Bidirectional PIM Deployment Guide

Last updated: February 2008

Deploying Bidirectional PIM for Many-to-Many Applications

This document attempts to provide self-standing guidelines on the deployment of bidirectional Protocol Independent Multicast (PIM), and includes an introduction to the protocol, design and configuration guidelines for a successful deployment, and some case studies of implementations.

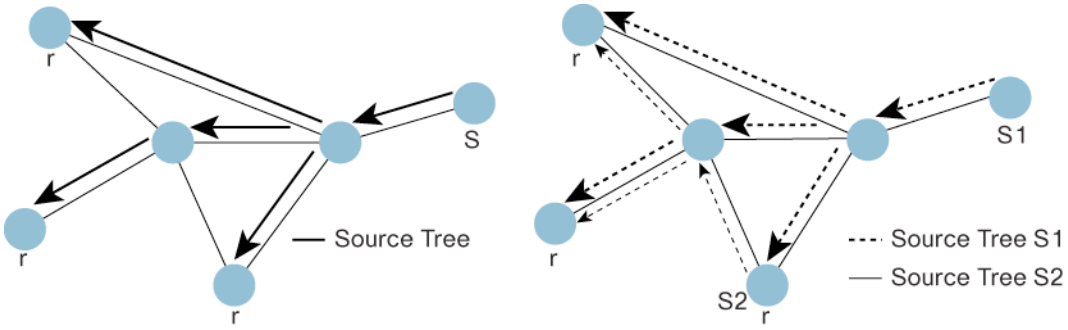
Bidirectional PIM

Bidirectional PIM is a member of the suite of multicast routing protocols supported in Cisco IOS® Software. The family of PIM protocols includes dense-mode, sparse-mode, source specific multicast (SSM), and bidirectional (Bidir) PIM. The initial set of protocols only included dense-mode and sparse-mode, but after a few years of deployment experience, the protocols have evolved and been optimized to better support the emerging multicast applications. Mainly, SSM was developed to easily support one-to-many applications by greatly simplifying the protocol mechanics for deployment ease. On the other hand, Bidir PIM was developed to help deploy emerging communication and financial applications that rely on a many-to-many applications model. Bidir PIM enables these applications by allowing them to easily scale to a very large number of groups and sources by eliminating the maintenance of source state. As you can see, these two protocols, SSM and Bidir, serve both ends of the spectrum of multicast applications: one-to-many and many-to-many. For this reason, neither one is a replacement for sparse-mode PIM, which is able to support the full spectrum of multicast applications but with a bit more complexity and overhead. Given the characteristics of the different variations of the protocols, they are meant to work side by side in the network. The network operator is now able to simultaneously deploy SSM for corporate communication applications, Bidir for “hoot-n-holler” and financial applications, and sparse-mode PIM for general IP Multicast connectivity. Later in this document, we will discuss how the IP Multicast address range is utilized to support concurrent deployment of the various PIM modes.

Bidir PIM Model

The traditional PIM protocols (dense-mode and sparse-mode) provided two models for forwarding multicast packets, source trees, and shared trees. Source trees are rooted at the source of the traffic while shared trees are rooted at the rendezvous point. Each model has its own set of characteristics and can be optimized for different types of applications. The source tree model provides optimum routing in the network, while shared trees provide a more scalable solution.

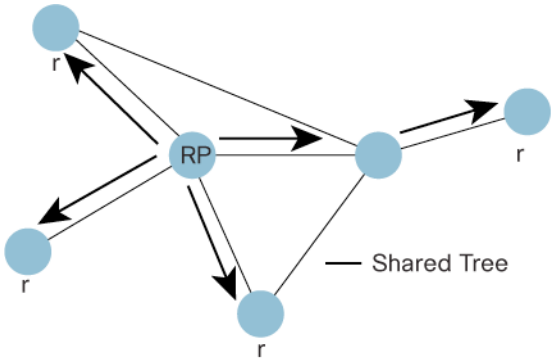
Figure 1.



Source trees (Figure 1) achieve the optimum path between each receiver and the source at the expense of additional routing information: an (S,G) routing entry per source in the multicast routing table. This is acceptable in applications with a limited number of sources. Applications like live broadcasts and distance learning are some examples where only one or a few sources are active. On the other hand, the shared tree provides a single distribution tree for all of the active sources. This means that traffic from different sources traverse the same distribution tree to reach the interested receivers, therefore reducing the amount of routing state in the network. This shared tree needs to be rooted somewhere, and the location of this root is the rendezvous point. In sparse-mode PIM the shared tree was utilized as an intermediate step while the protocol built the more efficient source trees. However, Bidir PIM will use these shared trees as their main forwarding mechanism.

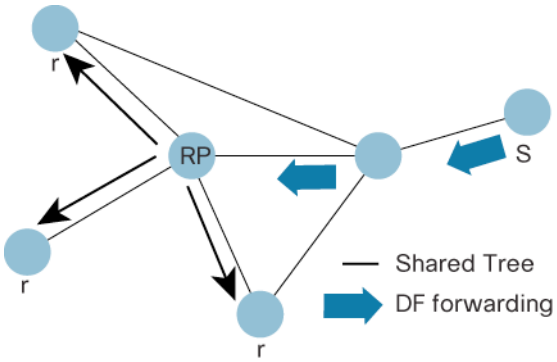
Shared trees only really explain the distribution of multicast data from the rendezvous point (the root of the tree) to the receivers (Figure 2). But this is only half the problem, there still needs to be a mechanism for the sources of the traffic to reach the rendezvous point. In sparse-mode this is solved by using the registration process. This process allows the routers with directly connected sources to communicate with the rendezvous point and inform it of the active sources. This would lead to the creation of a source tree between the first hop router and the rendezvous point and subsequent native multicast communication between sources and receivers. But we mentioned earlier that Bidir would not use source trees to better scale its routing table, so a new mechanism is required for Bidir sources to reach the rendezvous point. This new mechanism is called the designated forwarder (DF).

Figure 2.



Designated Forwarder

Figure 3.



Because Bidir PIM uses only shared trees for traffic distribution, this protocol needs a mechanism to get traffic from the sources to the rendezvous point. The designated forwarder provides that mechanism (Figure 3). The main responsibility of the designated forwarder is to decide what packets need to be forwarded upstream toward the rendezvous point. In the cases where the sources are also receivers, traffic originating from that host will be traveling against the direction of the shared tree (as shown in Figure 4(a)). This breaks the original assumption that shared trees only accept traffic on their Reverse Path Forwarding (RPF) interface to the rendezvous point. The same shared tree is now used to distribute traffic from the rendezvous point to receivers and from the sources to the rendezvous point, resulting in a bidirectional branch (Figure 4(b)). This assumes that for sources that are also receivers, the upstream traffic (from source to rendezvous point) will follow the same network path as the downstream traffic (from rendezvous point to receiver). In the ultimate case where all the hosts are sources and receivers, as is the case with many-to-many applications, the whole distribution tree becomes a bidirectional tree (Figure 5). The algorithm to elect the designated forwarder is straightforward, all the PIM neighbors in a subnet advertise their unicast route to the rendezvous point and the router with the best route is elected. This effectively builds a shortest path between every subnet and the rendezvous point without consuming any multicast routing state (no (S,G) entries are generated).

Figure 4.

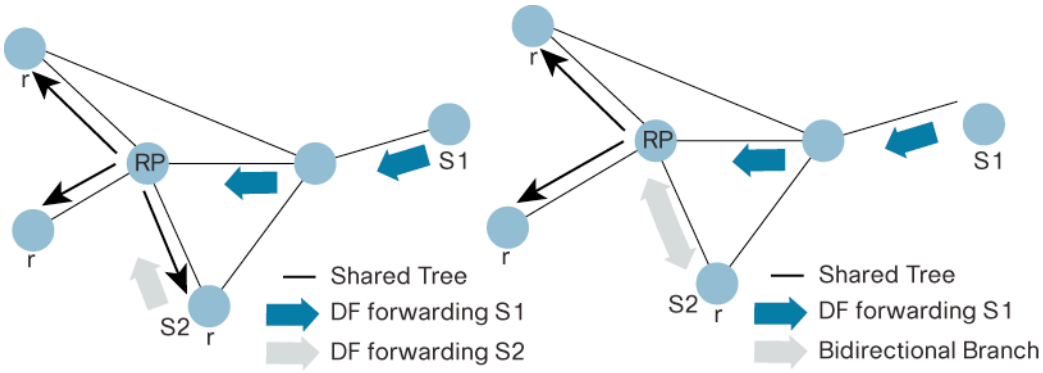
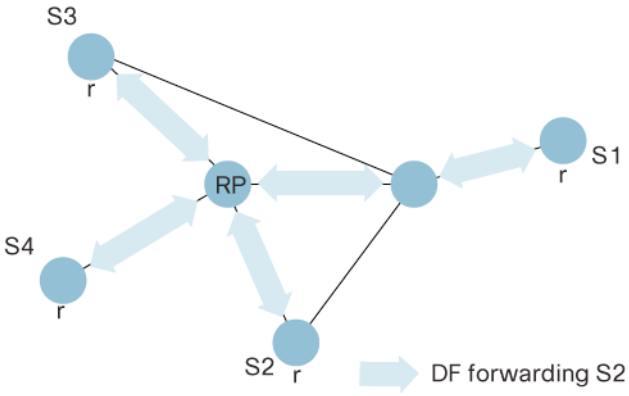


Figure 5.



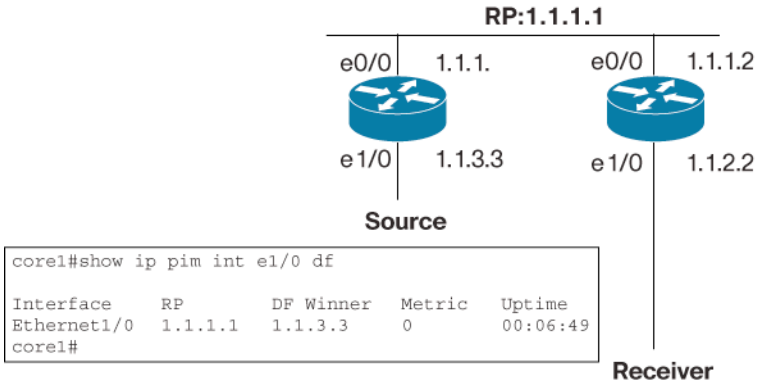
The designated forwarder election mechanism expects all of the PIM neighbors to be Bidir enabled. In the case where one of more of the neighbors is not a Bidir capable router the election fails and Bidir is disabled in that subnet. This can be a problem when incorporating Bidir in an existing multicast network because all the routers might not be upgraded simultaneously. In this case there is the [no] ip pim bidir-neighbor-filter command that allows you to explicitly specify the list of neighbors that should participate in the designated forwarder election process.

Bidir Rendezvous Point

Because Bidir PIM relies exclusively on shared trees to distribute multicast traffic, a rendezvous point is required. Remember that all shared trees are rooted at a rendezvous point, so the concept of the rendezvous point in sparse-mode is also used here. However, in sparse-mode the rendezvous point had some special functions that it needed to perform. Mainly it needed to handle the registration process and the creation of source trees between sources and the rendezvous point. Neither of these two functions exists in Bidir so the concept of the rendezvous point is a little different. The rendezvous point still serves the function of getting sources and receivers to learn about each other, but in the case of Bidir it can be thought of as a vector. That means that the rendezvous point address doesn't need to reside on a physical router interface but can just be an address in a subnet (it needs to be a routable address). This is not to say that the rendezvous point can't be a physical router, it just means that is no longer a requirement. Figure 6 illustrates the vector idea. In this case the rendezvous point address is just an IP address

(1.1.1.1) on a subnet that is not assigned to any physical interface, so in a way it's just acting as a destination vector. Traffic from the source is going to be forwarded hop by hop toward that destination (the rendezvous point) by the designated forwarder mechanism, while joins from the receivers will be sent to the rendezvous point. Let's start at the receiver end. Receivers, as usual, will send IGMP reports which will trigger a (*,G) join from the router toward the rendezvous point, creating a shared tree from the rendezvous point to the receiver as shown on the show ip mroute output. Note that in the "mroute" entry the incoming interface (or Bidir-Upstream interface) is Ethernet0/0 which is the RPF interface to the rendezvous point address; that means that any traffic received on Ethernet0/0 will be forwarded downstream through the shared tree. Moving to the source side of the topology, the state for the left router shows that it is the designated forwarder on Ethernet1/0. That means that whenever it receives traffic for a Bidir group on that interface it will forward it "upstream" toward the rendezvous point, out Ethernet 0/0. When the packets reach that network, the connectivity from the source to the receiver is achieved. So, in a way it is the subnet (1.1.1.0/24) and not a physical router that is acting as the rendezvous point.

Figure 6.



Deploying Bidir PIM

Configuring the Bidir Rendezvous Point

Deploying Bidir PIM in an existing multicast network that supports Bidir PIM is as simple as configuring a Bidir rendezvous point:

```
ip pim rp-address 1.1.1.1 <acl> bidir
```

where the “acl” specifies the range of group addresses that are to be treated as Bidir groups.

The above command is for static configuration of the rendezvous point. Auto-RP can be used to distribute the Bidir rendezvous point information in a similar fashion:

```
ip pim send-rp-announce 1.1.1.1 scope 32 group-list bidir-groups bidir
ip pim send-rp-discovery Loopback0 scope 32
```

where “bidir-groups” is a named access-list specifying which groups should be handled in Bidir mode. The RP address of 1.1.1.1 does not have to be a physical address on an interface but does need to belong to a directly connected subnet. The IP address option in the send-rp-announce command was added on the Catalyst 6500/7600 in Release12.2(18)SXF4.

Through this configuration mechanism it is possible to have dense-mode (for Auto-RP only), sparse-mode, SSM, and Bidir all running simultaneously in your network for different group ranges with a configuration like:

```
ip pim bidir-enable
ip pim rp-address 1.1.1.1 bidir-groups bidir
ip pim rp-address 1.1.1.10 sparse-groups
ip pim ssm default
ip access-list standard bidir-groups
permit 239.235.0.0 0.0.255.255
```

```
ip access-list standard sparse-groups
deny 224.0.1.39
deny 224.0.1.40
deny 239.235.0.0 0.0.255.255
permit 224.0.0.0 15.255.255.255
```

This example carves out the group ranges into nonoverlapping ranges, but the multicast code will perform a longest match lookup on group address to choose the rendezvous point in case of an overlap in the ranges.

Bidir Neighbors and Designated Forwarder Election

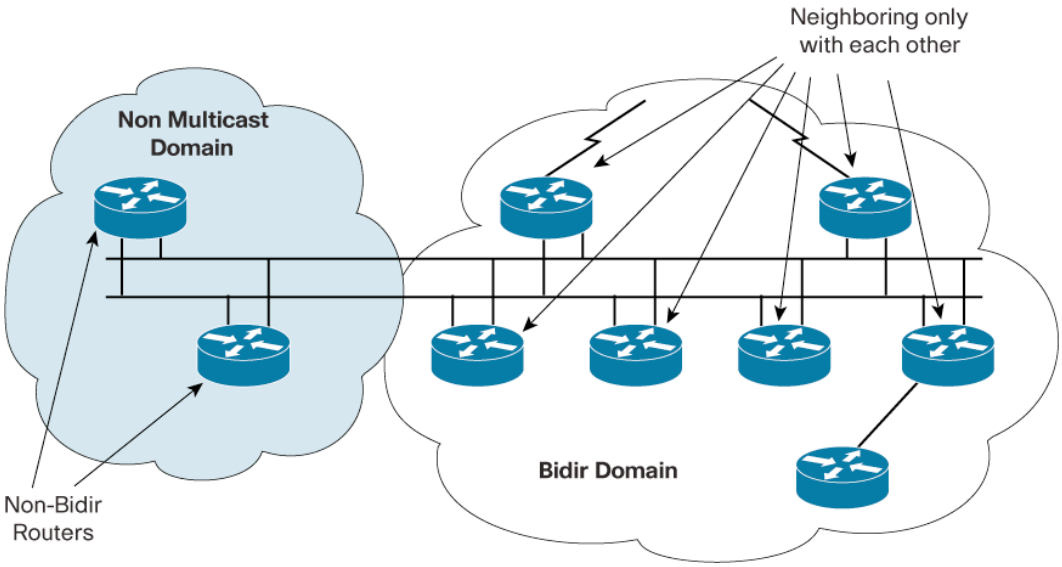
Bidir is enabled by default on software that supports the feature (find a recommended release), but if it has been previously disabled it can be re-enabled using the [no] ip pim bidir enable command. This is necessary for the routers to perform the designated forwarder election successfully. If a router detects through “PIM Hello” messages that one of its PIM neighbors is not Bidir capable, the designated forwarder election process is aborted and no Bidir traffic would be forwarded tos/from that interface. This protects the network from misconfigurations and possible loops. However, there are cases where a partial Bidir deployment is desired (one example is when not all routers support Bidir day one and the application is restricted to a certain part of the network).

Partial Deployments

There are two main cases where partial deployments are commonly seen. In the first case (Figure 7), a new Bidir application is deployed in a nonmulticast network. In this case the following per-interface neighbor-filter command is necessary on all the Bidir routers to specify all the other Bidir neighbors participating on the designated forwarder election:

```
[no] ip pim bidir-neighbor-filter <acl>
```

Figure 7.



where “acl” lists the IP addresses of all the Bidir neighbors off that interface. This is an interface configuration command and it needs to be applied to all the interfaces that have non-Bidir neighbors. The other case is where Bidir is partially deployed in an existing multicast (sparse-

mode) network. Figure 8 shows a likely scenario where a Bidir cloud overlaps an existing sparse-mode cloud. In this case we still want to maintain sparse-mode connectivity in the Bidir cloud, while adding the new Bidir capabilities. Here, the same approach needs to be taken for subnets that have non-Bidir networks as previously described with the [no] ip pim bidir-neighbor-filter command. Additionally, a boundary needs to be drawn between the sparse-mode and sparse-mode/Bidir clouds to isolate the Bidir groups and prevent traffic to those groups from entering or leaving the Bidir area. This is achieved by configuring the following command on all of the boundary interfaces:

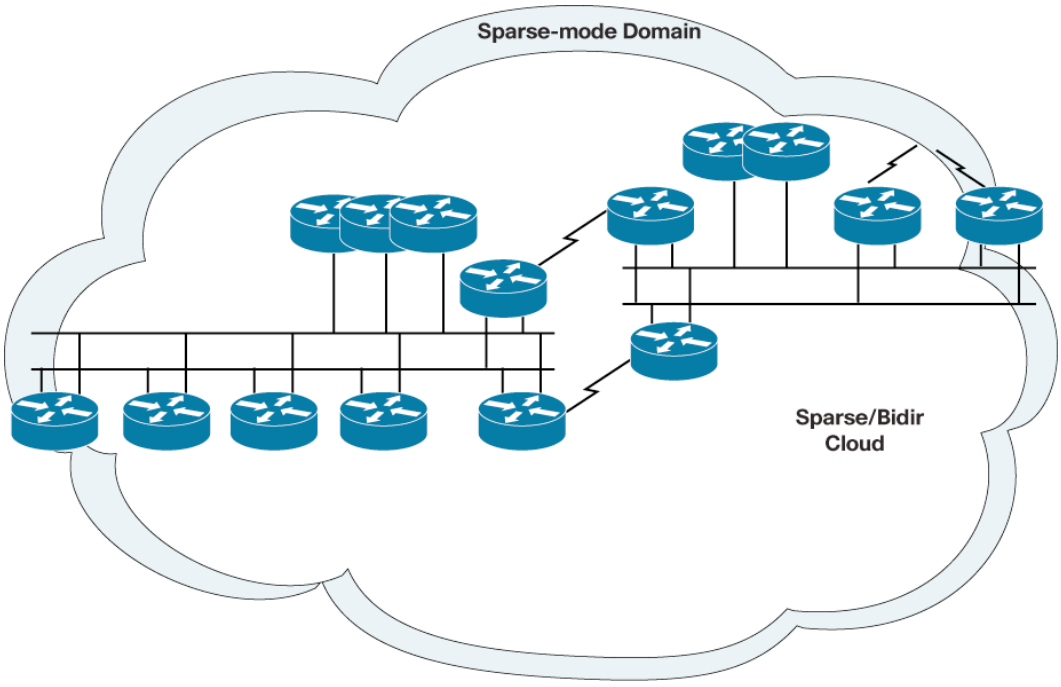
```
ip multicast boundary <bidir-groups-acl>
```

Now that the boundaries and the Bidir neighbors have been defined, the Bidir groups need to be enabled. For these partial deployments it is suggested that the Bidir rendezvous points be statically configured on all the Bidir-enabled routers with the command:

```
ip pim rp-address <ip-address> <bidir-groups-acl> bidir override
```

Note the “override” keyword; it is important to include this keyword when Auto-RP is used to distribute rendezvous point information in the sparse-mode domain. One of the reasons to statically configure the Bidir rendezvous point in partial deployments is that non-Bidir routers that receive an Auto-RP announcement for a Bidir rendezvous point will interpret that announcement as one for a sparse-mode rendezvous point. Furthermore, this configuration results in the sparse-mode cloud treating the Bidir groups as sparse-mode (with the rendezvous point as the sparse-mode rendezvous point). This is not a problem as the boundaries prevent both control and data traffic from those groups to interfere with each other. Effectively, the Bidir groups are running in Bidir mode inside the Bidir cloud, and in sparse-mode inside the sparse-mode cloud in a “ships in the night” fashion. Ideally we would have liked to disable the Bidir groups in the sparse-mode only cloud, but currently this would result in the groups defaulting to dense-mode (which must be avoided).

Figure 8.



Monitoring Bidir Traffic

Although Bidir PIM helps to greatly scale the multicast routing table, it does so by doing away with (S,G) source state. Some operators may rely on this information for monitoring and troubleshooting the network, but this information is no longer available from the multicast routing table. For Bidir, Multicast NetFlow can be use to monitor each individual flow, the equivalent of an (S,G) entry. Multicast NetFlow allows the operators to monitor and troubleshoot active Bidir sources in the network. Multicast NetFlow is supported in NetFlow v9 by just enabling netflow accounting.

Rendezvous Point Redundancy

While work on a general and simple mechanism for providing rendezvous point redundancy in Bidir is finalized, some topology dependent mechanisms are currently available. Because multicast source information is no longer available in Bidir, the Anycast/MSDP mechanism used to provide redundancy in sparse-mode is not an option for Bidir. That mechanism relied on the announcement of active sources through MSDP, which is not available in Bidir. Following is a set of solutions for providing rendezvous point redundancy for Bidir with our existing implementation. Keep in mind that the redundancy paradigm is different than that for sparse-mode. In Bidir, redundancy consists of a primary/ secondary model, there is no load sharing as was possible with the Anycast-rendezvous point sparse-mode case (it's impossible to load share without maintaining source information).

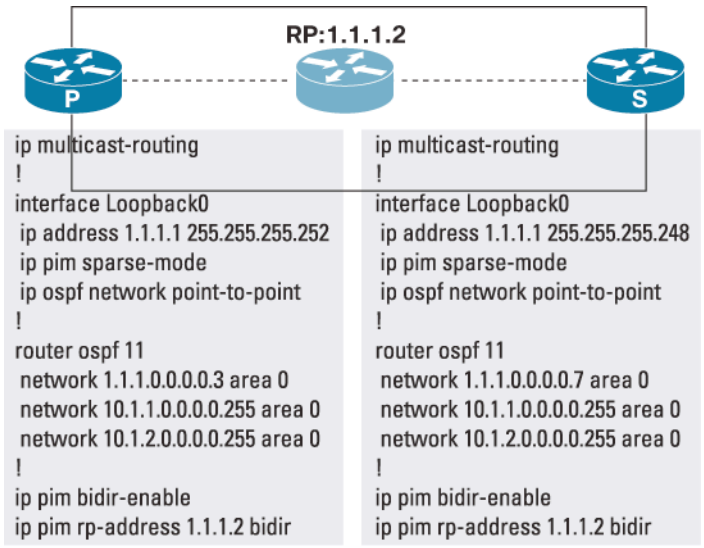
Redundant Phantom Rendezvous Point—Physical Interface

This is the simplest mechanism, but relies on placing the rendezvous point in a highly available network, like a collapsed Layer 2 backbone. As previously explained (see Figure 6), the phantom rendezvous point simply uses an IP address belonging to a particular subnet, but one that is not associated with any physical interface. This ensures that the rendezvous point address can be reached as long as one of the physical routers connected to that subnet remains up. The availability of the rendezvous point relies on the redundancy of the lower layers.

Redundant Phantom Rendezvous Point—Longest Match

The preferred method for providing Bidir rendezvous point redundancy can be designed using loopback networks with different mask length. This mechanism relies on unicast routing longest match route lookups to guarantee a consistent path to the rendezvous point. In this case the rendezvous point address is still a "phantom" address (that is, it is not associated with any physical entity). It is only necessary to ensure that a route to the rendezvous point exists, to maintain rendezvous point reachability. For this we will employ loopback interfaces in the primary and secondary routers with different netmask lengths. The way the primary/secondary relationship works in this situation is by advertising routes for the rendezvous point with different netmasks. Here we rely on unicast routing longest match algorithms to always pick the primary over the secondary. The primary router, by announcing a longest match route (that is, a /30 route for the rendezvous point address) will always be preferred over the less specific route being announced by the secondary router (that is, a /29 for the same rendezvous point address). Figure 9 shows an example of how this can be configured. In this example the primary router is advertising the /30 route of the rendezvous point, while the secondary router advertises a route with a shorter mask (a /29 that also includes the rendezvous point address). As long as both routes are present (both routers are up and available) unicast routing will choose the longest match and converge to the primary router. The secondary router's advertised route is chosen only when the primary router goes offline or all of its interfaces go down.

Figure 9.



Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
168 Robinson Road
#28-01 Capital Tower
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Europe Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: +31 0 800 020 0791
Fax: +31 0 20 357 1100

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA C17-455354-00 3/08