CCIE Professional Development

Routing TCP/IP

Volume II

Second Edition

ciscopress.com

Jeff Doyle, CCIE No. 1919

Contents

Chapter 2

Introduction to BGP

Now that you have a firm understanding of the key issues surrounding inter-domain routing from Chapter 1, "Inter-Domain Routing Concepts," it is time to begin tackling BGP. This chapter covers the basic operation of BGP, including its message types, how the messages are used, and the format of the messages. You also learn about the various basic attributes BGP can associate with a route and how it uses these attributes to choose a best path. Finally, this chapter shows you how to configure and troubleshoot BGP peering sessions.

Who Needs BGP?

If you answer "yes" to all four of the following questions, you need BGP:

- Are you connecting to another routing domain?
- Are you connecting to a domain under a separate administrative authority?
- Is your domain multihomed?
- Is a routing policy required?

The answer to the first question—are you connecting to another routing domain?—is obvious; BGP is an inter-domain routing protocol. But as the subsequent sections explain, BGP is not the only means of routing between separate domains.

Connecting to Untrusted Domains

An underlying assumption of an IGP is that, by definition, its neighbors are all under the same administrative authority, and therefore the neighbors can be trusted: Trusted to not be malicious, trusted to be correctly configured, and trusted to not send bad route information. All these things can still happen occasionally within an IGP domain, but they are rare. An IGP is designed to freely exchange route information, focusing more on performance and easy configuration than on tight control of the information.

BGP, however, is designed to connect to neighbors in domains out of the control of its own administration. Those neighbors cannot be trusted, and the information you exchange with those neighbors is (if BGP is configured properly) carefully controlled with route policies.

But if connection to an external domain is your only requirement—particularly if there is only one connection—BGP is probably not called for. Static routes serve you better in this case; you don't have to worry about false information being exchanged because no information at all is being exchanged. Static routes are the ultimate means of controlling what packets are routed into and out of your network.

Figure 2-1 shows a subscriber attached by a single connection to an ISP. BGP, or any other type of routing protocol, is unnecessary in this topology. If the single link fails, no routing decision needs to be made because no alternative route exists. A routing protocol accomplishes nothing. In this topology, the subscriber adds a static default route to the border router and redistributes the route into his AS.

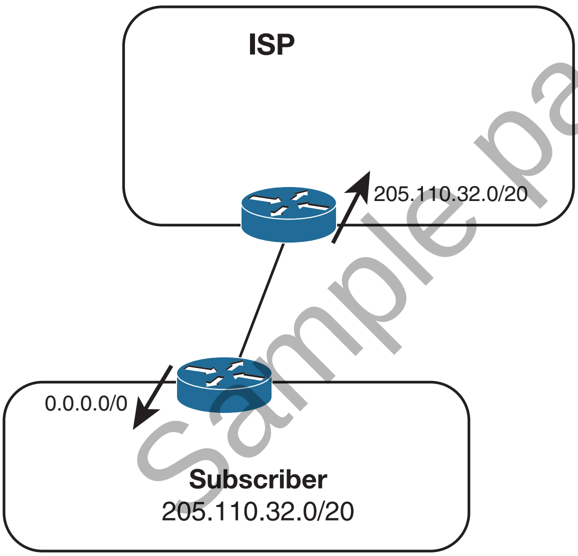

Figure 2-1 *Static Routes Are All That Is Needed in This Single-Homed Topology*

The ISP similarly adds a static route pointing to the subscriber's address range and advertises that route into its AS. Of course, if the subscriber's address space is a part of the ISP's larger address space, the route advertised by the ISP's router goes no farther than the ISP's own AS. "The rest of the world" can reach the subscriber by routing to the ISP's advertised address space, and the more-specific route to the subscriber can be picked up only within the ISP's AS.

An important principle to remember when working with inter-AS traffic is that each physical link actually represents two logical links: one for incoming traffic, and one for outgoing traffic, as shown in Figure 2-2.

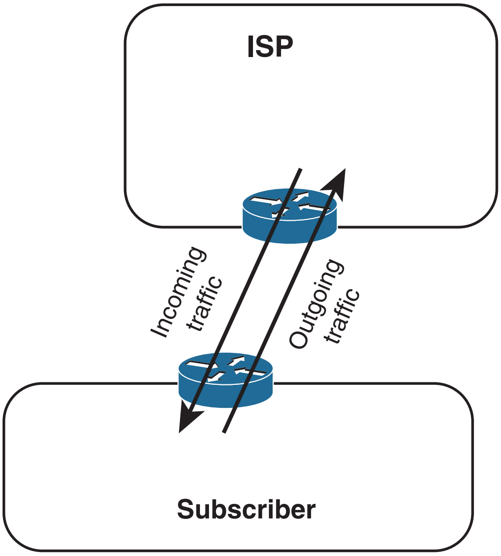

Figure 2-2 *Each Physical Link Between Autonomous Systems Represents Two Logical Links, Carrying Incoming and Outgoing Packets*

The routes you advertise in each direction influence the traffic separately. Avi Freedman, who has written many excellent articles on ISP issues, calls a route advertisement a promise to carry packets to the address space represented in the route. In Figure 2-1, the subscriber's router is advertising a default route into the local AS—a promise to deliver packets to any destination. And the ISP's router, advertising a route to 205.110.32.0/20, promises to deliver traffic to the subscriber's AS. The outgoing traffic from the subscriber's AS is the result of the default route, and the incoming traffic to the subscriber's AS is the result of the route advertised by the ISP's router. This concept may seem trivial and obvious at this point, but it is important to keep in mind as more complex topologies are examined and as we begin establishing policies for advertised and accepted routes.

The vulnerability of the topology in Figure 2-1 is that the entire connection consists of single points of failure. If the single data link fails, if a router or one of its interfaces fails, if the configuration of one of the routers fails, if a process within the router fails, or if one of the routers' all-too-human administrators makes a mistake, the subscriber's entire Internet connectivity can be lost. What is lacking in this picture is *redundancy*.

Connecting to Multiple External Neighbors

Figure 2-3 shows an improved topology, with redundant links to the same provider. How the incoming and outgoing traffic is manipulated across these links depends upon how the two links are used. For example, a frequent setup when multihoming to a single provider is for one of the links to be a primary, dedicated Internet access link and for the other link to be used only for backup.

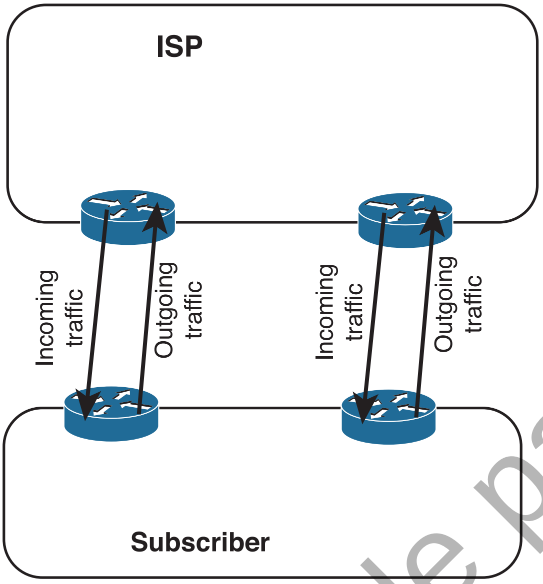

Figure 2-3 *When Multihoming You Must Consider the Incoming and Outgoing Advertisements and Resulting Traffic on Each Link*

When the redundant link is used only for backup, there is again no call for BGP. The routes can be advertised just as they were in the single-homed scenario, except that the routes associated with the backup link have the metrics set high so that they can be used only if the primary link fails.

Example 2-1 shows what the configurations of the routers carrying the primary and secondary links might look like.

Example 2-1 *Primary and Secondary Link Configurations for Multihoming to a Single Autonomous System*

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

```
ip route 0.0.0.0 0.0.0.0 205.110.168.108
```

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

In this configuration, the backup router has a default route whose administrative distance is set to 150 so that it will be only in the routing table if the default route from the primary router is unavailable. Also, the backup default is advertised with a higher metric than the primary default route to ensure that the other routers in the OSPF domain prefer the primary default route. The OSPF metric type of both routes is E2, so the advertised metrics remain the same throughout the OSPF domain. This ensures that the metric of the primary default route remains lower than the metric of the backup default route in every router, regardless of the internal cost to each border router. Example 2-2 shows the default routes in a router internal to the subscriber's OPSF domain.

Example 2-2 *The First Display Shows the Primary External Route; the Second Display Shows the Backup Route Being Used After the Primary Route Has Failed*

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

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

Although a primary/backup design satisfies the need for redundancy, it does not efficiently use the available bandwidth. A better design would be to use both paths, with each providing backup for the other if a link or router failure occurs. In this case, the configuration used in both routers is indicated in Example 2-3.

Example 2-3 *When Load Sharing to the Same AS, the Configuration of Both Routers Can Be the Same*

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

Note A key difference between building the simple peering of Figure 2-3 as a primary/backup configuration and as a load-sharing configuration is the consideration of bandwidth. If one link is a primary and one is a backup, the bandwidth of both links should be equal; if the primary fails, the load can be fully rerouted to the backup with no congestion. In some configurations, the backup link has considerably lower bandwidth, under the assumption that if the primary fails, the backup provides only enough bandwidth for critical applications to survive rather than full network functionality.

When a load-sharing configuration is used, the bandwidth of each of the two links should carry the total traffic load normally carried over both links. If one of the links fails, the other can then carry the full traffic load without packet loss.

The static routes in both routers have equal administrative distances, and the default routes are advertised with equal metrics (10). The default routes are now advertised with an OSPF metric type of E1. With this metric type, each of the routers in the OSPF domain takes into account the internal cost of the route to the border routers in addition to the cost of the default routes. As a result, every router chooses the closest exit point when choosing a default route, as shown by Figure 2-4.

In most cases advertising default routes into the AS from multiple exit points, and summarizing address space out of the AS at the same exit points, is sufficient for good internetwork performance. The one consideration is whether asymmetric traffic patterns will become a concern, as discussed in Chapter 1. If the geographical separation between the two (or more) exit points is large enough for delay variations to become significant, you might have a need for better control of the routing. BGP may now be a consideration.

For example, suppose the two exit routers in Figure 2-3 are located in Los Angeles and London. You might want all your exit traffic destined for the Eastern Hemisphere to use the London router, and all your exit traffic for the Western Hemisphere to use the Los Angeles router. Remember that the incoming route advertisements influence your outgoing traffic. If the provider advertises routes into your AS via BGP, your internal routers has more accurate information about external destinations.

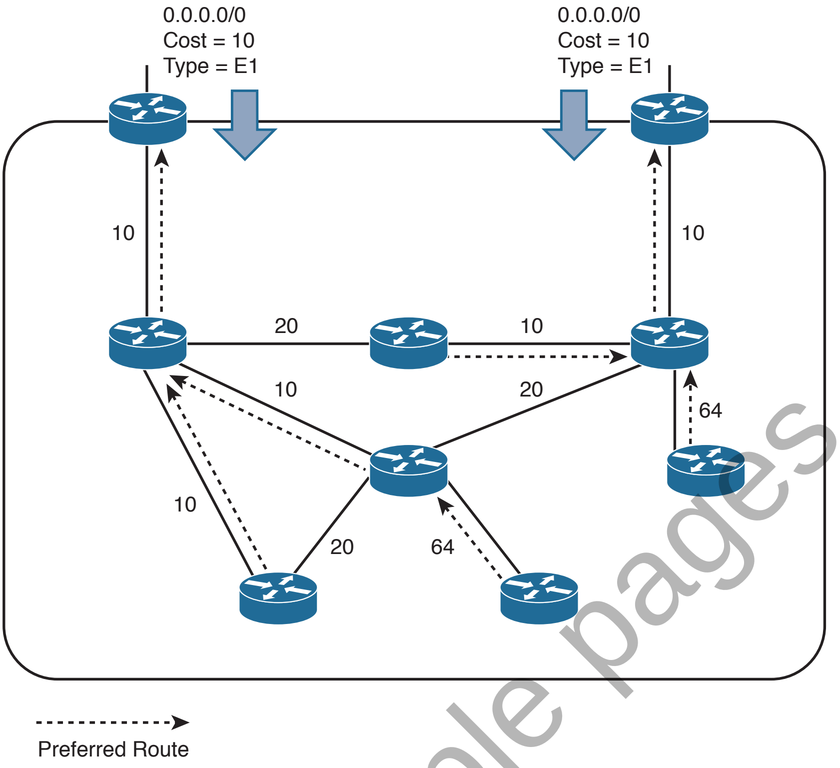

Figure 2-4 *The OSPF Border Routers Advertise a Default Route with a Metric of 10 and an OPSF Metric Type of E1*

Similarly, outgoing route advertisements influence your incoming traffic. If internal routes are advertised to the provider via BGP, you have influence over what routes are advertised at what exit point, and also tools for influencing (to some degree) the choices the provider makes when sending traffic into your AS.

When considering whether to use BGP, weigh the benefits gained against the cost of added routing complexity. BGP should be preferred over static routes only when an advantage in traffic control can be realized. Consider the incoming and outgoing traffic separately. If it is only important to control your incoming traffic, use BGP to advertise routes to your provider while still advertising only a default route into your AS.

However, if it is only important to control your outgoing traffic, use BGP just to receive routes from your provider. Consider the ramifications of accepting routes from your provider. "Taking full BGP routes" means that your provider advertises to you the entire Internet routing table. As of this writing, that is more than 500,000 IPv4 route entries, as shown in Example 2-4. The IPv6 Internet table is growing rapidly. You need a reasonably powerful router CPU to process the routes and enough router memory to store the entries. You also need sufficient TCAM or other forwarding plane memory to hold forwarding information. Example 2-4 shows that just the BGP routes require almost 155.7MB; the memory that BGP requires to process these routes, as shown in Example 2-5, is approximately 4.1GB. A simple default-routing scheme, however, can be implemented easily with a low-end router and a moderate amount of memory.

Example 2-4 *This Summary of the Full Internet Routing Table Shows 540,809 BGP Entries*[1]

```
route-views>show ip route summary
IP routing table name is default (0x0)
IP routing table maximum-paths is 32
Route Source    Networks    Subnets     Replicates  Overhead    Memory (bytes)
connected       0           2           0           192         576
static          1           57          0           5568        16704
application     0           0           0           0           0
bgp 6447        174172      366637      0           51917664    155752992
  External: 540809 Internal: 0 Local: 0
internal        7847                                            42922856
Total           182020      366696      0           51923424    198693128
route-views>
```

Example 2-5 *BGP Requires Approximately 4.1GB of Memory to Process the Routes Shown in Example 2-4*

```
route-views> show processes memory | include BGP
 117   0          0        232      41864        644        644 BGP Scheduler
 176   0 1505234352     262528     370120   14362638   14362638 BGP I/O
 299   0          0   10068312      41864          0          0 BGP Scanner
 314   0          0          0      29864          0          0 BGP HA SSO
 338   0 27589889144 2170064712 4102896864       3946       3946 BGP Router
 350   0          0          0      29864          0          0 XC BGP SIG RIB H
 383   0          0          0      41864          0          0 BGP Consistency
 415   0          0          0      41864          0          0 BGP Event
 445   0          0          0      29864          0          0 BGP VA
 450   0       3224          0      33160          1          0 BGP Open
 562   0     328104     262528     107440          0          0 BGP Task
 574   0       3248          0      33160          1          0 BGP Open
```


[1] This display was taken in 2014 from the public route server at University of Oregon (AS6447). The corresponding example in the first edition of this book, taken from an AT&T route server in 1999, showed 88,269 BGP entries.

```
 575    0        3120          0       33088          1           0 BGP Open
 577    0        3120          0       33040          1           0 BGP Open
 578    0        3120          0       33072          1           0 BGP Open
route-views>
```

Note The routing table summary in Example 2-4 and the related processes shown in Example 2-5 are taken from a route server at route-views.oregon-ix.net. By the time you read this chapter, the numbers shown in these two examples will have changed; telnet to the server, and see what they are now. There are a number of such publicly accessible route servers; for a good list, go to www.netdigix.com/servers.html.

Another consideration is that when running BGP, a subscriber's routing domain must be identified with an autonomous system (AS) number. Like IPv4 addresses, AS numbers are limited and are assigned only by the regional address registries when there is a justifiable need. And like IPv4 addresses, a range of AS numbers is reserved for private use: the AS numbers 64512 to 65534. As with private IPv4 addresses (RFC 1918), these AS numbers are not globally unique and must not be included in the AS_PATH of any route advertised into the public Internet. With few exceptions, subscribers that are connected to a single service provider (either single or multihomed) use an AS number out of the reserved range. The service provider filters the private AS number out of the advertised BGP path. Configuring and filtering private AS numbers is covered in Chapter 5, "Scaling BGP."

Although the topology in Figure 2-3 is an improvement over the topology in Figure 2-2 because redundant routers and data links have been added, it still entails a single point of failure. That point of failure is the ISP. If the ISP loses connectivity to the rest of the Internet, so does the subscriber. And if the ISP suffers a major internal outage, the single-homed subscriber also suffers.

Setting Routing Policy

Figure 2-5 shows a topology in which a subscriber has homed to more than one service provider. In addition to the advantages of multihoming already described, this subscriber is protected from losing Internet connectivity as the result of a single ISP failure. And with this topology BGP begins to become a better choice, in most cases, than static routes.

The subscriber in Figure 2-5 could still forego BGP. One option is to use one ISP as a primary Internet connection and the other as a backup only; another option is to default route to both providers and let the routing chips fall where they may. But if a subscriber has gone to the expense of multihoming and contracting with multiple providers, neither of these solutions is likely to be acceptable. BGP is the preferred option in this scenario.

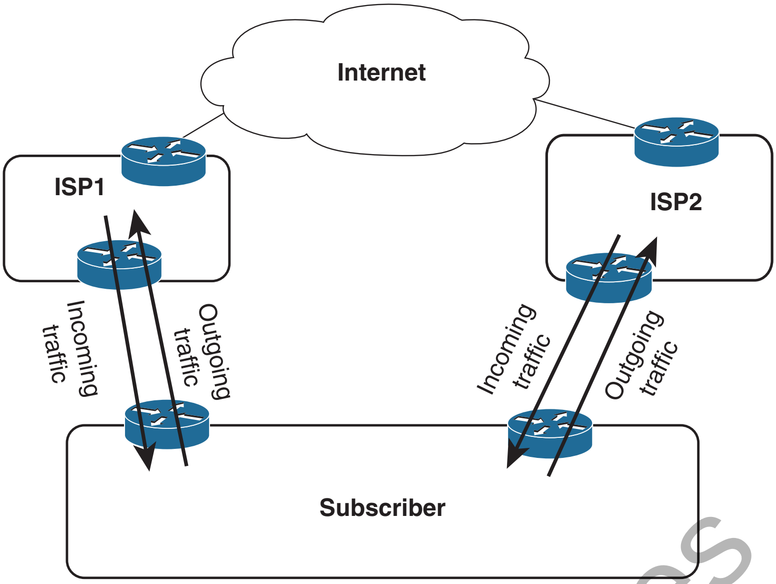

Figure 2-5 *Multihoming to Multiple Autonomous Systems*

Again, incoming and outgoing traffic should be considered separately. For incoming traffic, the most reliability is realized if all internal routes are advertised to both providers. This setup ensures that all destinations within the subscriber's AS are completely reachable via either ISP. Even though both providers are advertising the same routes, there are cases in which incoming traffic should prefer one path over another; such situations are discussed in the multihoming sections of Chapter 1. BGP provides the tools for communicating these preferences.

For outgoing traffic, the routes accepted from the providers should be carefully considered. If full routes are accepted from both providers, the best route for every Internet destination is chosen. In some cases, however, one provider might be preferred for full Internet connectivity, whereas the other provider is preferred for only some destinations. In this case, full routes can be taken from the preferred provider and partial routes can be taken from the other provider. For example, you might want to use the secondary provider only to reach its other subscribers and for backup to your primary Internet provider (see Figure 2-6). The secondary provider sends its customer routes, and the subscriber configures a default route to the secondary ISP to be used if the connection to the primary ISP fails.

The full routes sent by ISP1 probably include the customer routes of ISP2, learned from the Internet or perhaps from a direct peering connection. Because the same routes are received from ISP2, however, the subscriber's routers normally prefer the shorter path through ISP2. If the link to ISP2 fails, the subscriber uses the longer paths through ISP1 and the rest of the Internet to reach ISP2's customers.

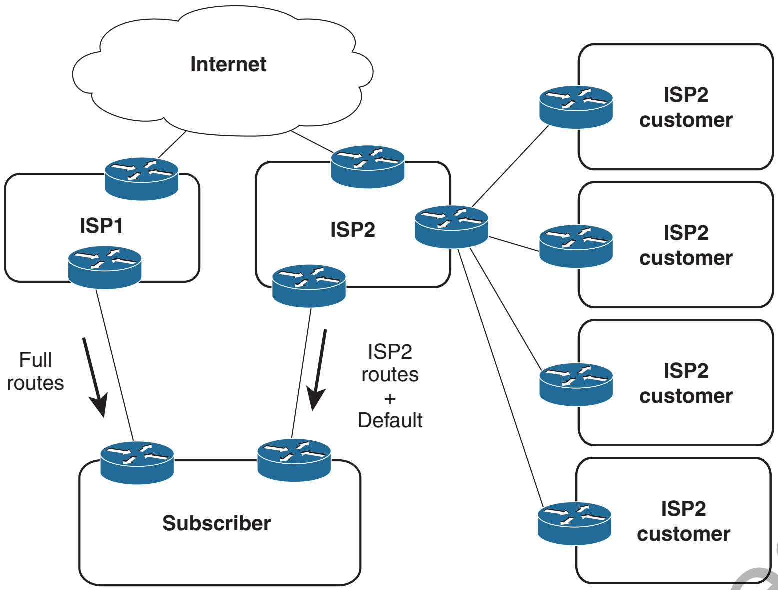

Figure 2-6 *ISP1 Is the Preferred Provider for Most Internet Connectivity; ISP2 Is Used Only to Reach Its Other Customers' Networks and for Backup Internet Connectivity*

Similarly, the subscriber normally uses ISP1 to reach all destinations other than ISP2's customers. If some or all of those more-specific routes from ISP1 are lost, however, the subscriber uses the default route through ISP2.

If router CPU and memory limitations prohibit taking full routes,[2] partial routes from both providers are an option. Each provider might send its own customer routes, and the subscriber points default routes to both providers. In this scenario, some routing accuracy is traded for a savings in router resources.

In yet another partial-routes scenario, each ISP might send its customer routes and also the customer routes of its upstream provider (which typically is a national or global backbone carrier such as Level 3 Communications, Sprint, NTT, or Deutsche Telekom). In Figure 2-7, for example, ISP1 is connected to Carrier1, and ISP2 is connected to Carrier2. The partial routes sent to the subscriber by ISP1 consist of all ISP1's customer routes and all Carrier1's customer routes. The partial routes sent by ISP2 consist of all ISP2's customer routes and all Carrier2's customer routes. The subscriber points to default routes at both providers. Because of the size of the two backbone carrier providers, the subscriber has enough routes to make efficient routing decisions on a large number of destinations. At the same time, the partial routes are still significantly smaller than a full Internet routing table.

[2] Taking full BGP routes from two sources doubles the number of BGP entries in all routers and consequently doubles the memory demand.

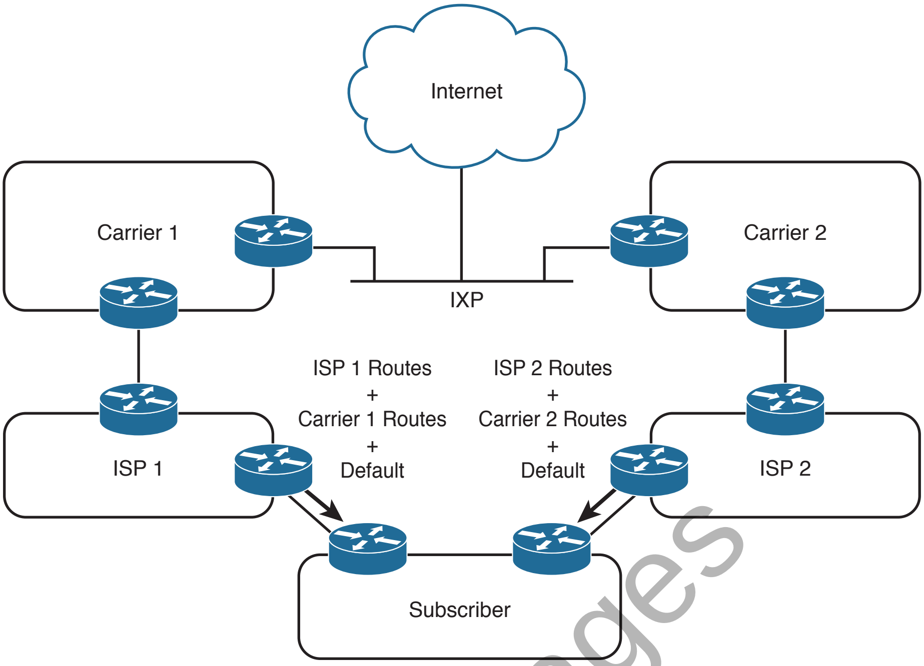

Figure 2-7 *The Subscriber Is Taking Partial Routes from Both ISPs, Consisting of All ISP's Customer Routes and the Customer Routes from Their Respective Upstream Providers*

All the examples here have shown a stub AS connected to one or more ISPs. Figures 2-5 through 2-7 begin introducing enough complexity that BGP and routing policy are probably called for. As the complexity of multihoming and its related policy issues grow, as illustrated in the transit AS examples in the previous chapter, the need for BGP becomes increasingly sure.

BGP Hazards

Creating a BGP peering relationship involves an interesting combination of trust and mistrust. The BGP peer is in another AS, so you must trust the network administrator on that end to know what she is doing. At the same time, if you are smart, you will take every practical measure to protect yourself if a mistake is made on the other end. When you implement a BGP peering connection, paranoia is your friend.

At the same time, you should be a good neighbor by taking practical measures to ensure that a mistake in your AS does not affect your BGP peers.

Recall the earlier description of a route advertisement as a promise to deliver packets to the advertised destination. The routes you advertise directly influence the packets you receive, and the routes you receive directly influence the packets you transmit. In a good

BGP peering arrangement, both parties should have a complete understanding of what routes are to be advertised in each direction. Again, incoming and outgoing traffic must be considered separately. Each peer should ensure that he is transmitting only the correct routes and should use route filters or other policy tools such as AS_PATH filters, described in Chapter 4, “BGP and Routing Policies,” to ensure that he receives only the correct routes.

Your ISP might show little patience with you if you make mistakes in your BGP configuration, but the worst problems can be attributed to a failure on both sides of the peering arrangement. Suppose, for example, that through some misconfiguration you advertise 207.46.0.0/16 to your ISP. On the receiving side, the ISP does not filter out this incorrect route, allowing it to be advertised to the rest of the Internet. This particular CIDR block belongs to Microsoft, and you have just claimed to have a route to that destination. A significant portion of the Internet community could decide that the best path to Microsoft is through your domain. You will receive a flood of unwanted packets across your Internet connection and, more important, you will have black-holed traffic that should have gone to Microsoft. It will be neither amused nor understanding.

This kind of thing happens frequently: Not long ago, Yahoo experienced a brief outage due to a company in Seoul mistakenly advertising a /14 prefix that included addresses belonging to Yahoo.

Figure 2-8 shows another example of a BGP routing mistake. This same internetwork was shown in Figure 2-6, but here the customer routes that the subscriber learned from ISP2 have been inadvertently advertised to ISP1.

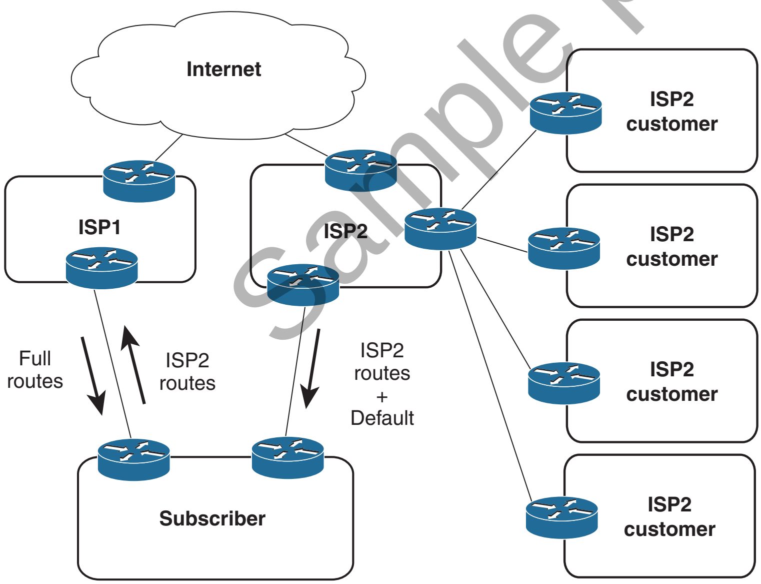

Figure 2-8 *This Subscriber Is Advertising Routes Learned from ISP2 into ISP1, Inviting Packets Destined for ISP2 and Its Customers to Transit His Domain*

Unless ISP1 and ISP2 have a direct peering connection, ISP1 and its customers probably see the subscriber's domain as the best path to ISP2 and its customers. In this case, the traffic is not black-holed because the subscriber does indeed have a route to ISP2. The subscriber has become a transit domain for packets from ISP1 to ISP2, to the detriment of its own traffic. And because the routes from ISP2 to ISP1 still point through the Internet, the subscriber has caused asymmetric routing for ISP2.

The point of this section is that BGP, by its nature, is designed to allow communication between autonomously controlled systems. A successful and reliable BGP peering arrangement requires an in-depth understanding of not only the routes to be advertised in each direction, but also the routing policies of each of the involved parties.

The remainder of this chapter introduces the technical basics of BGP and demonstrates how to configure and troubleshoot simple BGP sessions. With that foundation experience, you then get a good taste of configuring and troubleshooting policies in Chapter 4.

Operation of BGP

The section "BGP Basics" in Chapter 1 introduced you to the fundamental facts about BGP. To recap

- Unique among the common IP routing protocols, BGP sends only unicast messages and forms a separate point-to-point connection with each of its peers.
- BGP is an application layer protocol using TCP (port 179) for this point-to-point connection and relies on the inherent properties of TCP for session maintenance functions such as acknowledgment, retransmission, and sequencing.
- BGP is a vector protocol, although called a path vector rather than distance vector because it sees the route to a destination as a path through a series of autonomous systems rather than as a series of routers hops.
- A BGP route describes the path vector using a route attribute called the AS_PATH, which sequentially lists the autonomous system numbers comprising the path to the destination.
- The AS_PATH attribute is a shortest path determinant. Given multiple routes to the same destination, the route with an AS_PATH listing the fewest AS numbers is assumed to be the shortest path.
- The AS numbers on the AS_PATH list are used for loop detection; a router receiving a BGP route with its own AS number in the AS_PATH assumes a loop and discards the route.
- If a router has a BGP session to a neighbor with a different AS number, the session is called *external BGP (EBGP)*; if the neighbor has the same AS number as the router, the session is called *internal BGP (IBGP)*. The neighbors are called, respectively, *external* or *internal* neighbors.

This chapter builds on these basic facts to describe the operation of BGP.

BGP Message Types

Before establishing a BGP peer connection, the two neighbors must perform the standard TCP three-way handshake and open a TCP connection to port 179. TCP provides the fragmentation, retransmission, acknowledgment, and sequencing functions necessary for a reliable connection, relieving BGP of those duties. All BGP messages are unicast to the one neighbor over the TCP connection.

BGP uses four basic message types:

- Open
- Keepalive
- Update
- Notification

Note There is a fifth BGP message type: Route Refresh. But unlike the four presented here, this fifth message type is not a part of basic BGP functionality and might not be supported by all BGP routers. The Route Refresh message and its use are described in Chapter 4.

This section describes how these messages are used; for a complete description of the message formats and the variables of each message field, see the section "BGP Message Formats."

Open Message

After the TCP session is established, both neighbors send Open messages. Each neighbor uses this message to identify itself and to specify its BGP operational parameters. The Open message includes the following information:

- **BGP version number:** This specifies the version (2, 3, or 4) of BGP that the originator is running; the IOS default is BGP-4. Prior to IOS 12.0(6)T, IOS would autonegotiate the version: If a neighbor is running an earlier version of BGP, it rejects the Open message specifying version 4; the BGP-4 router then changes to BGP-3 and sends another Open message specifying this version. If the neighbor rejects that message, an Open specifying version 2 is sent. BGP-4 has now become so prevalent that as of 12.0(6)T IOS no longer autonegotiates, but you can still configure a session to speak to a neighbor running version 2 or 3 with **neighbor version.**
- **Autonomous system number:** This is the AS number of the originating router. It determines whether the BGP session is EBGP (if the AS numbers of the neighbors differ) or IBGP (if the AS numbers are the same).