementing some of the alternate ToS metrics described for OPSF and IS–IS. However, these would rely not only on consistent implementation among all ISPs but the proper setting of bits in IP packets. In the world of independent highly competitive ISPs, this consistency was next to impossible. So, BGP was developed as a *path-vector* protocol. This means that one of the most important attributes BGP uses to choose the active route is the length of the AS path reported in the NLRI.

To create this AS list, BGP routing updates carry a complete list of transit networks (ASs) that must be traversed between the AS receiving the update and the AS that can deliver the packet using its IGP. A loop occurs when an AS path list contains the same AS that is receiving the update, so this update is rejected and loops are prevented. If the update is accepted, that AS will add its own AS to the list when advertising the routing update to other ASs. This lets an AS apply routing policies to the updates and avoid using routes that lead through an AS that is not the preferred way to reach a destination.

Path vectors do not mean that all ASs are created equal. Numerous small ASs might get traffic through faster than one huge AS. But more aspects of a route are described in BGP than just the length of the AS path to the destination. The system allows each AS to represent the route with a different metric that means something to the AS originating the route.

But more ASs generate more and longer path information. RFC 1774 in 1995 estimated that 100,000 routes generated by 3000 ASs would have paths about 20 ASs long. There was a concern about router memory and processor requirements to store and maintain all of this information, especially in smaller routers.

Several mechanisms are built into BGP to address this. ISPs would not usually accept a BGP route advertisement with a mask more than 19 bits long (`/19`). This was called the *universally reachable* address level. The price for compact routing tables and maintenance was a loss of routing accuracy, and many ISPs relaxed this policy. Most today accept `/24` prefixes (although they can accept more specific addresses from their own customers, of course). The other BGP mechanisms to cut down on routing table size and maintenance complexity are route reflectors, confederations (also called sub-confederations), and route damping (or dampening). All of these are beyond the scope of this chapter, but should be mentioned.

## IBPG AND EBGP

BGP is an EGP that runs between individual routing domains, or ASs. When BGP *speakers* (the term for routers configured to peer with BGP neighbors) are in different ASs, the routers use an *exterior* BGP (EBPG) session to exchange information. When BGP peers are within the same AS, the routers use *interior* BGP (IBGP). These terms often appear as E-BPG/I-BGP or eBGP/iBGP.

IBGP is not some IGP version of BGP. It is used to allow BGP routers to exchange BGP routing information inside the same AS. IBGP sessions are usually only required when an AS is *multihomed* or has multiple links to other ASs. (However, we used them on the Illustrated Network anyway, and that's fine too.) An AS with only a single link to one other AS need only run EBGP on the border router and relies on the IGP to distribute routes learned by EBPG to the other routers. In the case where there is only one exit point for the entire AS, a single static default route to the border router can be used effectively instead. The reason that IBGP is needed is shown in Figure 15.2.

Without IBGP, all routes learned by EBGP must be dumped into the IGP to make sure all routes are known in the entire AS. This can easily overwhelm the IGP. For this reason, it is usual to create an IBGP mesh between routers on the backbone (other routers can make do with a handful of default routes).

EBGP sessions typically peer to the physical interface address of the neighbor router. These are often point-to-point WAN links, and are the only way to reach another AS. If the link is down, the other AS is unreachable over that link. So, there is little point in trying to keep a BGP session going to the peer.

On the other hand, IBGP sessions usually peer to the stable "loopback" interface address of the peer router. An IBGP peer can typically be reached over more than one physical interface within the AS, so even if an IBGP peer's "closest" interface is down the BGP sessions can stay up because BGP packets use the IGP routing table to find an alternate route to the peer.

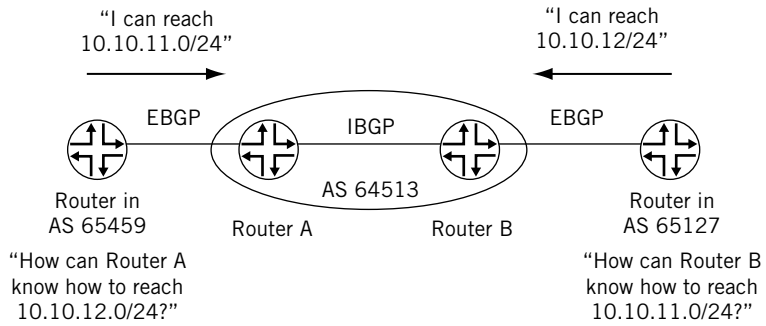



**FIGURE 15.2**

The need for IBGP. Note that if only EBGP is running, the AS in the middle must dump all BGP routes into the IGP to advertise them throughout the network.

Two BGP neighbors, EBGP or IBGP, first exchange their entire BGP routing tables—subject to the policies on each router. After that, only incremental or partial table information is exchanged when routing changes occur. BGP keepalives are exchanged because in stable networks long periods of time might elapse before something interesting happens.

## IGP Next Hops and BGP Next Hops

BGP uses NLRIs as the way one AS tells another, "I know how to reach IP address space `192.168.27.0/24` and `172.16.44.0/24` and…" The AS does not say that it is the AS that has assigned that IP address space locally. Many of the addresses might be from other ASs beyond the AS advertising the routes. The AS path allows an AS to figure out how far away a destination is through the AS that has advertised the route, or NLRI.

With an IGP, the next hop associated with a route is usually the IP address of the physical interface on the next hop router. But the BGP next hop (also sometimes called the "protocol next hop") is often the IP address of the *router* that is advertising the BGP NLRI information. The BGP next hop is the address of the BGP peer, most often the loopback interface address (the *BGP Identifier*) for IBGP and the physical interface address in the other AS for EBGP. The BGP next hop is the way one BGP router tells another, "If you have a packet for this IP address space, send it here."

The IGP has to know how to reach the next hop, whether it's a BGP next hop or not. But the next hop for EBGP is often at the end of a link to the other AS and is not running an IGP (it's not an internal link). So, how is the IGP to know about it? Well, BGP routes could be "dumped" into the IGP—but there are a lot more external routes than internal, and the whole point is to keep the IGP and EGP separate to some extent. This brings up an interesting point about the relationship of BGP and the IGP and a practice known as next hop self.

## BGP and the IGP

There is a well-known unreachable condition in BGP that must be solved with a simple routing policy know as *next hop self*, or just NHS. An EBGP route (NLRI) normally arrives from another AS with the physical address of the remote interface as the BGP next hop. If the EBGP route is readvertised through IBPG, it is likely that the BGP next hop will be completely unknown to the IGP routing tables inside the receiving AS. A router within an AS does not care how to reach a physical interface IP address in another AS. Next hop self is just a way to have the router advertising the route through IBGP use *itself* as the next hop for the EBGP route. The idea is not BGP "next-hop-is-the-physical-interface-in-another-AS" but BGP "next-hop-is-me-in-this-AS" or BGP "next-hop-self."

BGP is not a routing protocol built directly on top of IP. BGP relies on TCP connections to reach its peers, and so resembles an IP application more than an IGP routing protocol. Without the IGP to provide connectivity, TCP sessions for the BGP messages cannot be established except on links to adjacent routers. BGP does not flood information with IBPG. So, what an IBGP router learns from its IBGP peers is never passed along to another IBGP neighbor.

To fully distribute BGP information among the routers within an AS, a full mesh of IBGP connections (adjacencies) is necessary. Every IBGP router must send complete routing information to every other IBGP router in the AS. In a large AS with many external links to other ASs, this meshing requirement can add a lot of overhead traffic and configuration maintenance to the network. This is where route reflectors and confederations come in (these concepts are far beyond the scope of this chapter and will not be discussed further).

The main reasons BGP was built this way were to keep BGP as simple as possible and to prevent routing loops inside the AS. The dependency on TCP and the lack of flooding means that IBGP must communicate directly with every other router that needs to know BGP routing information. This does not mean that every router must be adjacent (connected by a direct link), because TCP can be routed through many routers to reach its destination. What it does mean is that routers connected by IBGP inside an AS must create a *full mesh* of IGBP peering sessions. This need to create a full mesh and *synchronize* BGP with the IGP is shown in Figure 15.3.

In the figure, Ace ISP and Best ISP are no longer peers. Now they are both customers of National ISP. Naturally, everyone on LAN2 still has to know how to reach LAN2 at `10.10.11.0/24` (and vice versa, of course). EBGP advertises LAN1 to National ISP, and IBGP from border router to border router makes sure that LAN2 on Best ISP can reach `10.10.11.0/24`. But what about an internal router inside National ISP's AS? There are only two ways to allow everyone in National ISP's service area to access LAN1 (presumably to buy something, although there *are* cases concerning LAN1 security where the route might not be advertised everywhere). With a full mesh of IBGP sessions in National ISP, there is no need to dump all external routes into the IGP (the IGP should only handle routes within the AS).

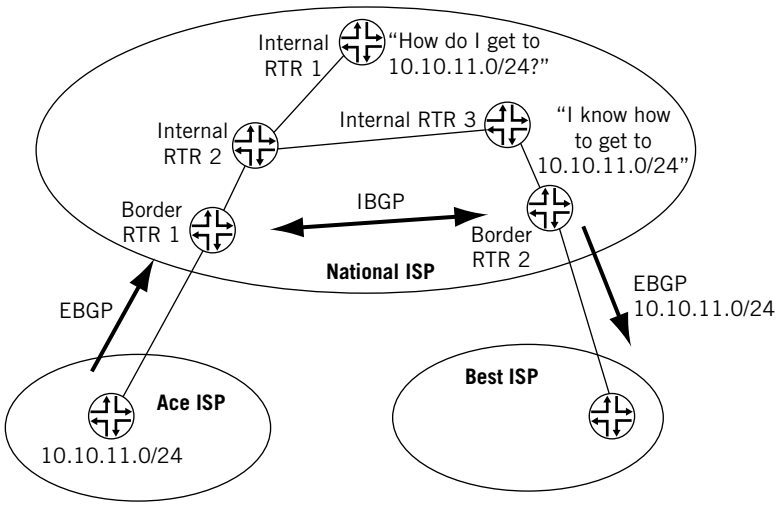



**FIGURE 15.3**

The need for a full IBGP mesh. Note that the routers inside National ISP do not necessarily know how to reach 10.10.11.0/24 (LAN1).

## OTHER TYPES OF BGP

The major types of BGP are EBGP for external peers outside the AS and IBGP for internal peers within the same AS. These are usually the only types of BGP mentioned in most sources. But there are other variations of BGP used in other situations.

One BGP variation that is becoming very important, especially where VPNs are concerned, is Multiprotocol BGP (often seen as MBGP or MP-BGP). Multiprotocol BGP originally extended BGP to support IP multicast routes and routing information. But MBGP is also used to support IP-based VPN information and to carry IPv6 routing information, such as from RIPng and OSPF for IPv6. MBGP work on IPv6 is just starting, so no special consideration of using BGP for IPv6 appears in this chapter other than to note than MBGP is used for this purpose. MBGP is currently defined in RFC 4760.

There is also Multihop BGP, sometimes seen as *EBGP multihop*. Multihop BGP is only used with EBGP and allows an EBGP peer in another AS to be more than one hop away. Usually, EBGP peers are directly connected by a point-to-point WAN link. But sometimes it is necessary to peer with a router beyond the border router that actually terminates the link. Normally, BGP packets have a TTL of 1 and thus never travel beyond the adjacent router. Multihop BGP packets have a TTL greater than 1 and the peer is beyond the adjacent router. Multihop BGP is also used in load balancing situations when there is more than one link between two border routers, and for "route-view"–style route collectors.

Finally, there is a slight change in behavior of the BGP that runs between confederations. In most cases, the version of BGP that runs between confederations is just called EBGP. However, there are slight differences in the EBGP that runs between ASs and the EBGP that runs between confederations—which are always inside the

same AS. Sometimes the variant of BGP that runs between confederations is known as Confederation BGP, or CBGP, although use of this term is not common.

## BGP ATTRIBUTES

The information that all forms of BGP carry is associated with a route (NLRI) as a series of *attributes*. This is the major difference between BGP and IGPs. IGP routes carry the route, next hop, metric, and maybe an optional tag (or two). BGP routes can carry a considerable amount of information, all intended to allow an AS to choose the "best" way to reach a destination.

Most implementations of BGP will understand 10 attributes, and some use and understand even more. Every BGP attribute is characterized by two major parameters. An attribute is either well known or optional. Well-known attributes must be understood and processed by every implementation of BGP regardless of vendor. Optional attributes are exactly that: there is no guarantee that a given BGP implementation will understand or process that particular attribute. BGP implementations that do not support an optional attribute simply pass that information on if that is what is called for, or ignore it.

In addition, a well-known BGP attribute is either mandatory or discretionary. Mandatory BGP attributes must be present in every BGP update message for EBGP, IBGP, or something else. Discretionary BGP attributes appear only in some types of BGP update messages, such as those used by EBGP only.

Finally, optional BGP attributes are transitive or nontransitive. Transitive BGP optional attributes are passed from peer to peer even if the router does not support that option. Nontransitive BGP optional attributes can be ignored by the receiver BGP process if not supported and not sent along to peers. The ten BGP attributes discussed in this chapter are listed in Table 15.1 and their characteristics are described in the list that follows.

**Table 15.1** BGP Attributes

| Attribute and Type Code | Well-Known Mandatory | Well-Known Discretionary | Optional Transitive | Optional Nontransitive |
|---|---|---|---|---|
| ORIGIN (1) | X | | | |
| AS_PATH (2) | X | | | |
| NEXT_HOP (3) | X | | | |
| LOCAL_PREF (4) | | X | | |
| ATOMIC_AGGR (5) | | X | | |
| AGGREGATOR (6) | | | X | |
| COMMUNITY (7) | | | X | |
| MED (8) | | | | X |
| ORIGINATOR_ID (9) | | | | X |
| CLUSTER_LIST (10) | | | | X |

ORIGIN—This attribute reflects where BGP obtained knowledge of the route in the first place. This can be the IGP, EGP, or "incomplete."

AS_PATH—This forms a sequence of AS numbers that leads to the originating AS for the NLRI. The main use of the AS Path is for loop avoidance among ASs, but it is common to artificially extend the AS Path attribute through a routing policy so that a particular path through a certain router looks very unattractive. The AS Path attribute can consist of an ordered list of AS numbers (AS_SEQUENCE) or just a collection of AS numbers in no particular order (AS_SET).

NEXT_HOP—The BGP Next Hop (or "protocol next hop") is quite distinct from an IGP's next hop. Outside an AS, the BGP Next Hop is most likely the border router—not the actual router inside the other AS that has this network on a local interface. Next Hop Self is the typical way to make sure that the BGP Next Hop is reachable.

LOCAL_PREF—The Local Preference of the NLRI is relative to other routes learned by IBGP within an AS and therefore is not used by EBGP. When routes are advertised with IBGP, traffic will flow toward the AS exit point (border router) that advertised the highest Local Preference for the route. It is used to establish a preferred exit link to another AS.

MULTI_EXIT_DISC (MED)—The Multi-Exit Discriminator (MED) attribute is the way one AS tries to influence another when it goes to choosing among multiple exit points (border routers) that link to the AS. A MED is the closest thing to a purely IGP metric that BGP has. Changing MEDs is one of the most common ways one ISP tries to make another ISP use the links it wants between the ISPs, such as higher speed links ("use this address on this link to reach me, unless it's down, then use this one…"). MED values are totally arbitrary.

ATOMIC_AGGREGATE and AGGREGATOR—These two attributes work together. Both are used when routing information is aggregated for BGP. A common goal on the Internet today is to represent as many networks (routes) with as few routing table entries as possible. So, as routing information makes its way through the Internet each AS will often try to condense (aggregate) the routing information as much as possible with as short a VLSM as can be properly contrived.

COMMUNITY—The BGP Community attribute is sort of a "club for routes." Communities make it easier to apply policies to routes as a group. There might be a community that applies to an ISP's customers. In that case, it is not necessary to list every customer's IP address in a policy to set Local Pref or MED (for example) as long as they all are assigned to a unique "customer" community value. Community values are often used today as a way for one ISP to inform a peer ISP of the value of the Local Pref for the route inside the originating ISP's

AS (Local Pref is not present in EBGP). The Community attribute was originally Cisco specific, but was standardized in RFC 1997. Communities just make it easier for a router to find all NLRIs associated with (for example) a particular VPN.

ORIGINATOR_ID and CLUSTER_LIST—These attributes are used by BGP route reflectors. Both of these attributes are used to prevent routing loops when route reflectors are in use. The Originator ID is a 32-bit value created by the route reflector and is the originator of the route within the local AS. If the originator router sees that its own ID is a received route, a loop has occurred and the route is ignored. The Cluster List is a list of the route reflection cluster IDs of the clusters through which the route has passed. If a route reflector sees it own cluster ID in the Cluster List, a loop has occurred and the route is ignored.

## BGP AND ROUTING POLICY

BGP is a policy-driven protocol. What BGP does and how BGP does it can be almost totally determined by routing policy. It is difficult to make BGP do exactly what an ISP wants without the use of routing policies.

Want BGP to advertise customers on static routes or running OSPF, IS–IS, or RIP? Redistribute statics, OSPF, IS–IS, and RIP into BGP? Want to artificially extend an AS path to make an AS look very unattractive for transit traffic? Write a routing policy to prepend the AS multiple times. Want to change the community attribute to add or subtract information? Use a routing policy. Concerned about the shear amount of routes advertised? Write a routing policy to aggregate the routes any way that makes sense. Want to advertise a more specific route along with a more general aggregate (called "punching a hole" in the advertised address space)? Write a routing policy. BGP depends on routing policy to behave the way it should.

### BGP Scaling

A global corporation today might have 3000 routers large and small spread around the world. Even with multiple ASs, there could be 1000 routers within an AS that might all need IBGP information—no matter how the routes have been aggregated. To fully mesh 1000 IBGP routers within an AS requires 499,500 IBGP sessions. A network 100 times larger than a 10-router network requires more than 10,000 times more IBGP sessions. Adding one router adds 1000 additional IBGP sessions to the network.

This problem with the exponential growth of IBGP sessions is the main BGP scaling issue. There are two ways to deal with this issue: the use of router reflectors (RR) and confederations.

What is the difference between RRs and confederations? At the risk of offending BGP purists, it can be loosely stated that RRs are a way of grouping BGP routers inside

an AS and running IBGP between the RR clusters. Confederations are a way of grouping BGP routers inside an AS and running EBGP between the confederation "sub-ASs." Because of the differences between RRs and confederations, it is even possible to have both configured at the same time in the same AS. There is also BGP route damping, which is not a way of dealing with BGP scaling directly but rather a way to deal with the effects of BGP scaling in terms of the amount of routing information that needs to be distributed to IBGP and EBGP peers when a router or link fails.

## BGP MESSAGE TYPES

BGP messages types are simpler than those used by OSPF and IS–IS because of the presence of TCP. TCP handles all of the details of connection setup and maintenance, and before a BGP peering session is established the router performs the usual TCP three-way handshake using TCP port 179 on one router. The other router uses a port that is not well known, and it is just a matter of whose TCP SYN message arrives first that determines which BGP peer is technically the "server." All BGP messages are then unicast over the TCP connection. There are only four BGP message types.

*Open*—Used to exchange version numbers (usually four, but two routers can agree on an earlier version), AS numbers (same for IBGP, different for EBGP), hold time until a Keepalive or Update is received (the smaller value is used if they differ), the BGP identifier (Router ID, usually the loopback interface address), and options such as authentication method (if used).

*Keepalive*—Keepalive messages are used to maintain the TCP session when there are no Updates to send. The default time is one-third of the hold time established in the Open message exchange.

*Update*—This advertises or withdraws routes. The Update has fields for the NLRI (both prefix and VLSM length), path attributes, and withdrawn routes by prefix and length.

*Notification*—These are for errors and always close a BGP connection. For example, a BGP version mismatch in the Open message closes the connection, which must then be reopened when one router or the other adjusts its version support.

The maximum TCP segment size for a BGP message is 4096 bytes and the minimum is 19 bytes. All BGP messages have a common header, as shown in Figure 15.4.

The Marker is a 16-byte field used for synchronizing BGP connections and in authentication. If no authentication is used and the message is an Open, this field is set to all 1s. The Length is a 16-bit field that contains the length of the message, including the header, in bytes. Finally, the Type is an 8-bit field set to 1 (Open), 2 (Update), 3 (Notification), or 4 (Keepalive).

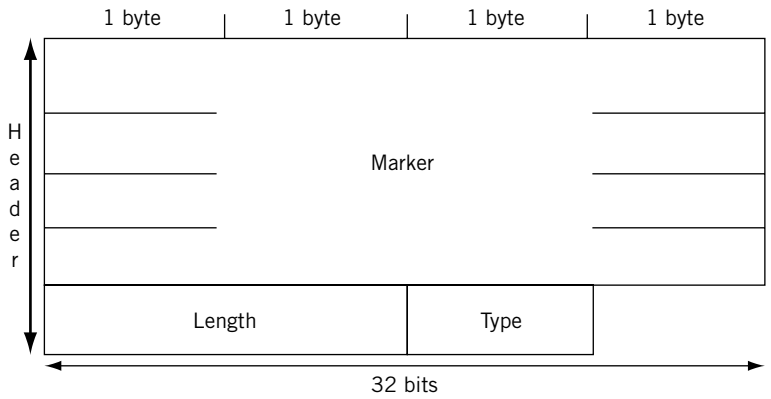



**FIGURE 15.4**

The BGP message header carried inside a TCP segment.

## BGP MESSAGE FORMATS

A data portion follows the header in all but the Keepalive messages. Keepalives consist of only the BGP message headers and so need not be discussed further in this section.

### The Open Message

Once a TCP connection has been established between two BGP speakers, Open messages are exchanged between the BGP peers. If the Open is acceptable to a router, a Keepalive is sent to confirm the Open. Once Keepalives are exchanged, peers can exchange Updates, Keepalives, and Notification messages. The format of the Open message is shown in Figure 15.5.

The Open message has an 8-bit Version field, a 2-byte My Autonomous System field, a 2-byte Hold Time value (0 or at least 3 seconds), a 32-bit BGP Identifier (router ID), an 8-bit Optional Parameters Length field (set to 0 if no options are present), and the optional parameters themselves in the same TLV format used by IS–IS in the previous chapter. BGP options are not discussed in this chapter.

### The Update Message

The Update message is used to advertise NLRIs (routes) to a BGP peer, to withdraw multiple routes that are now unreachable (or *unfeasible*), or both. The format of the Update message is shown in Figure 15.6. Because of the peculiar "skew" the 19-byte BGP header puts on subsequent fields, this message is shown in a different format than the others. There are two distinct sections to the Update message. They are used to Withdraw and Advertise routes.

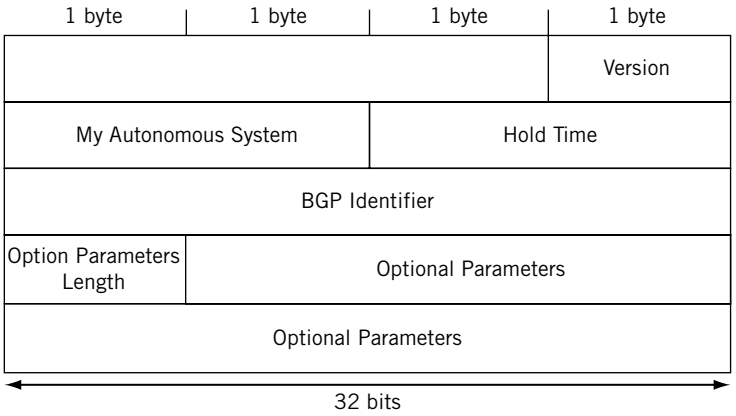



**FIGURE 15.5**

The BGP Open message showing optional fields at the end.

| Unfeasible Routes Length (2 bytes) |
|---|
| Withdrawn Routes (variable length) |
| Total Path Attribute Length (2 bytes) |
| Path Attribute (variable length) |
| Network Layer Reachability Information (variable length) |

**FIGURE 15.6**

The BGP Update message. This is the main way routes are advertised with BGP.

The Update message starts with a 20-byte field indicating the total length of the Withdrawn Routes field in bytes. If there are no Withdrawn Routes, this field is set to zero. If there are Withdrawn Routes, the routes follow in a variable-length field with the list of Withdrawn Routes. Each route is a Length/Prefix pair. The length indicates the number of bits that are significant in the following prefix and form a mask/prefix pair.

The next field is a 2-byte Total Path Attribute Length field. This is the length in bytes of the Path Attributes field that follows. A value of zero means that nothing follows.

The variable-length Path Attributes field lists the attributes associated with the NRLIs that follow. Each Path attribute is a TLV of varying length, the first part of which

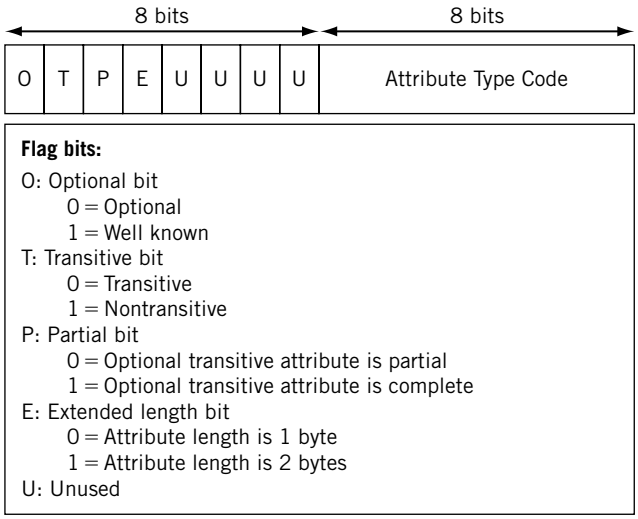



**FIGURE 15.7**

The BGP Attribute Type format. This is how NRLIs are grouped.

is the 2-byte Attribute Type. There is a structure to the Attribute Type field, as shown in Figure 15.7. There are four flag bits, four unused bits, and then an 8-bit Attribute Type code.

There are other attribute codes in use with BGP, but these are not discussed in this chapter. One of the most important of these other attributes is the Extended Community attribute used in VPNs.

The Update message ends with a variable-length NLRI field. Each NLRI (route) is a Length/Prefix pair. The length indicates the number of bits that is significant in the following prefix. There is no length field for this list that ends the Update message. The number of NLRIs present is derived from the known length of all of the other fields.

So, instead of saying "here's a route and these are its attributes…" for every NLRI advertised the Update message basically says "here's a group of path attributes and here are the routes that these apply to…" This cuts down on the number of messages that needs to be sent across the network. In this way, each Update message forms a unit of its own and has no further fragmentation concerns.

## The Notification Message

Error messages in BGP have an 8-bit Error Code, an 8-bit Subcode, and a variable-length Data field determined by the Error Code and Subcode. The format of the BGP Notification message is shown in Figure 15.8.

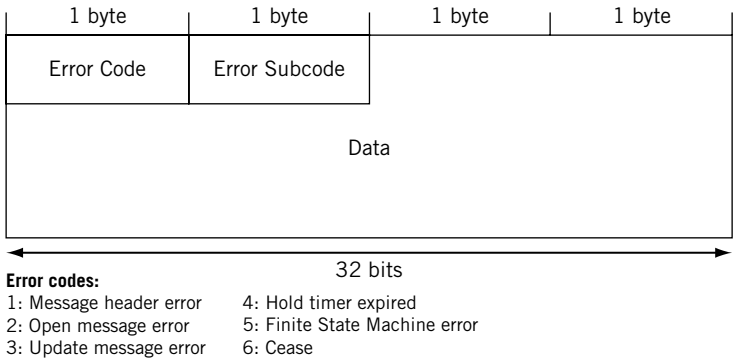



**FIGURE 15.8**

The BGP Notification message format. BGP benefits from using TCP as a transport protocol.

A full discussion of BGP Notification codes and subcodes is beyond the scope of this chapter. The major Error Codes are Message Header Error (1), Open Message Error (2), Update Message Error (3), Hold Timer Expired (4), Finite State Machine Error (5), used when the BGP implementation gets hopelessly confused about what it should be doing next, and Cease (6), used to end the session.

## QUESTIONS FOR READERS

Figure 15.9 shows some of the concepts discussed in this chapter and can be used to help you answer the following questions.

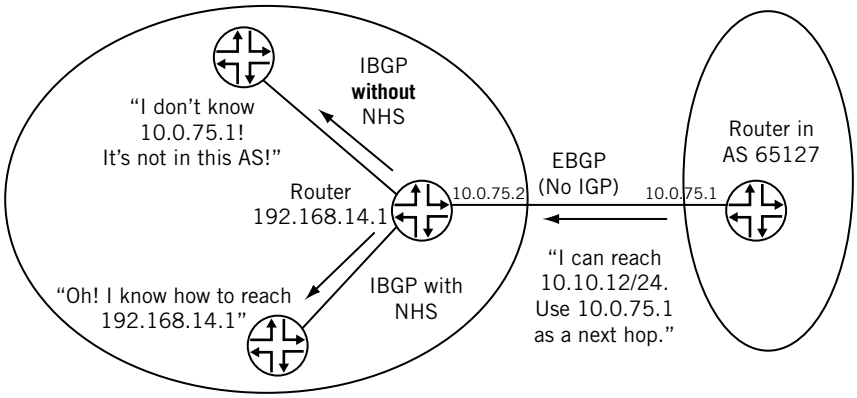



**FIGURE 15.9**

How Next Hop Self allows internal routers to forward packets for BGP routes. Border router 192.168.14.1 substitutes its own address for the "real" next hop.

1. BGP distributes "reachability" information and not routes. Why doesn't BGP distribute route information?
2. What does it mean to say that the BGP is a "path-vector" protocol?
3. What is "next hop self" and why is it important in BGP?
4. Which two major BGP router configurations are employed to deal with BGP scaling?
5. What are the ten major BGP attributes?