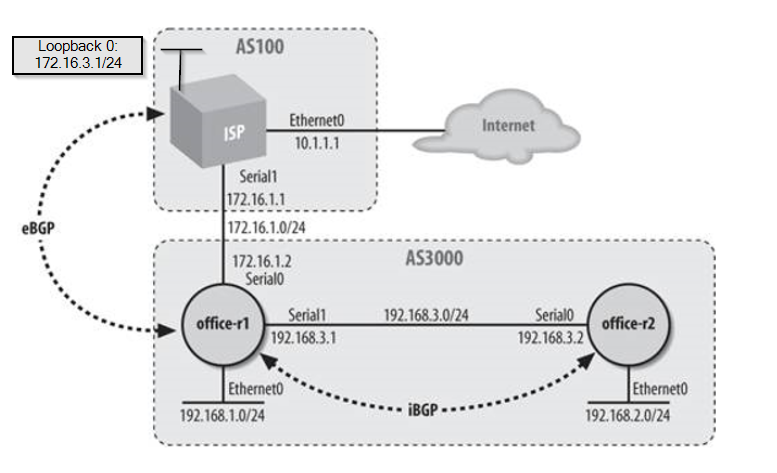
# **Simple iBGP and eBGP Network**

In this lab we'll look at a simple BGP configuration that includes both eBGP and iBGP configurations. A realistic example would be much more complex (particularly for the ISP), but this will help you see how things work. There are two office routers (office-r1 and office-r2); office-r1 connects to the Internet via an ISP, whose router is named (logically enough) "ISP".

#**Note:** this lab is not compatible with Packet Tracer. This lab should be built on physical hardware or in a virtualized environment like CML.#



All the basic configuration is up to you. First build the topology and configure the basic settings like hostnames and interfaces. When the basic settings are complete, proceed with the configuration details below.   
Essential to the lab are the *show* commands near the end of the lab.

Here's the BGP configuration for office-r1:

Configure BGP for our local-AS 3000

router bgp 3000

The networks we want to advertise

network 192.168.1.0

network 192.168.3.0

Our eBGP peers

neighbor 172.16.1.1 remote-as 100

For our iBGP peers, we'll set us as the default-originate

And we'll set us as the next hop using the next-hop-self command

neighbor 192.168.3.2 remote-as 3000

neighbor 192.168.3.2 next-hop-self

neighbor 192.168.3.2 default-originate

Our iBGP peers expect us to be the default route, so we need a local default route

ip route 0.0.0.0 0.0.0.0 172.16.1.1

The BGP configuration for office-r2:

router bgp 3000

Only one network to define

network 192.168.2.0

Only one neighbor to define

neighbor 192.168.3.1 remote-as 3000

The BGP configuration for ISP:

If this were a real ISP configuration, we would be fired!

But it shows the concepts :-)

router bgp 100

network 172.16.3.0 mask 255.255.255.0

neighbor 10.1.1.2 remote-as 200

neighbor 172.16.1.2 remote-as 3000

To demonstrate some of the BGP show commands, let's look at the office-r2 router. show ip route gives us a quick look at what's going on:

office-r2#**show ip route**

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, \* - candidate default

U - per-user static route, o - ODR

Gateway of last resort is 192.168.3.1 to network 0.0.0.0

B\* 0.0.0.0/0 [200/0] via 192.168.3.1, 01:05:08

172.16.0.0/24 is subnetted, 1 subnets

B 172.16.3.0 [200/0] via 192.168.3.1, 00:04:23

B 192.168.1.0/24 [200/0] via 192.168.3.1, 01:05:08

192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.2.0/24 is directly connected, FastEthernet1/0

L 192.168.2.1/32 is directly connected, FastEthernet1/0

192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.3.0/24 is directly connected, FastEthernet1/1

L 192.168.3.2/32 is directly connected, FastEthernet1/1

Everything here should be familiar. The gateway of last resort is set, because we have default-originate set on the office-r1 router (192.168.3.1). Note that the route for 172.16.3.0/24 is via 192.168.3.1. This route is set to office-r1's interface, because we used the next-hop-self option in one of the neighbor commands for 192.168.3.2 on office-r1. Therefore, office-r1 rewrote the BGP route for 172.16.3.0, making itself the next hop. If we hadn't put that command in, the route would have looked like this:

B 172.16.3.0/24 [200/0] via 172.16.1.1, 00:00:17

In this configuration, this route would work as well as the route to 192.168.3.2 because the default route tells our router how to get to that address. If we didn't have the default route, we would have to add an extra network statement, defining 172.16.3.0, to office-r1's configuration. next-hop-self makes the configuration a little easier.

Next, let's look at the output of show ip bgp on office-r2:

Office-r2#**show ip bgp**

BGP table version is 7, local router ID is 192.168.3.2

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal

Origin codes: i - IGP, e - EGP, ? - incomplete

Network Next Hop Metric LocPrf Weight Path

\*>i0.0.0.0 192.168.3.1 0 100 0 i

\*>i172.16.3.0/24 192.168.3.1 0 100 0 100 i

\*>i192.168.1.0 192.168.3.1 0 100 0 i

\*> 192.168.2.0 0.0.0.0 0 32768 i

r>i192.168.3.0 192.168.3.1 0 100 0 i

The output from this show command gives us a lot of useful information. The left-hand side lists the known networks with different codes, indicating the route's status. > indicates the best route to the given network. Then we have the next-hop address , the metric, the local preference (LocPrf), the weight, and finally the AS path.

| Table 10-1. Route status codes | |
| --- | --- |
| **Key** | **Route status** |
| s | Suppressed |
| d | Damped |
| \* | Valid |
| h | History |
| > | Best |
| i | Internal |

The Path column is particularly important. Most of the entries in this column have a path of i, which means that the route was learned through an interior protocol and therefore doesn't cross autonomous system boundaries. The only exception is the 172.16.3.0 network, which is in another autonomous system (AS 100). For this route to reach office-r1, BGP must learn the route from some sort of interior protocol. Therefore, the path for this network is 100 i. AS paths can obviously be much more complex. For a slightly more complex example, imagine that network 172.30.0.0 is attached to the ISP router and has an AS number of 200. The route might look like this:

Office-r2#**show ip bgp**

...

\*>i172.30.0.0 192.168.3.1 100 0 100 200 i

...

This path shows that to reach 172.30.0.0, you must cross AS 100, then enter AS 200, which learned the route through an interior protocol such as RIP. Therefore, you don't need to cross any more AS boundaries.