

Deploying Cisco Service Provider Network Routing

Volume 2

Version 1.01

Student Guide

Text Part Number: 97-3148-03



Americas Headquarters

Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters

Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters

Cisco Systems International BV Amsterdam,
The Netherlands

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.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

DISCLAIMER WARRANTY: THIS CONTENT IS BEING PROVIDED "AS IS" AND AS SUCH MAY INCLUDE TYPOGRAPHICAL, GRAPHICS, OR FORMATTING ERRORS. CISCO MAKES AND YOU RECEIVE NO WARRANTIES IN CONNECTION WITH THE CONTENT PROVIDED HEREUNDER, EXPRESS, IMPLIED, STATUTORY OR IN ANY OTHER PROVISION OF THIS CONTENT OR COMMUNICATION BETWEEN CISCO AND YOU. CISCO SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE. This learning product may contain early release content, and while Cisco believes it to be accurate, it falls subject to the disclaimer above.

Table of Contents

Volume 2

<u>Implement BGP in the Service Provider Network</u>	<u>4-1</u>
Overview	4-1
Module Objectives	4-1
<u>Enterprise Connectivity to Service Providers.....</u>	<u>4-3</u>
Overview	4-3
Objectives	4-3
Enterprise Connectivity Requirements.....	4-4
Routing Method Selection	4-5
Connection Options.....	4-8
Single-Homed Enterprises	4-9
Dual-Homed Enterprises.....	4-10
Multihomed Enterprises	4-11
Dual-Multihomed Enterprises.....	4-12
Summary	4-13
<u>Introducing BGP Routing.....</u>	<u>4-15</u>
Overview	4-15
Objectives	4-15
BGP Terminology.....	4-16
Autonomous System and BGP	4-18
BGP Routing Between Autonomous Systems	4-19
BGP Path Vector Functionality.....	4-20
BGP Routing Policies.....	4-21
Features of BGP	4-22
BGP Databases	4-23
BGP Message Types	4-24
Multiprotocol Extensions for BGP4	4-26
MPLS VPN Overview	4-28
Summary	4-29
<u>Implementing Basic BGP Routing.....</u>	<u>4-31</u>
Overview	4-31
Objectives	4-31
Planning for BGP	4-33
Configure Basic EBGp.....	4-34
Advertising BGP Networks.....	4-36
Configure Basic IBGP	4-38
Full-Mesh IBGP	4-40
BGP Support for IPv6.....	4-41
Shutting Down a BGP Neighbor	4-42
BGP Next-Hop Behavior	4-43
BGP Next Hop Self	4-44
Cisco IOS XR BGP Configuration Templates.....	4-45
BGP Neighbor Authentication and Neighbor Reset.....	4-47
BGP Neighbor Authentication	4-50
Clearing the BGP Session	4-51
Monitoring BGP Routes	4-53
BGP Route Selection Process and Path Attributes	4-54
BGP Weight Attribute	4-57
BGP Local Preference Attribute.....	4-58
BGP AS Path Attribute	4-59
BGP Multi-Exit Discriminator Attribute	4-60
Summary	4-61
Module Summary	4-63
Module Self-Check.....	4-65
Module Self-Check Answer Key.....	4-70

Routing Protocol Tools and Route Manipulation 5-1

Overview	5-1
Module Objectives	5-1
Introducing Routing Protocol Tools	1-3
Overview	1-3
Objectives	1-3
Routing Protocol Tools Overview	1-5
Example: OSPF Filtering	1-7
Example: BGP Filtering	1-8
Route Filtering Tools	1-9
Typical Filtering Objectives in OSPF	1-10
Typical Filtering Objectives in IS-IS	1-11
Typical Filtering Objectives in BGP	1-12
Typical BGP Routing Objectives	1-13
Prefix Lists Overview	1-15
Prefix Lists Syntax in Cisco IOS/IOS XE Software	1-16
Prefix List Example: Match Any Host Route	1-17
Prefix List Example: Match Default Route	1-18
Prefix List Example: Match All Routes	1-19
Prefix List Example: Match Small Routes	1-20
Prefix List Example: Match Core Loopbacks	1-21
Prefix List Example: Match Private Networks	1-22
Prefix Lists Syntax in Cisco IOS XR Software	1-23
Example: Prefix Lists in Cisco IOS XR Software	1-24
AS Path-Based Filtering	1-25
AS Path Access List Syntax in Cisco IOS/IOS XE Software	1-26
Regular Expressions, Special Characters	1-27
Commonly Used Regular Expressions	1-28
AS Path Access List Example: Permit All Routes	1-29
AS Path Access List Example: Permit Local Routes	1-30
AS Path Access List Example: Permit Routes From a Neighbor	1-31
AS Path Access List Example: Permit Routes Originating in a Specific AS	1-32
AS Path Access List Example: Permit Neighboring Local Routes	1-33
AS Path Access List Example: Allow AS Path Prepending	1-34
Summary	1-35
Introducing Route Maps and Routing Policy Language	1-37
Overview	1-37
Objectives	1-37
Route Maps Overview	1-39
Route Map Processing	1-40
Route Maps Syntax	1-41
Route Maps Example	1-43
Routing Policy Language	1-45
RPL Example	1-47
RPL Pass and Drop Actions	1-48
RPL Conditions	1-49
RPL Operators	1-50
RPL Boolean Operators	1-51
RPL Nesting	1-53
RPL Setting Attributes and Parameters	1-54
RPL Setting BGP Attributes and Parameters	1-57
Example: Setting BGP Attributes and Parameters	1-59
RPL Setting OSPF and IS-IS Parameters	1-61
RPL Parameterization	1-62
Applying Routing Policies	1-65
Maintaining Routing Policies	1-68
Value Sets	1-71
AS Path Sets	1-72

Standard Community Sets	1-75
Prefix Sets	1-79
Monitoring Routing Policies.....	1-80
Testing Routing Policies.....	1-82
Translating Route Maps to Routing Policies	1-83
Summary	1-85
Implementing Route Redistribution.....	5-87
Overview	5-87
Objectives	5-87
Need for Redistribution	5-89
Route Redistribution.....	5-92
Using Seed Metrics in Route Redistribution	5-93
Default Seed Metrics in Route Redistribution	5-95
One-Point Route Redistribution	5-96
Multipoint Route Redistribution	5-97
Route Redistribution Techniques	5-98
Redistribution Implementation Considerations	5-100
Implement Redistribution	5-104
Redistribution into OSPF	5-105
Redistribution into IS-IS	5-107
Redistribution into BGP	5-108
Administrative Distance.....	5-109
Modifying OSPF Administrative Distance	5-110
Modifying IS-IS and BGP Administrative Distance	5-111
Example: Redistribution Using Administrative Distance	5-112
Redistribution to Prevent Routing Loops	5-117
Summary	5-119
Influencing BGP Route Selection	5-121
Overview	5-121
Objectives	5-121
BGP Weight	5-123
Configuring Per-Neighbor Weights	5-125
Changing Weights with RPLs or Route Maps.....	5-126
BGP Weight Attachment Points	5-127
BGP Local Preference	5-128
Changing Local Preference.....	5-130
Monitoring Local Preference	5-131
AS Path Prepending.....	5-132
AS Path Prepending Design Considerations	5-137
Configuring AS Path Prepending	5-139
Monitoring AS Path Prepending.....	5-140
AS Path Filtering Concerns: AS Path Prepending	5-141
BGP Multi-Exit Discriminators	5-143
Changing the MED	5-145
Monitoring the MED	5-147
BGP Communities.....	5-148
Configuring BGP Communities	5-151
BGP Named Community Lists	5-155
BGP Support for Sequenced Entries in Extended Community Lists	5-156
Matching BGP Communities	5-157
Monitoring BGP Communities.....	5-158
BGP Community Examples.....	5-159
Summary	5-165
Module Summary	5-167
Module Self-Check.....	5-169
Module Self-Check Answer Key	5-173

Implement BGP in the Service Provider Network

Overview

This module describes Border Gateway Protocol (BGP) and its importance to the service provider environment. The primary goal of the module is to describe the BGP implementation steps that connect an enterprise to a service provider, and connect a service provider to upstream service providers. The use of BGP as a routing protocol requires that an administrator understand how to properly configure BGP for scalable internetworking.

Module Objectives

Upon completing this module, you will be able to implement and verify BGP to connect a service provider with customers and other service providers. This ability includes being able to meet these objectives:

- Describe enterprise requirements when implementing connectivity to a service provider
- Use BGP to connect an enterprise network to service providers, and from the service providers to upstream service providers
- Implement internal and external BGP sessions to enable service provider routing

Enterprise Connectivity to Service Providers

Overview

To properly plan the enterprise connectivity to a service provider, the designer must understand different connectivity options. Requirements, such as IP addressing, link type and bandwidth availability, routing options, and redundancy should be considered to choose from different connectivity types.

This lesson first describes enterprise connectivity requirements and routing methods that are available. The lesson concludes with different types of connectivity that affect routing method selection, link and bandwidth availability, IP addressing, and redundancy.

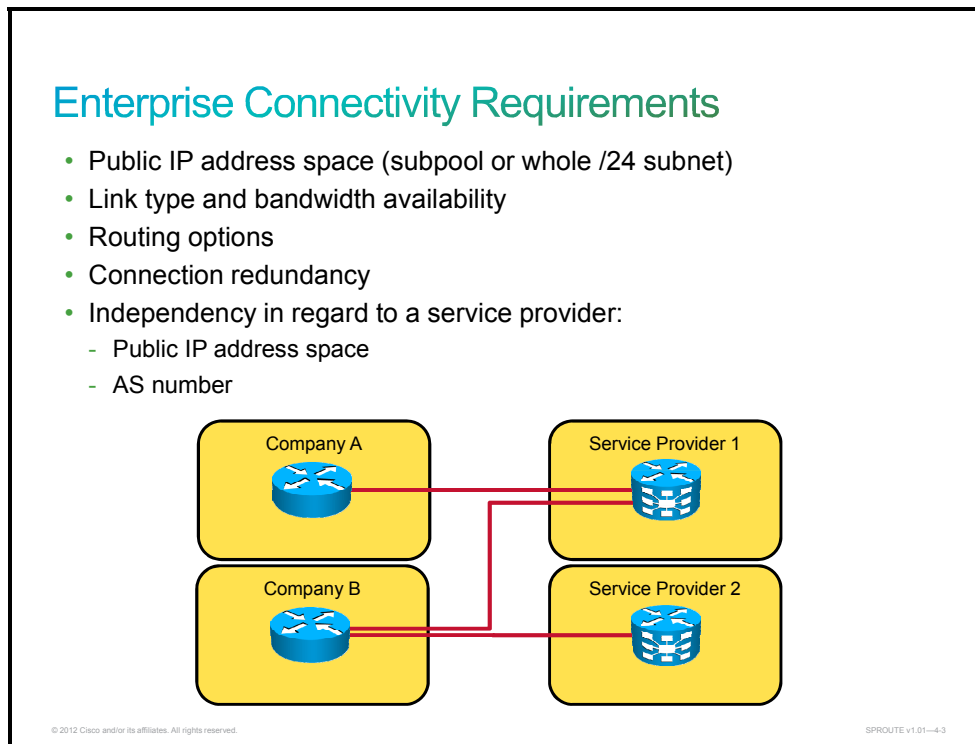
Objectives

Upon completing this lesson, you will be able to use different connectivity options to connect an enterprise network to a service provider. You will be able to meet these objectives:

- Describe connectivity requirements when connecting an enterprise network to a service provider
- Describe routing method selection when connecting an enterprise to a service provider
- Describe different connection options between enterprises and service providers
- Describe single-homed enterprises
- Describe dual-homed enterprises
- Describe multihomed enterprises
- Describe dual-multihomed enterprises

Enterprise Connectivity Requirements

This topic describes connectivity requirements when connecting an enterprise network to a service provider.



Modern corporate networks are connected to the Internet and make use of the Internet for some of their data transport needs. Establishing enterprise connectivity to a service provider requires knowledge and careful planning. The first parameter to be determined is the number of public IP addresses that are needed. Public IP addresses are used to translate client private addresses for those clients that need to access resources on the Internet. These public IP addresses are also used for those enterprise servers that need to be accessible from the Internet. These servers are either equipped with public addresses or with addresses that are statically translated from private to public addresses.

The link type and speed, which depend on the service provider, are the second parameter to be determined. The link types that are available include leased line, Ethernet over fiber optics or copper, and xDSL. The bandwidth assignment must be done properly to address the connectivity requirements.

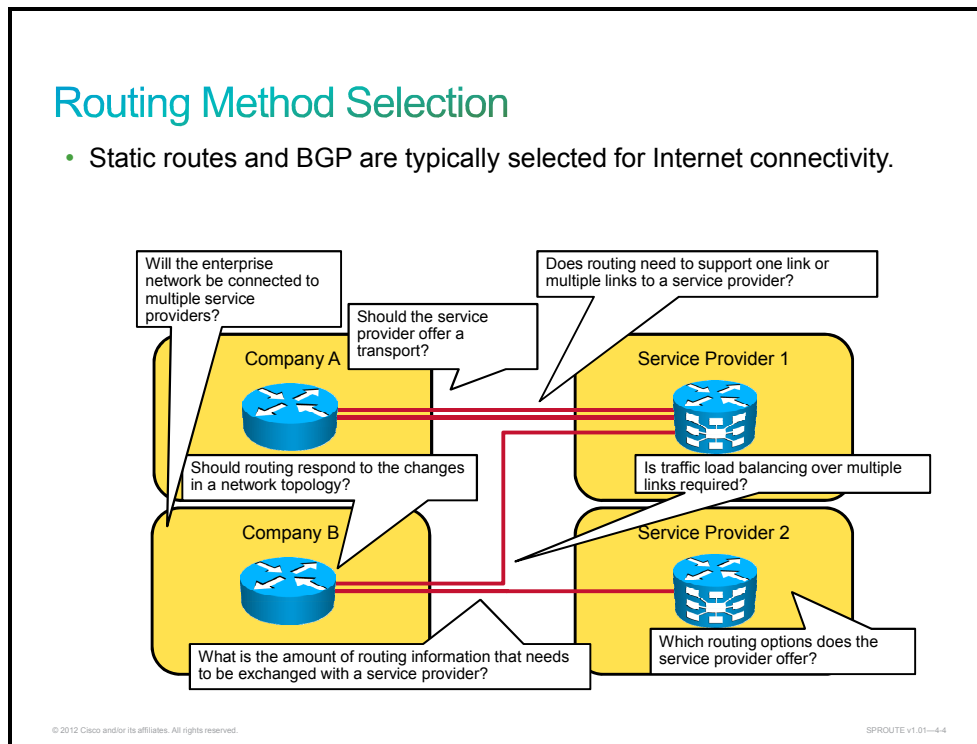
In the third parameter, a proper routing protocol must be selected. The selection is typically made between static and dynamic routing.

The fourth parameter relates to the issue of connectivity redundancy. An evaluation is needed to determine which kind of redundancy is required for the connectivity. Redundancy includes edge router redundancy, link redundancy, and service provider redundancy.

When assessing these parameters, it must be taken into account whether an enterprise network needs to be independent of the selected service providers. If independency is required, the public IP address space should not be used from the service provider public address space, but should instead be acquired from the regional Internet authority. Similarly, independency is required for the enterprise network AS. The AS number must be a public assigned number, and not from the private AS number pool that can be assigned by the service provider.

Routing Method Selection

This topic describes routing method selection when connecting an enterprise to a service provider.

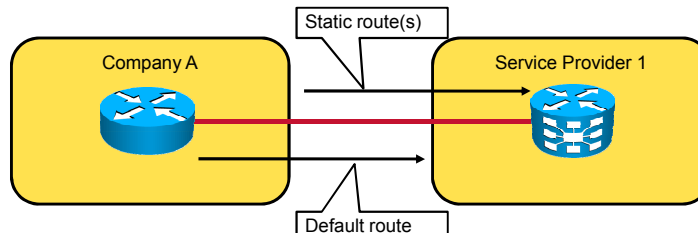


Connecting an enterprise network to a service provider requires routing information to be exchanged. The selection of the routing means depends on the answers to these questions:

- Should the routing methodology respond to the changes in a network topology?
- Does the routing methodology need to support one link, or multiple links, to a service provider?
- Is traffic load balancing over multiple links required?
- Will the enterprise network be connected to multiple service providers?
- Should the service provider offer merely a transport capability that requires connecting different customer locations, perhaps via certain Layer 2 technologies?
- What is the amount of routing information that needs to be exchanged with a service provider?
- Which routing options does the service provider offer?

Routing Method Selection - Static Routes

- Customer is connected through a single connection to a service provider.
- The customer uses the default route toward the service provider.
- The service provider uses static route(s) for customer public networks.
- There is no automatic adjustment to any changes in the network.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4.6

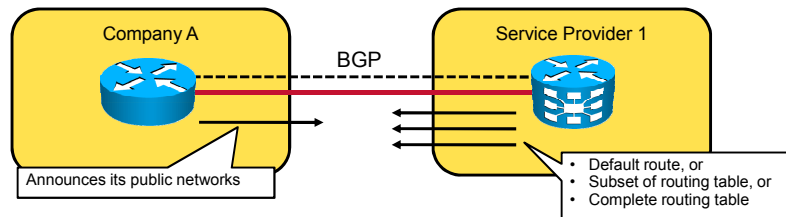
Static routes are the simplest way of exchanging routing information with a service provider. These static routes must agree with the service provider about routing configuration, and they do not change afterward; or, if they do change, these changes should not occur often.

Static routes are typically used when a customer is connected through a single connection to a service provider. The customer can use the default route toward the service provider, and the service provider must deploy a static route or routes that encompass the public networks of the customer. The service provider typically would also redistribute this information into its BGP.

Although static routes are a simple solution, they also have some drawbacks, especially in terms of flexibility and adaptability. For example, if there were a change in a network topology, beyond a directly connected link failure, the static routes would not adapt. If adaptation is required, either the static routes must be combined with an IP service level agreement (SLA) functionality that, in turn, would indicate that a static route is down if a certain condition is met, or dynamic routing is used. Note that using an IP SLA cannot completely substitute for dynamic routing, because it cannot react to all changes in the topology in the Internet.

Routing Method Selection - BGP

- Typically used for inter-AS routing



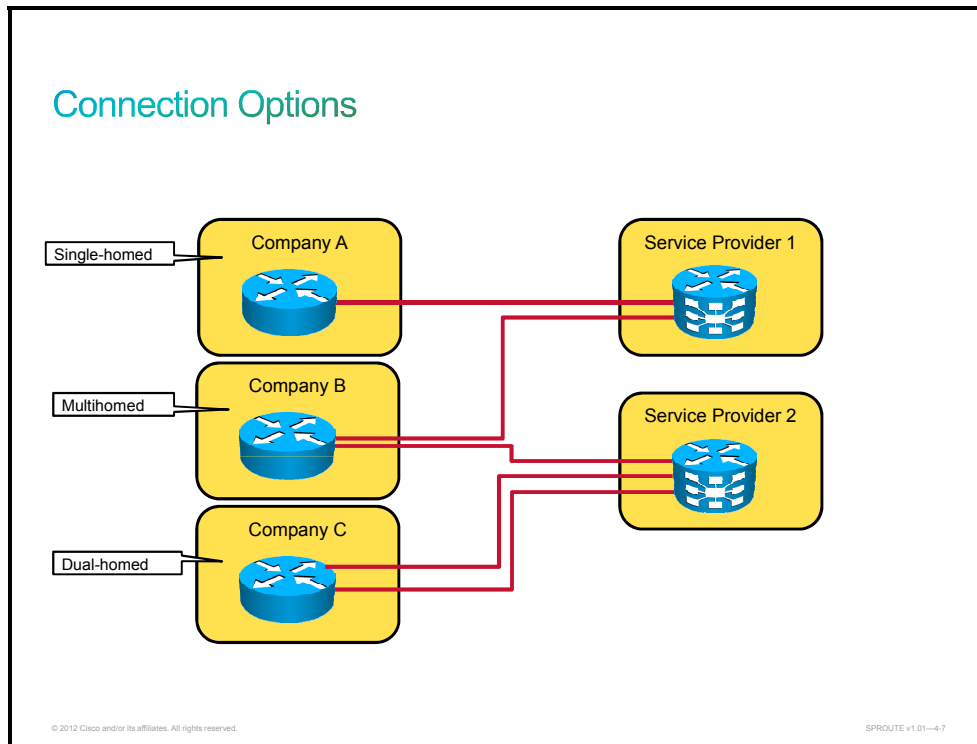
A typical option that is used to provide dynamic routing exchange when Internet connectivity is deployed is BGP. BGP dynamically exchanges routing information, and thus reacts to topology changes, including those changes beyond a link failure.

From the routing perspective, three options can be used:

- A service provider announces a default route only. If two service providers are used, one service provider will be the primary and the second service provider will serve as the backup.
- A service provider announces a default route and a subset of Internet routes, typically from its own public address space. This results in the shortest path being used to the directly connected public networks. For public networks that are not directly connected, one of the service providers is the primary and the second service provider is the backup.
- A service provider announces a complete Internet routing table. The shortest path to any destination will be chosen according to the routing table.

Connection Options

This topic describes different connection options between enterprises and service providers.



When you are connecting an enterprise network to a service provider, redundancy is a serious concern. These different aspects can be addressed to achieve connectivity redundancy:

- Deployment of redundant links
- Deployment of redundant equipment
- Use of redundancy within a single router

From the connection perspective, a customer can be connected to a single service provider or to multiple service providers. With a single service provider connection, redundancy can still be achieved if two links toward the same service provider are used effectively, making a customer dual-homed. When no link redundancy is used, the customer is simply single-homed, and in the case of a failure within a service provider network, connectivity to the Internet is interrupted.

With multiple service provider connections, redundancy is built into the design, because the customer is multihomed and thus is resistant to a single service provider failure. To enhance the resiliency further, a customer can have two links toward a single service provider, making the solution dual-multihomed.

Single-Homed Enterprises

This topic describes single-homed enterprises.

Single-Homed Enterprises

Link failure results in broken connectivity.

Option 1: Routing with static routes

- A default route from an enterprise network is used.
- A static route(s) from a service provider is used for customer networks.

Option 2: Routing with BGP

- The customer announces its public network or networks.
- The service provider announces the default route to the customer.



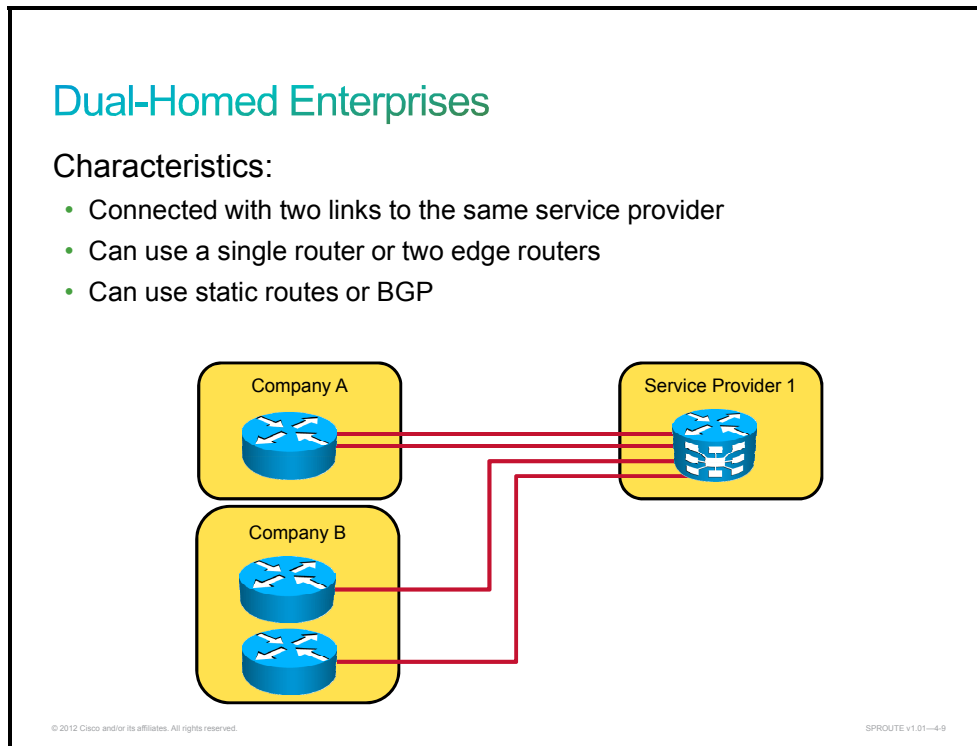
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4-8

Single-homed service provider connectivity is used in cases when a loss in Internet connectivity is not vital or highly problematic to a customer. Such Internet access to a single service provider does not require BGP. Static routes are typically used to manage the routing. If BGP is used, the customer uses it to dynamically announce its public networks to a service provider; the service provider announces only a default route to the customer, because that is sufficient to provide the connectivity through a single link to the Internet.

Dual-Homed Enterprises

This topic describes dual-homed enterprises.



When a customer is connected to a single service provider only, resiliency can still be achieved by deploying a second link to the same service provider. With a second link being used, the routing must be properly configured to allow usage of such a link.

Depending on the SLA that is signed with the service provider, the routing that is deployed could achieve these benefits:

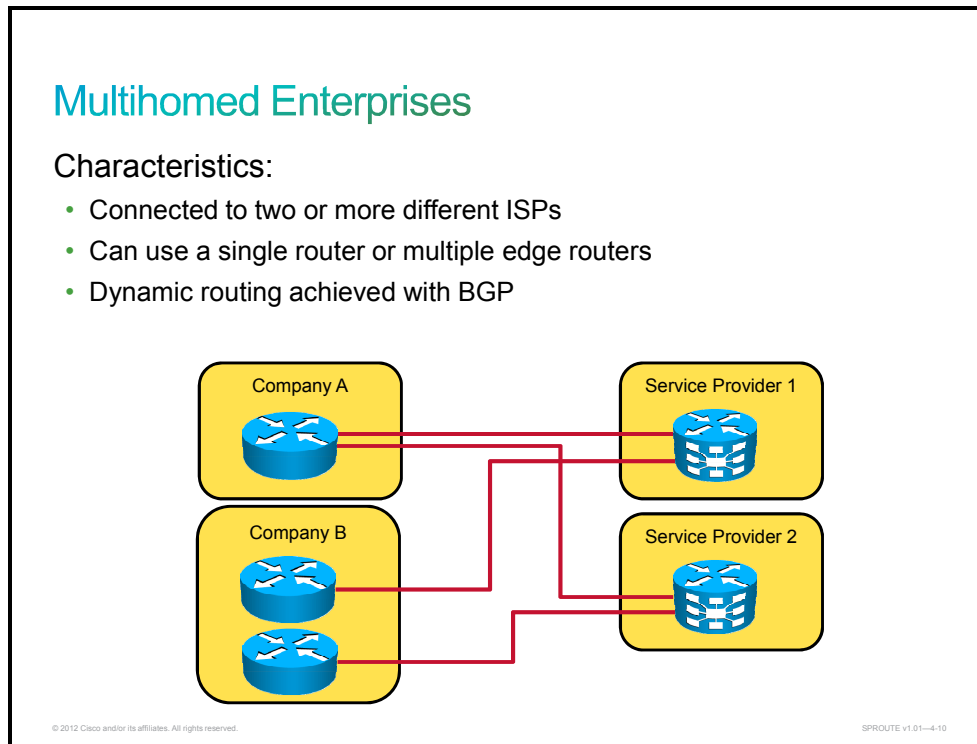
- Primary and backup link functionality, where a single primary link is used to forward and receive traffic to and from the service provider, and the secondary link is used only when the first link fails
- Load sharing between the links, achieved with Cisco Express Forwarding switching

In both cases, the routing that is used can be based on either static or dynamic routes, which would typically include BGP.

To enhance the resiliency further, the two links can terminate at separate customer routers.

Multihomed Enterprises

This topic describes multihomed enterprises.



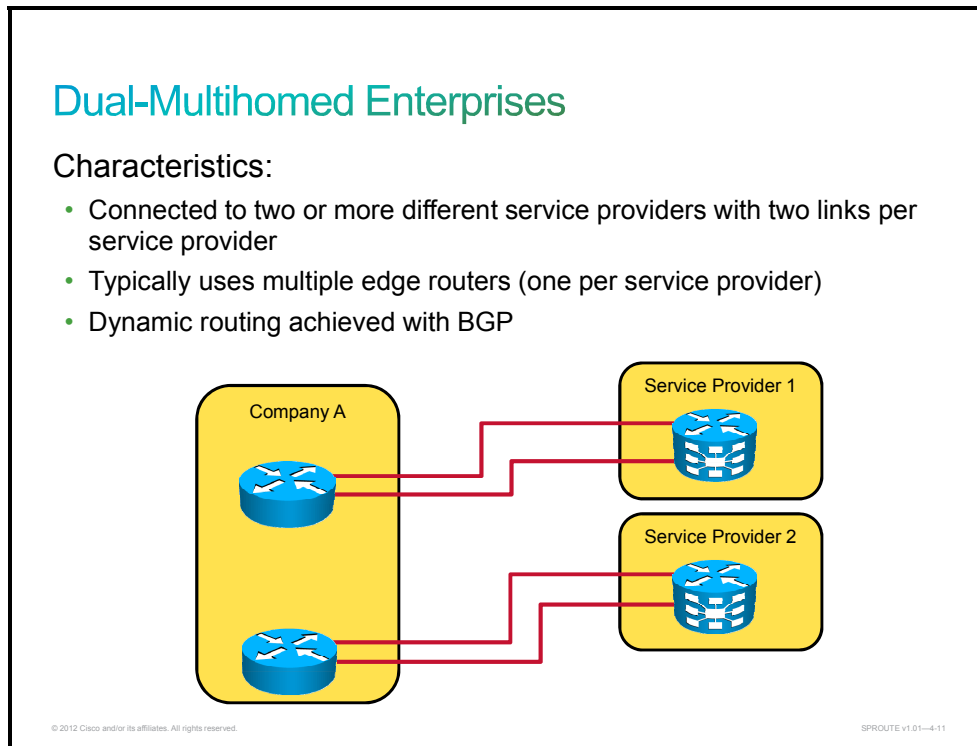
Robust resiliency is achieved by connecting to two or more different service providers. The benefits of connecting to two or more different service providers are significant:

- Greater resistance to a failure than with a directly connected link to a single service provider
- Sharing the load for different destination networks between service providers, based on the network proximity
- Scaling the solution beyond two service providers
- Achieving a solution independent of a service provider. A service provider change requires an update of routing configuration and change of a link. The public IP address space that is used remains the same.

Connections from different service providers can terminate on the same router or different routers to further enhance the resiliency. The routing methodology must be capable of reacting to dynamic changes, and BGP is typically used to achieve this flexibility.

Dual-Multihomed Enterprises

This topic describes dual-multihomed enterprises.



Multihoming exists when an AS has more than one connection to the Internet. There are two typical reasons for multihoming:

- To increase the reliability of the connection to the Internet. If one connection fails, the other connection remains available.
- To increase the performance of the connection. Better paths can be used to certain destinations.

The benefits of BGP are apparent when an AS has multiple External Border Gateway Protocol (EBGP) connections to either a single AS or multiple autonomous systems. Having multiple connections allows an organization to have redundant connections to the Internet, so that if a single path becomes unavailable, connectivity can still be maintained.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- The way in which the customer connects to a service provider depends on the redundancy requirements.
- You can use static routing or BGP to provide routing between an enterprise and a service provider.
- You can achieve redundancy between an enterprise and a service provider by using redundant links and/or redundant equipment.
- Single-homed service provider connectivity is used in cases when a loss in Internet connectivity is not vital or highly problematic to a customer.
- Dual-homed customer is connected to single service provider using two links.
- Multihomed customer is connected to two service providers.
- Dual-multihomed customer is connected to two service providers using two links.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4-12

Introducing BGP Routing

Overview

To properly plan the enterprise connectivity to a service provider, the designer must understand many aspects of the Border Gateway Protocol (BGP), including the session origination point, the prerequisites for successfully implementing such connectivity, the available routing options, and which routing option should be used in certain cases. The connectivity redundancy considerations must also be taken into account.

BGP is an exterior gateway protocol (EGP) and this lesson introduces BGP routing. BGP routes between autonomous systems by using path vector attributes upon which routing policy decisions at the autonomous system (AS) level are enforced.

Objectives

Upon completing this lesson, you will be able to explain the advantages of using BGP to connect an enterprise network to a service provider. You will be able to meet these objectives:

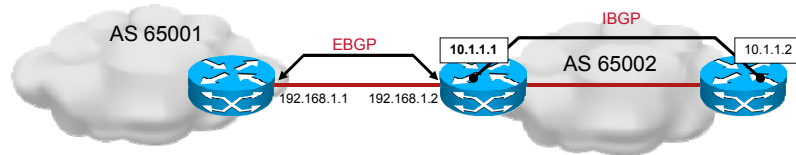
- Describe BGP terminology
- Describe autonomous systems in BGP networks
- Describe BGP routing between autonomous systems
- Describe BGP path vectors
- Describe BGP routing policies
- Describe features of BGP
- Describe the tables used by a router to store BGP information
- Describe the four BGP message types
- Describe the multiprotocol extensions add to BGP to support IPv6
- Describe MPLS VPNs

BGP Terminology

This topic describes BGP terminology.

BGP Terminology

- **Autonomous system:** A collection of networks under a single administrative domain
- **Interdomain routing:** Routing between the customer and the service provider
- **Internal routing:** Uses IGP protocol (RIP, OSPF, IS-IS, and so on) to exchange routing information inside the autonomous system
- **External routing:** Uses EGP protocol (BGP) to exchange routes between autonomous systems
- Two BGP implementations:
 - **Internal BGP (IBGP):** When BGP is used inside an AS
 - **External BGP (EBGP):** When BGP is used between autonomous systems



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4.3

The Internet is a collection of autonomous systems that are interconnected to allow communication among these autonomous systems. BGP provides the routing between these autonomous systems.

Enterprises that want to connect to the Internet do so through one or more service providers. If your organization has only one connection to one service provider, then you probably do not need to use BGP; instead, you would use a default route. However, if you have multiple connections to one or multiple service providers, then BGP might be appropriate, because it allows manipulation of path attributes so that the optimal path can be selected.

One way to categorize routing protocols is by whether the protocols are interior or exterior:

- **Interior gateway protocol (IGP)** is a routing protocol that exchanges routing information within an AS. Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Intermediate System-to-Intermediate System (IS-IS) are examples of IGPs.
- **EGP** is a routing protocol that exchanges routing information between different autonomous systems. BGP is an example of an EGP.

BGP is an Interdomain Routing Protocol (IDRP), also known as an EGP. BGP version 4 (BGP4) is defined in RFC 4271. As noted in this RFC, the classic definition of an AS is “a set of routers under a single technical administration, using an IGP and common metrics to route packets within the AS, and using an interautonomous system routing protocol (also called an EGP) to determine how to route packets to other autonomous systems.”

Autonomous systems can use more than one IGP, potentially with several sets of metrics. From the point of view of BGP, the most important characteristic of an AS is that it appears to other autonomous systems to have a single coherent interior routing plan and presents a consistent picture of reachable destinations. All parts of an AS must connect to each other.

When BGP is running between routers in different autonomous systems, it is called EBGp. When BGP is running between routers in the same AS, it is called IBGP. BGP allows the path that packets take to be manipulated by the AS. It is important to understand how BGP works to avoid creating problems for the AS that result from running BGP.

Autonomous System and BGP

This topic describes autonomous systems in BGP networks.

Autonomous System

- An autonomous system (AS) is a collection of networks under a single technical administration.
 - 16-bit numbers (as of January 2009, 32-bit numbers are available)
 - Ranging from 1 to 65535
 - Private AS: 64512–65535
- The IANA allocates AS numbers.
- IGPs operate within an AS.
- BGP is used between autonomous systems.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--4.4

Recall that an AS is a collection of networks under a single technical administration. IGPs operate within an AS, and BGP is used between autonomous systems on the Internet.

The Internet Assigned Numbers Authority (IANA) is the organization that is responsible for allocating AS numbers. Specifically, the American Registry for Internet Numbers (ARIN) has the jurisdiction to assign numbers for the Americas, the Caribbean, and Africa. Réseau IP Européens Network Coordination Centre (RIPE NIC) administers AS numbers for Europe, and the Asia Pacific Network Information Center (APNIC) administers the numbers for the Asia Pacific region.

AS numbers traditionally were 16-bit numbers ranging from 1 to 65535. RFC 1930 provides guidelines for the use of AS numbers. A range of AS numbers, 64512 through 65535, is reserved for private use, much like private IP addresses.

As the 16-bit AS range was coming to exhaustion, the IANA extended in 2007 the AS number registry to a 32-bit range (0 to 4,294,967,295). Most networks can still use the “old” AS 16-bit range. Routers that are using the new range need their code to be updated to extend the AS number field size to 32 bits. Routers that are using the old system still use the 16-bit AS number field. When a router that is using the new AS range communicates with a router that is using the old AS range, it uses the reserved AS transition number 23456 in 16-bit form.

For readability purposes, the 32-bit AS range is written in two parts. The old AS ranges from 0.1 to 0.65535; the new AS ranges from 1.0 to 65535.65535.

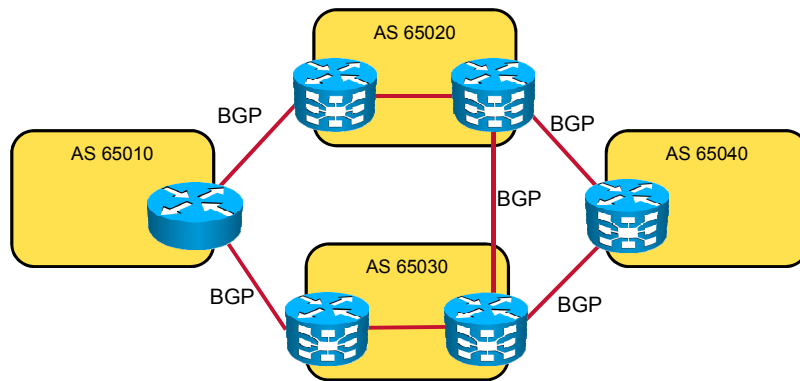
AS numbers from 65512 to 65535 are still reserved for private usage in the new 16-bit system.

BGP Routing Between Autonomous Systems

This topic describes BGP routing between autonomous systems.

BGP Routing Between Autonomous Systems

- BGP is used to provide an interdomain routing system.
- BGP guarantees the exchange of loop-free routing information.
- BGP works differently than IGPs.
 - BGP is a policy-based routing protocol.
 - BGP controls traffic flow using multiple BGP path attributes.

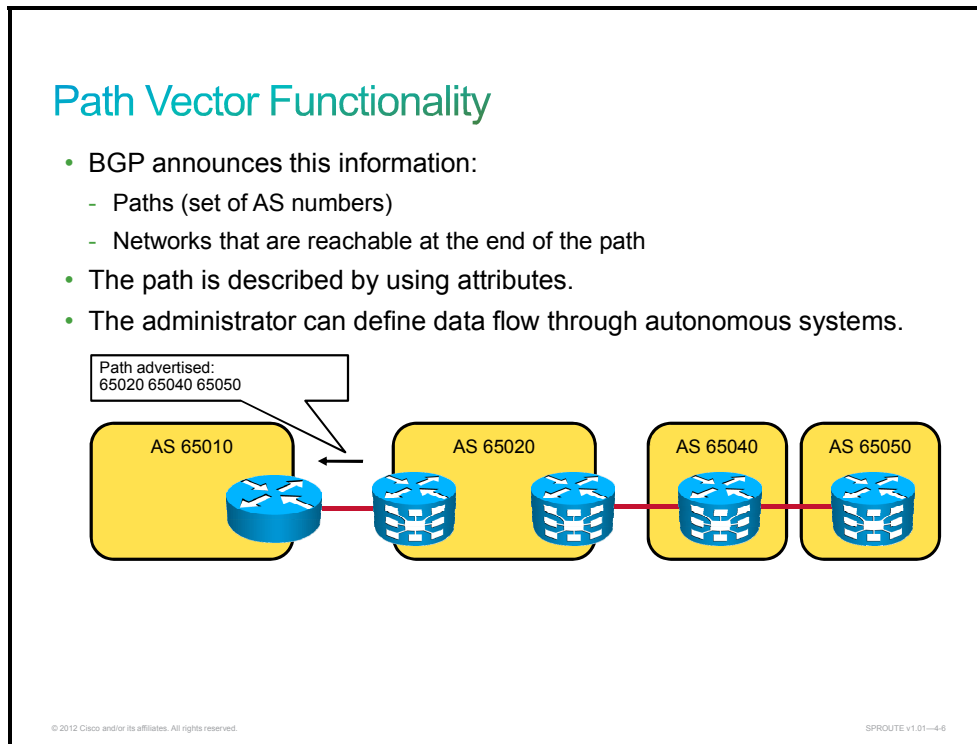


The main goal of BGP is to provide an interdomain routing system that guarantees a loop-free exchange of routing information between autonomous systems. Routers exchange information about paths to destination networks. BGP is a successor of EGP, which was developed to isolate networks from each other as the Internet grew.

BGP works differently than IGPs. An internal routing protocol looks for the quickest path from one point in a network to another, based on certain metrics. In the internal routing protocol, the next hop is the next router; in the BGP, the next hop is the next AS. Rather than treating a router as a single point in the path to any given destination, BGP treats each AS as a single point on the path to the destination. The primary reason that BGP treats an entire autonomous system as a single hop in the AS path is to hide topological details of the AS. No AS can tell what the path through another AS looks like, only that the destination is reachable through that AS. BGP is a policy-based routing (PBR) protocol that allows an AS to control traffic flow using multiple BGP path attributes. BGP allows a provider to use all its bandwidth by manipulating these path attributes.

BGP Path Vector Functionality

This topic describes BGP path vectors.



Internal routing protocols announce a list of networks and the metrics to get to each network. In contrast, BGP routers exchange network reachability information, called path vectors, which are made up of path attributes (like metrics). The path vector information includes a list of the complete path of BGP AS numbers (hop by hop) that are necessary to reach a destination network and the networks that are reachable at the end of the path. Other attributes include the IP address to get to the next AS (the next-hop attribute), and an indication of how the networks at the end of the path were introduced into BGP (the origin code attribute).

This AS path information is useful to construct a graph of loop-free autonomous systems, and is used to identify routing policies so that restrictions on routing behavior can be enforced, based on the AS path.

The AS path is always loop-free. A router that is running BGP does not accept a routing update that already includes the router AS number in the path list, because the update has already passed through its AS, and accepting it again would result in a routing loop.

An administrator can define policies or rules about how data will flow through the autonomous systems.

BGP Routing Policies

This topic describes BGP routing policies.

BGP Routing Policies

- BGP can support any policy conforming to the hop-by-hop (AS-by-AS) routing paradigm.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4.7

BGP allows routing policy decisions at the AS level to be enforced. These policies can be implemented for all networks that are owned by an AS, for a certain classless interdomain routing (CIDR) block of network numbers (prefixes), or for individual networks or subnetworks.

BGP specifies that a BGP router can advertise to neighboring autonomous systems only those routes that it actually uses. This rule reflects the hop-by-hop routing paradigm that the Internet generally uses.

The hop-by-hop routing paradigm does not support all possible policies. For example, BGP does not enable one AS to send traffic to a neighboring AS, intending that the traffic takes a different route from that taken by traffic that originates in that neighboring AS. In other words, the way that a neighboring AS routes traffic cannot be influenced, but the way that traffic gets to a neighboring AS can be influenced. However, BGP supports any policy that conforms to the hop-by-hop routing paradigm.

Because the Internet currently uses the hop-by-hop routing paradigm only, and because BGP can support any policy that conforms to that paradigm, BGP is highly applicable as an interautonomous system (inter-AS) routing protocol.

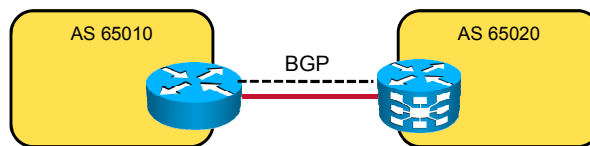
Features of BGP

This topic describes features of BGP.

Features of BGP

BGP is a path vector protocol with the following properties:

- Reliable updates: BGP runs on top of TCP (port 179)
- Incremental, triggered updates only
- Periodic keepalive messages to verify TCP connectivity
- Rich metrics (called path vectors or attributes)
- Designed to scale to huge internetworks



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4.8

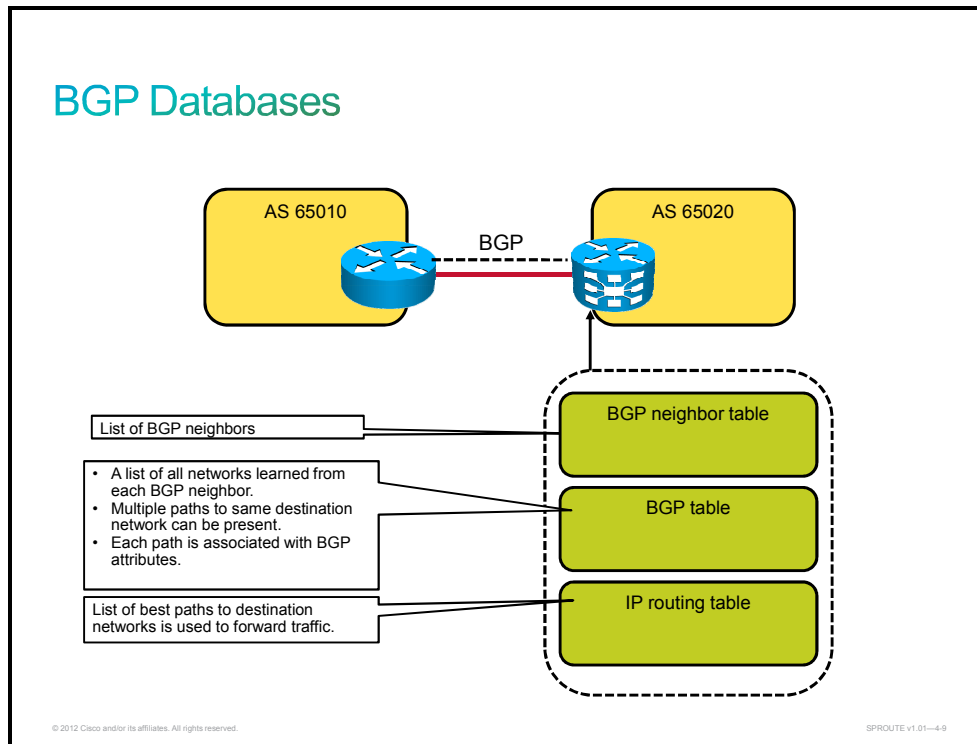
BGP uses TCP as its transport protocol, which provides reliable connection-oriented delivery. BGP assumes that its communication is reliable; therefore, it does not have to implement retransmission or error recovery mechanisms. BGP uses TCP port 179.

A router that runs the BGP protocol is called a BGP “Speaker”. Two routers that are using BGP form a TCP connection with one another and exchange messages to open and confirm the connection parameters. These two BGP routers are called “peer routers,” or “neighbors.” No routing information is exchanged until the TCP connection has been established.

After the connection is made, BGP peers exchange complete routing tables. However, because the connection is reliable, BGP peers send only changes (incremental, or triggered, updates) after that. Reliable links do not require periodic routing updates; therefore, routers use triggered updates instead. BGP sends keepalive messages.

BGP Databases

This topic describes the tables used by a router to store BGP information.



A router that is running BGP keeps its own tables to store BGP information that it receives from and sends to other routers. The tables include a neighbor table, a BGP table (also called a forwarding database or topology database), and an IP routing table.

For BGP to establish an adjacency, it must be explicitly configured for each neighbor. BGP forms a TCP relationship with each of the configured neighbors and keeps track of the state of these relationships by periodically sending a BGP/TCP keepalive message. The BGP sends BGP/TCP keepalives by default every 60 seconds.

After establishing an adjacency, the neighbors exchange the BGP routes that are in their IP routing table. Each router collects these routes from each neighbor that successfully establishes an adjacency, and then places the routes in its BGP forwarding database. All routes that have been learned from each neighbor are placed into the BGP forwarding database. The best routes for each network are selected from the BGP forwarding database, using the BGP route selection process, and are then offered to the IP routing table.

Each router compares the offered BGP routes to any other possible paths to those networks, and the best route—based on administrative distance—is installed in the IP routing table.

EBGP routes (BGP routes that are learned from an external AS) have an administrative distance of 20. IBGP routes (BGP routes that are learned from within the AS) have an administrative distance of 200.

BGP Message Types

This topic describes the four BGP message types.

BGP Message Types

BGP defines the following message types:

- **Open**
 - Includes hold time and BGP router ID
- **Keepalive**
- **Update**
 - Information for one path only (could be to multiple networks)
 - Includes path attributes and networks
- **Notification**
 - When an error is detected
 - BGP connection closed after message is sent

© 2012 Cisco and/or its affiliates. All rights reserved.SPROUTE v1.01—4-10

The four BGP message types are open, keepalive, update, and notification.

After a TCP connection is established, the first message that is sent by each side is an open message. If the open message is acceptable, the side that receives the message sends a keepalive message confirming the open message. After the receiving side confirms the open message and establishes the BGP connection, the BGP peers can exchange any update, keepalive, and notification messages.

BGP peers initially exchange their full BGP routing tables. Incremental updates are sent only after topology changes in the network occur. BGP peers send keepalive messages to ensure that the connection between the BGP peers still exists, and they send notification packets in response to errors or special conditions.

Here are more details about the different types of BGP messages:

- **Open message:** An open message includes the following information:
 - **Version number:** The suggested version number. The highest common version that both routers support is used. Most BGP implementations today use BGP4.
 - **AS number:** The AS number of the local router. The peer router verifies this information. If it is not the AS number that is expected, the BGP session is ended.
 - **Hold time:** Maximum number of seconds that can elapse between the successive keepalive and update messages from the sender. On receipt of an open message, the router calculates the value of the hold timer by using whichever is smaller: its configured hold time or the hold time that was received in the open message.
 - **BGP router ID:** This 32-bit field indicates the BGP ID of the sender. The BGP ID is an IP address that is assigned to that router, and it is determined at startup. The BGP router ID is chosen in the same way that the OSPF router ID is chosen—it is

the highest active IP address on the router, unless a loopback interface with an IP address exists. In this case, the router ID is the highest loopback IP address. The router ID can also be statically configured.

- **Optional parameters:** These parameters are type, length, value (TLV) encoded. An example of an optional parameter is session authentication.
- **Keepalive message:** BGP keepalive messages are exchanged between BGP peers frequently enough to keep the hold timer from expiring. If the negotiated holdtime interval is 0, then periodic keepalive messages are not sent. A keepalive message consists of only a message header.
- **Update message:** A BGP update message has information on one path only; multiple paths require multiple update messages. All the attributes in the update message refer to that path, and the networks are those that can be reached through that path. An update message can include the following fields:
 - **Withdrawn routes:** This list displays IP address prefixes for routes that are withdrawn from service, if any.
 - **Path attributes:** These attributes include the AS path, origin, local preference, and so on (as described later in this module). Each path attribute includes the attribute TLV. The attribute type consists of the attribute flags, followed by the attribute type code.
 - **Network-layer reachability information:** This field contains a list of IP address prefixes that are reachable by this path.
- **Notification message:** A BGP notification message is sent when an error condition is detected; the BGP connection is closed immediately after this is sent. Notification messages include an error code, an error subcode, and data that is related to the error.

Multiprotocol Extensions for BGP4

This topic describes the multiprotocol extensions add to BGP to support IPv6.

Multiprotocol Extensions for BGP4

- BGP originally designed for IPv4:
 - Carries IPv4 prefix reachability information
 - Uses IPv4 for transport
- Multiprotocol extensions for BGP4:
 - Enables other protocols besides IPv4
 - New identifier for the address family
 - Most often used in MPLS networks for MPLS VPN
- IPv6-specific extensions:
 - Scoped addresses: NEXT_HOP contains a global IPv6 address and potentially a link-local address.
 - NEXT_HOP and NLRI are expressed as IPv6 addresses and prefixes in the multiprotocol attributes.
- Still uses TCP for transport:
 - TCP can run over IPv4, transporting IPv6 information.
 - TCP can run natively over IPv6.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01-4.11

Original BGP4 (RFC 1771) carries IPv4-specific information. Multiprotocol BGP (RFC 2283) uses extensions to add capability to transport routing information for other protocols. Multiprotocol extensions for BGP are defined as new attributes. These new attributes define Network Layer Reachability Information (NLRI) and a next hop (the next router in the path to the destination) where IPv6 addresses can be used.

BGP4, with multiprotocol extensions, enables the use of many address families. Address families define the type of addresses being carried. The most common address families are IPv4, IPv6, and VPNv4 and VPNv6 for Multiprotocol Label Switching Virtual Private Network (MPLS VPN) routes.

An address family is activated within BGP using the Cisco IOS/IOS XE/IOS XR **address-family** router BGP command.

Two new attributes support multiprotocol BGP (MP-BGP):

- Multiprotocol reachable NLRI (MP_REACH_NLRI)
- Multiprotocol unreachable NLRI (MP_UNREACH_NLRI)

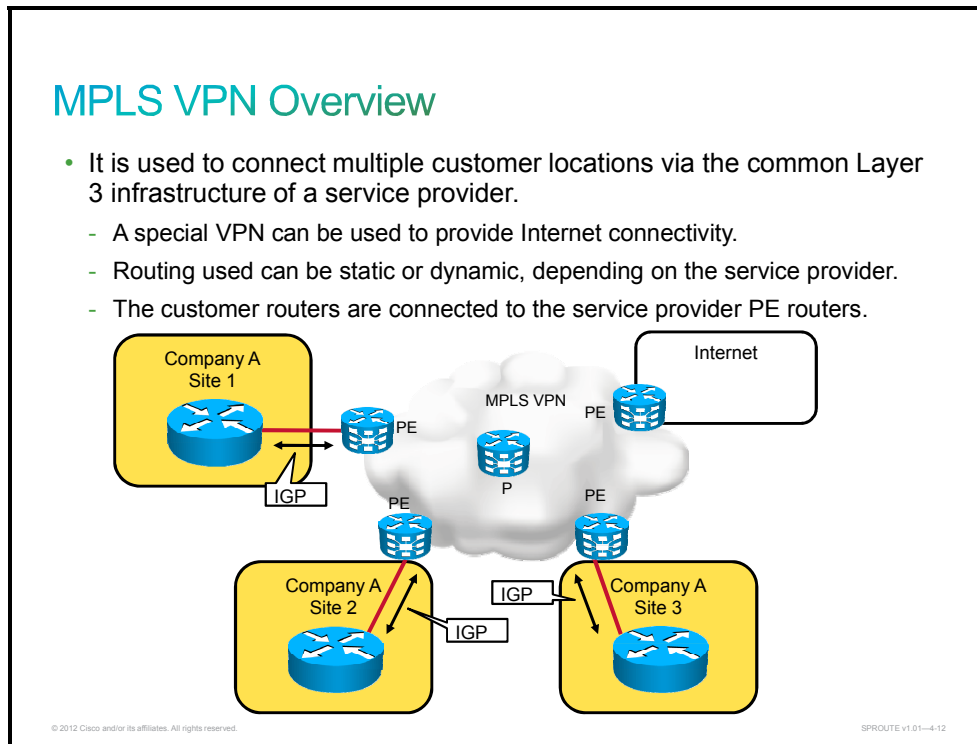
The MP_REACH_NLRI attribute describes reachable destinations and includes attributes that contain information about network-layer protocol prefixes (specifically IPv6), and the next destination hop-to-reach prefixes.

Here are the IPv6-specific extensions for MP-BGPv4:

- **Scoped addresses:** NEXT_HOP contains a global IPv6 address, or potentially a link-local address, when there is link-local reachability with the peer. When link-local addresses are used for peering with a neighbor BGP router, these link-local IPv6 addresses are used as next hop IP addresses for the routes that are carried by BGP. In the majority of cases, the next hop IPv6 addresses need to be changed to global IPv6 by attaching a route map or route policy to the neighbor.
- **IPv6 address format:** NEXT_HOP and NLRI are expressed as IPv6 addresses and prefixes in the multiprotocol attributes. You can run MP-BGP over IPv4 or IPv6 transport and can exchange routes for IPv4, IPv6, or both. BGP uses the TCP protocol for peering and this has no relevance to the routes that are carried inside the BGP exchanges. Either IPv4 or IPv6 can be used to transport a TCP connection on the network layer.

MPLS VPN Overview

This topic describes MPLS VPNs.



MPLS VPNs are used when a customer has multiple locations that need to be interconnected through a service provider and the customer does not want to use expensive Layer 2 technologies, such as dedicated leased lines.

With MPLS VPNs, the service provider uses a common IP-based core network that is enhanced with MPLS technology to provide secure and manageable connectivity for different customers to their geographically diverse sites. In this way, the traffic from different customers can share the same physical wire, but at the same time, traffic is tagged with the labels in such way that the traffic cannot intermix.

When a customer uses MPLS VPN functionality, the routing between the customer and the service provider is required to provide connectivity between the customer locations. Routing options range from static to dynamic and include RIP, Enhanced Interior Gateway Routing Protocol (EIGRP), OSPF, IS-IS, or even BGP, depending on what the service provider offers. Different locations could use different routing protocols, though this situation typically would not occur.

With an MPLS VPN deployment, the service provider can also offer Internet connectivity through the same MPLS core network, either through a special Internet VPN or through a global routing table. To exchange the Internet routing information, either BGP or a default route is used.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- BGP is the external routing protocol used between autonomous systems.
- Autonomous system is a collection of networks under a single administration and is represented by 16-bit or 32-bit number.
- Forwarding is based on policies and not on best path. BGP routers exchange network reachability information, called path vectors, which are made up of path attributes.
- BGP announces set of AS numbers and networks that are reachable at the end of the path. The path is described by attributes.
- BGP router can advertise to neighboring autonomous systems only those routes that it actually uses.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4-13

Summary (Cont.)

- BGP routers establish a TCP session and then exchange routing tables. After that, BGP peers send only incremental triggered updates.
- BGP uses three databases: neighbor table, BGP table, and routing table.
- The four BGP message types are open, keepalive, update, and notification.
- BGP4, with multiprotocol extensions, enables the use of many address families. Address families define the type of addresses being carried.
- MPLS VPNs are used by customers with multiple locations that do not want to use expensive Layer 2 technologies.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4-14

Implementing Basic BGP Routing

Overview

This lesson presents the commands and configuration examples that are needed to configure Border Gateway Protocol (BGP) properly. After a successful configuration, BGP will be able to establish a neighbor relationship, set the next-hop address, set the source IP address of a BGP update, and announce networks to other BGP routers.

A router that is running BGP goes through several neighbor states, through which BGP progresses to establish a BGP session, and offers hints for troubleshooting BGP, because the session can be stuck in the stuck-in-active (SIA) or idle state. External BGP (EBGP) configuration, Internal BGP (IBGP) configuration, and authentication between neighbors are explained.

BGP is used to perform policy-based routing (PBR). To manipulate the best paths that are chosen by BGP, you need to understand the different attributes that BGP uses and how BGP selects the best path that is based on these attributes. This lesson also explains the BGP path selection process and the BGP attributes and their characteristics.

Objectives

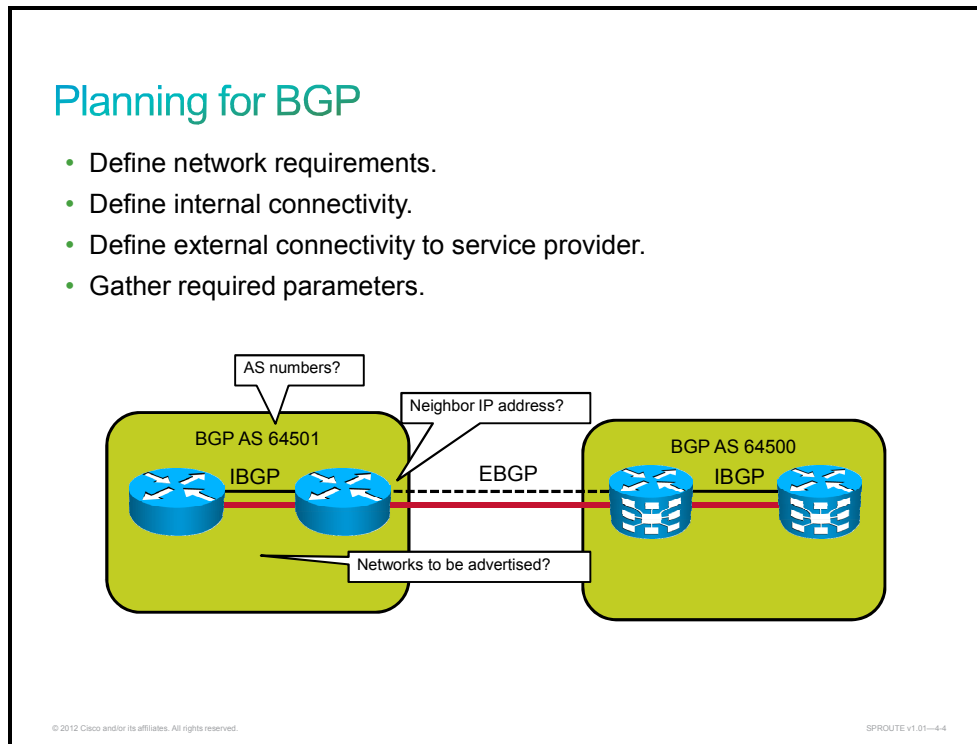
Upon completing this lesson, you will be able to implement internal and external BGP sessions to enable service provider routing. You will be able to meet these objectives:

- Describe planning for BGP deployments
- Describe the basic configuration steps for EBGP
- Describe how networks are advertised in BGP networks
- Describe the basic configuration steps for IBGP
- Describe full-mesh IBGP networks
- Describe BG support for IPv6
- Describe how to shut down a BGP neighbor
- Describe next-hob behavior in BGP
- Describe the BGP next hop self feature

- Describe configuration template support in Cisco IOS XR Software
- Describe the BGP neighbor states
- Describe BGP neighbor authentication
- Describe clearing the BGP session
- Describe how to monitor BGP routes
- Describe the importance of the BGP path attributes in the path selection
- Describe the BGP weight attribute
- Describe the BGP local preference attribute
- Describe the BGP AS path attribute
- Describe the BGP multi-exit discriminator attribute

Planning for BGP

This topic describes planning for BGP deployments.



Before initiating BGP configuration, a network administrator must define the network requirements, including the internal connectivity for IBGP design and configuration, as well as the external connectivity to the service provider for EBGP design and configuration. The next step is to gather all the parameters that are needed to provide enough details for a network operator to start the BGP configuration.

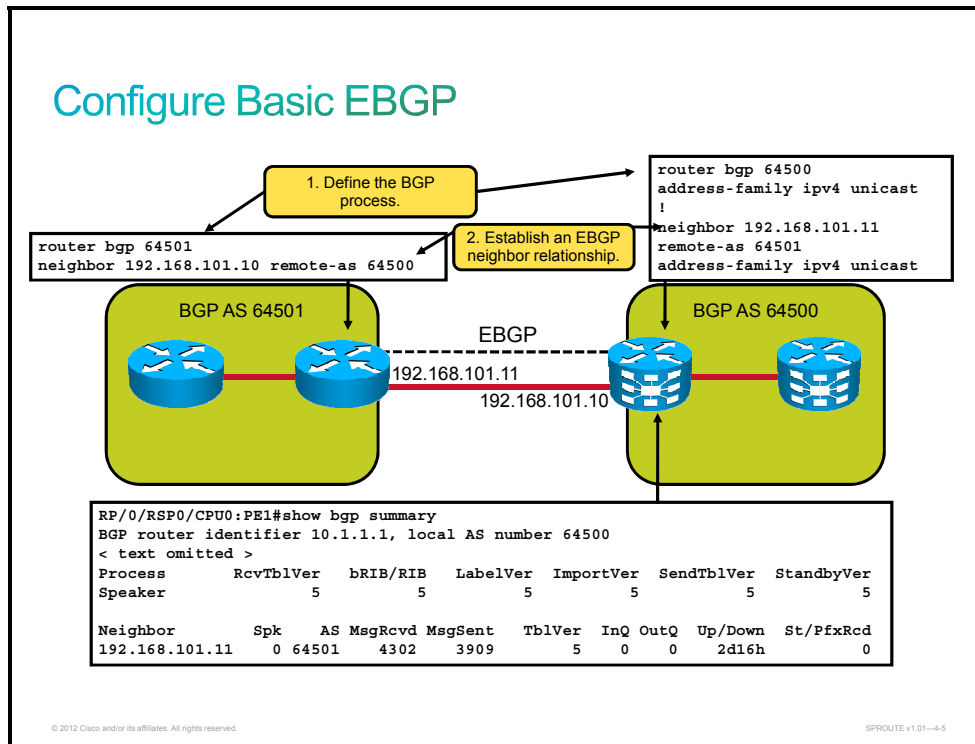
The requirements to configure basic BGP include these details:

- Autonomous system (AS) numbers (your own, and all remote AS numbers)
- All the neighbors (peers) that are involved in BGP, and IP (IPv4 or IPv6) addressing that is used among the BGP neighbors
- Networks that need to be advertised into BGP

A typical BGP configuration involves configuring BGP between a customer network and a service provider. This process is called EBGP. Many times, IBGP is required, as well as all the collected details for a complete configuration.

Configure Basic EBGP

This topic describes the basic configuration steps for EBGP.



Basic EBGP configuration requires three main steps:

- Define the BGP process.
- Establish the neighbor relationship.
- Advertise the networks into BGP. (next figure)

To perform these steps, information is needed about the neighbors: which AS numbers are used, which IP address is used as the IP address on a remote router (neighbor), and which network will be advertised.

The syntax of the basic BGP configuration commands is like the syntax for configuring internal routing protocols. However, there are significant differences in how BGP functions.

Use the Cisco IOS/IOS XE/IOS XR **router bgp as-number** command to notify the router that any subsequent subcommands belong to this routing process. This command also identifies the local AS in which this router belongs. The router needs to be informed of the AS number so that it can determine whether the BGP neighbors that are to be configured next are IBGP or EBGP neighbors. To establish a connection to another AS, insert the AS number with a Cisco IOS XR **neighbor** router BGP command or Cisco IOS/IOS XE **neighbor remote-as** router BGP command so that the router can properly identify the relationship between the neighboring router and itself.

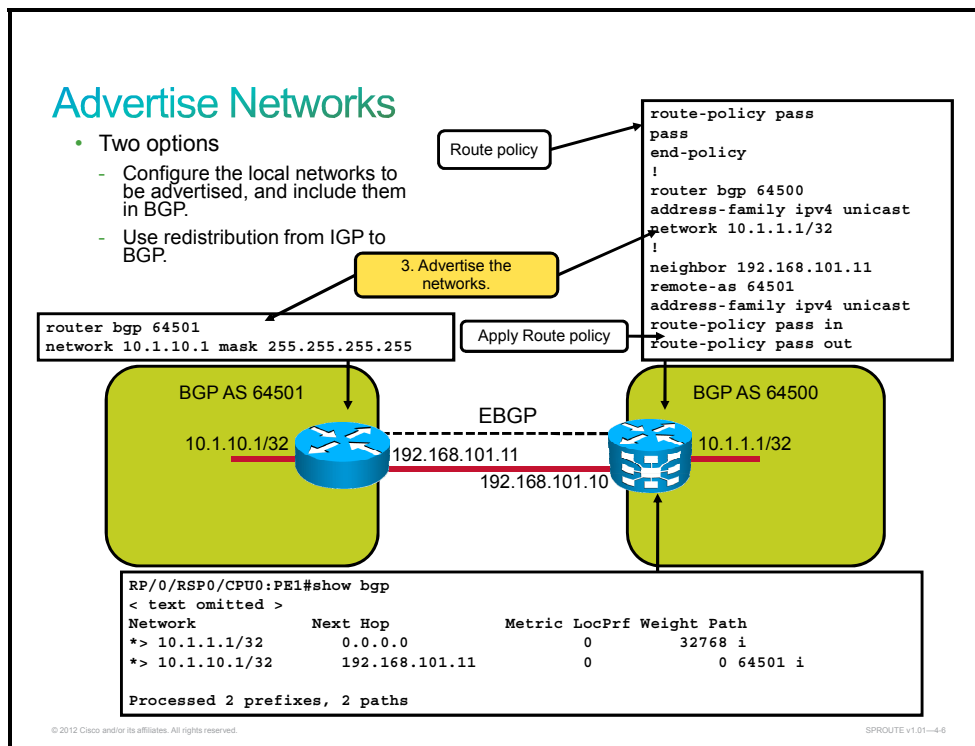
Only one instance of BGP can be configured on the router at a single time. On the Cisco IOS XR router, enable **address-family** for IPv4 unicast to enable IPv4 unicast capability. The router will start sending and receiving IPv4 routes.

The Cisco IOS XR **show bgp summary** or Cisco IOS/IOS XE **show ip bgp summary** command is one way to verify the neighbor relationship. The figure presents the output from a Cisco IOS XR router. Some of the details of this command output are listed:

- **BGP router identifier:** The IP address that all other BGP speakers recognize as representing this router
- **Neighbor:** The IP address that is used in the neighbor statement with which this router has a relationship
- **AS:** The AS number of the listed neighbor
- **Messages received (MsgRcvd):** The number of BGP messages that have been received from this neighbor
- **Messages sent (MsgSent):** The number of BGP messages that are sent to this neighbor
- **Table version (TblVer):** The BGP table version
- **In queue (InQ):** The number of messages waiting to be processed from this neighbor
- **Out queue (OutQ):** The number of messages that are queued and waiting to be sent to this neighbor. TCP flow control prevents this router from overwhelming a neighbor with a large update.
- **Up/Down:** The length of time that this neighbor has been in the current BGP state (established, active, or idle)
- **State (established, active, idle, open sent, open confirm, or idle [admin]):** The BGP state. A neighbor can be set to administratively shut down (admin state) by using the Cisco IOS XR **shutdown** router BGP command or Cisco IOS/IOS XE **neighbor shutdown** router BGP command.
- **Prefix received (PfxRcd):** When the session is in the established state, this value represents the number of BGP network entries that are received from the listed neighbor.

Advertising BGP Networks

This topic describes how networks are advertised in BGP networks.



Two options exist when advertising networks into the BGP. The first option is using the Cisco IOS/IOS XE/IOS XR **network** router BGP command to define the networks that are required. The second option is the redistribution of interior gateway protocol (IGP) routes into the BGP routing process.

Use the Cisco IOS/IOS XE/IOS XR **network** router BGP command to permit BGP to advertise a prefix if it is present in the IP routing table. The **network** command determines the networks that the router originates. This concept is different from using the **network** command when configuring IGP. Unlike IGP, the **network** command does not start BGP on specific interfaces; rather, it indicates to BGP which networks it should originate from this router.

The Cisco IOS XR route policy must be applied in order to pass routes in the inbound and outbound directions. Configure route-policy to pass all routes and use Cisco IOS XR **route-policy** router BGP command to apply route policy to the inbound and outbound directions.

Use the Cisco IOS XR **show bgp** or Cisco IOS/IOS XE **show ip bgp** command to display the BGP topology database (BGP table). The figure shows a partial sample output of the Cisco IOS XR **show bgp** command; the complete output is shown here:

```
RP/0/RSP0/CPU0:PE1#show bgp
```

```
Mon Jun 19 22:11:52.515 UTC
```

```
BGP router identifier 10.1.1.1, local AS number 64500
```

```
BGP generic scan interval 60 secs
```

```
BGP table state: Active
```

```
Table ID: 0xe0000000 RD version: 7
```

```
BGP main routing table version 7
```

BGP scan interval 60 secs

Status codes: s suppressed, d damped, h history, * valid, > best
i - internal, r RIB-failure, S stale

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 10.1.1.1/32	0.0.0.0	0		32768	i
*> 10.1.10.1/32	192.168.101.11	0		0	64501 i

Processed 2 prefixes, 2 paths

The status codes are shown at the beginning of each line of output, and the origin codes are shown at the end of each line. In this output, there is an asterisk (*) in most of the entries in the first column. This asterisk means that the next-hop address is valid. The next-hop address is not always the router that is directly connected to this router. Here are some other options:

- An “s,” for suppressed, indicates that the specified routes are suppressed (usually because routes have been summarized and only the summary route is being sent).
- A “d,” for dampening, indicates that the route is being dampened (penalized) for going up and down too often. Although the route might be up right now, it is not advertised until the penalty has expired.
- An “h,” for history, indicates that the route is unavailable and is probably down; historic information about the route exists, but a best route does not exist.
- An “r,” for Routing Information Base (RIB) failure, indicates that the route was not installed in the RIB.
- An “S,” for stale, indicates that the route is stale (this symbol is used in the nonstop forwarding-aware router).

The second column shows “>” when BGP has selected the path as the best path to a network.

The third column is either blank or shows “i.” If it is blank, BGP learned that route from an external peer. An “i” indicates that an IBGP neighbor advertised this path to the router.

The fourth column lists the networks that the router learned.

The Next Hop column lists all the next-hop addresses for each route. This column may contain the entry 0.0.0.0, which signifies that this router is the originator of the route.

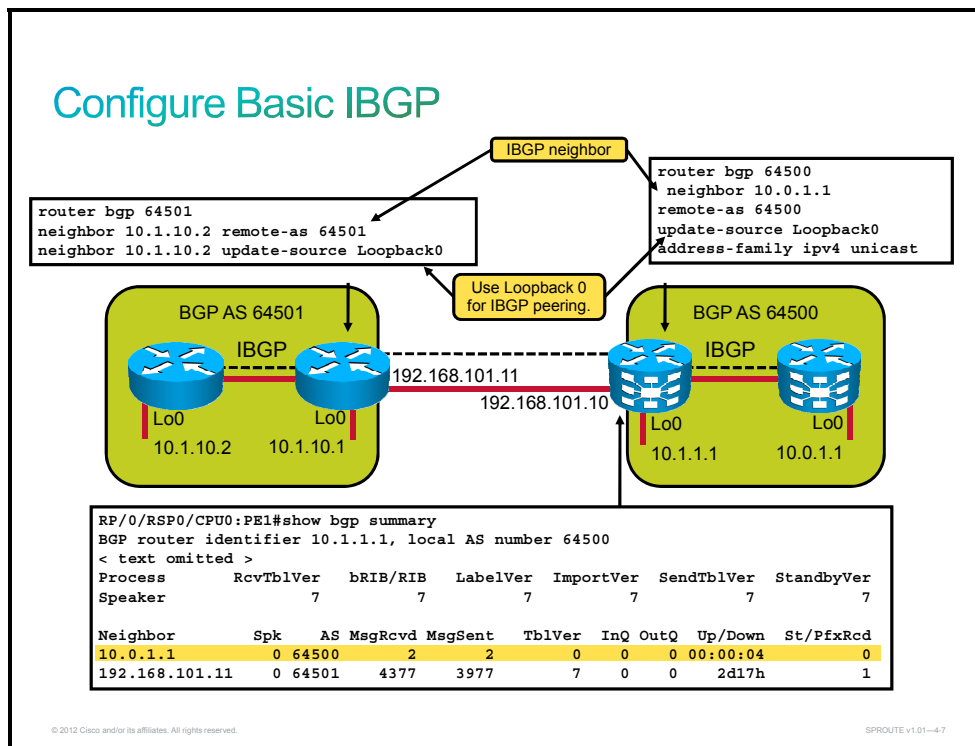
The three columns to the left of the Path column list three BGP path attributes that are associated with the path: metric (multi-exit discriminator [MED]), local preference, and weight.

The column with the Path header may contain a sequence of autonomous systems in the path. From left to right, the first AS that is listed is the adjacent AS, from which this network was learned. The last number (the rightmost AS number) is the originating AS of this network. The AS numbers between these two numbers represent the exact path that a packet takes back to the originating AS. If the path column is blank, the route is from the current AS.

The last column signifies how this route was entered into BGP on the original router. If the last column has an “i” in it, the originating router probably used a network statement to introduce this network into BGP. If the character is an “e,” the originating router learned this network from an exterior gateway protocol (EGP), which is the historical predecessor to BGP. A question mark (?) signifies that BGP cannot absolutely verify the availability of this network because it is redistributed from an IGP into BGP.

Configure Basic IBGP

This topic describes the basic configuration steps for IBGP.



To establish an IBGP session between routers, use the same neighbor AS number that is configured on the local router. To establish the IBGP session, a loopback interface is used in order to keep the BGP session stable. As long as the loopback interface is up and reachable, the BGP session will stay up.

The BGP neighbor statement informs the router of the destination IP address for each update packet. The router must decide which IP address to use as the source IP address in the BGP routing update. When a router creates a BGP packet for a neighbor, it checks the routing table for the destination network to reach that neighbor. The IP address of the outbound interface, as the routing table indicates, is used as the source IP address of the BGP packet. When a BGP packet is received for a new BGP session, the source address of the packet is compared to the list of neighbor statements. This source IP address must match the address in the corresponding neighbor statement on the other router. Otherwise, the routers will not be BGP peers because they are not able to establish the BGP session.

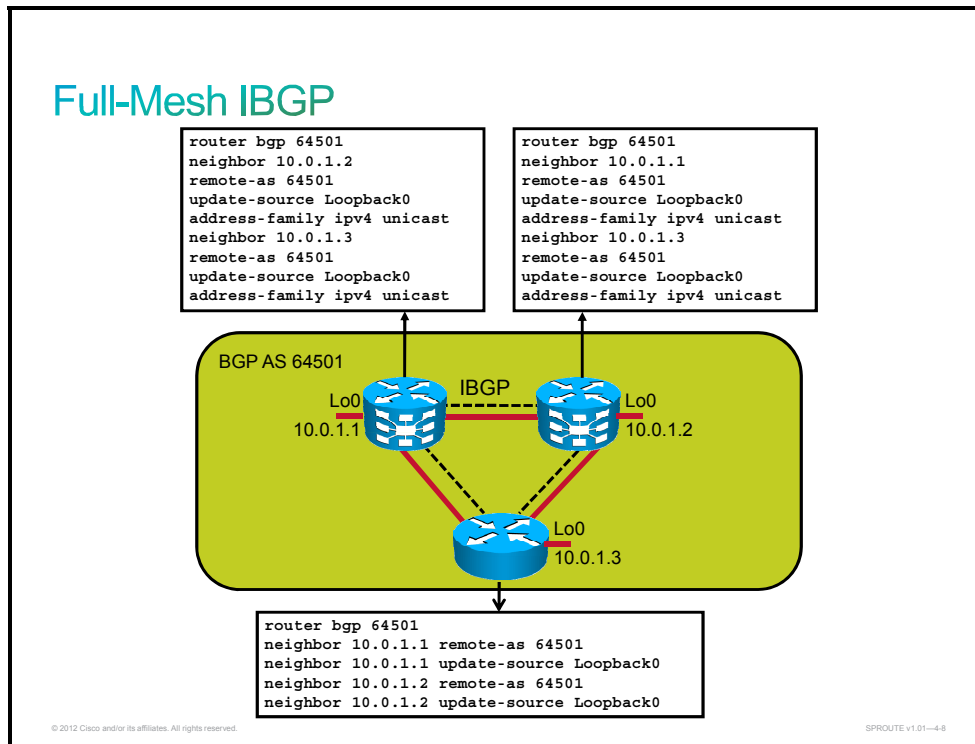
Multiple paths can exist to reach each neighbor when peering with IBGP neighboring routers. If the BGP router is using a neighbor address that is assigned to a specific interface on another router, and that interface goes down, the router that is pointing to this address loses its BGP session with that neighbor. If the router peers instead with the loopback interface of the other router, the loopback interface will always be available as long as the router itself does not fail. This peering arrangement adds resiliency to the IBGP sessions because the routers are not tied into a physical interface, which may fail for any number of reasons.

To peer with the loopback of another internal neighbor, the first router would point the neighbor statement to the loopback address of the other internal neighbor. Ensure that both routers have a route to the loopback address of the other neighbor in their routing table. Also ensure that both routers are announcing their loopback addresses into their local routing protocol.

The Cisco IOS XR **update-source** router BGP command or Cisco IOS/IOS XE **neighbor update-source** router BGP command overrides the default source IP address that is used for BGP packets. It is necessary to tell the router which IP address to use as the source address for all BGP packets, if a loopback interface is to be used instead of the physical interface.

Full-Mesh IBGP

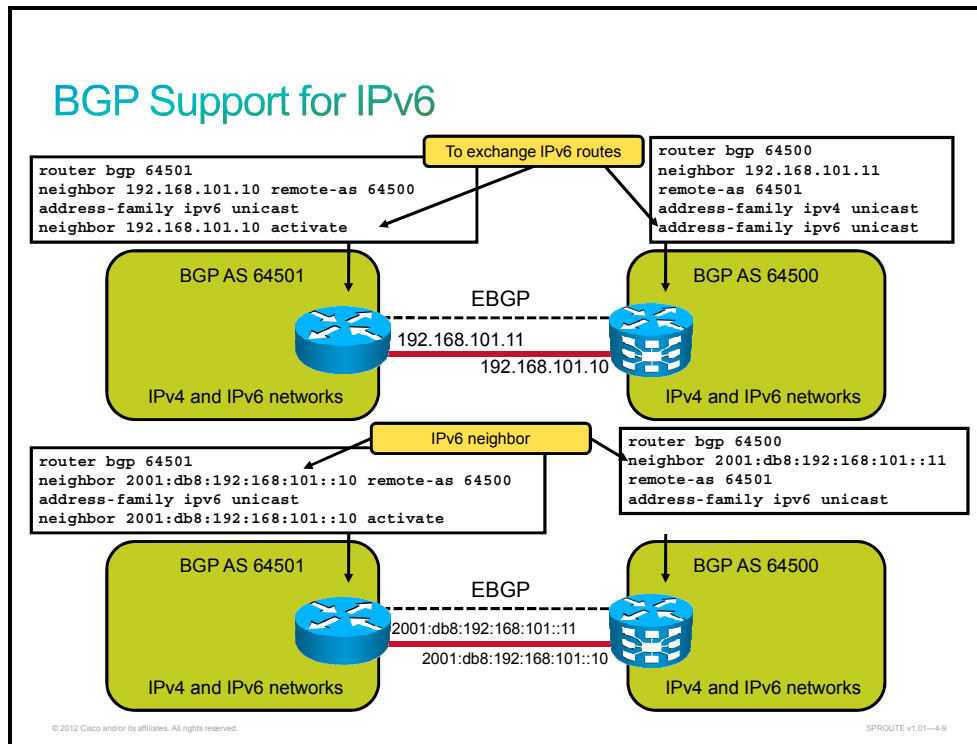
This topic describes full-mesh IBGP networks.



The figure is showing full-mesh IBGP, which is required because of the BGP split horizon rule. Instead of using full-mesh IBGP, you can also implement BGP route reflector or BGP confederation. These topics are beyond the scope of this course.

BGP Support for IPv6

This topic describes BG support for IPv6.

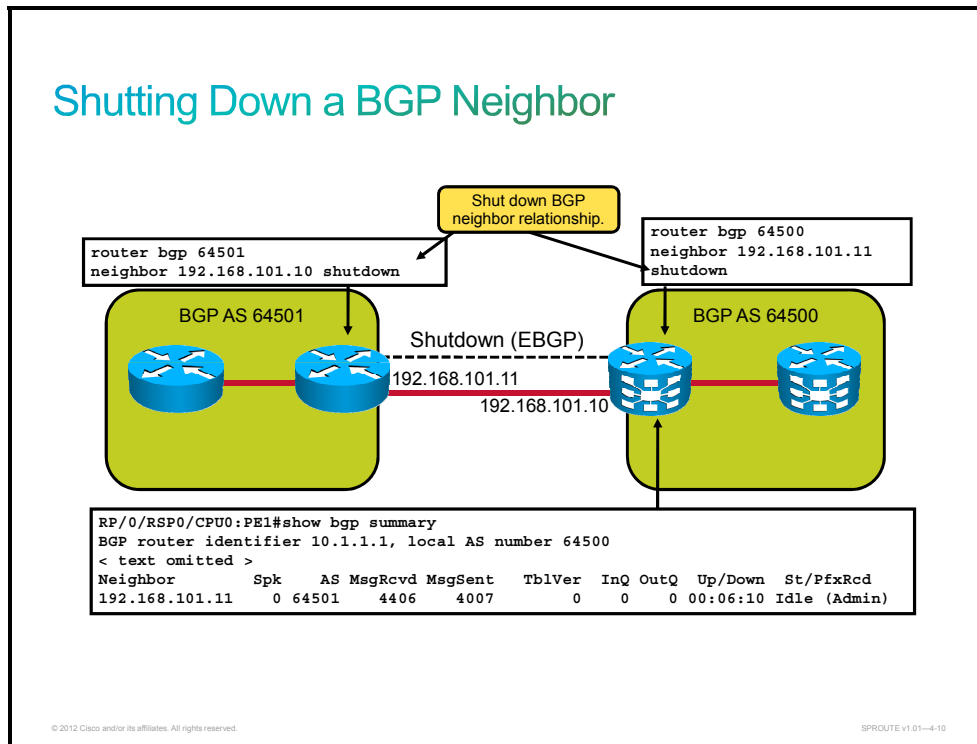


To exchange IPv4 and IPv6 routes across the BGP session, enable IPv4 and IPv6 unicast address families. The IPv4 unicast address family is already enabled by default in the Cisco IOS/IOS XE router, but you must enable it in the Cisco IOS XR router.

When a neighbor is reachable on the IPv6 address, use IPv6 instead of the IPv4 neighbor address in the Cisco IOS XR **neighbor** router BGP command or the Cisco IOS/IOS XE **neighbor remote-as** router BGP command. When the BGP session is configured by using IPv6 addresses in the Cisco IOS/IOS XE router, only IPv4 unicast routes are exchanged by default; to enable exchange for IPv6 routes, the IPv6 unicast address family needs to be enabled. On the Cisco IOS XR Software, the exchange of IPv4 prefixes is not supported for IPv6 prefixes. Only IPv6 prefixes can be exchanged between IPv6 prefixes.

Shutting Down a BGP Neighbor

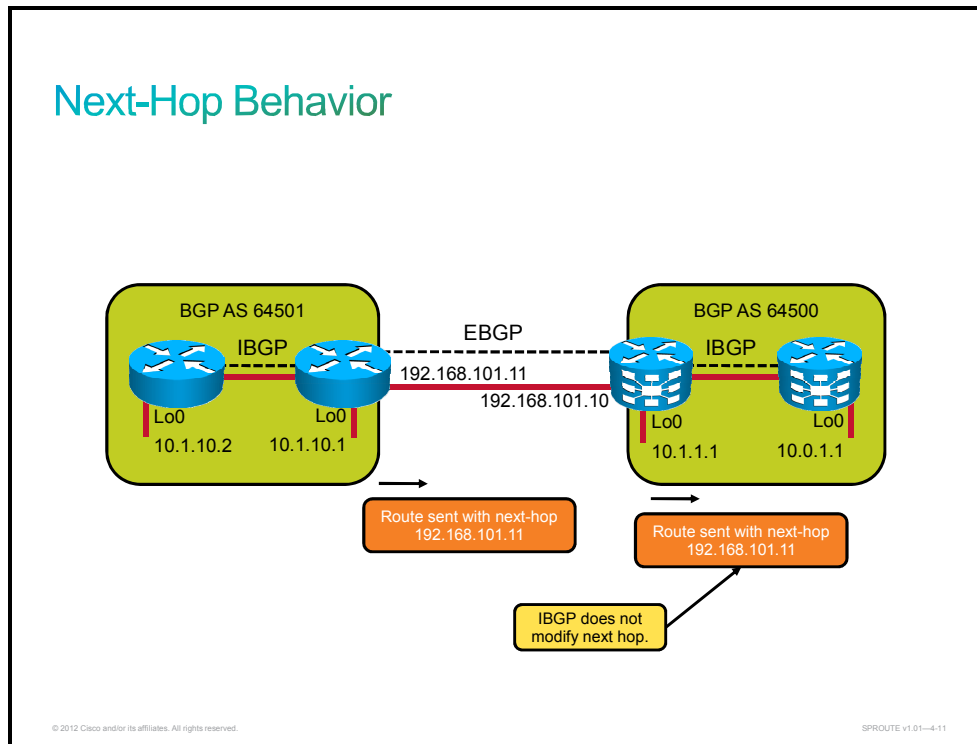
This topic describes how to shut down a BGP neighbor.



Use the Cisco IOS XR **shutdown** router BGP command or Cisco IOS/IOS XE **neighbor shutdown** router BGP command to administratively shut down a BGP neighbor. If major policy changes are implemented for a neighboring router, and multiple parameters are changed, the neighboring router must be administratively shut down; then the changes must be implemented, and the neighboring router must be brought back up with the **no** version of this command. The usage of this command not only terminates the session, but removes all associated routing information as well.

BGP Next-Hop Behavior

This topic describes next-hop behavior in BGP.



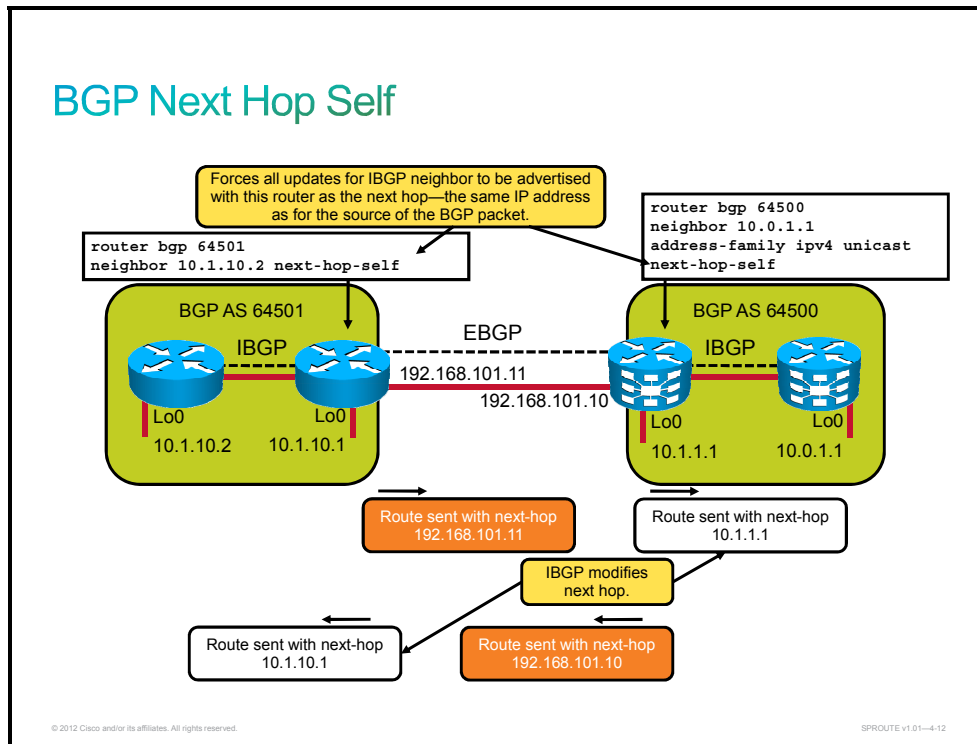
The way in which BGP establishes an IBGP relationship is very different from the way that IGPs behave. BGP is an AS-by-AS routing protocol and not a router-by-router routing protocol. Next hop is the IP address that is used to reach the next AS. The method that BGP uses to denote its next-hop address is also very different from the way that an IGP performs the same function. The default next hop is explained:

- **EBGP:** IP address of the neighbor router that is sending the update
- **IBGP:** IP address that is advertised by EBGP, and should be carried in IBGP

In the figure, the next hop for a route sent via EBGP is set to the IP address of the outgoing interface (192.168.101.11). When same route is sent to the IBGP neighbor, next hop is not changed.

BGP Next Hop Self

This topic describes the BGP next hop self feature.



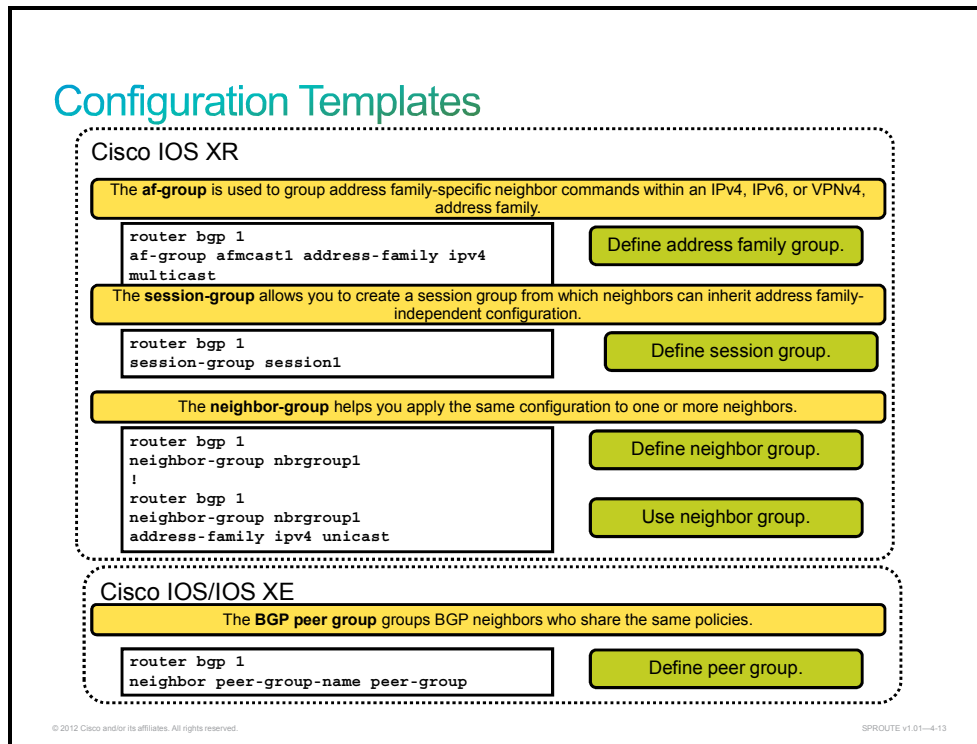
It is sometimes necessary to override the default next-hop behavior of a router and force it to advertise itself as the next-hop address for routes that are sent to a neighboring router. The Cisco IOS XR **next-hop-self** router BGP command or Cisco IOS/IOS XE **neighbor next-hop-self** router BGP command forces BGP to use its own IP address as the next-hop address for each network that it advertises to its IBGP neighbor, rather than letting the protocol choose the next-hop address to use.

An internal protocol—such as Routing Information Protocol (RIP), EIGRP, or OSPF—always uses the source IP address of a routing update as the next-hop address for each network that is placed in the routing table. This command forces BGP to use the source IP address of the update as the next-hop address for each advertised network.

The figure shows the **next-hop-self** set to the IBGP neighbors on the Cisco IOS XR and Cisco IOS/IOS XE routers. Because the loopback 0 interface is used in the IBGP peering, the update to the IBGP peer is sent with the loopback 0 IP address in the next-hop field.

Cisco IOS XR BGP Configuration Templates

This topic describes configuration template support in Cisco IOS XR Software.



The **af-group**, **session-group**, and **neighbor-group** configuration commands provide template support for the neighbor configuration in Cisco IOS XR Software.

- The **af-group** command is used to group address family-specific neighbor commands within an IPv4, IPv6, or VPNv4, address family. Neighbors that have the same address family configuration are able to use the address family group (af-group) name for their address family-specific configuration. A neighbor inherits the configuration from an address family group by way of the **use** command. If a neighbor is configured to use an address family group, the neighbor (by default) inherits the entire configuration from the address family group. However, a neighbor does not inherit all of the configuration from the address family group, if items are explicitly configured for the neighbor. The address family group configuration is entered under the BGP router configuration mode.
- The **session-group** command allows you to create a session group from which neighbors can inherit address family-independent configuration. A neighbor inherits the configuration from a session group by way of the **use** command. If a neighbor is configured to use a session group, the neighbor (by default) inherits the entire configuration of the session group. A neighbor does not inherit all of the configuration from a session group if a configuration is done directly on that neighbor.
- The **neighbor-group** command helps you apply the same configuration to one or more neighbors. Neighbor groups can include session groups and address family groups and can comprise the complete configuration for a neighbor. After a neighbor group is configured, a neighbor can inherit the configuration of the group using the **use** command. If a neighbor is configured to use a neighbor group, the neighbor inherits the entire BGP configuration of the neighbor group. However, a neighbor does not inherit all of the configuration from the neighbor group, if items are explicitly configured for the neighbor. In addition, some part of the configuration of the neighbor group could be hidden if a session group or address family group was also being used.

In the Cisco IOS/IOS XE you can group BGP neighbors who share the same policies together in what is called a **BGP peer group**. Instead of configuring each neighbor with the same policy individually, a peer group allows you to group the policies that can be applied to individual peers, thus making an efficient update calculation along with simplified configuration.

BGP Neighbor Authentication and Neighbor Reset

This topic describes BGP neighbor states.

BGP States

When establishing a BGP session, BGP goes through the following states:

1. **Idle**: The router is searching the routing table to see whether a route exists to reach the neighbor.
2. **Connect**: The router found a route to the neighbor and has completed the three-way TCP handshake.
3. **Open sent**: The open message is sent, with the parameters for the BGP session.
4. **Open confirm**: The router received an agreement on the parameters for establishing a session.
 - Alternatively, the router goes into the **active** state if there is no response to the open message.
5. **Established**: Peering is established; routing begins.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4.14

After the TCP handshake is complete, the BGP application tries to set up a session with the neighbor. Several steps must occur for the session to be established.

After the **neighbor** command is entered in BGP, BGP takes the IP address that is listed and checks the local routing table for a route to this address. At this point, BGP is in the “idle” state. If BGP does not find a route to the IP address, it stays in the idle state. If it finds a route, it goes to the “connect” state when the TCP handshaking synchronization-acknowledgment (SYN-ACK) packet returns.

After the TCP connection is complete, BGP creates a BGP open packet and sends it to the neighbor. Once BGP sends this open packet, the BGP peering session changes to the “open sent” state. If there is no response for five seconds, the state changes to the “active” state.

If a response does come back in a timely manner, BGP goes to the “open confirm” state and starts scanning (evaluating) the routing table for the paths to send to the neighbor. When those paths have been found, BGP then goes to the “established” state and begins routing between the neighbors.

BGP Established and Idle States

- **Idle:** The router cannot find the address of the neighbor in the routing table.
 - **Solution:** Check for an IGP problem. Is the neighbor announcing the route?
- **Established:** This is the proper state for BGP operations.

```

RP/0/RSP0/CPU0:PE1#show bgp summary
BGP router identifier 10.1.1.1, local AS number 64500
< text omitted >
Process      RcvTblVer    bRIB/RIB    LabelVer    ImportVer    SendTblVer    State      R
Speaker      7            7           7           7           7           7

```

A number in the state column indicating the number of routes learned from this neighbor.

Neighbor	Spk	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	St/PfxRcd
10.0.1.1	0	64500	355	343	36	0	0	00:16:28	4
192.168.101.11	0	64501	4444	4051	36	0	0	00:16:18	1


```

RP/0/RSP0/CPU0:PE1#show bgp neighbor
< text omitted >
BGP neighbor is 192.168.101.11
Remote AS 64501, local AS 64500, external link
Remote router ID 10.1.1.10.1
BGP state = Established, up for 00:16:18
Last read 00:00:49, Last read before reset 00:16:52
Hold time is 180, keepalive interval is 60 seconds
Configured hold time: 180, keepalive: 60, min acceptable hold time: 3
< text omitted >

```

The idle state is an indication that the router does not know how to reach the IP address that is listed in the neighbor statement. The router is idle due to one of these scenarios:

- It is waiting for a static route to that IP address or network to be configured.
- It is waiting for the local routing protocol (IGP) to learn about this network through an advertisement from another router.

The most common reason for a router to enter the idle state is that the neighbor is not announcing the IP address or the network toward which the neighbor statement of the router is pointing. Check these two conditions first to correct this problem:

- Ensure that the neighbor announces the route in its local routing protocol (IGP).
- Verify that an incorrect IP address has not been entered in the neighbor statement.

The established state is the desired state for the neighbor relationship. This state means that both routers have agreed to exchange BGP updates with one another and routing has begun.

Use the Cisco IOS XR **show bgp summary** command or Cisco IOS/IOS XE **show ip bgp summary** command to examine the number of prefixes received. Use the Cisco IOS XR **show bgp neighbor** command or Cisco IOS/IOS XE **show ip bgp neighbor** command to display information about the BGP connections to neighbors.

BGP Active State

Active: The router has sent an open packet and is waiting for a response.

- The state may cycle between active and idle.
- The neighbor may not know how to get back to this router due to the following reasons:
 - There is no route to the source IP address of the BGP open packet.
 - The neighbor is peering with the wrong address.
 - There is no neighbor statement for this router.
 - The AS number is misconfigured.

Active does not mean that BGP peering is established.

```
RP/0/RSP0/CPU0:PE1#show bgp summary
BGP router identifier 10.1.1.1, local AS number 64500
< text omitted >
Neighbor      Spk    AS MsgRcvd MsgSent  TblVer  InQ  OutQ  Up/Down  St/PfxRcd
10.0.1.1      0 64500      0    343      0     0     0 00:00:00  Active
```

If the router is in the active state, it has found the IP address in the neighbor statement and has created and sent out a BGP open packet. However, the router has not received a response (open confirm packet).

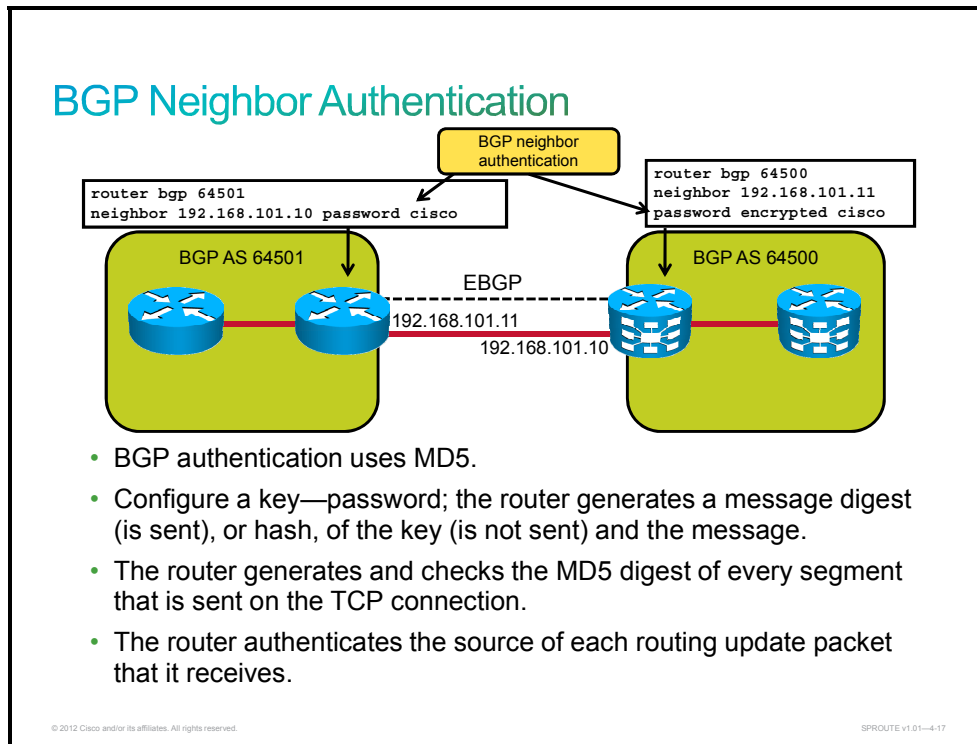
One common problem in this case is that the neighbor may not have a return route to the source IP address. Ensure that the source IP address or network of the packets has been announced to the local routing protocol (IGP).

Another common problem that is associated with the active state occurs when a BGP router attempts to peer with another BGP router that does not have a neighbor statement peering back to the first router, or when the other router is peering with the wrong IP address on the first router. Check to ensure that the other router has a neighbor statement that is peering to the correct address of the router that is in the active state.

If the state toggles between the idle state and the active state, one of the most common problems is AS number misconfiguration.

BGP Neighbor Authentication

This topic describes BGP neighbor authentication.



BGP neighbor authentication can be configured on a router so that the router authenticates the source of each routing update packet that it receives. This authentication is accomplished by the exchange of an authentication key (sometimes referred to as a password) that is known to both the sending and the receiving router.

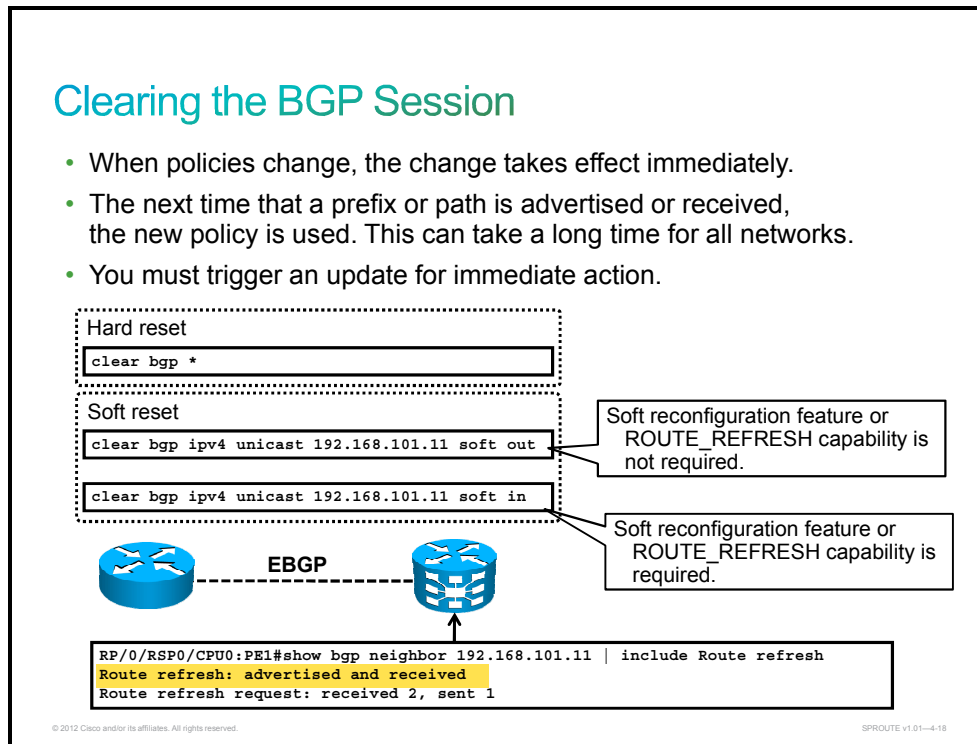
BGP supports Message Digest 5 (MD5) neighbor authentication. MD5 sends a message digest (also called a “hash”) that is created using the key and a message. The message digest is then sent instead of the key. The key itself is not sent to prevent it from being read by an eavesdropper on the line while it is being transmitted.

To enable MD5 authentication on a TCP connection between two BGP peers, use the Cisco IOS XR **password** router BGP command or Cisco IOS/IOS XE **neighbor password** router BGP command.

MD5 authentication can be configured between two BGP peers, which means that each segment that is sent on the TCP connection between the peers is verified. MD5 authentication must be configured with the same password on both BGP peers; otherwise, the connection between the peers will not be made. Configuring MD5 authentication causes Cisco IOS/IOS XE/IOS XR Software to generate and check the MD5 digest of every segment that is sent on the TCP connection.

Clearing the BGP Session

This topic describes clearing the BGP session.



BGP can potentially process huge volumes of routing information. When a policy configuration change occurs, the router cannot go through the huge table of BGP information and recalculate which entry is no longer valid in the local table; also, the router cannot determine which route or routes, already advertised, should be withdrawn from a neighbor.

There is an obvious risk that the first configuration change will be immediately followed by a second, which would cause the whole process to start all over again. To avoid such a problem, Cisco IOS/IOS XE/IOS XR Software applies changes only to those updates that are received or transmitted after the BGP policy configuration change has been performed. The new policy, enforced by the new filters, is applied only on routes that are received or sent after the change.

A network administrator who would like the policy change to be applied on all routes must trigger an update to force the router to let all routes pass through the new filter. If the filter is applied on outgoing information, the router has to resend the BGP table through the new filter. If the filter is applied on incoming information, the router needs its neighbor to resend its BGP table so that it passes through the new filters.

There are three ways to trigger an update:

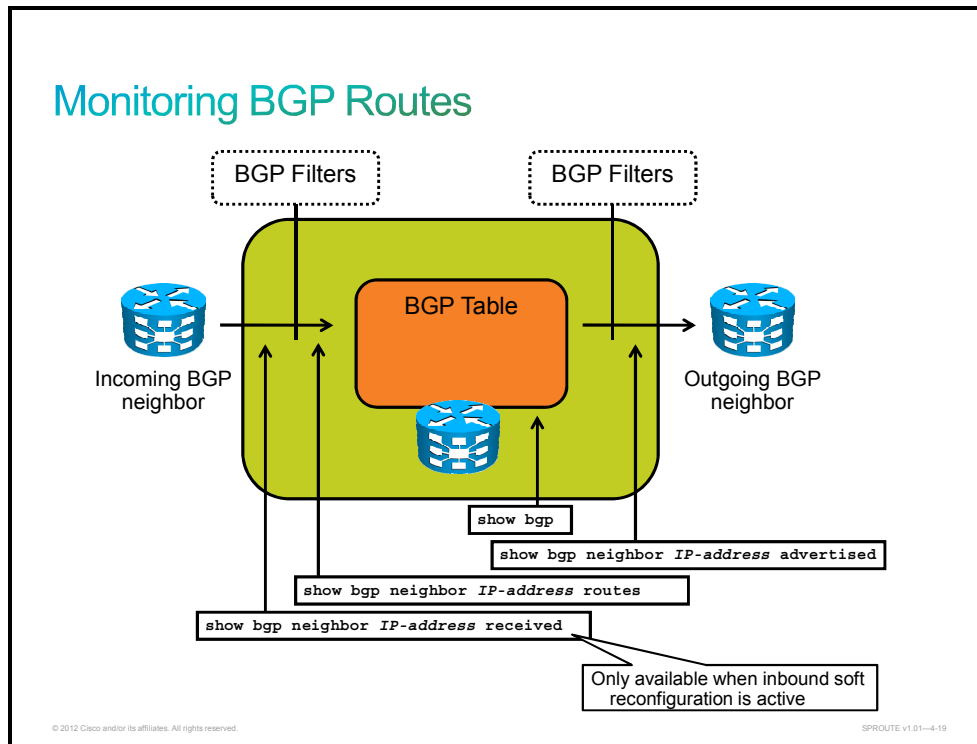
- **Hard reset:** The Cisco IOS XR `clear bgp *` command or Cisco IOS/IOS XE `clear ip bgp *` command causes the BGP forwarding table on the router that issued this command to be deleted, and all networks must be relearned from every neighbor. If a router has multiple neighbors, this action is a very dramatic event. This command forces all neighbors to resend their entire tables simultaneously.
- **Soft reset:** The Cisco IOS/IOS XE/IOS XR `soft out` option of the previous command causes BGP to do a soft reset for outbound updates. The router that is issuing the BGP soft reset does not reset the BGP session; instead, the router creates a new update and sends the whole table to the specified neighbor. This update includes withdrawal commands for the networks that the other neighbor will not see any more, based on the new outbound policy.

The Cisco IOS/IOS XE/IOS XR **soft in** option of the previous command causes BGP to do a soft reset for inbound updates; this option requires the soft reconfiguration feature to be enabled for every neighbor. This feature is very memory consuming, because the router has to store the BGP routing table received from every neighbor.

- **Route refresh:** The Cisco IOS/IOS XE/IOS XR **soft in** option of the previous command also works without the soft reconfiguration feature. If an inbound soft reset is triggered, BGP sends a REFRESH request to the neighbor, if the neighbor has advertised the ROUTE_REFRESH capability. To determine whether the neighbor has advertised the ROUTE_REFRESH capability, use the Cisco IOS XR **show bgp neighbors** command or Cisco IOS/IOS XE **show ip bgp neighbors** command.

Monitoring BGP Routes

This topic describes how to monitor BGP routes.



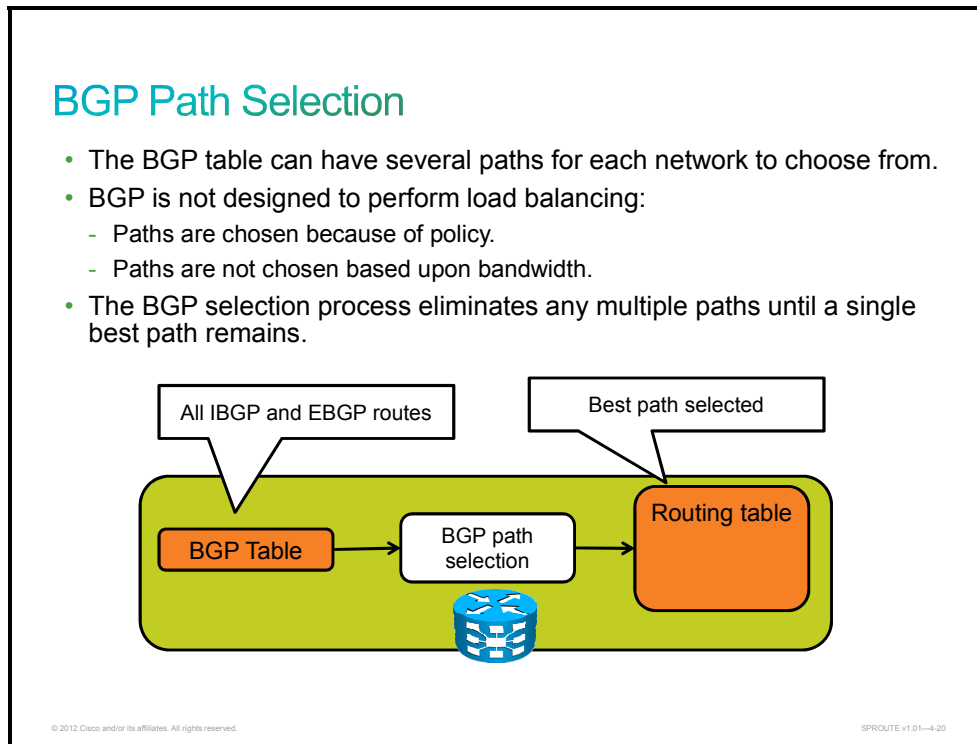
When a BGP session is reset and soft reconfiguration is used, several commands exist to monitor the BGP routes that are received, sent, or filtered.

The following commands can be used:

- Use the Cisco IOS XR **show bgp neighbor IP-address received** command or Cisco IOS/IOS XE **show ip bgp neighbor IP-address received** command to list BGP routes that are received from the BGP neighbor, before filters are applied. This command is available only when inbound soft reconfiguration is enabled for the BGP neighbor.
- Use the Cisco IOS XR **show bgp neighbor IP-address routes** command or Cisco IOS/IOS XE **show ip bgp neighbor IP-address routes** command to list BGP routes that are received from the BGP neighbor, after inbound filters are applied.
- Use the Cisco IOS XR **show bgp** command or Cisco IOS/IOS XE **show ip bgp** command to list all BGP routes that are installed into the BGP routing table.
- Use the Cisco IOS XR **show bgp neighbor IP-address advertised** command or Cisco IOS/IOS XE **show ip bgp neighbor IP-address advertised** command to list BGP routes that are sent to the BGP neighbor.

BGP Route Selection Process and Path Attributes

This topic describes the importance of the BGP path attributes in the path selection.



Routers often have several neighbors and receive routing updates from each neighbor. All routing updates enter the BGP forwarding table; as a result, multiple paths may exist to reach a given network. The entire BGP forwarding table can be displayed using the Cisco IOS XR **show bgp** command or Cisco IOS/IOS XE **show ip bgp** command.

Next, paths for the network are evaluated to determine which path is best. Paths that are not the best are eliminated from the selection criteria but kept in the BGP forwarding table in case the best path becomes inaccessible. If one of the best paths is not accessible, a new best path must be selected.

BGP is not designed to perform load balancing; paths are chosen based on policy, not based on bandwidth. The BGP selection process eliminates any multiple paths until a single best path remains.

The best path is submitted to the routing table manager process and is evaluated against any other routing protocols that can also reach that network. The router usually runs BGP and one of the interior gateway protocols (IGP)—OSPF, IS-IS, and so on—which are sending candidates for the routing table to the routing table manager. The route from the source with the lowest administrative distance is installed in the routing table. The entire routing table can be displayed using the Cisco IOS XR **show route** command or Cisco IOS/IOS XE **show ip route** command.

Route Selection Decision Process

Consider only (synchronized) routes with no AS loops and a valid next hop. The next steps in the evaluation process are:

1.	Prefer the highest weight (local to router).
2.	Prefer the highest local preference (global within AS).
3.	Prefer the route originated by the local router (next hop = 0.0.0.0).
4.	Prefer the shortest AS path.
5.	Prefer the lowest origin code (IGP < EGP < incomplete).
6.	Prefer the lowest MED (exchanged between autonomous systems).
7.	Prefer the EBGp path over the IBGP path.
8.	Prefer the path through the closest IGP neighbor.
9.	Prefer the oldest route for EBGp paths.
10.	Prefer the path with the lowest neighbor BGP router ID.
11.	Prefer the path with the lowest neighbor IP address.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4-21

After BGP receives updates about different destinations from different autonomous systems, it chooses the single best path to reach a specific destination.

The decision process is based on the BGP attributes. When faced with multiple routes to the same destination, BGP chooses the best route for routing traffic toward the destination. BGP considers only synchronized routes with no AS loops and a valid next hop.

Note BGP synchronization is disabled by default. With BGP synchronization, enabled routes learned via IBGP must be validated by the interior routing table before they can be advertised to EBGp peers.

The following process summarizes how BGP chooses the best route on a Cisco router:

1. Prefer the route with the highest weight. (Weight is proprietary to Cisco and is local to the router only.)
2. If multiple routes have the same weight, prefer the route with the highest local preference. (The local preference is used within an AS.)
3. If multiple routes have the same local preference, prefer the route that the local router originated. A locally originated route has a next hop of 0.0.0.0 in the BGP table.
4. If none of the routes was locally originated, prefer the route with the shortest AS path.
5. If the AS path length is the same, prefer the lowest origin code (IGP < EGP < incomplete).
6. If all origin codes are the same, prefer the path with the lowest multi-exit discriminator (MED). (The MED is exchanged between autonomous systems.) The MED comparison is made only if the neighboring AS is the same for all routes that are considered, unless the Cisco IOS XR **bgp bestpath med always** command or Cisco IOS/IOS XE **bgp always-compare-med** command is enabled. The Internet Engineering Task Force (IETF) decision regarding BGP MED assigns a value of infinity to the missing MED, making the route that is lacking the MED variable the least preferred. The default behavior of BGP routers is to treat routes without the MED attribute as having a MED of 0, making the route that is lacking the MED variable the most preferred. To configure the router to conform to the IETF standard, use the Cisco IOS XR **bgp bestpath med missing-as-worst** router BGP command or Cisco IOS/IOS XE **bgp bestpath missing-as-worst** router BGP command.

7. If the routes have the same MED, prefer external paths (External BGP, or EBGP) to internal paths (Internal BGP, or IBGP).
8. If synchronization is disabled and only internal paths remain, prefer the path through the closest IGP neighbor. This step means that the router will prefer the shortest internal path within the AS to reach the destination (the shortest path to the BGP next hop).
9. For EBGP paths, select the oldest route to minimize the effect of routes that are going up and down (flapping).
10. Prefer the route with the lowest neighbor BGP router ID value.
11. If the BGP router IDs are the same, prefer the router with the lowest neighbor IP address.

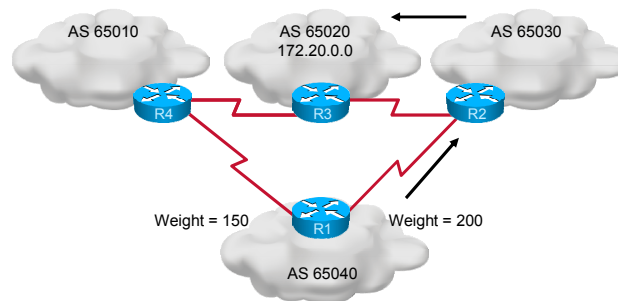
Only the best path is entered in the routing table and propagated to the BGP neighbors of the router.

BGP Weight Attribute

This topic describes the BGP weight attribute.

BGP Weight

- Weight is an attribute that is proprietary to Cisco.
- Weight is not sent to any BGP neighbors.
- It is local to the router only.
- Paths with the highest weight value are preferred.



The weight is an attribute that Cisco defines for the path selection process. The weight is configured locally on a router and is not propagated to any other routers. This attribute applies when one router is used with multiple exit points out of an AS, as opposed to the local preference attribute, which is used when two or more routers provide multiple exit points.

The weight can have a value from 0 to 65535. Paths that the router originates have a weight of 32768 by default, and other paths have a weight of 0 by default.

Routes with a higher weight are preferred when multiple routes exist to the same destination.

In the figure, R2 and R4 learn about network 172.20.0.0 from AS 65020 and propagate the update to R1. R1 has two ways to reach 172.20.0.0, and it must decide which route to take—the path through R2 or the path through R4.

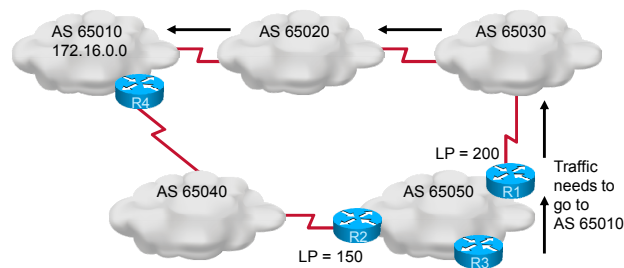
R1 sets the weight of updates that are coming from R2 to 200, and the weight of those updates that are coming from R4 to 150. Because the weight for the route that is pointing to R2 is higher than the weight for the route that is pointing to R4, R1 uses R2 as a next hop to reach 172.20.0.0.

BGP Local Preference Attribute

This topic describes the BGP local preference attribute.

BGP Local Preference

- Used to select the outbound EGP path
- Sent to IBGP neighbors only (and only within the AS)
- Stripped in the outgoing EGP updates except in the EGP updates with confederation peers
- Local preference attribute is **well known** and **discretionary**
- Default value = 100
- Paths with highest local preference value are preferred



Local preference is a well-known discretionary attribute that provides information to routers in the AS about the path that is preferred for exiting the AS. A path with a higher local preference is preferred.

The local preference is an attribute that is configured on a router and exchanged among routers within the same AS only. The default value for local preference on a Cisco router is 100.

To change the default local preference value of 100, use the Cisco IOS/IOS XE/IOS XR **bgp default local-preference** router BGP command.

In the figure, AS 65050 receives updates about network 172.16.0.0 from two neighbors. Each of the networks is advertising a different path to the destination. The local preference on R1 for network 172.16.0.0 is set to 200, and the local preference on R2 for network 172.16.0.0 is set to 150.

Because the local preference information is exchanged within AS 65050, all routers are aware of the exit point for network 172.16.0.0 out of AS 65050. In the figure, R1 is configured with a higher local preference than R2, and all the traffic that is destined for network 172.16.0.0 will be sent to R1 as an exit point from AS 65050.

BGP AS Path Attribute

This topic describes the BGP AS path attribute.

BGP AS Path

- **Fourth BGP path selection criteria**
- Prefer shorter AS paths (only length is compared)
- Influences the inbound path selection in a multihomed AS
- Manual manipulation of AS path length—AS path prepending
- AS path prepending specified per neighbor by complex criteria

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-4-24

When connections to multiple providers are required, it is important that BGP selects the optimum route for traffic to use. The optimum, or best, route may not be what the network designer intended, based on design criteria, administrative policies, or corporate mandate.

It is fairly easy for an AS to select the appropriate path for outgoing traffic. It is much more complicated to influence other autonomous systems to select the appropriate path for traffic that is returning to a specific AS. It is unlikely that the operator of an AS can request changes in router configurations in another AS. This limitation makes it virtually impossible to influence another AS to select the desired path that is based on the weight and local preference attributes, because both options would require configuration changes in the neighboring AS.

If no BGP path selection tools are configured on the route to influence the traffic flow, BGP will use the shortest AS path—the fourth option in the path selection process. If the AS path is not manually manipulated by some administrative means, the path that is going over the fewest number of autonomous systems is selected by the router, regardless of available bandwidth.

However, if the AS that is attempting to influence the incoming traffic flow is sending out EBGp updates with a manipulated AS path attribute over that undesired path, the receiver of this update is less likely to select it as the best, because the AS path now appears to be longer. AS path prepending potentially allows the customer to influence the route selection of its service providers. The AS path is extended with multiple copies of the AS number of the sender. There is no exact mechanism to calculate the required prepended AS path length.

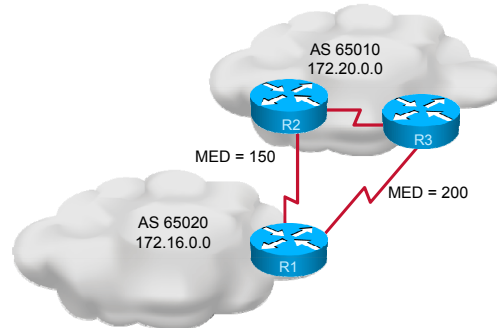
The benefit of manipulating AS paths to influence route selection is that the configuration that is needed is done in the AS that is requesting a desired return path.

BGP Multi-Exit Discriminator Attribute

This topic describes the BGP multi-exit discriminator attribute.

BGP Multi-Exit Discriminator

- The paths with the lowest MED (also called the metric) value are the most desirable.
- MED is used to advertise an exit path to be used by EBGp neighbors to reach networks owned by this AS.
- The MED attribute is **optional** and **nontransitive**.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01-4.25

The MED attribute, also called the metric, is an optional nontransitive attribute.

The MED is an indication to EBGp neighbors about the preferred path into an AS. The MED attribute is a dynamic way to influence another AS about which path it should choose to reach a certain route, when multiple entry points into an AS exist. A lower metric is preferred.

Unlike local preference, the MED is exchanged between autonomous systems. The MED is sent to EBGp peers. Those routers propagate the MED within their AS, and the routers within the AS use the MED but do not pass it on to the next AS. When the same update is passed on to another AS, the metric is set back to the default of 0. To change this value, use the Cisco IOS/IOS XE/IOS XR **default-metric** router BGP command. All routes that are advertised to an EBGp neighbor are set to the value that is specified using this command.

MED influences inbound traffic to an AS, and local preference influences outbound traffic from an AS. By default, a router compares the MED attribute only for paths from neighbors in the same way AS. The MED attribute means that BGP is the only protocol that can affect how routes are sent into an AS.

In the figure, the R2 MED attribute is set to 150, and the R3 MED attribute is set to 200. When R1 receives updates from R2 and R3, it picks R2 as the best next hop because its MED of 150 is less than the R3 MED of 200.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- For a BGP configuration, the following must be defined: BGP requirements, BGP parameters, and connectivity.
- Basic EBGP configuration requires three main steps: define the BGP process, establish the neighbor relationship, and advertise the networks into BGP.
- Networks can be advertised into BGP using the network command or by redistribution.
- It is recommended to use Loopback interfaces when establishing IBGP sessions.
- Full-mesh IBGP is required because of the split horizon rule.
- To exchange IPv6 networks over BGP, you have to activate a neighbor for IPv6 address family.
- You can use the shutdown command to manually shut down a BGP neighbor.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4.26

Summary (Cont.)

- By default, next-hop over EBGP session is IP address of a router that is sending the update. Next-hop over IBGP session is as advertised by EBGP, and should not change.
- You can use the next-hop-self command to change the default next-hop behavior.
- You can use configuration templates to group configuration that can be applied to several BGP neighbors.
- When establishing a BGP session, the BGP goes through the following states: idle, connect, open sent, open confirm, and established.
- BGP supports MD5 authentication to authenticate each received routing packet.
- You should trigger a BGP update by resetting a BGP session when you change routing policy.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4.27

Summary (Cont.)

- Use various show commands to verify BGP operations.
- After BGP receives updates about multiple destinations from different autonomous systems, it follows a multiple-step process for selecting the best route to reach a destination; the best route is a candidate for the routing table.
- Weight is Cisco's proprietary attribute that is configured locally on a router and is not propagated to any other routers.
- Local preference is a well-known discretionary attribute that provides information to routers in the AS about the path that is preferred for exiting the AS.
- AS path length is the fourth path selection criteria. The shortest AS path is preferred.
- MED is an indication to EBGP neighbors about the preferred path into an AS.

Module Summary

This topic summarizes the key points that were discussed in this module.

Module Summary

- Enterprise requirements should be considered when implementing connectivity to a service provider. Type of connectivity then influences whether static routing or BGP will be used.
- BGP is a path-vector, policy-based routing protocol that allows routing policy decisions at the AS level to be enforced.
- BGP forms EBGP relationships with external neighbors and IBGP relationships with internal neighbors. BGP controls traffic flow using multiple BGP path attributes. BGP uses a multistep process for selecting the best path to reach a destination.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—4-1

The Internet has proven to be a valuable tool to many companies and service providers, resulting in multiple redundant connections between autonomous systems. The function of Border Gateway Protocol (BGP) is to provide alternatives to using default and static routes to control path selections.

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

Q1) What is *not* a requirement for enterprise-to-service provider connectivity? (Source: Enterprise Connectivity to Service Providers)

- A) public IP address space
- B) link type and bandwidth availability
- C) AS routing policy
- D) connection redundancy

Q2) List three routing update exchange options for enterprise-to-service provider connectivity. (Source: Enterprise Connectivity to Service Providers)

Q3) Which one of these statements is *not* the reason for using BGP as a routing update exchange mechanism? (Source: Enterprise Connectivity to Service Providers)

- A) A customer deploys BGP to announce its public networks.
- B) A BGP is typically used for inter-AS routing.
- C) Customer routers are connected to service provider PE routers.
- D) Customer network implementation requires a complete Internet routing table.

Q4) What are four enterprise-to-service provider connection options? (Source: Enterprise Connectivity to Service Providers)

Q5) What is a characteristic of dual-multihomed service provider connectivity? (Source: Enterprise Connectivity to Service Providers)

- A) a connection to two or more different service providers, with two links per service provider
- B) a connection to multiple service providers, with one link per service provider
- C) the default route pointing to each service provider from an enterprise network
- D) each service provider announcing a default route with a different metric to the enterprise network

Q6) What are three common ways to perform multihoming? (Choose three.) (Source: Enterprise Connectivity to Service Providers)

- A) Each service provider passes only a default route to the AS.
- B) Each service provider passes a default route and provider-owned specific routes to the AS.
- C) Each service provider passes selected provider-owned routes, but no default routes, to the AS.
- D) Each service provider passes all routes to the AS.

- Q7) Which statement about the AS is true? (Source: Introducing BGP Routing)
- A) The AS is a collection of networks under a single administrative domain.
 - B) The AS is a collection of networks that belong to one enterprise network.
 - C) The AS requires IGP protocol to exchange routing information between autonomous systems.
 - D) EBGP neighbors must be configured within the same AS.
- Q8) What are the two typical reasons for multihoming? (Choose two.) (Source: Introducing BGP Routing)
- A) to increase the reliability of the connection to the Internet
 - B) to increase the performance of the connection
 - C) to increase the bandwidth of the connection
 - D) to simplify the IGP protocol configuration
- Q9) What is a drawback of having all of your connections to a single service provider? (Source: Introducing BGP Routing)
- A) It has redundancy with the multiple connections.
 - B) Connectivity issues in that single service provider can cause your autonomous system to lose connectivity to the Internet.
 - C) It is not tied into the routing policy of a single connection.
 - D) It has more paths to the same networks for better policy manipulation.
- Q10) Which two conditions are valid reasons to run BGP in an AS? (Choose two.) (Source: Introducing BGP Routing)
- A) The AS has only a single connection to another AS.
 - B) Path and packet flow manipulation is required in this AS.
 - C) You have a limited understanding of BGP routing and route filtering.
 - D) The AS is a service provider.
- Q11) Which routing method best describes BGP? (Source: Introducing BGP Routing)
- A) distance vector
 - B) link state
 - C) path vector
 - D) hybrid of link state and distance vector
- Q12) Which protocol does BGP use? (Source: Introducing BGP Routing)
- A) IP protocol number 88
 - B) IP protocol number 89
 - C) UDP port 520
 - D) TCP port 179
- Q13) Which four message types are defined by BGP? (Source: Introducing BGP Routing)
- _____
- _____
- _____
- _____

- Q14) By default, what are two conditions for routers to be EBGp neighbors? (Choose two.) (Source: Implementing Basic BGP Routing)
- A) Routers must be in the same AS.
 - B) Routers must be in different autonomous systems.
 - C) Routers are running an IGP between them to establish an adjacency.
 - D) Routers are directly connected.
- Q15) What does BGP use during the best path selection process? (Source: Implementing Basic BGP Routing)
- A) speed
 - B) AS routing policy
 - C) bandwidth plus delay
 - D) number of routers to reach a destination network
- Q16) Which four parameters are required for a basic BGP configuration? (Source: Implementing Basic BGP Routing)
-
-
-
-
- Q17) Which Cisco IOS XR router BGP command indicates to a BGP router whether an IP address belongs to an IBGP neighbor or to an EBGp neighbor? (Source: Implementing Basic BGP Routing)
- A) **shutdown**
 - B) **update-source** *Loopback0*
 - C) **remote-as** *65010*
 - D) **next-hop-self**
- Q18) Which Cisco IOS XR router BGP command sets the source IP address of a BGP update to be the IP address of a specific interface? (Source: Implementing Basic BGP Routing)
- A) **shutdown**
 - B) **update-source** *Loopback0*
 - C) **remote-as** *65020*
 - D) **next-hop-self**
- Q19) What is the result of using this command: **router bgp 65010**? (Source: Implementing Basic BGP Routing)
- A) The BGP process starts in the router.
 - B) The BGP process starts on the interface.
 - C) The neighboring router AS is defined.
 - D) The router enters into BGP configuration mode with AS number 65010 used locally.
- Q20) The **network** command that is used in the router BGP process identifies the interfaces from which to advertise BGP updates. (Source: Implementing Basic BGP Routing)
- A) true
 - B) false

Q21) Which five states does BGP go through during the establishment of a BGP session?
(Source: Implementing Basic BGP Routing)

Q22) Which state indicates that the router does not have a path to the neighbor IP address?
(Source: Implementing Basic BGP Routing)

- A) active
- B) idle
- C) established
- D) open confirm

Q23) Which Cisco IOS/IOS XE **clear ip bgp** command is the least intrusive for resetting a BGP session after changing outbound policy for neighbor 10.5.5.5? (Source: Implementing Basic BGP Routing)

- A) **clear ip bgp ***
- B) **clear ip bgp 10.5.5.5 soft out**
- C) **clear ip bgp 10.5.5.5**
- D) **clear ip bgp 10.5.5.5 soft in**

Q24) Number the 11 BGP selection criteria in the order in which they are used, from the first to the last, when selecting the BGP path that is submitted to the IP routing table.
(Source: Implementing Basic BGP Routing)

- A) _____ prefer the path with the lowest neighbor BGP router ID
- B) _____ prefer the lowest MED
- C) _____ prefer the shortest AS path
- D) _____ prefer the oldest route for EBGp paths
- E) _____ prefer the lowest origin code (IGP < EGP < incomplete)
- F) _____ prefer the highest weight
- G) _____ prefer the path through the closest IGP neighbor
- H) _____ prefer the highest local preference
- I) _____ prefer the route that was originated by the local router
- J) _____ prefer an EBGp path over an IBGP path
- K) _____ prefer the lowest neighbor IP address

Q25) Which description applies to the local preference attribute? (Source: Implementing Basic BGP Routing)

- A) well-known mandatory
- B) well-known discretionary
- C) optional transitive
- D) optional nontransitive

Q26) Which description applies to the MED attribute? (Source: Implementing Basic BGP Routing)

- A) well-known mandatory
- B) well-known discretionary
- C) optional transitive
- D) optional nontransitive

- Q27) Which description applies to the weight attribute? (Source: Implementing Basic BGP Routing)
- A) weight is a well-known mandatory BGP attribute
 - B) weight is a well-known discretionary BGP attribute
 - C) weight is an optional transitive BGP attribute
 - D) weight is proprietary to Cisco and not advertised to other BGP routers
- Q28) Which two statements regarding local preference are true? (Choose two.) (Source: Implementing Basic BGP Routing)
- A) The higher value for local preference is preferred.
 - B) Local preference is used only between EBGp neighbors.
 - C) The lower value for local preference is preferred.
 - D) Local preference is used only between IBGP neighbors.
- Q29) Which two statements regarding weight are true? (Choose two.) (Source: Implementing Basic BGP Routing)
- A) The lower value for weight is preferred.
 - B) The higher value for weight is preferred.
 - C) Weight is used only between IBGP neighbors.
 - D) Weight is used only locally inside the router.
- Q30) Which two statements regarding the MED are true? (Choose two.) (Source: Implementing Basic BGP Routing)
- A) The higher value for the MED is preferred.
 - B) The lower value for the MED is preferred.
 - C) The MED is exchanged between autonomous systems.
 - D) The MED is local to an AS.
- Q31) Which two statements regarding the AS path are true? (Choose two.) (Source: Implementing Basic BGP Routing)
- A) The shorter AS path is preferred.
 - B) The longer AS path is preferred.
 - C) The AS path is prepended and exchanged between autonomous systems.
 - D) The AS path is local to an AS.
- Q32) Which Cisco IOS/IOS XE/IOS XR router BGP command changes the MED for all routes? (Source: Implementing Basic BGP Routing)
- A) **bgp med number**
 - B) **default-metric number**
 - C) **bgp default-metric number**
 - D) **set med number**
- Q33) The MED is used to decide how to enter an AS from neighboring autonomous systems, when multiple paths exist between two autonomous systems. (Source: Implementing Basic BGP Routing)
- A) true
 - B) false
- Q34) The MED is set inbound when a router is receiving router updates from an EBGp neighbor. (Source: Implementing Basic BGP Routing)
- A) true
 - B) false

Module Self-Check Answer Key

- Q1) C
- Q2) static routes, MPLS VPNs, BGP
- Q3) C
- Q4) single-homed, dual-homed, multihomed, dual-multihomed
- Q5) A
- Q6) A, B, D
- Q7) A
- Q8) A, B
- Q9) B
- Q10) B, D
- Q11) C
- Q12) D
- Q13) open, keepalive, update, notification
- Q14) B, D
- Q15) B
- Q16) neighbors (peers) that are involved; AS numbers that are used; IP addresses that are used; networks, which need to be advertised
- Q17) C
- Q18) B
- Q19) D
- Q20) B
- Q21) idle, connect, open sent, open confirm, established
- Q22) B
- Q23) B
- Q24) 1-F
2-H
3-I
4-C
5-E
6-B
7-J
8-G
9-D
10-A
11-K
- Q25) B
- Q26) D
- Q27) D
- Q28) A, D
- Q29) B, D
- Q30) B, C
- Q31) A, C
- Q32) B
- Q33) A
- Q34) B

Routing Protocol Tools and Route Manipulation

Overview

This module describes tools that are used for routing protocol manipulation, route redistribution, and BGP route selection. This module also presents the mechanisms that are available to service provider administrators; these mechanisms, in combination with routing protocols, are used to filter routing information or to implement desired routing policies.

Module Objectives

Upon completing this module, you will be able to describe the tools, features, and implementation steps for routing protocol manipulation. This ability includes being able to meet these objectives:

- Identify the mechanisms that are available, in combination with routing protocols, to filter routing information or to implement desired routing policies
- Describe route maps and routing policy language (RPL) and how to use them to implement routing policies
- Describe the importance of route redistribution in a complex service provider network, and list the implementation steps to successfully redistribute routes between routing protocols
- Configure BGP to influence route selection by using different BGP attributes

Introducing Routing Protocol Tools

Overview

The lesson focuses on the mechanisms that are available, in combination with routing protocols, to filter routing information or to implement desired routing policies using Cisco IOS/IOS XE and Cisco IOS XR routers. The lesson describes prefix lists and autonomous system (AS) path access lists.

Objectives

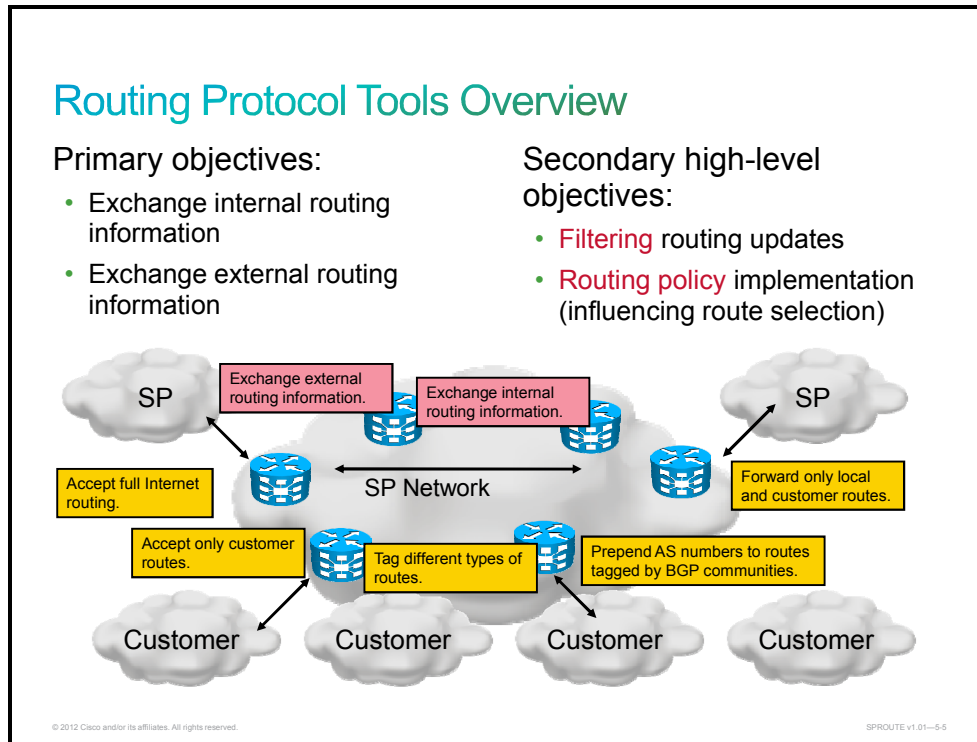
Upon completing this lesson, you will be able to identify the main characteristics of routing protocols that are used in service provider environments. You will be able to meet these objectives:

- Describe the characteristics and requirements for routing policies in service provider environments
- Provide examples of OSPF filtering
- Provide examples of BGP filtering
- Describe the most common tools used to implement route filtering
- Describe typical filtering objectives in OSPF networks
- Describe typical filtering objectives in IS-IS networks
- Describe typical filtering objectives in BGP networks
- Describe the routing policy requirements that are typically implemented using BGP
- Provide an overview of prefix lists
- Describe the syntax for prefix lists in Cisco IOS and IOS XE software
- Show an example of matching any host route in Cisco IOS and IOS XE software
- Show an example of matching a default route in Cisco IOS and IOS XE software
- Show an example of matching all routes in Cisco IOS and IOS XE software
- Show an example of matching all small prefixes in Cisco IOS and IOS XE software
- Show an example of matching all host routes in a range in Cisco IOS and IOS XE software

- Show an example of matching private networks in Cisco IOS and IOS XE software
- Describe the syntax for prefix lists in Cisco IOS XR software
- Show examples of prefix lists in Cisco IOS XR Software
- Describe the characteristics and usage scenarios for AS path-based filtering in service provider environments
- Describe the syntax for the AS path access list command in Cisco IOS and IOS XE software
- Describe the special characters used to create regular expressions
- List commonly used regular expressions
- Show an example as-path access-list to permit all routes in Cisco IOS and IOS XE software
- Show an example as-path access-list to permit local routes in Cisco IOS and IOS XE software
- Show an example as-path access-list to permit routes from a specific neighbor in Cisco IOS and IOS XE software
- Show an example as-path access-list to permit routes from a specific AS in Cisco IOS and IOS XE software
- Show an example as-path access-list to permit routes that originate in a neighboring AS in Cisco IOS and IOS XE software
- Show an example as-path access-list to permit local prefixes with AS path prepending in Cisco IOS and IOS XE software

Routing Protocol Tools Overview

This topic describes the characteristics and requirements for routing policies in service provider environments.



The figure illustrates various actions that are performed on routing updates in a typical service provider environment. The actions can be divided into two main categories:

- Exchanging routing information (the primary objective of routing protocols)
- Implementing a routing policy and filtering routing information

To exchange routing information, a typical service provider would use two routing protocols:

- An interior gateway protocol (IGP) such as Open Shortest Path First (OSPF), or Intermediate System-to-Intermediate System (IS-IS) to exchange local routing information
- Border Gateway Protocol (BGP) to exchange external routing information (for example, customer routing information and complete Internet routing information from other service providers)

BGP will always be combined with advanced filtering and policy mechanisms for security and performance reasons. This lesson will discuss various mechanisms that can be used for filtering or for routing policy implementation.

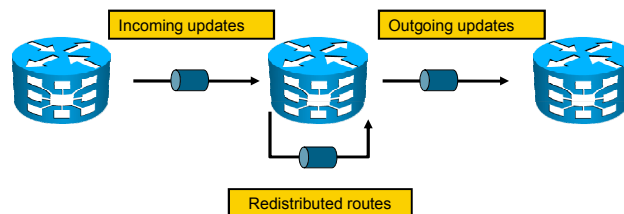
Typical Filtering Objectives

Filter:

- Incoming updates
- Outgoing updates
- Redistributed routes from other routing protocols

Filter based on:

- Prefix and prefix length (subnet mask)
- Update parameters (routing protocol-specific)



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6.6

Filtering of routing information can be performed in these three locations:

- On incoming updates, as they are received from a neighboring router
- On outgoing updates, before they are sent to a neighboring router
- On redistribution from another routing protocol, including static and connected route redistribution

Filtering can be based on the prefix itself or based on some other parameter that is available in the routing protocol or source routing protocol (when you are doing redistribution).

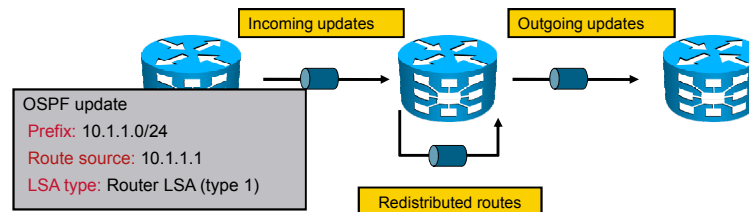
Example: OSPF Filtering

This topic provides examples of OSPF filtering.

Example: Typical OSPF Filtering Objectives

Filter OSPF based on:

- Prefix and prefix length (subnet mask)
- LSA type (internal, external, NSSA-external)
- Route source



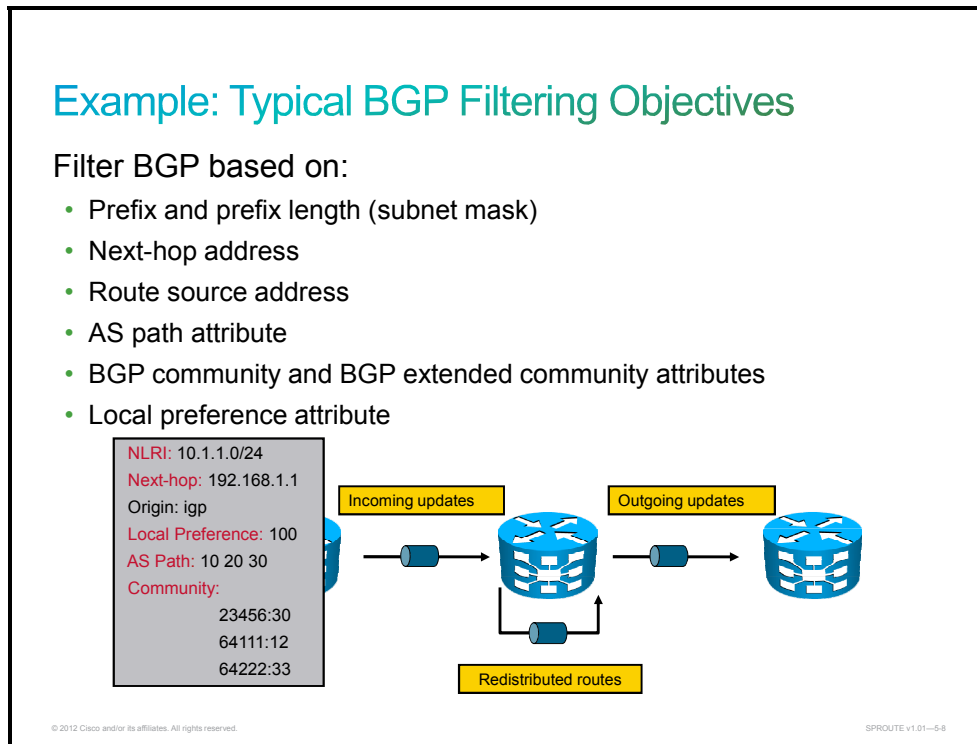
The figure illustrates an OSPF update that carries information that can be used for filtering purposes:

- Prefix and prefix length
- Route source (that is, the advertising IP address of the router)
- OSPF link-state advertisement (LSA) type

Filtering of redistributed routes can be performed on any router that effectively becomes an Autonomous System Boundary Router (ASBR), while regular filtering of OSPF updates can only be performed on Area Border Routers (ABRs) for routes forwarded from one area into another.

Example: BGP Filtering

This topic provides examples of BGP filtering.



The figure illustrates a BGP update, which has a much richer metric (that is, collection of BGP attributes) that can also be used for filtering purposes.

Filtering of BGP updates can be performed based on these attributes:

- Prefix and prefix length (subnet mask) found in the BGP Network Layer Reachability Information (NLRI)
- Next-hop address found in the BGP next-hop attribute
- Route source address (that is, the IP address of the neighbor)
- AS path attribute
- BGP community and BGP extended community attributes
- Local preference attribute

Route Filtering Tools

This topic describes the most common tools used to implement route filtering.

Filtering Tools

Prefix lists:

- Used for prefix-based filtering or matching of routes
- Can be used to match on the prefix, route source, or next-hop address

AS path access lists:

- Used in BGP for filtering or route matching based on BGP AS Path attribute

Route maps:

- Primarily used to implement complex routing policies
- Can also be used as a powerful filtering tool

Routing policy language:

- Replaces route maps in Cisco IOS XR Software
- Feature-rich language for complex routing policies

© 2012 Cisco and/or its affiliates. All rights reserved.

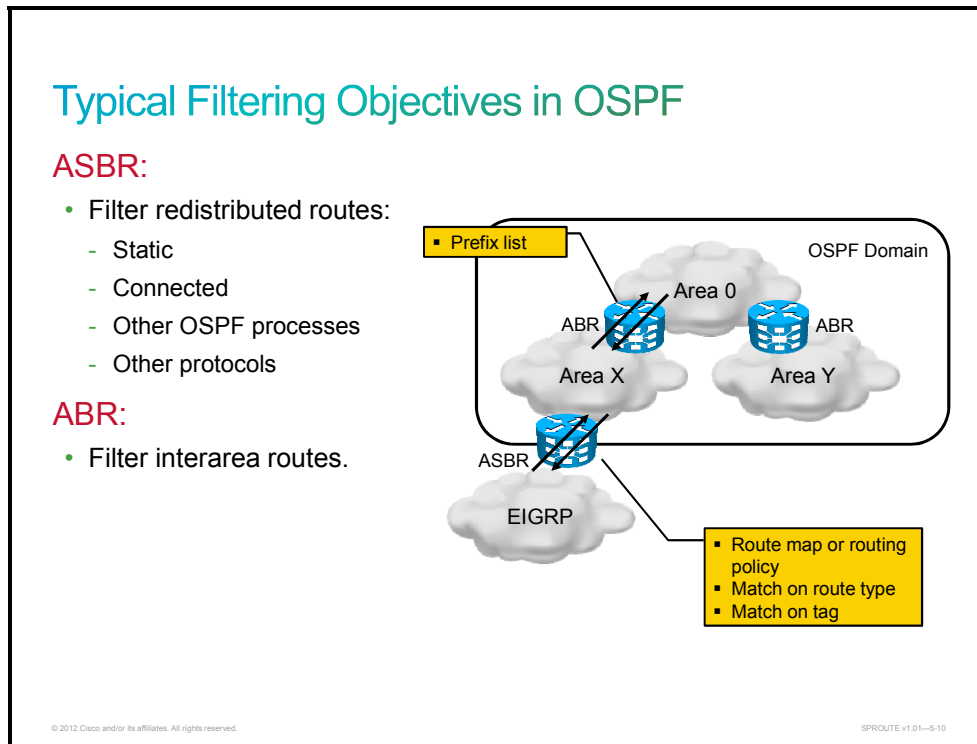
SPROUTE v1.01--5.9

The following tools are most commonly used to implement filtering and routing policies in Cisco IOS/IOS XE and/or Cisco IOS XR Software:

- **Prefix lists** can be used to implement filtering or matching routing updates that are based on IP address or IP network information; some examples are a prefix, a next-hop address, or the address of a neighbor. Prefix lists are available in Cisco IOS/IOS XE Software. Prefix lists are also available Cisco IOS XR Software with slight differences.
- **AS path access lists** can be used with BGP to implement filtering or matching of routing updates that are based on the contents of the AS path attribute. A regular expression is used to process the AS path as a string of characters. AS path access lists are only available in Cisco IOS/IOS XE Software. Cisco IOS XR Software matches AS path attributes directly in routing policies.
- **Route maps** are primarily used to implement routing policies that can modify routing protocol parameters as well as perform filtering. Route maps are only available in Cisco IOS/IOS XE Software.
- **Routing policies** are a more powerful and flexible version of the route maps that are available in Cisco IOS XR Software.

Typical Filtering Objectives in OSPF

This topic describes typical filtering objectives in OSPF networks.



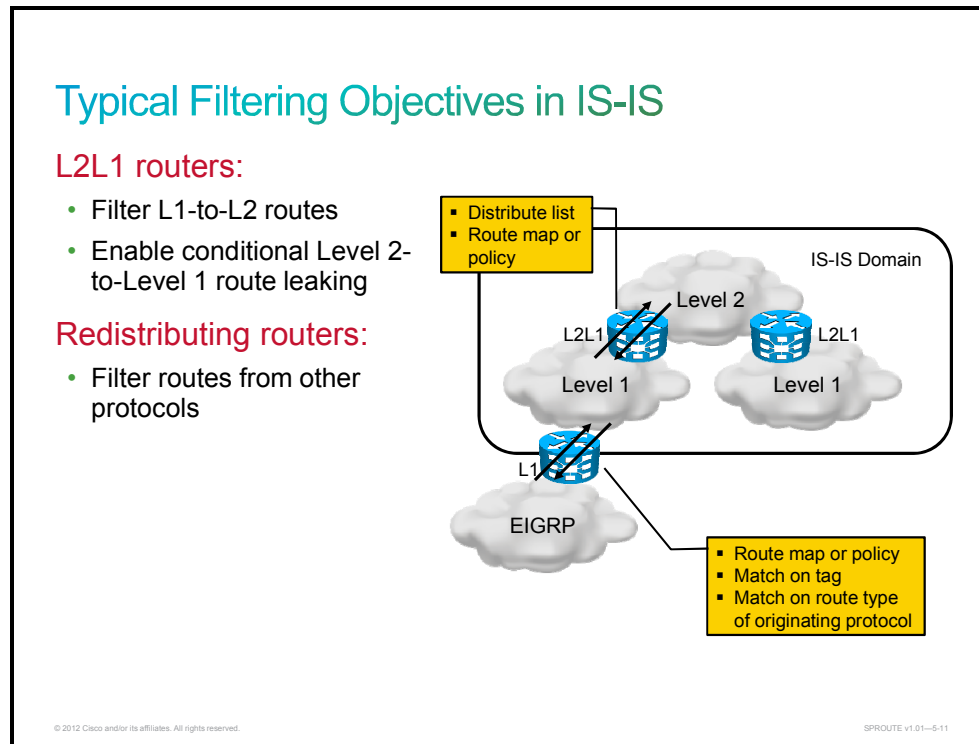
The figure illustrates an OSPF domain that is using multiple OSPF areas and a connection to an external EIGRP AS.

ASBRs perform routing exchange with other routing protocols or OSPF domains by performing redistribution of routing information. ASBRs can filter redistributed routes using route maps or routing policies from connected routes, static routes, other OSPF processes, IS-IS, Enhanced Interior Gateway Routing Protocol (EIGRP), Routing Information Protocol (RIP), or even BGP (not recommended).

ABRs perform an exchange of routing information between OSPF areas within the same OSPF domain, according to OSPF rules. Prefix lists can be used to control the exchange of routing information between areas.

Typical Filtering Objectives in IS-IS

This topic describes typical filtering objectives in IS-IS networks.



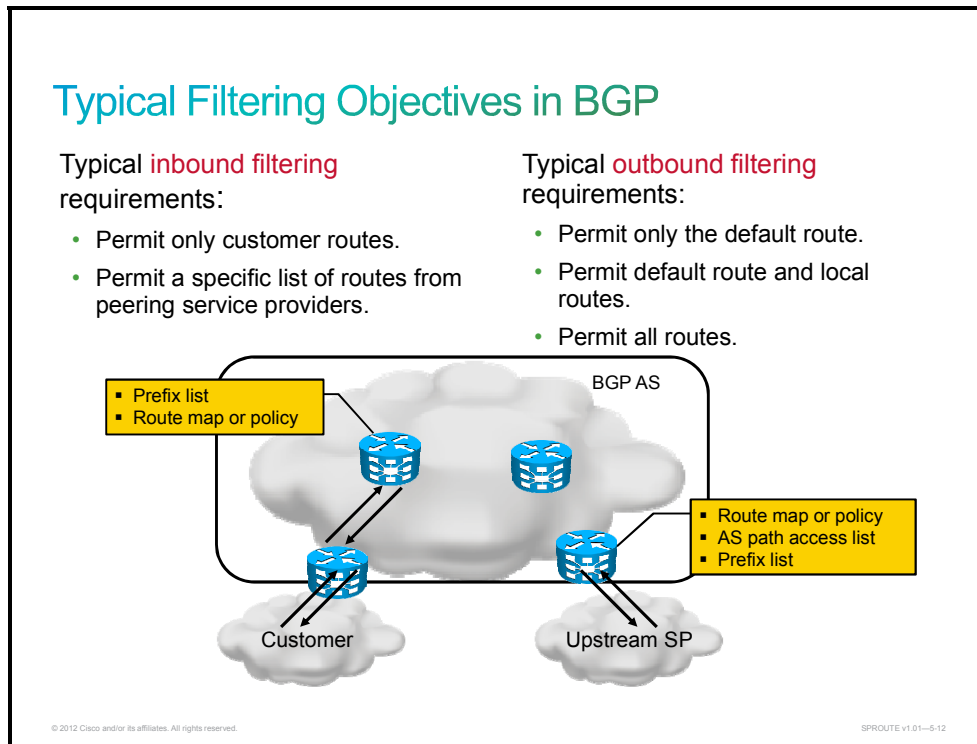
The figure illustrates an IS-IS domain that is using multiple IS-IS levels and a connection to an external EIGRP AS.

L2L1 routers (like ABRs) perform routing exchange for both IS-IS levels. Prefix lists, route maps, or routing policies can be used to filter an exchange of routing information between IS-IS levels. Route leaking can also be used to control the distribution of Level 2 routes into Level 1.

Any IS-IS router can perform redistribution from other routing protocols, using a route map or routing policy to control the redistribution of routes.

Typical Filtering Objectives in BGP

This topic describes typical filtering objectives in BGP networks.



The figure illustrates a sample BGP AS using BGP route reflectors (to reduce the full-mesh Internal Border Gateway Protocol [IBGP] requirements) and edge BGP routers to implement routing for external destinations.

Inbound filtering can depend on the type of neighboring AS:

- Permit only customer routes for end customers.
- Permit a specific list of routes from subordinate service providers, service providers that are peering at an exchange point.
- Permit the complete Internet routing information from upstream service providers.

These are outbound filtering requirements:

- Permit only the default route (for example, single-homed customers do not require more specific information; most single-homed customers do not even require a routing protocol).
- Permit default route and local routes (for example, multihomed customers that are using this service provider as a backup provider, but still want to access local destinations directly).
- Permit all routes (for example, multihomed customers that are requiring complete Internet routing information).

Typical BGP Routing Objectives

This topic describes the routing policy requirements that are typically implemented using BGP.

Typical Routing Objectives

- Complex routing policies are most often implemented using BGP.
- Influencing route selection for:
 - Outgoing traffic
 - Incoming traffic
- Routing decision influenced:
 - Locally
 - Remotely (e.g. by customer or downstream service provider)

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-13

Routing policies are most commonly implemented for external routing information, using BGP. A routing policy can address the outgoing path or the return path. Additionally, BGP can be used to implement a policy locally within an AS, or used to have a neighboring AS influence the route selection in an AS. (One example is using AS path prepending, or signaling a policy using BGP communities, which are translated to local preference in an AS).

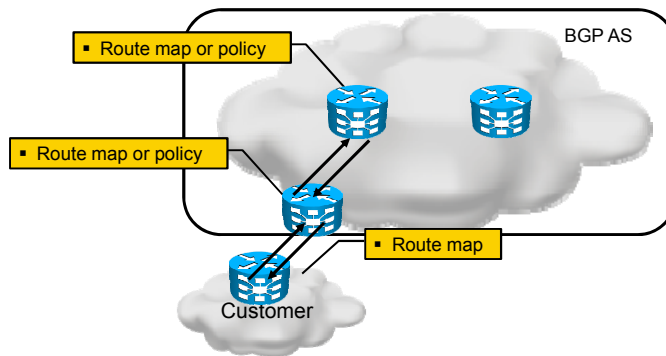
Typical Routing Objectives in BGP

Customer selecting primary or backup ISP:

- AS path prepending by customer
- BGP community sent by customer

Policy implemented by service provider:

- Setting local preference
- Translating BGP community to local preference



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5-14

A routing policy is always implemented by using route maps in Cisco IOS/IOS XE Software or routing policies in Cisco IOS XR Software. It is recommended that you implement policies that are consistent across the entire AS (that is, implement policies on edge routers).

The figure lists some commonly implemented policies in service provider environments:

- Customers often use AS path prepending to artificially lengthen the AS path attribute, thus making it less desirable (that is, to signal that this service provider is the backup service provider).
- Customers can alternatively signal their service provider preference by using some BGP communities that are offered by a service provider. The service provider will then translate the BGP communities received from the customers to some other BGP attribute (for example, AS path prepending or local preference) to influence the outbound traffic to the customers.
- Service providers can use the BGP local preference attribute to influence route selection internally, within the service provider AS (for example, to select preferred upstream service providers).

Prefix Lists Overview

This topic provides an overview of prefix lists.

Prefix List Overview

- Designed for route filtering/matching
- Replaces access-lists that were designed for packet filtering/matching
- Available in Cisco IOS/IOS XE and Cisco IOS XR Software with slight differences

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-5-15

Prefix lists are designed to simplify the filtering of routing updates. They are available in Cisco IOS, IOS XE (for the ASR router family), and IOS XR Software (with some slight differences).

Prefix Lists Syntax in Cisco IOS/IOS XE Software

This topic describes the syntax for prefix lists in Cisco IOS and IOS XE software.

Prefix Lists Syntax

Cisco IOS/IOS XE Software

- Each prefix list is identified using a case-sensitive *name*.
- Each prefix list can have one or more lines.
- Edit and order prefix list entries by using line numbers.
- The *network/length* pair identifies the bits in prefixes to match.
- The **ge** and **le** operators identify the length of prefixes to match:
 - **le** :“less or equal” matches any prefix that is shorter or equal in length to the specified value.
 - **ge** :“greater or equal” matches any prefix that is longer or equal in length to the specified value.
 - **ge x le x** :“equal” (There is no “eq” operator in Cisco IOS/IOS XE Software.)

Router(config)#

```
ip prefix-list name [seq num] {deny|permit} net/length [ge len] [le len]
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--6-16

Each prefix list is identified using a case-sensitive name (like all other named objects in Cisco IOS/IOS XE and Cisco IOS XR Software). A prefix list can have multiple lines that are ordered using line numbers.

The network/length pair identifies the bits in prefixes to match. The **ge** and **le** operators identify the length of the prefixes to match. A combination of both operators can be used to match a range of prefix lengths or a specific length: **ge x le x** ~ “equal” (There is no **eq** operator in Cisco IOS/IOS XE Software).

Full syntax:

ip prefix-list {*list-name* [*seq number*] {**deny** | **permit**} *network/length* [**ge** *ge-length*] [**le** *le-length*] | **description** *description* [*sequence-number*]

For IPv6, the **ipv6 prefix-list** *name* command is used to match IPv6 routing information.

Prefix List Example: Match Any Host Route

This topic shows an example of matching any host route in Cisco IOS and IOS XE software.

Example: Match Any Host Route

Cisco IOS/IOS XE Software

- Host routes are often filtered out to minimize the size of the routing table.

```
ip prefix-list Host_Routes deny 0.0.0.0/0 ge 32
```

Not interested in any bit in the prefix

Prefix must be of length 32 (e.g. host route)

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-17

The sample prefix list shows how to match any host route:

- The “0” in the prefix length indicates that we are not interested in any bit in the prefix itself.
- The “ge 32” indicates that the length of the prefix (that is, subnet mask) must be 32 (that is, 255.255.255.255), thus matching host routes.

Prefix List Example: Match Default Route

This topic shows an example of matching a default route in Cisco IOS and IOS XE software.

Example: Match Default Route

Cisco IOS/IOS XE Software

- Single-homed customers running BGP or multi-homed customers that do not require full Internet routing should receive only the default route.

```
ip prefix-list Default_Route permit 0.0.0.0/0
```

Not interested in any bit in the prefix

Omitted operator implies “eq *match-length*” (e.g. “eq 0” in this example)

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-18

The sample prefix list shows how to match a default route:

- The “0” in the prefix length indicates that we are not interested in any bit in the prefix itself.
- The omitted operator indicates that the prefix length should be the same as the number of bits in the prefix we are trying to match (that is, 0, which is the length of the subnet mask in a default route).

Prefix List Example: Match All Routes

This topic shows an example of matching all routes in Cisco IOS and IOS XE software.

Example: Match All Routes

Cisco IOS/IOS XE Software

- There is no keyword **any** as in access lists.
- Use this example instead, to match any route.

```
ip prefix-list All_Prefixes permit 0.0a.0.0/0 le 32
```

Not interested in any bit in the prefix

Prefix can be of any length from 0 to 32 (e.g. any route)

The sample prefix list shows how to match any route:

- The “0” in the prefix length indicates that we are not interested in any bit in the prefix itself.
- The “le 32” operator indicates that the prefix length can be anything from 0 to 32, thus matching any subnet mask.

Prefix List Example: Match Small Routes

This topic shows an example of matching all small prefixes in Cisco IOS and IOS XE software.

Example: Match Small Routes

Cisco IOS/IOS XE Software

- Small prefixes (e.g., prefix length greater than 24) are sometimes filtered out to minimize the size of the full Internet routing table.

```
ip prefix-list Small_Prefixes permit 0.0.0.0/0 ge 25
```

Not interested in any bit in the prefix

Prefix can be of any length from 25 to 32 (e.g. any prefix smaller than a Class C network)

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5.20

The sample prefix list shows how to match all small prefixes that have a prefix length greater than a Class C equivalent:

- The “0” in the prefix length indicates that we are not interested in any bit in the prefix itself.
- The “ge 25” operator indicates that the prefix length can be anything from 25 to 32, thus matching any subnet mask from 255.255.255.128 to 255.255.255.255.

Prefix List Example: Match Core Loopbacks

This topic shows an example of matching all host routes in a range in Cisco IOS and IOS XE software.

Example: Match Core Loopbacks

Cisco IOS/IOS XE Software

- Host routes are sometimes matched (e.g., loopback addresses).
- Match the address range used for loopback interfaces.
- Match /32 prefix lengths.

```
ip prefix-list Core_Loopbacks permit 172.16.1.0/24 ge 32
```

Interested in the first 24 bits of the prefix

Prefix can be of length 32 only (e.g. host route)

The sample prefix list matches all host routes in a given range of prefixes (for example, 172.16.1.1/32, 172.16.1.2/32, and so on). This type of prefix list is useful for matching (for example, loopback addresses).

Prefix List Example: Match Private Networks

This topic shows an example of matching private networks in Cisco IOS and IOS XE software.

Example: Match Private Networks

Cisco IOS/IOS XE Software

- Private networks are always filtered out when sending updates to other autonomous systems.

```
ip prefix-list Private_Prefixes permit 10.0.0.0/8 le 32
ip prefix-list Private_Prefixes permit 172.16.0.0/12 le 32
ip prefix-list Private_Prefixes permit 192.168.0.0/16 le 32
```

Interested in the first 16 bits of the prefix

Prefix can be of any length (e.g. any subnet)

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5.22

The sample prefix list matches any network or subnet in the RFC 1918 range of the IP address (that is, private address space). These private networks are typically filtered out on a routing exchange between autonomous systems.

The “le 32” is used whenever you are not interested in the size of the prefix (that is, to match any subnet).

Prefix Lists Syntax in Cisco IOS XR Software

This topic describes the syntax for prefix lists in Cisco IOS XR software.

Prefix Lists Syntax

Cisco IOS XR Software

- It is similar to Cisco IOS/IOS XE Software syntax, except it is modular.
- Each prefix list is identified using a case-sensitive *name*.
- Edit and order prefix list using line numbers.
- The *network/length* pair identifies the bits in the prefix that must match.
- The **ge**, **le** and **eq** operators identify the length of prefixes to match:
 - **le**: "less or equal" matches any prefix that is shorter or equal in length to the specified value.
 - **ge**: "greater or equal" matches any prefix that is longer or equal in length to the specified value.
 - **eq**: "equal" matches any prefix of the exact specified length.

RP/0/RP0/CPU0:CRS(config)#

```
ipv4 prefix-list name  
[seq num] {deny | permit} network/length [ge len] [le len] [eq len]  
...
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-523

Prefix list syntax in Cisco IOS XR Software is different from Cisco IOS/IOS XE Software, only in that it also implements the **eq** operator to match an exact prefix length.

For IPv6, the **ipv6 prefix-list** *name* command is used to match IPv6 routing information.

Example: Prefix Lists in Cisco IOS XR Software

This topic shows examples of prefix lists in Cisco IOS XR Software.

Example: Prefix Lists

Cisco IOS XR Software

```
ipv4 prefix-list Private_Prefixes
deny 10.0.0.0/8 le 32
deny 172.16.0.0/12 le 32
deny 192.168.0.0/16 le 32
permit 0.0.0.0/0 le 32
!
ipv4 prefix-list Core_Loopbacks
permit 172.16.1.0/24 eq 32
!
ipv4 prefix-list Host_Routes
permit 0.0.0.0/0 eq 32
!
ipv4 prefix-list Default_Route
permit 0.0.0.0/0
!
ipv4 prefix-list All_Prefixes
permit 0.0.0.0/0 le 32
!
ipv4 prefix-list Small_Prefixes
permit 0.0.0.0/0 ge 25
!
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5.24

In the figure, prefix lists examples are shown for Cisco IOS XR Software.

The Private_Prefixes prefix list shows how to filter out all RFC 1918 prefixes. These types of filters are commonly used on incoming and outgoing updates on External Border Gateway Protocol (EBGP) sessions.

The Core_Loopbacks prefix list illustrates how to match host routes that can be used to match loopback addresses from a given address range.

The Host_Routes prefix list illustrates how to match any host route.

The Default_Route prefix list illustrates how to match only the default route.

The All_Prefixes shows how to match any (all) prefixes. (The prefix list line equals the **any** keyword that is used in access lists to match any network).

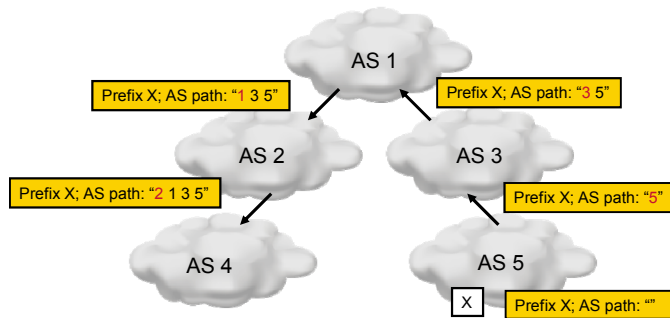
The Small_Prefixes illustrates how to filter out all small prefixes that have a prefix length equal to or greater than 25 (small prefixes).

AS Path-Based Filtering

This topic describes the characteristics and usage scenarios for AS path-based filtering in service provider environments.

AS Path Based Filtering Overview

- BGP uses autonomous systems to identify the origin and path of a prefix.
- Each path is identified using a sequence of AS numbers.
- AS path attribute is used to carry the AS path in BGP updates.
- Each egress BGP router prepends its own AS number to the AS path attribute.
- AS path access lists are used to match prefixes based on AS path characteristics.



The figure illustrates the automatic prepending that is done by all egress routers when they send updates to a neighboring AS. It shows that the first number in the AS path is always the number of the neighboring AS from which the update was received. The last number in the AS path is the number of the originating AS.

An AS path access list can be used to identify various updates that are based on the characteristics of their AS path attribute. Regular expressions are used to process AS path attributes.

AS Path Access List Syntax in Cisco IOS/IOS XE Software

This topic describes the syntax for the AS path access list command in Cisco IOS and IOS XE software.

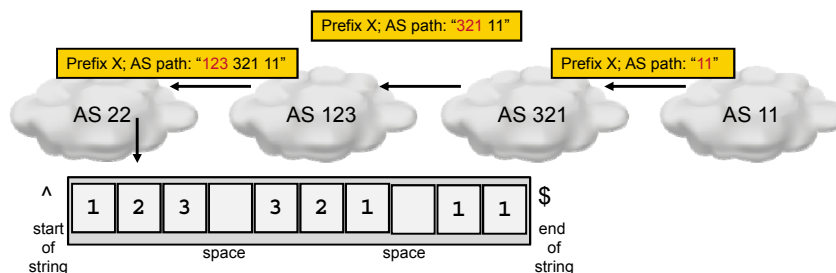
AS Path Access List Syntax

Cisco IOS/IOS XE Software

- Each AS path access list is identified using a unique number.
- Regular expressions are used to match prefixes based on the contents of the AS path attribute.
- The AS path is processed as a string of characters.

Router(config)#

```
ip as-path access-list acl-number {permit | deny} regexp
```



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5-28

Each AS path access list is identified using a unique number in the range from 1 to 500. Regular expressions are used to match prefixes that are based on contents of the AS path attribute that is converted to a string of characters.

The figure illustrates an AS path attribute as seen in AS 22. The AS path is converted to a string of characters that starts with character "1" and also ends with character "1" in this AS path example. Regular expressions must be written to take into account that we typically want to identify AS numbers and not individual characters.

Regular Expressions, Special Characters

This topic describes the special characters used to create regular expressions.

Regular Expressions, Special Characters

Character	Description
^	matches the start of AS path (e.g. " ^ 20_")
\$	matches the end of AS path (e.g. "^20 \$ ")
-	matches any delimiter (start, end, or space; e.g. "_20_")
.	matches any single character
*	matches preceding character any number of times including zero (e.g. ". * " "^20(_20) * \$")
+	matches preceding character once or more times (e.g. "^[0-9] + \$")
?	matches preceding character zero or one time (e.g. "^20(_20) ? \$")

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-5-27

Regular Expressions, Special Characters (Cont.)

Character	Description
 	logical OR operator (e.g. "_100_ _200_")
()	groups characters for precedence or to capture matched values into \n (e.g. "_100_(200 300)_")
[range]	matches a single character from the defined range of characters (e.g. "[0-9]", "[13579]")
\n	matches again what was found within the n-th pair of parentheses (e.g. "([0-9]+)(\1) * ")
\X	removes the special meaning of character X (e.g. " \(" or " \) ")

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-5-28

Commonly Used Regular Expressions

This topic lists commonly used regular expressions.

Commonly Used Regular Expressions

Regular Expression	Description
^\$	matches locally originated prefixes
^number\$	matches prefixes originating in the specified neighboring AS
_number\$	matches prefixes originating in the specified AS
^number_	matches prefixes learned through the specified neighboring AS
^([0-9]+)(_\1)*\$	matches prefixes originating in any neighboring AS and allowing prepending
.*	matches all prefixes (e.g. "any")
.	matches nonlocal prefixes (e.g. all except empty AS path)

© 2012 Cisco and/or its affiliates. All rights reserved.

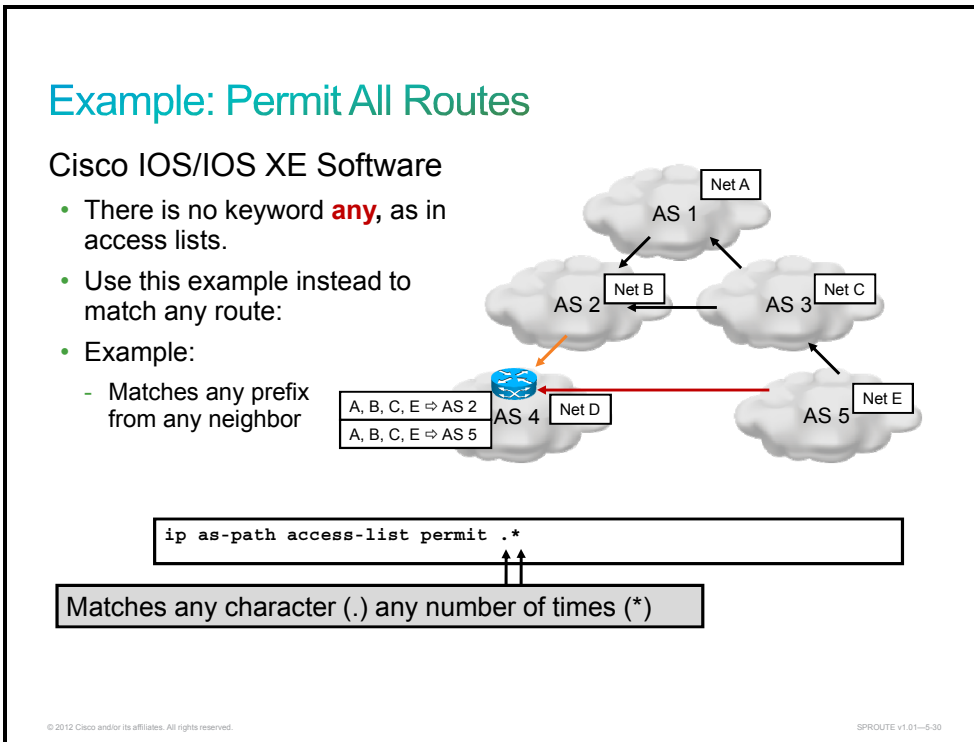
SPROUTE v1.01--5.29

Commonly Used Regular Expressions

Regular Expression	Description
^\$	Matches all local routes (local routes have an empty AS path attribute).
^10\$	Matches any route originating in a neighboring AS 10.
_20\$	Matches any route originating in AS 20.
^10_	Matches any route received from a neighboring AS 10.
([0-9]+)(_\1)*	Matches any AS number, which can optionally repeat any number of times (that is, prepending). \1 references whatever is matched in the first pair of parentheses.
.*	Matches any character any number of times. This regular expression is used to match any prefix.
.	Matches any single character.
20	Matches any route originating or passing through AS 20.
[0-9]+	Matches any AS number from 0 to 65335 (maximum range in BGP).
[13579]\$	Matches routes originating in odd-numbered AS.
[02468]\$	Matches routes originating in even-numbered AS.

AS Path Access List Example: Permit All Routes

This topic shows an example as-path access-list to permit all routes in Cisco IOS and IOS XE software.

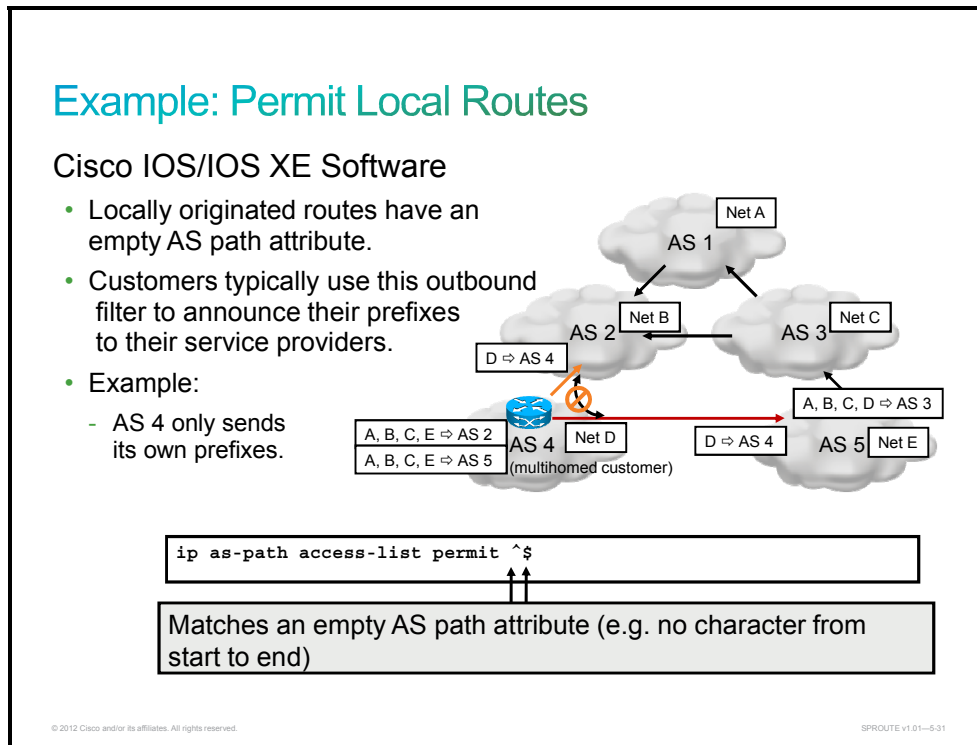


The sample regular expression (**ip as-path access-list permit .***) matches any character any number of times. This AS path access list entry is used to permit any route (that is, the equivalent of the **any** keyword in access lists).

The figure illustrates five autonomous systems, each represented by one prefix that it advertises. AS 1, for example, advertises prefix “A” which can be learned by AS 4 from AS 2, AS 5, or both. If we apply this filter in AS 4 to incoming updates from AS 2 or AS 5, we would accept all routes.

AS Path Access List Example: Permit Local Routes

This topic shows an example as-path access-list to permit local routes in Cisco IOS and IOS XE software.

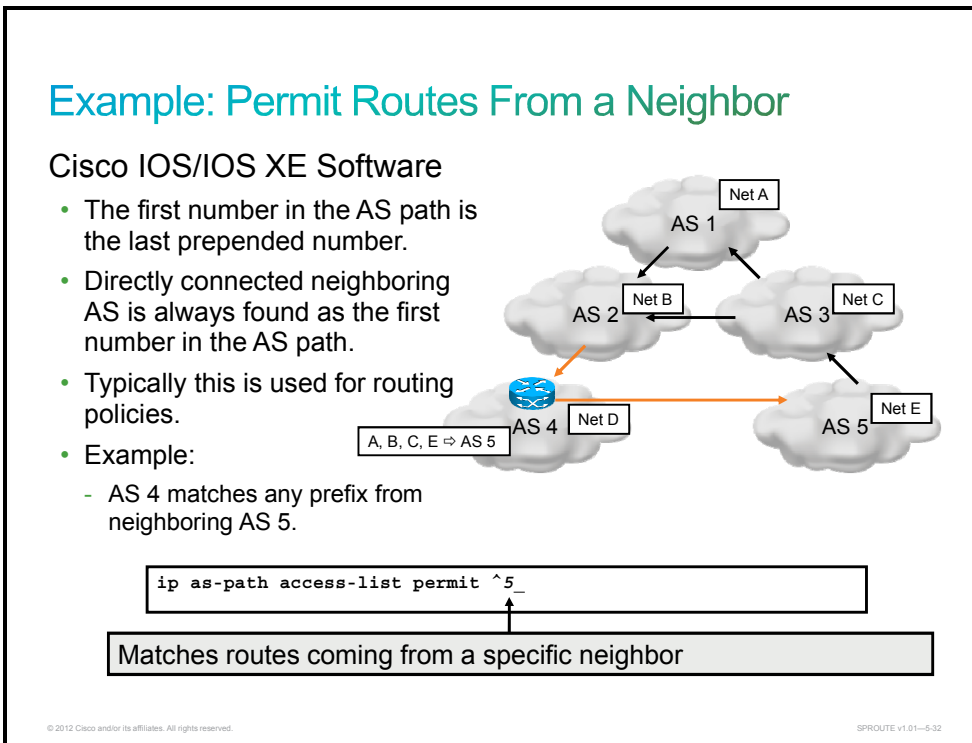


The sample regular expression (**ip as-path access-list permit ^\$**) matches any route that has an empty AS path attribute (that is, no character from start to end). Only locally originated routes have an empty AS path attribute, hence this regular expression is used when matching local routes.

This type of filter is used by multihomed customers to send only their address space to their service providers, to prevent them from becoming a transit AS. In the figure, AS 4 only advertises its own prefix ("D") to its providers. Other prefixes that are received from one provider are not forwarded to the other provider.

AS Path Access List Example: Permit Routes From a Neighbor

This topic shows an example as-path access-list to permit routes from a specific neighbor in Cisco IOS and IOS XE software.

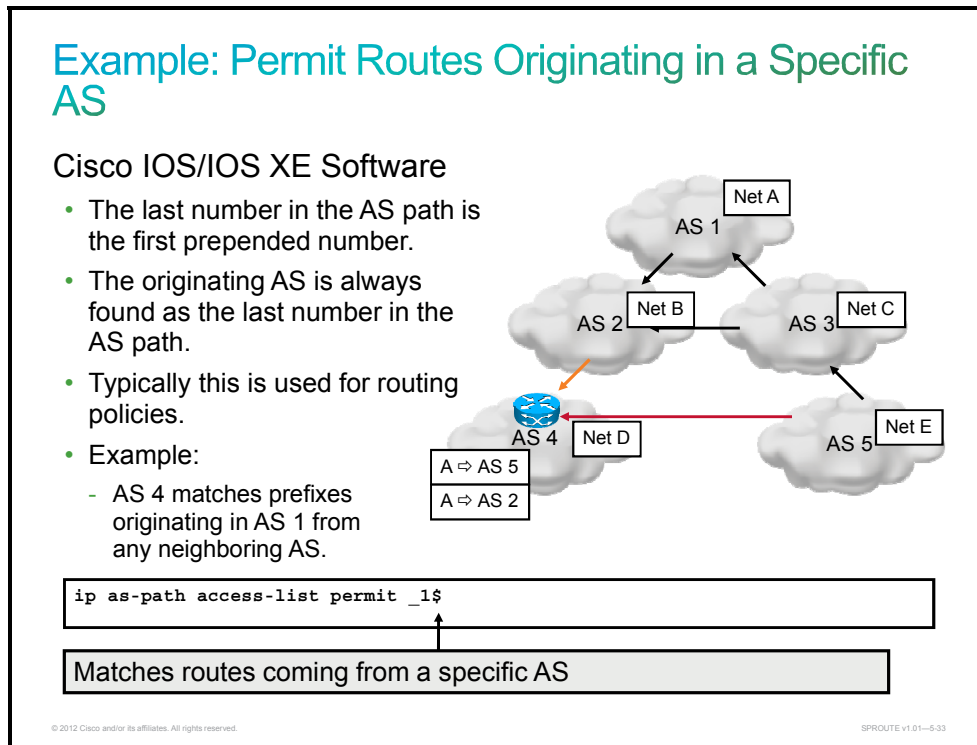


The figure illustrates a regular expression (**ip as-path access-list permit ^5_**) that matches any route received from a neighboring AS 5, the first number in the AS path. All prefixes received from AS 5 are accepted. If the same filter is applied to incoming updates from AS 2, the prefixes would be denied.

This type of filter is typically used when creating routing policies (for example, assigning different local preference values).

AS Path Access List Example: Permit Routes Originating in a Specific AS

This topic shows an example as-path access-list to permit routes from a specific AS in Cisco IOS and IOS XE software.



The figure illustrates a regular expression (**ip as-path access-list permit _1\$**) that matches any route originated in AS 1, the last number in the AS path. If this filter is applied to incoming updates from AS 2 or AS 5, it will permit prefix “A” originating in AS 1. This type of filter is commonly used to implement routing policies where you can assign preference for certain prefixes coming from a preferred service provider.

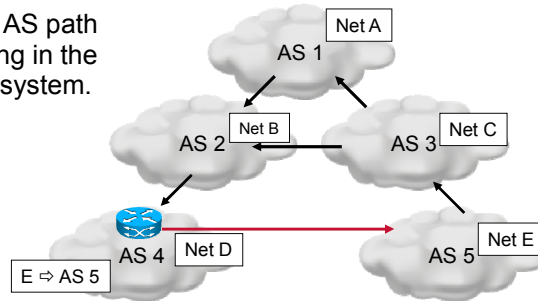
AS Path Access List Example: Permit Neighboring Local Routes

This topic shows an example as-path access-list to permit routes that originate in a neighboring AS in Cisco IOS and IOS XE software.

Example: Permit Neighboring Local Routes

Cisco IOS/IOS XE Software

- A single AS number in an AS path denotes prefixes originating in the neighboring autonomous system.



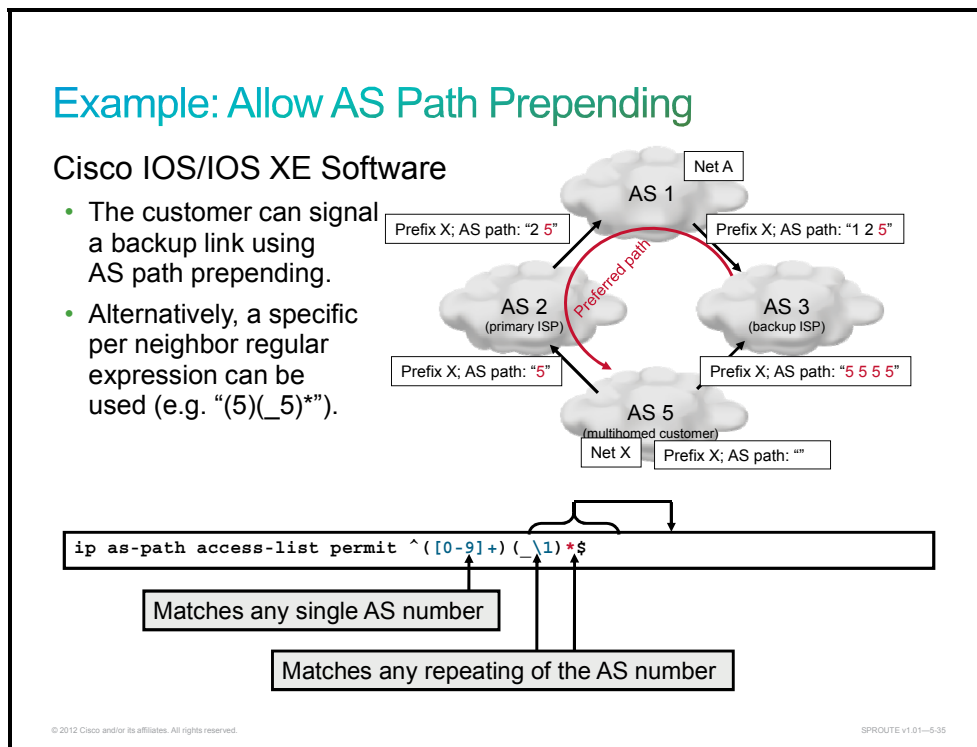
```
ip as-path access-list permit ^5$
```

Matches a single AS number in the AS path (e.g. prefix originating in a neighboring AS)

The figure illustrates a regular expression (**ip as-path access-list permit ^5\$**) that matches any route originated in the neighboring AS 5, the first number in the AS path. In the example, AS 4 only accepts prefix “E” from AS 5, because other prefixes originate in other autonomous systems.

AS Path Access List Example: Allow AS Path Prepending

This topic shows an example as-path access-list to permit local prefixes with AS path prepending in Cisco IOS and IOS XE software.



The figure illustrates a generic filter (**ip as-path access-list permit ^([0-9]+)([_1])*\$**) that can be used on any neighboring AS where you wish to accept the local prefixes of the neighbors, while still allowing them to perform AS path prepending.

Enclosed within the first parentheses is a range of digits that can appear multiple times. In BGP, this effectively matches any number from 0 to 65535. Enclosed within the second pair of parentheses, you reference whatever was matched in the first parentheses, and allow that number to repeat zero or more times.

AS 3 can reach prefix X via AS 1 or via AS 5 directly. However, since AS 5 is using AS path prepending for its updates to AS 3, AS 3 will prefer the seemingly shorter AS path, which is through AS 1 to reach prefix X.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- Routing protocol tools are used to implement a routing policy and to filter routing information.
- OSPF routing updates can be filtered based on prefix and prefix length, LSA type, and route source.
- BGP routing updates can be filtered based on prefix and prefix length, next-hop address, route source, AS path, BGP communities, and local preference.
- You can use the following route filtering tools: prefix lists, AS path access lists, route maps, and route policy language.
- Typical filtering objective in OSPF is to filter redistributed routes on the ASBR.
- Typical filtering objective in IS-IS is to filter redistributed routes.
- Typical filtering objective in BGP is to implement inbound or outbound filtering based on customer requirements.
- Routing policies are most commonly implemented for external routing information, using BGP.
- Prefix lists are designed to simplify the filtering of routing updates.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-36

Summary (Cont.)

- Each prefix list is identified using a case-sensitive name and can have multiple lines that are ordered using line numbers.
- You can use a prefix list to match any host route.
- You can use a prefix list to match a default route.
- You can use a prefix list to match all routes.
- You can use a prefix list to match small routes.
- You can use a prefix list to match core loopbacks.
- You can use a prefix list to match private networks.
- Prefix list syntax in Cisco IOS XR Software is similar to Cisco IOS and IOS XE Software.
- Prefix list syntax in Cisco IOS XR Software is modular.
- An AS path access list can be used to identify various updates that are based on the characteristics of their AS path attribute.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-37

Summary (Cont.)

- Each AS path access list is identified using a unique number in the range from 1 to 500.
- Regular expressions are used to identify AS numbers in AS path access lists.
- Some commonly used regular expressions can be used to match specific AS paths.
- You can use AS path access list to match all routes.
- You can use AS path access list to match local routes.
- You can use AS path access list to match routes from a specific neighbor.
- You can use AS path access list to match routes originating in a specific AS.
- You can use AS path access list to match neighboring local routes.
- You can allow AS path prepending using an AS path access list.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-38

Introducing Route Maps and Routing Policy Language

Overview

The lesson focuses on two mechanisms that are available, in combination with routing protocols, to filter routing information or to implement desired routing policies using Cisco IOS/IOS XE and Cisco IOS XR routers.

The lesson describes route maps that are available on Cisco IOS and IOS XE routers, and the routing policy language (RPL), which is available on Cisco IOS XR routers. The lesson also describes how to manipulate BGP attributes using RPL and how to use value sets.

Objectives

Upon completing this lesson, you will be able to describe route maps and routing policy language. You will be able to meet these objectives:

- Provide an overview of route maps
- Describe route maps processing when processing a routing update
- Describe route maps syntax
- Provide a route map example
- Describe the characteristics of RPL
- Provide an RPL example
- Describe RPL pass and drop actions
- Describe RPL conditions
- Describe RPL operators
- Describe RPL boolean operators
- Describe how to nest statements in RPL
- Describe how to set attributes and parameters in RPL
- Describe how to set BGP attributes and parameters using RPL
- Show an example of setting BGP attributes and parameters using RPL

- Describe how to set OSPF and IS-IS parameters using RPL
- Describe how to use parameterization in RPL
- Describe how to apply routing policies
- Describe how to maintain routing policies
- Describe value sets that can be used in RPL
- Describe AS path sets
- Describe standard community sets
- Describe prefix sets
- Describe how to monitor routing policies
- Describe how to test routing policies
- Describe how to translate route maps to routing policies

Route Maps Overview

This topic provides an overview of route maps.

Route Maps Overview

- Route maps are a simple language to support complex routing policies, in addition to filtering.
- Route maps are uniquely identified by a case-sensitive name.
- Each route map consists of one or more statements.
- Each statement contains zero or more **match** commands.
- Each statement contains zero or more **set** commands used to modify routing updates.
- Route maps are available in Cisco IOS/IOS XE Software. (Cisco IOS XR Software uses the Routing Policy Language.)

© 2012 Cisco and/or its affiliates. All rights reserved.

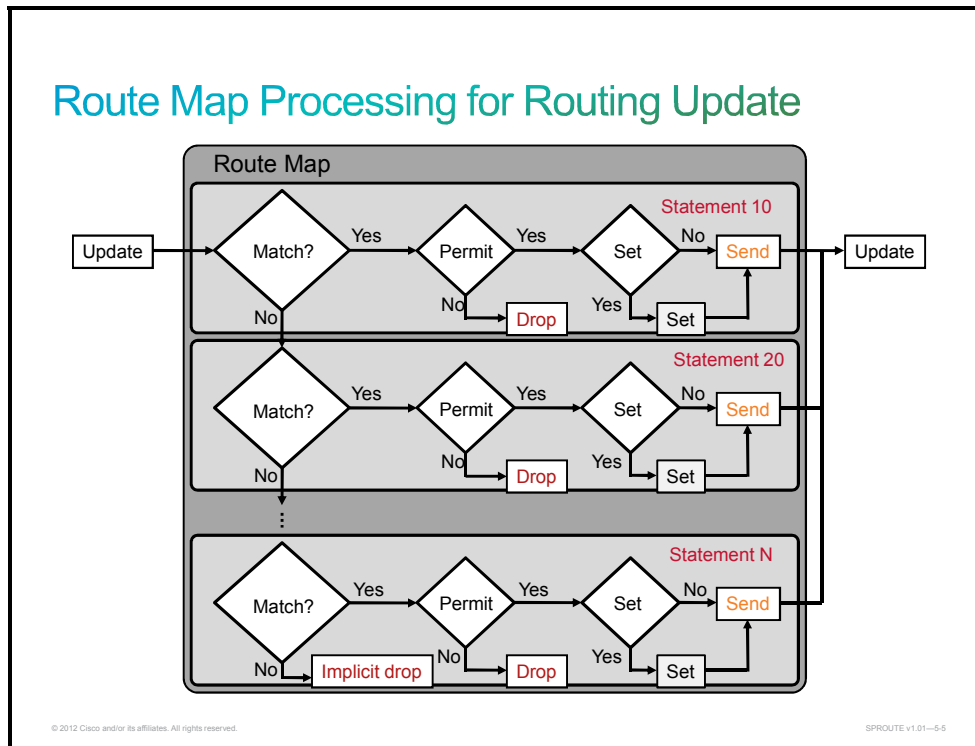
SPROUTE v1.01—5-4

Route maps are a simple language to support complex routing policies, in addition to filtering. Route maps are uniquely identified by a case-sensitive name and consist of one or more statements. Each statement contains zero or more **match** and **set** commands. The **match** command is used to identify which routes should be processed in a given statement. The **set** command specifies which parameters should be modified or added in a routing update.

Route maps are not available in Cisco IOS XR Software. Instead, Cisco IOS XR Software uses the RPL.

Route Map Processing

This topic describes route maps processing when processing a routing update.



The diagram illustrates the components and processing of a route map:

- A route is processed by route map statements in the order defined by sequence numbers.
- If a route matches the match conditions, it is processed by that statement.
- If the statement uses the “deny” options, the route is immediately dropped.
- If a route has one or more **set** commands, the **set** commands are processed, resulting in modified or added parameters and attributes.

Route Maps Syntax

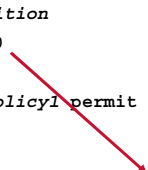
This topic describes route maps syntax.

Route Maps Syntax

Additional route-map options:

- The **continue** command can be used to jump to another statement instead of exiting.
- **Policy lists** can be used to modularize and group **match** statements.

```
route-map Policy1 permit 10
match condition
continue 40
!
route-map Policy1 permit 20
...
!
route-map Policy1 permit 40
...
!
route-map Policy1 permit 1000
```



```
route-map Policy2 permit 10
match policy-list Policy3
```

```
route-map Policy3 permit 10
...
route-map Policy3 permit 20
...
route-map Policy3 permit 30
```



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5.6

In addition to the components shown in the previous diagram, route maps also have the **continue** command, allowing the processing to continue in another statement (that is, the **jump** command).

Complex match options can be grouped in policy lists and then reused in various route maps for more modularity and reusability.

Route Maps Syntax (Cont.)

- Each route map is identified using a **case-sensitive name**.
- Each route map can have one or more **ordered statements** identified using the **sequence number**.
- Each route-map statement can filter updates using **permit** or **deny** options.
- Each statement processes updates matched by the **match** command
- Each statement can optionally modify or set parameters in an update.
- Match conditions of the same type are evaluated using a logical **OR** operator; match conditions of different types are evaluated using a logical **AND** operator.

Router(config)#

```
route-map map-tag [permit | deny] [sequence-number]
match condition
match condition
set parameter value
set parameter value
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-67

A route map statement processes routes that are matched by a match condition (for example, any prefix permitted by a prefix list will be matched; any prefix denied by a prefix list will not be processed by the statement, and is instead evaluated by the next route map statement). If a route is matched it can then be permitted or denied by the route map statement.

If there are multiple match conditions, they are evaluated using the following rules:

- Match conditions of the same type are evaluated using the logical OR operator (that is, the prefix must be matched by at least one condition).
- Match conditions of different types are evaluated using the logical AND operator (that is, the prefix must be matched by all conditions).

Routes matched and permitted by a statement can optionally be modified using **set** commands.

Route Maps Example

This topic provides a route map example.

Example: Route Maps

- Preferred paths for specific prefixes
- Backup paths for specific prefixes
- Preferred paths for prefixes based on AS path
- Backup paths for prefixes based on AS path
- Explicit permit at the end

```
route-map Policy1 permit 10
  match ip address prefix-list PL1
  set local-preference 200
!
route-map Policy1 permit 20
  match ip address prefix-list PL2
  set local-preference 50
!
route-map Policy1 permit 30
  match as-path APACL1
  set local-preference 200
!
route-map Policy1 permit 40
  match as-path APACL2
  set local-preference 50
!
route-map Policy1 permit 1000
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5.8

The sample route map configuration consists of five route map statements. The first two process routes are matched by a prefix list based on the prefix and set appropriate BGP local preference attributes. The next two statements match routes using AS path access lists and also set appropriate BGP local preference values. All non-matched routes are passed unchanged by explicitly permitting them at the end.

Example: Route Maps (Cont.)

- The first route-map statement processes routes matched by prefix list PL1 or PL2 and AS path access list APACL1.
- These routes are assigned local preference 100 and MED 1000.
- All other routes are passed unchanged.

```
route-map Policy1 permit 10
  match ip address prefix-list PL1 PL2
  match as-path APACL1
  set local-preference 200
  set metric 1000
!
route-map Policy1 permit 1000
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-9

The sample configuration illustrates the logical processing of multiple match conditions. The first condition uses two prefix lists where a route must be matched by at least one of the prefix lists (logical OR). In order for a route to be processed by this statement, it must also match the second match command, which uses an AS path access list (logical AND).

A single match statement may contain multiple conditions of the same type (prefix lists PL1 and PL2 in this example). At least one condition in the match statement must be true for that match statement to be considered a match (logical OR).

A route map statement may also contain multiple match statements of different types (prefix lists and AS path access lists in this example). All match statements must be true for the route map statement to be considered a match (logical AND).

This example can be illustrated as “(PL1 **OR** PL2) **AND** APACL1”.

Routing Policy Language

This topic describes the characteristics of RPL.

Routing Policy Language

- RPL replaces route maps in Cisco IOS XR Software.
- RPL is a simple, yet powerful language, designed to process routing updates.
- RPL addresses the deficiencies of route maps in Cisco IOS/IOS XE Software:
 - Better modularity
 - Better reusability
 - Parameterization
 - Nesting of policies and conditions
 - Powerful match options
 - Reusable value sets

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-5-10

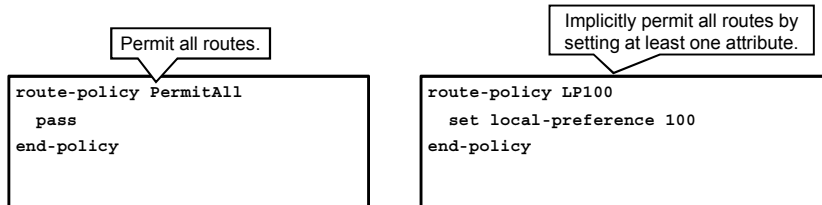
The RPL is a newer mechanism that was introduced into Cisco IOS XR Software as a replacement and improvement upon the route maps used in Cisco IOS/IOS XE Software.

RPL offers a more powerful set of tools to process routes:

- Modularity allows policies to reference other objects such as prefix list, value sets, and other policies (that is, nesting of policies).
- Parameterization permits optimization and better reusability of policies.

RPL Modularity

- Each routing policy is identified by a case-sensitive name.
- Entire policy is defined between **route-policy** and **end-policy** commands.
- Main RPL functions:
 - Filtering of updates (**pass** and **drop** commands)
 - Modification of attributes (**set** commands)



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-11

Like route maps or many other objects, routing policies are identified by using a case-sensitive name. Each routing policy is a single object (no sequence numbers or multiple lines or statements).

Like route maps, routing policies are also a filtering tool that allows you to permit or deny routing updates. The explicit commands to permit or deny are **pass** and **drop** respectively.

Like route maps, routing policies can also modify or add parameters or attributes using the **set** command. A single **set** command also implicitly includes the **pass** command.

RPL Example

This topic provides an RPL example.

RPL Example

EBGP

- **Note:** Cisco IOS XR Software does not automatically send BGP updates to external peers.
- A routing policy is required to forward updates.

Permit all routes to
EBGP peers.

```
route-policy PermitAll
  pass
end-policy
!
router bgp 1
  neighbor 1.2.3.4
    remote-as 64111
    address-family ipv4 unicast
      route-policy PermitAll out
    !
  !
!
```

This sample configuration shows how to enable the forwarding of routing updates to an external BGP neighbor. In Cisco IOS XR Software, updates are not forwarded to an external neighbor unless an outbound policy is attached to the neighbor.

The sample configuration uses a simple policy to permit all routes.

RPL Pass and Drop Actions

This topic describes RPL pass and drop actions.

RPL Pass and Drop Actions

- Using the explicit **pass** command continues the processing of route policy.
- Using the explicit **drop** command stops processing of route policy.
- The default action is **drop**.
- If any modification is applied to a route (e.g. **set**), it is an implicit **pass**.

```
route-policy DropOrPass1
end-policy                                     Drop!

route-policy DropOrPass2
  pass                                       Pass!
end-policy

route-policy DropOrPass3
  drop                                     Drop!
end-policy

route-policy DropOrPass4
  set med 100                             Pass!
end-policy

route-policy DropOrPass5
  pass
  drop
  pass                                     Drop!
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-13

The sample configurations illustrate the routing policy rules:

- An empty policy implicitly denies all routes.
- An explicit **pass** without any conditions will forward all routes without any modifications.
- An explicit **drop** will do the same as an implicit **drop** (deny all routes).
- A **set** command will modify the attribute accordingly and forward all routes.
- An explicit **drop** command will stop the processing of a policy and deny a route.

RPL Conditions

This topic describes RPL conditions.

RPL Conditions

- RPL uses various match options for conditional update processing.
- Condition syntax:

```
if attribute operator value then  
    ... do something ...  
elseif attr operator value then  
    ... do something else ...  
else  
    ... do something else ...  
endif
```

```
route-policy SetLP  
  if med eq 10 then  
    set local-preference 200  
  elseif med eq 20 then  
    set local-preference 150  
  else  
    set local-preference 50  
  endif  
end-policy
```

RPL uses conditional statement syntax that is found in many programming languages:

if condition **then** operation1 **else** operation2 **endif**

if condition1 **then** operation1 **elseif** condition2 **then** operation2 **else** operation3 **endif**

The sample configuration illustrates how the MED attribute can be used to influence a routing policy by setting a more powerful local preference attribute.

RPL Operators

This topic describes RPL operators.

RPL Operators

Comparing attributes against values supports these operators:

- **eq** : An attribute numerically equal to specified value
- **le** : An attribute numerically lower than or equal to a specified value
- **ge** : An attribute numerically greater than or equal to a specified value
- **is** : An attribute equal to a specified value
- **in** : An attribute contained in a value set
- Many other attribute-specific options

Simple
conditions

```
route-policy SetLP
  if med le 19 then
    set local-preference 200
  elseif med eq 20 then
    set local-preference 150
  elseif med ge 21 then
    set local-preference 50
  endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-15

RPL conditions can use a variety of operators:

- **eq** : The attribute numerically equal to a specified value
- **le** : The attribute numerically lower than or equal to a specified value
- **ge** : The attribute numerically greater than or equal to a specified value
- **is** : The attribute equal to a specified value (used for non-numerical values)
- **in** : The attribute contained in a value set
- Many attribute-specific conditions (for example, AS path matching)

RPL Boolean Operators

This topic describes RPL boolean operators.

RPL Boolean Operators

- Multiple match options can be combined using Boolean operators:
 - **and** : both conditions must match
 - **or** : at least one condition must match
 - **not** : negate the following condition

Using composite conditions

```
route-policy SetLP
  if med eq 10 and not local-preference eq 100 then
    set local-preference 200
  elseif med eq 20 or local-preference eq 200 then
    set local-preference 150
  else
    set local-preference 150
  endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-16

Boolean operators can be used to create complex compound conditions:

- Use the **and** operator if two or more conditions must match.
- Use the **or** operator if at least one of two or more conditions must match.
- Use the **not** operator to negate a condition.

RPL Boolean Operators (Cont.)

- Multiple match options can be combined using Boolean operators:
 - **not** : highest precedence
 - **and** : higher precedence than **or**, lower than **not**
 - **or** : lowest precedence
- Influence precedence by grouping using parentheses.

```
if med eq 10 and not local-preference eq 100 or med eq 50 then
```

VS.

```
if med eq 10 and (not local-preference eq 100 or med eq 50) then
```

VS.

```
if med eq 10 and not (local-preference eq 100 or med eq 50) then
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--6-17

Use parentheses to influence the precedence of operators and achieve the desired result. The operators have the following precedence:

- **not** is always evaluated first.
- **and** is evaluated second.
- **or** is evaluated last.

The first example is not using any parentheses. It can be written with parentheses to ensure the proper understanding of the condition:

```
if ((med eq 10) and (not (local-preference eq 100))) or (med eq 50)
then
```

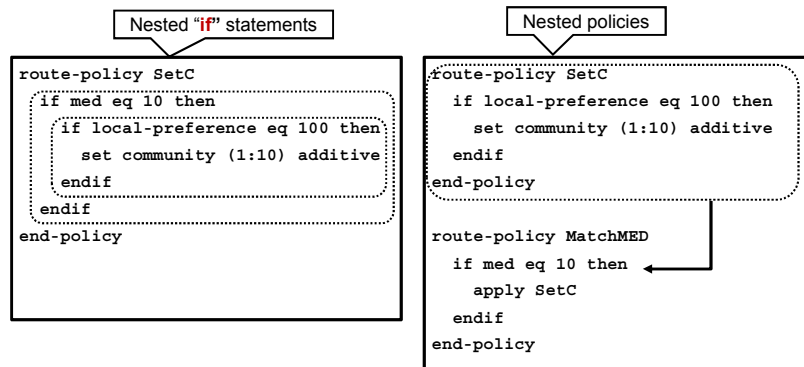
The second and third example will result in different conditions.

RPL Nesting

This topic describes how to nest statements in RPL.

RPL Nesting

- Two types of nesting are supported:
 - “if” statement within another “if” statement
 - A routing policy within another routing policy
- Multiple levels of nesting are supported.



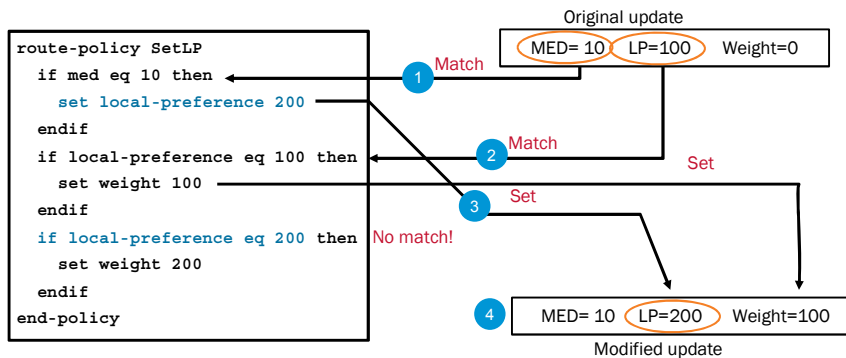
Large and complex routing policies should preferably be optimized by using modularization as much as possible. The two samples show a nested **if** statement in the left example and a nested route-policy in the right example. The **SetC** policy in the right example can be reused in multiple policies to conditionally assign a BGP community. The **apply** command is used within a route-policy to call another route-policy.

RPL Setting Attributes and Parameters

This topic describes how to set attributes and parameters in RPL.

RPL Setting Attributes and Parameters

- Use the **set** command to assign values to attributes and parameters.
- **Note:** All **set** statements are processed when the processing of policy completes (e.g. matching on a previously set attribute is not possible).



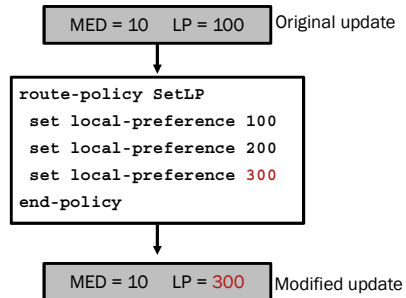
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-19

The figure illustrates a policy with multiple conditions and modifications that are based on the same parameter (BGP local preference). It is important to remember that the modifications of an attribute are only executed when the policy completes, and that conditions that are based on a previously modified value cannot be used.

RPL Setting Attributes and Parameters (Cont.)

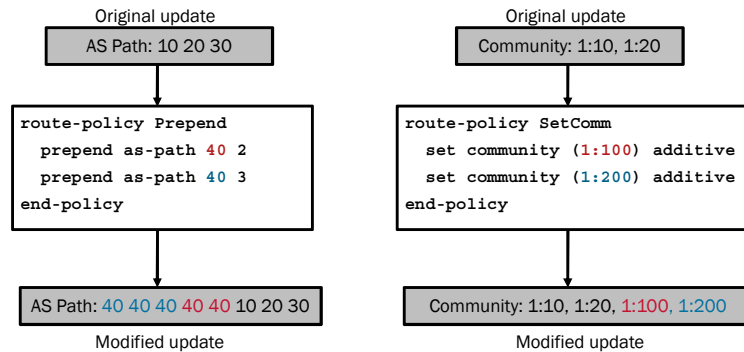
- **Note:** Last set wins when multiple sets are evaluated for a unique parameter.



If multiple **set** commands are processed for the same attribute, the last one will be used when the processing of the policy completes.

RPL Setting Attributes and Parameters (Cont.)

- **Note:** All **set** commands are evaluated in the same order for nonunique attributes and operations.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5.21

If a **set** command is processing an attribute that is a set of values, it may happen that all **set** commands will have effect.

The left example shows how the first **prepend** command modifies the AS path attribute by prepending 40 twice. The second **prepend** command then additionally did prepend 40 three times.

The right example shows how two **set** commands add two values to a set of BGP Community attributes.

RPL Setting BGP Attributes and Parameters

This topic describes how to set BGP attributes and parameters using RPL.

RPL Setting BGP Attributes and Parameters

- Standard BGP community attribute:
`set community (value [value2 ...]) [additive]`
- Extended BGP community attribute:
`set extcommunity (value [value2 ...]) [additive]`
- BGP dampening parameters:
`set dampening [half-life value] [max-suppress value] [reuse value] [suppress value]`
- Local preference attribute:
`set local-preference value`
- MED attribute:
`set med {[+|-]value | igp-cost | max-reachable}`

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5.22

Setting Standard and Extended BGP Community attributes:

- One or more values can be assigned to the BGP community attribute.
- If the **additive** keyword is used, the new communities will be added to the existing BGP communities.
- Omitting the **additive** keyword will result in the existing BGP communities being overwritten.

BGP route-flap dampening can be enabled and tuned using the **set dampening** command, where the dampening parameters can be specified to determine the aggressiveness of the dampening mechanism.

BGP local preference can be set on incoming updates or upon redistribution. The default BGP local preference is 100 and is set on all updates coming from external peers or being redistributed.

The BGP MED attribute can be set to a specific value or modified using the "+" or "-" options. MED can also be set to the underlying IGP cost or to a maximum value.

RPL Setting BGP Attributes and Parameters (Cont.)

- Delete standard BGP community attributes:
delete community {all | [not] in *community-set*}
- Delete extended BGP community attributes:
delete extcommunity rt {all | [not] in *extcomm-set*}
- Prepend AS path:
prepend as-path {AS | most-recent} [count]
- Replace a sequence of AS numbers with local AS:
replace as-path {private-as | 'AS1 AS2 ...'}
- Suppress route if aggregated:
suppress-route
- Unsuppress route if aggregated:
unsuppress-route

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-23

The **delete** command can be used in combination with standard and extended BGP communities to delete some or all of the BGP community attributes.

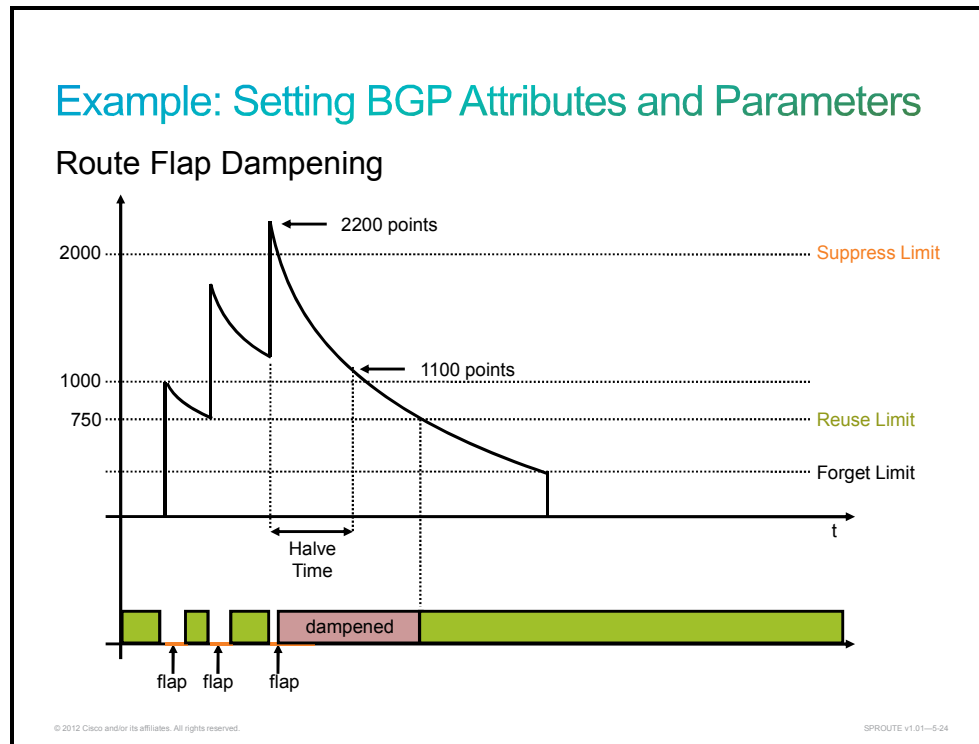
The **prepend as-path** command can be used to prepend an arbitrary number to the AS path a number of times.

The **replace as-path** command can be used to replace all occurrences of private AS numbers with the local AS number, or to arbitrarily replace specified AS numbers with the local AS number.

Policies can be used in combination with summarization (aggregation) in order to set various parameters to the summary, but also to specify which individual routes are suppressed or unsuppressed.

Example: Setting BGP Attributes and Parameters

This topic shows an example of setting BGP attributes and parameters using RPL.



BGP route flap dampening is a feature designed to make BGP more stable and consequently scale better by “punishing” routes that flap (disappear and reappear) more often. The default behavior of dampening results in stopping propagation of routes that consecutively flap three or more times in a short period, for a certain period of time.

The default behavior can be summarized:

- Each flap is penalized by adding 1000 penalty points to the penalty.
- If a cumulative penalty exceeds the **suppress limit** (2000 points by default), the route is dampened (for example, it is stored in the BGP table, but is not evaluated in the best-path selection and is consequently not installed into the routing table nor forwarded to any neighbor). The penalty is remembered by routers when the route is not reachable by storing it as a “history” entry.
- The penalty is gradually decreased. The penalty reduction is determined by the **halve-time**, which is 15 minutes by default.
- When a penalty drops below the **reuse limit** (750 by default) or when the route has been dampened for more than the **maximum suppress time** (one hour by default), the route becomes valid again.
- When the penalty drops below one half of the **reuse limit**, all flap history and penalty is forgotten.

Example: Setting BGP Attributes and Parameters

Conditional BGP Dampening

Conditional BGP dampening, where smaller prefixes are more aggressively punished than larger prefixes

```
router bgp 1
  address-family ipv4 unicast
    bgp dampening route-policy BDamp
  !
!
route-policy BDamp
  if destination in (0.0.0.0/0 ge 25) then
    set dampening max-suppress 30 halflife 10 reuse 750 suppress 1000
  elseif destination in (0.0.0.0/0 ge 21) then
    set dampening max-suppress 15 halflife 7 reuse 750 suppress 2000
  elseif destination in (0.0.0.0/0 ge 17) then
    set dampening max-suppress 10 halflife 5 reuse 750 suppress 3000
  else
    set dampening max-suppress 5 halflife 3 reuse 750 suppress 4000
  endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5.25

The sample configuration illustrates how graded BGP route-flap dampening is configured:

- Small prefixes (/25 to /32) are assumed to be more likely to flap, and are hence more aggressively punished if they flap several times.
- Larger prefixes (/21 to /24) are assumed to be slightly more stable and are less aggressively punished (allow more flaps before suppression and become unsuppressed faster when they stabilize).
- Large prefixes (/17 to /20) are even less aggressively punished if they flap. The largest prefixes (/0 to /16) are assumed to be the most stable (large summaries belonging to service providers) and are suppressed after more than four consecutive flaps and are unsuppressed within ten minutes after stabilizing.

RPL Setting OSPF and IS-IS Parameters

This topic describes how to set OSPF and IS-IS parameters using RPL.

RPL Setting OSPF and IS-IS Parameters

- OSPF metric type:
set metric-type {type-1 | type-2}
- OSPF metric:
set ospf-metric *value*
- IS-IS metric type:
set metric-type {external | internal}
- IS-IS metric type:
set isis-metric *value*
- IS-IS level for redistributed routes:
set level {level-1 | level-2 | level-1-2}

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01—5-28

Routing policies can also be used in combination with OSPF and IS-IS to modify the routing information.

RPL Parameterization

This topic describes how to use parameterization in RPL.

RPL Parameterization

RPL supports two types of parameters:

- **Global parameters:**
 - Defined globally using the **policy-global** command
 - Available for use in all routing policies
- **Parameters passed to a nested routing policy:**
 - Defined when creating a routing policy
 - Available in **match** and **set** statements within a policy or when calling another nested routing policy

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-27

In order to make policies modular and reusable, parameters can be used in place of fixed values when calling nested policies.

A policy can reference global parameters or parameters passed to it from a calling policy.

RPL Global Parameters

- Parameters are defined using the **policy-global** command, and are separated by commas.
- Values are defined within single quotes.
- Parameters are referenced by prepending the **\$** sign to the name of the parameter.

Defining global variables

```
policy-global
# Global variables
AS '65001',
Lo0 '10.1.2.3',
EBGP1 '192.168.1.1',
EBGP2 '192.168.2.1',
DefWeight '0',
DefLP '100',
DefMED '0'
end-global
```

Using global variables

```
route-policy SetMED
if as-path originates-from '$AS' then
set med $DefMED
endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-28

The left example illustrates the usage of the **policy-global** commands where all the global variables should be defined. These variables can then be referenced by any routing policy.

The right example illustrates a routing policy referencing two global variables.

RPL Passed Parameters

- Declare parameters when creating a routing policy.
- Nesting policies with parameters allows for greater modularization and optimization of policies.

Policy using passed parameters

```
route-policy SetMED($med, $as)
  if as-path originates-from '$as' then
    set med $med
  else
    set med max-reachable
  endif
end-policy
!
```

Using a nested policy and passing parameters to it

```
route-policy ProcessUpdates
  if as-path neighbor-is '100' then
    apply SetMED(50,100)
  elseif as-path neighbor-is '200' then
    apply SetMED(150,200)
  endif
end-policy
!
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5-29

The sample configuration illustrates a modular approach to creating routing policies, where one policy (on the right) calls another policy (on the left) that applies a different MED value, based on whether the route originates in a given AS. Note that matching based on the AS path is always done using regular expressions, which must be enclosed within single quotes.

The left routing policy is defined with two parameters: **\$med** and **\$as**. When you are calling this policy from within another policy, using the **apply** command, you should supply two parameters.

Applying Routing Policies

This topic describes how to apply routing policies.

Applying Routing Policies

- Design a routing policy.
- Configure the policy.
- Test the policy by using **show** commands.
- Apply the policy if it is correct.
- Use routing policies in many places (attach points):
 - Routing updates (e.g. BGP, OSPF, EIGRP, IS-IS, RIP)
 - Route origination (e.g. redistribution, network commands)
 - Route insertion into routing table
 - **show** commands to filter output

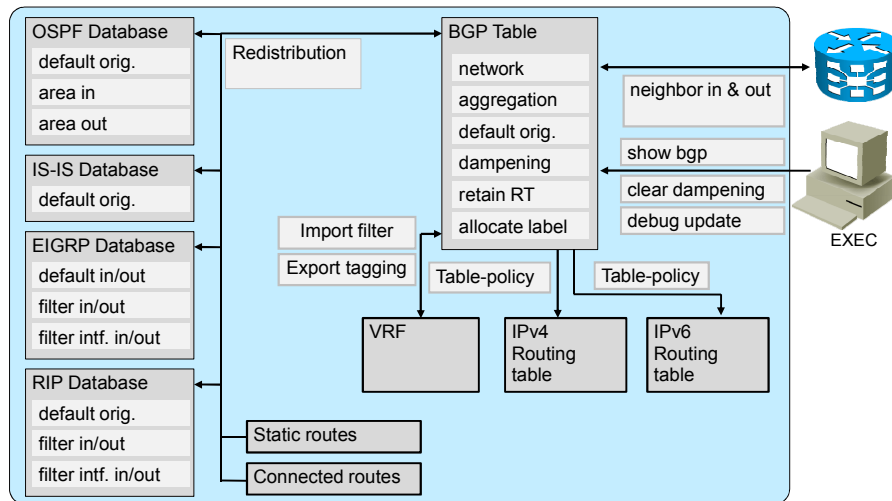
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-30

When building a routing policy it is important to clearly define the requirements for the route-policy. Often a route-policy will be derived from existing routers using Cisco IOS/IOS XE Software, in which case it requires a route map to be “translated” to RPL. Reviewing a route-policy and optimizing it is recommended in order to simplify its maintenance.

Applying Routing Policies (Cont.)

Attach Points



The figure illustrates the many attach points for routing policies:

- Redistribution between any pair of routing protocols
- Received or sent updates, depending on the limitations of routing protocols (for example, ABRs in OSPF)
- Origination of routes in BGP by using network statements or summarization
- Injecting routes into the routing table from BGP
- Using **show** commands in BGP to filter the output or test the effect of the routing policy

Applying Routing Policies (Cont.)

Validity Checking

- RPL validity checking is done in two phases:
 - Syntax checking and value checking are performed during policy configuration.

```
RP/0/RP1/CPU0:CRS(config-rpl)#set med 289314790283408912634789
                                     ^
% Invalid input detected at '^' marker.
```

- Applicability of a policy for a given attach point is checked during configuration commit.

```
RP/0/RP1/CPU0:CRS(config-bgp-af)#commit
% Failed to commit one or more configuration items during an atomic operation, no
changes have been made. Please use 'show configuration failed' to view the errors
RP/0/RP1/CPU0:CRS(config)# show config failed
!! CONFIGURATION FAILED DUE TO SEMANTIC ERRORS
router bgp 1
  address-family ipv4 unicast
    redistribute connected route-policy t9
  !!% Could not find entry in list: Policy [t9] uses the 'ospf-metric' attribute.
  There is no 'ospf-metric' attribute at the BGP redistribution-dflt attach point.
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-6-32

Cisco IOS XR Software performs validity checks in two phases:

- Basic syntax and value checking are performed when a command is entered. The first example illustrates the configuration of **MED** with a value that is out of range. The syntax checking will reject this command immediately.
- The applicability of a routing policy is verified for a given attach point when the configuration is committed. The second example illustrates that the configuration of a route policy was successful, and the policy was successfully applied to redistribution into BGP. However, when you try to commit the configuration, the router will reject the configuration because there was an attempt to set OSPF parameters inside BGP.

Maintaining Routing Policies

This topic describes how to maintain routing policies.

Maintaining Routing Policies

- Trying to edit an existing routing policy through configuration-mode CLI will result in the policy being rewritten:

```
RP/0/RP1/CPU0:CRS(config)#route-policy R1
% WARNING: Policy object 'route-policy R1' exists! Reconfiguring it via CLI
will replace current definition. Use 'abort' to cancel.
RP/0/RP1/CPU0:CRS(config-rpl)# abort
```

- Use EXEC-mode editor instead.
- Three editors are available:
 - GNU Nano
 - Emacs
 - VIM
- Modify the policy, and complete the process:
 - Save changes.
 - Exit editor.
 - Commit changes.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6.33

Editing a routing policy requires the use of one of two available editors. Using the configuration mode approach will result in the policy being rewritten.

Cisco IOS XR Software comes with three types of editors that are accessible through EXEC mode:

- GNU Nano (the default editor since Cisco IOS XR Release 3.6)
- Micro Emacs
- VIM

Upon exiting from the editor, you will be prompted to save and commit the changes.

The example shows the warning that is displayed when you try to go into policy configuration mode for an already configured policy. If this happens, the entire policy will be overwritten by the new configuration.

Maintaining Routing Policies (Cont.)

Using an Editor

- An editor can be used for routing policies and sets.

```
RP/0/RP1/CPU0:CRS#edit ?
as-path-set      edit an as-path-set
community-set    edit a community-set
extcommunity-set edit an extended-community-set
policy-global    edit policy-global definitions
prefix-set       edit a prefix-set
rd-set           edit a rd-set
route-policy     edit a route-policy
```

- Invoke the desired editor.

```
RP/0/RP1/CPU0:CRS#edit route-policy RP1 ?
emacs  to use Micro Emacs editor
inline to use command line
nano   to use nano editor
vim    to use Vim editor
<cr>
```

Use the **edit** command in EXEC mode to start editing a configuration object. Select the preferred editor.

The built-in editors are available for route policies and other objects (covered later in this lesson) such as various sets that are used in combination with route policies.

The following list contains some of the most commonly used keystroke combinations within the Emacs editor:

- **Ctrl-F**: Move cursor forward (right).
- **Ctrl-B**: Move cursor backward (left).
- **Ctrl-N**: Move cursor to next line (down).
- **Ctrl-P**: Move cursor to previous line (up).
- **Ctrl-E**: Move to the end of the line.
- **Ctrl-A**: Move to the start of the line.
- Backspace: Delete character to the left of the cursor.
- **Ctrl-D**: Delete character to the right.
- **Ctrl-X** followed by **Ctrl-S**: Save changes.
- **Ctrl-X** followed by **Ctrl-C**: Exit and commit saved changes.

The following list contains some of the most commonly used commands within the VIM editor:

- `←, ↓, ↑, →`: Move cursor left, down, up, right.
- `h, j, k, l`: Move cursor left, down, up, right.
- `i`: Start editing at the cursor position.
- `a`: Start editing after the cursor position.
- `ESC`: Stop editing (return to command mode).
- `x`: Delete the character at cursor position.
- `dd`: Delete a line.
- `u`: Undo a single action.
- `ESC` followed by `:w`: Save changes.
- `ESC` followed by `:q`: Exit and commit saved changes.

After exiting the editor, you will be asked to save and commit the changes.

Refer to the Cisco IOS XR Software command reference for a detailed list of all commands and options for all the available editors.

Value Sets

This topic describes value sets that can be used in RPL.

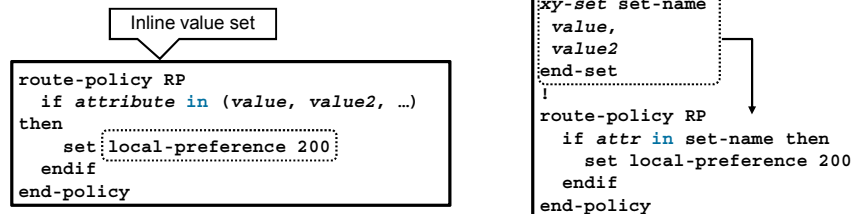
Value Sets

RPL can match attributes against a set of multiple values:

- Inline sets using parentheses for one-time use
- Named value sets for reusability

Value sets:

- AS path in AS path set
- Standard community in community set
- Extended community in extcommunity set
- Prefix in prefix set
- Route distinguisher in route distinguisher set



Value sets are objects that are used to modularize routing policies. Various types of sets exist for different types of parameters and attributes.

Each set can contain multiple values. The **in** operator can be used for the existence of a value in the set.

The example on the left illustrates a generic condition where an inline value set is used. The example on the right references a preconfigured value set.

AS Path Sets

This topic describes AS path sets.

AS Path Sets

- Define an AS-path set using the **as-path-set** command.
- Use one or more comma-separated **ios-regex** commands to define regular expression that define set membership.
- Use the **in** operator in routing policy to test for membership of AS path in AS path set.

Match prefixes originating in defined autonomous systems.

```
as-path-set PreferredOriginators
ios-regex '_10$',
ios-regex '_20$',
ios-regex '_30$',
ios-regex '_40$'
end-set
```

Use an AS path set in a policy to match prefixes based on AS path attribute.

```
route-policy RP
if as-path in PreferredOriginators then
  set local-preference 200
endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-38

An AS path set can contain one or more regular expressions. A condition can be used to check for an AS path attribute against the set of regular expressions.

The sample configuration uses a policy to set the local preference to 200 for all preferred originating autonomous systems that are listed in the AS path set.

AS Path Sets (Cont.)

Predefined matching criteria	Description
is-local	matches any prefix with an empty AS path attribute (equals regular expression '^\$')
neighbor-is path	matches based on first ASN in the AS Path attribute (equals regular expression '^path_')
originates-from path	matches based on last ASN in the AS Path attribute (equals regular expression '_path\$')
passes-through ASN	matches based on ASN anywhere in the AS Path (equals regular expression '_path_')
length len	matches AS paths based on number of ASNs in the path
unique-length len	matches AS paths based on number of unique ASNs in the path

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-97

Instead of using regular expressions, some of the more common AS path checks can be performed using built-in conditions:

- **is-local** : Identifies whether a prefix is local to the AS; it performs the same function as a regular expression checking for an empty AS path attribute (“^\$”).
- **neighbor-is path** : Identifies whether a prefix was received from a neighboring AS; equals regular expression '^path_'.
- **originates-from path** : Identifies whether a prefix was originated by a specified AS; equals regular expression '_path\$'.
- **passes-through ASN** : Identifies whether a prefix passed through the specified AS; equals regular expression '_ASN_'.
- **length len** : Matches AS paths based on number of ASNs in the path.
- **unique-length len** : Matches AS paths based on number of unique ASNs in the path.

AS Path Set Examples

Using built-in AS-Path
match options

```
route-policy RP
  if as-path is-local then
    set local-preference 200
  endif
  if as-path neighbor-is '20'
  then
    set local-preference 190
  endif
  if as-path originates-from '20'
  then
    set local-preference 180
  endif
  if as-path passes-through '20'
  then
    set local-preference 170
  endif
end-policy
```

Equivalent regular
expressions

```
route-policy RP
  if as-path in (ios-regex '^$')
  then
    set local-preference 200
  endif
  if as-path in (ios-regex '^20_')
  then
    set local-preference 190
  endif
  if as-path in (ios-regex '_20$')
  then
    set local-preference 180
  endif
  if as-path in (ios-regex '_20_')
  then
    set local-preference 170
  endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-38

The two samples show configurations equivalent in result but using different approaches:

- The left sample uses built-in conditions
- The right example uses regular expressions

Notice in the example:

- The regular expression '^\$' can be replaced by the built-in operator **is-local**.
- The regular expression '^20_' can be replaced by the built-in operator **neighbor-is**.
- The regular expression '_20\$' can be replaced by the built-in operator **originates-from**.
- The regular expression '_20_' can be replaced by the built-in operator **passes-through**.

Standard Community Sets

This topic describes standard community sets.

Standard Community Sets

- Define a standard community set using the **community-set** command.
- Use one or more comma-separated match options:
 - **ios-regex** commands to define regular expressions that define set membership
 - numbered membership matching
 - membership matching using well-known standard communities
- Use the **matches-any** operator to match routes that have at least one community in the community set.
- Use the **matches-every** operator in routing policy to match routes that have all communities in the community set.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-5.39

Multiple BGP communities can also be grouped into a community set. Communities can be matched:

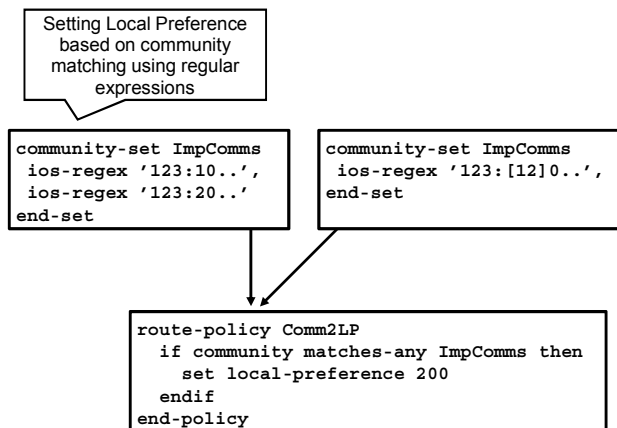
- Regular expression matching, where a regular expression is used against an ordered list of communities; converted into a string of characters
- Numbered matching, where community attributes are matched against a list of values in a community set
- Named matching, where community attributes are matched against a list of communities, including named well-known communities

Community matching can use modifiers that define how the matching is performed:

- The **matches-any** operator should be used to match routes that have at least one community in the community set.
- The **matches-every** operator should be used to match routes that have all communities listed in the community set.

Standard Community Set RegExp Matching

- Use one or more comma-separated **ios-regex** commands to define regular expressions that define set membership.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--640

The sample configuration illustrates two community sets, based on regular expressions.

The left ImpComms community-set uses two regular expressions; the right ImpComms community-set uses a single regular expression. Either approach can be used in the Comm2LP route-policy so that a route will be assigned local preference 200 if it contains BGP community 123:10xx or 123:20xx.

Standard Community Set Numbered Matching

Use numbered matching:

- *AS:num*
- *AS:[range]*
- *AS:**

Setting Local Preference
based on numbered
community matching

```
community-set ImpComms
123:1010
123:[2000..2099]
999:*
end-set
!
route-policy Comm2LP
if community matches-any ImpComms then
    set local-preference 200
endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-541

The sample configuration illustrates numbered matching where explicit BGP communities are matched. Additionally, ranges and wildcards can be used in sets:

- “*AS:num*” is used to match a specific community.
- “*AS:[range]*” is used to match a range of values.
- “*AS:**” is used to match all values for a given AS.

The sample community set tries to match at least one community from a range that is defined using different options.

Standard Community Set Named Matching

Use identifiers for well-known communities:

- **Internet** : Match all communities.
- **local-as** :Keep tagged prefixes in the local AS.
- **no-advertise** :Prevent tagged prefixes from being advertised to any peer.
- **no-export** :Prevent tagged prefixes from being announced to EBGp peers.

Prevent sending of core subnets to external peers.

```
router bgp 1
 address-family ipv4 unicast
  redistribute connected route-policy NoExport
!
route-policy NoExport
 set community no-export
!
```

Delete all communities on incoming updates.

```
route-policy DeleteAllComms
 delete community in (internet)
end-policy
!
router bgp 1
 neighbor 1.2.3.4
 address-family ipv4 unicast
  route-policy DeleteAllComms in
!
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--6-42

The figure illustrates the third matching option for BGP communities; it is based on the names of well-known communities.

The left example assigns the **no-export** community to all redistributed routes. The right example matches all communities by using the **internet** keyword, and deletes them.

Prefix Sets

This topic describes prefix sets.

Prefix Sets

- Used to match prefixes in routing protocol updates: **Prefix[/length [{le | ge | eq} mask-len]]**

Various prefix sets

```
prefix-set PrivatePrefixes
 10.0.0.0/8 le 32,
 172.16.0.0/12 le 32,
 192.168.0.0/16 le 32
end-set
!
prefix-set CoreLoopbacks
 172.16.1.0/24 eq 32
end-set
!
prefix-set HostRoutes
 0.0.0.0/0 eq 32
end-set
```

Various prefix sets

```
prefix-set DefaultRoute
 0.0.0.0/0
end-set
!
prefix-set AllPrefixes
 0.0.0.0/0 le 32
end-set
!
prefix-set SmallPrefixes
 0.0.0.0/0 ge 24
end-set
!
prefix-set
SmallPrefixesExceptHostRoutes
 0.0.0.0/0 ge 24 le 31
end-set
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-543

A prefix set is used to match routes based on prefix-list-like criteria in a prefix set. The same syntax is used as with prefix lists.

Monitoring Routing Policies

This topic describes how to monitor routing policies.

Monitoring Routing Policies

- Use the **show rpl route-policy [policy-name] [detail]** commands to display the policies.
- Detailed output also displays all referenced objects (e.g. sets and nested route policies).

Display a policy and all other associated objects.

```
RP/0/RP1/CPU0:CRS# show rpl route-policy MgmtRTExport detail
extcommunity-set rt MgmtRT
 23456:100,
 23456:200
end-set
!
prefix-set MgmtLoopbacks
 10.1.1.0/24 le 32
end-set
!
route-policy MgmtRTExport
 if destination in MgmtLoopbacks then
   set extcommunity rt MgmtRT
 endif
end-policy
!
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6.44

The **show rpl** command with the **detail** keyword can be used to display the policy configuration, including all the dependencies.

In the slide, the output shows the configurations of the MgmtRTExport route-policy as well as the configurations of the prefix set and the extended community-set referenced within the route policy.

Determining Attach Points

- Use the **show rpl route-policy policy-name attachpoints** commands to list the attach points of the policy.
- Detailed output also displays all referenced objects (e.g. sets and nested route policies).

Display attach points
for a routing policy.

```
RP/0/RSP0/CPU0:PE1#show rpl route-policy pass attachpoints
Thu Nov 17 19:50:52.025 UTC
```

BGP Attachpoint: Neighbor

Neighbor/Group	type	afi/safi	in/out	vrf name
192.168.101.11	--	IPv4/uni	in	default
192.168.101.11	--	IPv4/uni	out	default

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-545

The **attachpoints** option can be used to display all references to the specified policy.

In the example, the **show** command shows that the specified route policy **pass** is attached to **BGP neighbor in the in and out direction**.

Testing Routing Policies

This topic describes how to test routing policies.

Testing Routing Policies

- Some policies can be tested (e.g. outbound BGP filter).
- Use the **show bgp route-policy policy-name** command to list BGP entries permitted by the policy.
- **Note:** Attributes modified by the policy are not displayed.

Test a new policy to filter outgoing updates.

```
RP/0/RP1/CPU0:CRS# show bgp route-policy FilterOut
BGP router identifier 0.0.0.0, local AS number 1
BGP generic scan interval 60 secs
BGP table state: Active
BGP main routing table version 30
BGP scan interval 60 secs
Status codes: s suppressed, d damped, h history, * valid, > best
               i - internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network             Next Hop           Metric LocPrf Weight Path
*> 10.4.100.0/30        0.0.0.0               0      200  32768 ?

Processed 1 prefixes, 1 paths
RP/0/RP1/CPU0:CRS#
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--648

Policies can be combined with the **show bgp** command to display only those BGP entries that are permitted by the policy.

This approach can be used to test the performance of a newly configured policy or to limit the display of a large BGP table for troubleshooting purposes.

In the example, the policy FilterOut only displays one entry (10.4.100.0/30) in the BGP table.

Translating Route Maps to Routing Policies

This topic describes how to translate route maps to routing policies.

Translating Route Maps to Routing Policies

- When you migrate from Cisco IOS/IOS XE Software to Cisco IOS XR Software, use the following guidelines to translate route maps to policies:
 - Each numbered entry is one **if** statement.
 - Each **match** option is one condition:
 - Match conditions of the same type should be joined using the **OR** logical operator.
 - Match conditions of different types should be joined using the **AND** logical operator.
 - Use parentheses to maintain proper precedence.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-547

When you translate route maps to routing policies it is important to understand the relationship between multiple conditions in a single route map statement. Recall that multiple conditions of the same type are combined by using a logical **OR**. Therefore, you should use the **OR** operator in the **if** statement of the routing policy. Multiple conditions of the different types are combined using a logical **AND**. Therefore, you should use the **AND** operator in the **if** statement of the routing policy. Make sure you use parentheses for proper operator precedence.

Translating Route Maps to Routing Policies (Cont.)

```
route-map RM permit 10
  match ip address prefix-list PL1
  set local-preference 200
!
route-map RM permit 20
  match ip address prefix-list PL2
  set local-preference 150
!
```

Sample route map

Translated routing
policy

```
route-policy RP
  if destination in PL-Set1 then
    set local-preference 200
  elseif destination in PL-Set2 then
    set local-preference 150
  endif
end-policy
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--648

The two sample configurations show how two route map statements can be translated into a routing policy.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- Route maps are a simple language to support complex routing policies.
- A route map is processed in a top-down fashion.
- Route maps support modularity and reusability.
- There is implicit deny at the end of a route map.
- The RPL is a newer mechanism that was introduced into Cisco IOS XR Software as a replacement and improvement upon the route maps.
- In Cisco IOS XR Software, BGP updates are not forwarded to an external neighbor unless an outbound policy is attached to the neighbor.
- Set action in a route policy also implicitly allows an update.
- RPL uses conditional statement syntax that is found in many programming languages.
- RPL conditions can use a variety of operators, such as eq, le, ge and others.
- You can use boolean operators to create complex compound conditions.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-49

Summary (Cont.)

- You can nest one routing policy inside another to achieve modularity and reusability.
- You can use the set RPL action to modify attributes.
- You can use RPL to manipulate BGP attributes.
- BGP route flap dampening is a feature designed to make BGP more stable and scalable by punishing routes that flap.
- You can use RPL to change OSPF or IS-IS parameters.
- In order to make policies modular and reusable, you can use parameters in place of fixed values when calling nested policies.
- You can apply routing policies to many attach points, such as redistribution between any pair of routing protocols.
- Editing a routing policy requires the use of one of three available editors.
- Value sets are objects that are used to modularize routing policies. Various types of sets exist for different types of parameters and attributes.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-50

Summary (Cont.)

- AS path set can contain one or more regular expressions which describe a set of AS paths.
- You can use community sets to group multiple BGP communities.
- You can use a prefix set to match routes based on prefix-list-like criteria.
- You can use the show rpl command to display a policy configuration, including all the dependencies.
- Policies can be combined with the show bgp command to display only those BGP entries that are permitted by the policy.
- When you translate route maps to routing policies it is important to understand the relationship between multiple conditions in a single route map statement.

Implementing Route Redistribution

Overview

Simple routing protocols work well for simple networks, but as networks grow and become more complex, it may be necessary to change routing protocols. Often, the transition between routing protocols takes place gradually, so there are multiple routing protocols that are operating in the network for variable lengths of time. It is important to understand how to exchange routing information between these routing protocols, and how Cisco routers operate in an environment with multiple routing protocols.

Configuring route redistribution can be simple or complex, depending on the mix of routing protocols that you want to redistribute. The commands that are used to enable redistribution and assign metrics vary slightly, depending on the routing protocols that are being redistributed. Before configuring the exchange of routing information between routing protocols, you must understand the procedures for, and requirements of, each routing protocol. Redistribution must be configured correctly for each routing protocol to obtain proper results.

This lesson examines several reasons for using more than one routing protocol, describes redistribution from one routing protocol to another, and explains how Cisco routers make route selections when multiple protocols are active in the network. This lesson also describes how to configure route redistribution between various routing protocols. The commands for each protocol are covered. In addition, the impact of route redistribution is analyzed.

Objectives

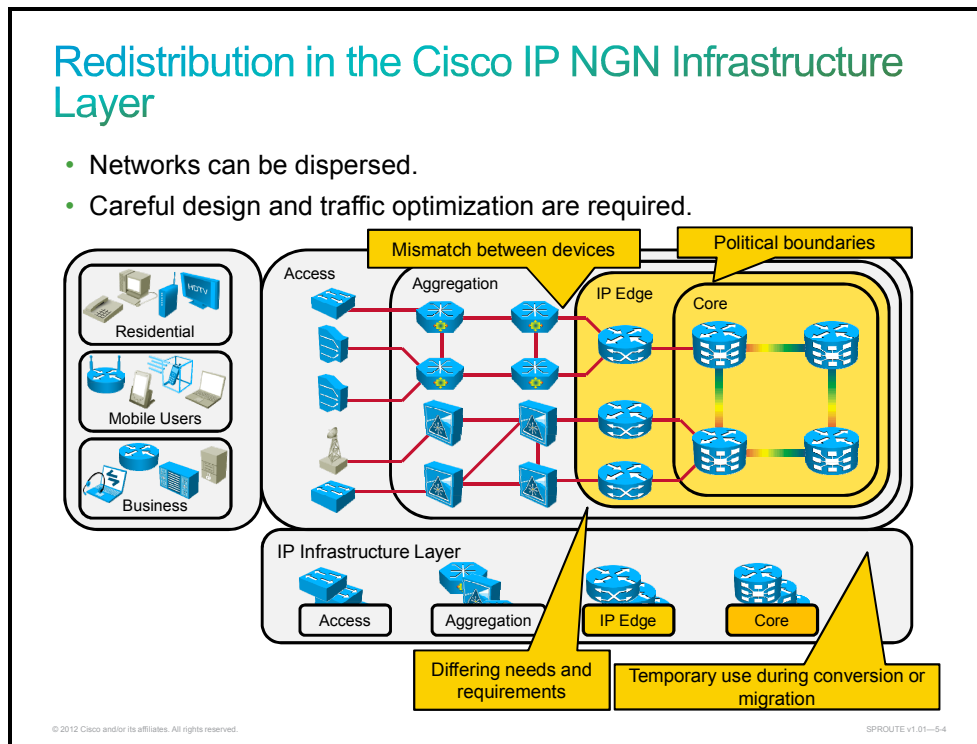
Upon completing this lesson, you will be able to explain what route redistribution is, why it is necessary, and how to configure route redistribution. You will be able to meet these objectives:

- Describe the need to use redistribution in the multiple IP routing protocol environment
- Describe route redistribution
- Describe using seed metrics in route redistribution
- List the default seed metrics
- Describe redistribution on a single router
- Describe redistribution on multiple devices that are running both routing protocols
- Describe techniques for performing route redistribution

- Describe potential problems redistributing routes and identifies potential solutions
- Describe the procedures that are necessary to configure route redistribution in networks that use Cisco IOS, IOS XE, and IOS XR Software
- Describe how to redistribute routes into OSPF
- Describe how to redistribute routes into IS-IS
- Describe how to redistribute routes into BGP
- Describe administrative distance
- Describe how to modify the administrative distance for OSPF routes
- Describe how to modify the administrative distance for IS-IS and BGP routes
- Provide an example of using the administrative distance to modify route redistribution
- Describe how to prevent routing loops with multipoint two-way redistribution

Need for Redistribution

This topic describes the need to use redistribution in the multiple IP routing protocol environment.



There are many reasons for complex routing scenarios. When a network grows, it typically becomes more complex as network administrators adjust it based on factors like political borders, geographical borders, and mergers with other companies. In all such situations, network administrators face complex routing scenarios, in which the use of multiple IP routing protocols is not unusual. With large numbers of routers and routing protocols running, the number of routing updates increases. As the network grows larger, traffic from those updates can slow down the network, indicating that a change is required.

There are many possible solutions when the design of a network must be changed or optimized. A scalable routing protocol may be necessary, requiring careful design and traffic optimizations like these:

- Redistribution
- Routing traffic filtering
- Summarization

Multiple routing protocols may be necessary in situations like these:

- When you are migrating from an older interior gateway protocol (IGP) to a new IGP, multiple redistribution boundaries may exist until the new protocol has completely displaced the old protocol. The same applies to company mergers between companies that are each using a different routing protocol.
- Although you want or need a new protocol, the old routing protocol may still be needed for host systems, such as UNIX host-based routers that are running Routing Information Protocol (RIP).
- Some departments do not want to upgrade their routers to support a new routing protocol.

- You work with mixed-router vendor environments. In these environments, you can use a routing protocol that is specific to Cisco, such as Enhanced Interior Gateway Routing Protocol (EIGRP), in the Cisco portion of the network; you can use a common standards-based routing protocol, like Open Shortest Path First (OSPF), to communicate with devices from other vendors.

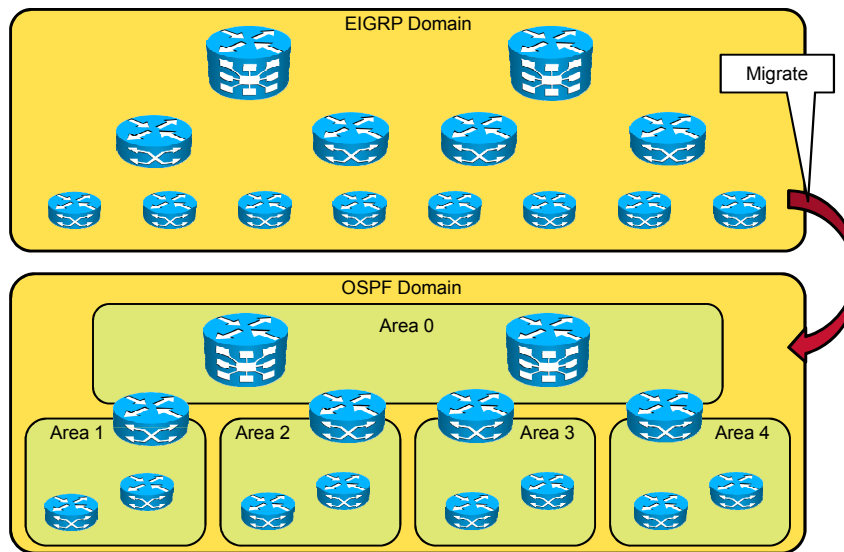
When multiple routing protocols are running in different parts of the network, there may be a need for hosts in one part of the network to reach hosts in the other part. One solution is to advertise a default route into each routing protocol, but this is not always the best policy. The network design may not allow default routes.

If there is more than one way to get to a destination network, routers may need information about routes in other parts of the network to determine the best path to that destination. Additionally, if there are multiple paths, a router must have sufficient information to determine a loop-free path to the remote networks.

Cisco routers allow internetworks that use different routing protocols, referred to as routing domains or autonomous systems, to exchange routing information through a feature called route redistribution.

Redistribution is the way that routers connect different routing domains, so that they can exchange and advertise routing information between the different autonomous systems.

Using Multiple IP Routing Protocols



Whatever the reason for running multiple protocols, network administrators must conduct the migration from one routing protocol to another, or the redistribution of routing information between them, carefully and thoughtfully.

It is important for network administrators to understand what must be changed, and to create a detailed plan before making any changes. An accurate topology map of the network and an inventory of all network devices are also critical for success.

Network administrators must keep in mind the requirements and capabilities of differing routing protocols when they are planning redistribution. Link-state routing protocols, such as OSPF, require a hierarchical network structure. Decisions must be made about which routers will reside in the backbone area, and how to divide the other routers into areas. While EIGRP does not require a hierarchical structure, it operates much more effectively within one. Network administrators must carefully plan the redistribution strategy to avoid disrupting network traffic or causing outages.

Route Redistribution

This topic describes route redistribution.

Redistribution

- Routes learned by some other means are selectively redistributed into a routing protocol from one of three sources:
 - Another routing protocol
 - Static routes
 - Directly connected routes
- Routing loop prevention:
 - Only routes used by the router itself are redistributed.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-6

Redistribution is the process of using a routing protocol to advertise routes that are learned by the usual means of learning routes, such as by another routing protocol, static routes, or directly connected routes.

While it is desirable that you run a single routing protocol throughout your entire IP internetwork, multiprotocol routing is common for many reasons. These reasons include company mergers, multiple departments that are managed by multiple network administrators, and multivendor environments. Running different routing protocols is often part of a network design. Whatever the reason, if you have a multiprotocol environment, redistribution is a necessity.

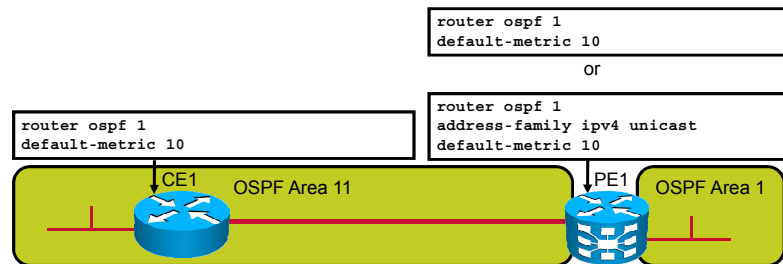
To have a scalable solution and limit the amount of routing update traffic, the redistribution process must selectively insert the routes that are learned. Redistribution can lead to routing loops, which must be avoided. Only routes that are used by the router itself should be redistributed.

Using Seed Metrics in Route Redistribution

This topic describes using seed metrics in route redistribution.

Using Seed Metrics

- The initial, or seed, metric for a route is derived from the directly connected router interface.
- When a compatible metric is established, the metric will increase in increments, just like any other route.
- Use the **default-metric** command to establish the seed metric for the route, or specify the metric when redistributing.



© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUT v1.01-5.7

When a router advertises a link that is directly connected to one of its interfaces, the initial metric (also called the seed metric or default metric) is derived from the characteristics of that interface; this metric increases in increments as the routing information is passed to other routers.

For OSPF, the seed metric is based on the bandwidth of the interface. For EIGRP, the seed metric is based on the interface bandwidth and delay. For RIP, the seed metric starts with a hop count of 0, and increases in increments from router to router.

Redistributed routes are not physically connected to a router; they are learned from other routing protocols. It is difficult to translate from one metric to another, i.e., from hops to bandwidth. If a boundary router is to redistribute information between routing protocols, it must be able to translate the metric of one routing protocol into the metric of the other routing protocol.

For example, if a boundary router receives an RIP route, the route will have a hop count as its metric. To redistribute the route into OSPF, the router must translate the hop count into a cost metric that the OSPF routers will understand. This seed metric is defined during redistribution configuration. Once the seed metric for a redistributed route is established, the metric will increase in increments, normally within the autonomous system (AS).

There is an exception to the rule that metrics increase in normal increments within the AS. OSPF external type 2 (E2) routes maintain their initial metric, regardless of how far they are propagated across an AS.

The Cisco IOS/IOS XE/IOS XR **default-metric** command, which is used in routing process configuration mode, establishes the seed metric for all redistributed routes. Cisco routers also allow the seed metric to be specified as part of the **redistribution** command, either with the **metric** option, by using a route map, or RPL.

Note	Whichever way you set the initial seed metric, you should set it to a value that is larger than the largest metric within the receiving AS, to help prevent suboptimal routing and routing loops.
-------------	---

Default Seed Metrics in Route Redistribution

This topic lists the default seed metrics.

Default Seed Metrics in Route Redistribution

Source	into RIP	into EIGRP	into OSPF	into IS-IS	into BGP (MED)
Connected	1	Interface metric	20 (E2)	0	0
Static	1	Interface metric	20 (E2)	0	0
RIP		Infinite	20 (E2)	0	IGP metric
EIGRP	Infinite		20 (E2)	0	IGP metric
OSPF	Infinite	Infinite		0	IGP metric
IS-IS	Infinite	Infinite	20 (E2)		IGP metric
BGP	Infinite	Infinite	1 (E2)	0	

© 2012 Cisco and/or its affiliates. All rights reserved.

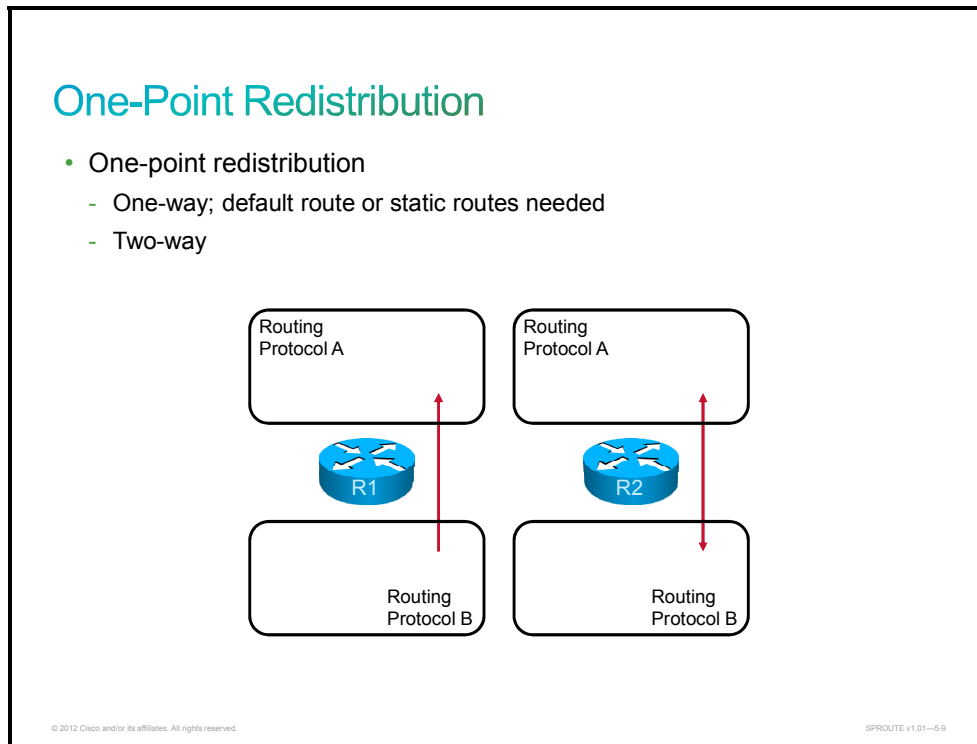
SPROUTE v1.01-5.8

The figure shows the default seed metric value for redistributed routes for each IP routing protocol. RIP and EIGRP do not advertise a redistributed route unless a seed metric is configured.

These protocols interpret the seed metric of 0 as infinity by default. A metric of infinity tells the router that the route is unreachable; therefore, it should not be advertised.

One-Point Route Redistribution

This topic describes redistribution on a single router.



One-point redistribution defines only one redistribution point between two routing protocols. Routes are redistributed on one router only. The redistribution can be one of two types:

- One-way
- Two-way

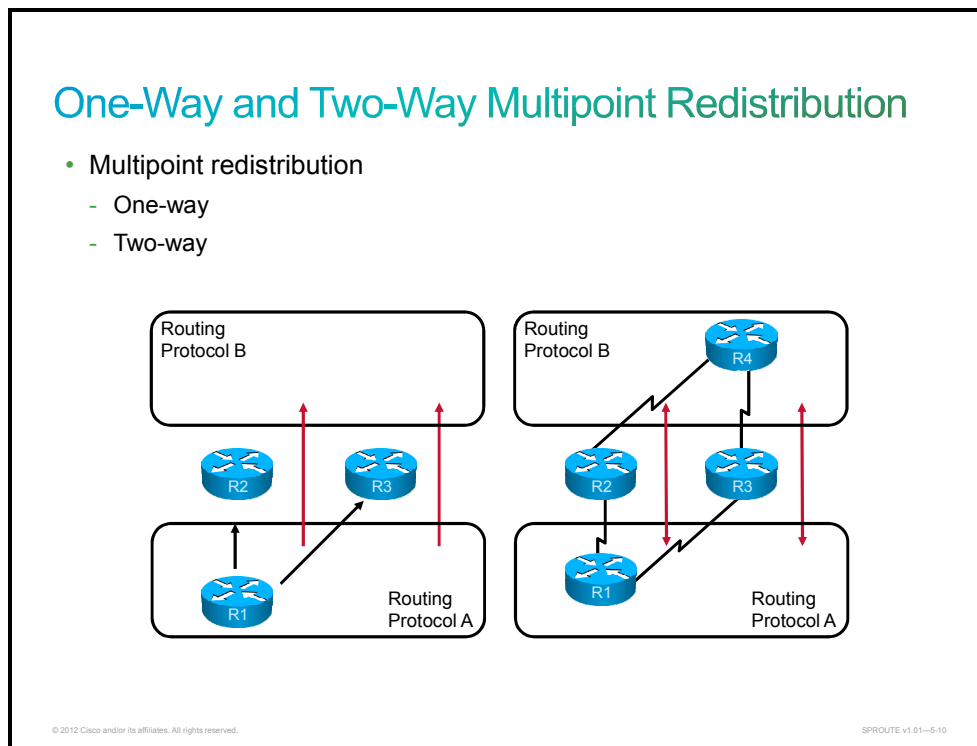
Redistribution is one-way if routes from routing protocol A are redistributed into routing protocol B, but not vice versa. Redistribution is two-way if routes from routing protocol A are redistributed into routing protocol B, and routes from routing protocol B are also redistributed into routing protocol A.

One-way redistribution requires the use of a default route or static routes. If routes are redistributed from routing protocol A into routing protocol B, routing protocol B devices are aware of all the routing information. At the same time, devices in the routing protocol A autonomous system are aware of routing information for their AS only, and reachability for destinations that are outside the routing protocol A autonomous system requires the use of a default route or one or more static routes.

One-way or two-way redistribution at one point is always safe, because one-point redistribution represents the only exit and entrance from one routing protocol to another. Routing loops cannot be inadvertently created.

Multipoint Route Redistribution

This topic describes redistribution on multiple devices that are running both routing protocols.



Multipoint redistribution is redistribution between two routing protocols that takes place on two or more separate devices that are running both routing protocols. Two possibilities exist:

- Multipoint one-way redistribution
- Multipoint two-way redistribution

Multipoint redistribution is likely to introduce routing loops. Even one-way multipoint redistribution is dangerous, and generic multipoint two-way redistribution is highly problematic. Problems often result from differences in administrative distance between the two protocols, and from incompatible metrics. Statically assigned metrics are used in redistribution points.

Multipoint one-way redistribution only works well under these circumstances:

- The receiving routing protocol supports different administrative distances for internal and external routes. Routing protocols that support different administrative distances include EIGRP, BGP, and recent maintenance releases of OSPF.
- The external administrative distance of the receiving routing protocol is higher than the administrative distance of the sending routing protocol.

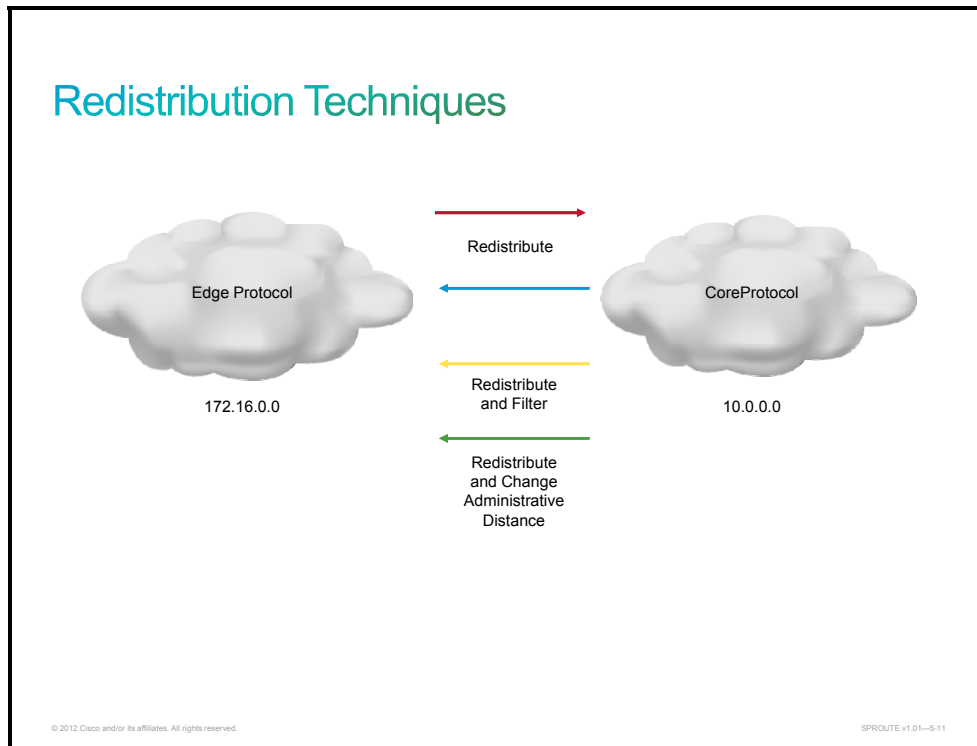
Multipoint two-way redistribution includes difficulties:

- Suboptimal routing (only part of the total cost is considered in routing decisions)
- Self-sustained routing loops on route loss

In multipoint redistribution scenarios, preventing routing loops is a main concern. The redistribution configuration should insert only internal routes from routing protocol A to B and vice versa. Routes at the redistribution points should be tagged and then filtered, based on the tags that are used when doing redistribution in the other direction. Propagation of the metric from A to B and vice versa is recommended, even though it is not sufficient to prevent loops. The easiest way to avoid loops when using two-way redistribution is to use a default route.

Route Redistribution Techniques

This topic describes techniques for performing route redistribution.



The safest way to perform redistribution is to redistribute routes in only one direction, on only one boundary router within the network. To do this, you must first determine which routing protocol is the core routing protocol, and which ones are edge routing protocols.

The core routing protocol is the main routing protocol that is running in the network. During a transition between routing protocols, the core is the new routing protocol and the edge is the old routing protocol. In networks that run multiple routing protocols all the time, the core is usually the more advanced routing protocol.

If redistribution must be done in both directions, or on multiple boundary routers, the redistribution should be tuned to avoid problems like suboptimal routing and routing loops.

Depending on your network design, you may use any of the following redistribution techniques:

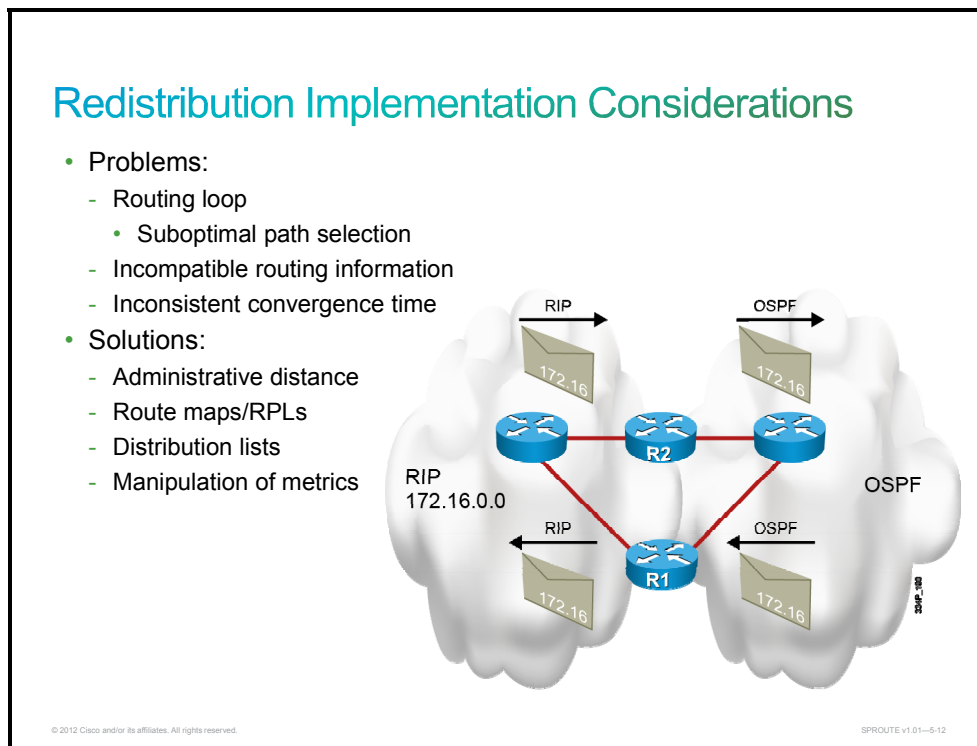
- Redistribute a default route about the core AS into the edge AS. In one-way redistribution, routes from the edge routing protocols are redistributed into the core routing protocol, and a default route is sent back to the edge routers. This technique helps prevent route feedback, suboptimal routing, and routing loops.
- Redistribute multiple static routes about the core AS into the edge AS. The edge routes are still redistributed into the core, but static routes for the core networks are redistributed into the edge protocol and sent to the edge routers. This method works if there is one redistribution point only, but it may cause route feedback if there are multiple points.
- Redistribute routes from the core AS into the edge AS with filtering to block out inappropriate routes. For example, routes from the edge should not be redistributed back into the edge routers from the core via another redistribution point (when there are multiple boundary routers).

- Redistribute all routes from the core AS into the edge AS and from the edge AS into the core AS. Then, modify the administrative distance that is associated with the external routes so that they are not the selected routes when multiple routes exist for the same destination. In some cases, the route that is learned by the native (local) routing protocol is better, but it may have a higher (less believable) administrative distance.

If two routing protocols advertise routes to the same destination, information from the routing protocol with the lowest administrative distance is placed in the routing table. By default, a route that is redistributed into a routing protocol inherits the default administrative distance of that routing protocol.

Redistribution Implementation Considerations

This topic describes potential problems redistributing routes and identifies potential solutions.

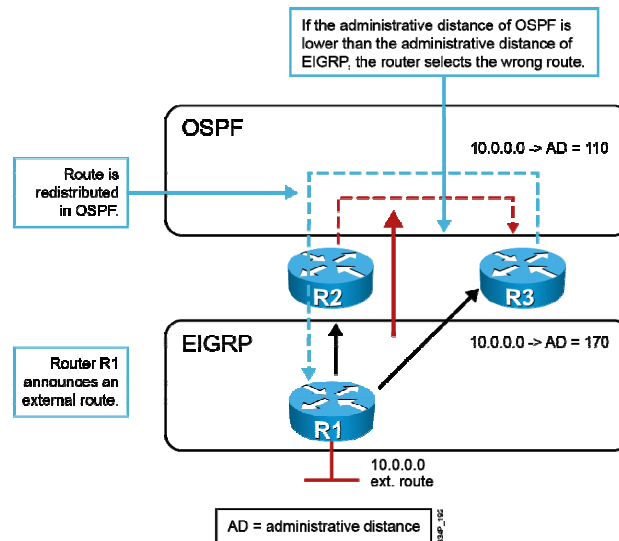


Redistribution of routing information adds to the complexity of a network and increases the potential for routing confusion, so it should be used only when necessary. Several important issues arise when you are using redistribution:

- **Routing feedback (routing loops):** Depending on how you employ redistribution, routers may send routing information that is received from one AS back into that same AS. The feedback is similar to the routing loop problem that occurs in distance vector topologies.
- **Incompatible routing information:** Because routing protocols use different metrics to determine the best path, path selection using the redistributed route information may be suboptimal. The metric information about a route cannot be translated exactly into a different protocol, so the path that a router chooses may not be the best. Generally, to prevent suboptimal routing, you should assign to redistributed routes a seed metric that is higher than any routes that are native to the redistributing protocol. For instance, if RIP routes are being redistributed into OSPF and the highest OSPF metric is 50, the redistributed RIP routes should be assigned an OSPF metric that is higher than 50.
- **Inconsistent convergence time:** Different routing protocols converge at different rates. For example, RIP converges more slowly than EIGRP, so if a link goes down, the EIGRP network will learn about it before the RIP network does.

Good planning will ensure that these issues do not cause problems in your network. Good planning can eliminate the majority of issues, but additional configuration might be required. Some issues may be solved by changing the administrative distance, manipulating the metrics, and filtering using route maps, RPLs, and distribute lists.

One-Way Redistribution Issue



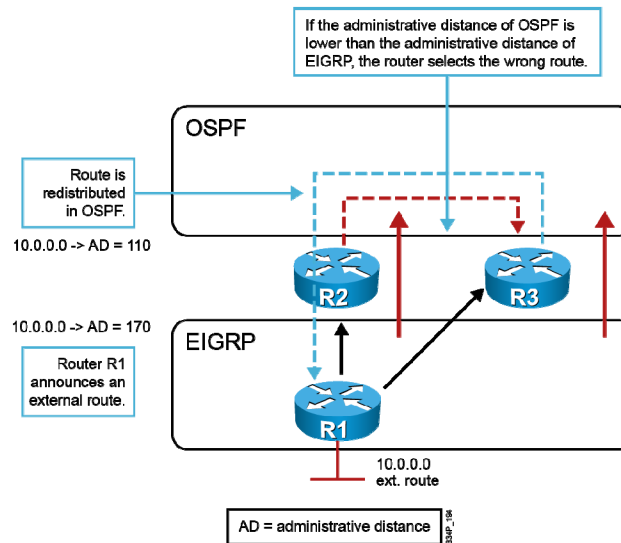
Multipoint one-way redistribution can cause routing confusion if routing protocols are using different administrative distances. It works properly only in these circumstances:

- If the receiving routing protocol supports different administrative distances for internal and external routes
- If the external administrative distance of one routing protocol (OSPF in the figure) is higher than the administrative distance of the second routing protocol (EIGRP in the figure)

Routing protocols that support different administrative distances include EIGRP, BGP, and OSPF.

In the figure, R1 from EIGRP is announcing an external route by sending routing updates to R2 and R3. Both neighboring routers are running two routing protocols, and the redistribution between EIGRP and OSPF takes place on R2. R3 receives routing update information for the same route directly from R1 and via R2, which is sending a redistributed route through OSPF updates. The administrative distance of OSPF (110) is lower than the administrative distance of external EIGRP routes (170), so R3 selects the wrong route. Instead of sending packets directly to R1, R3 prefers the path via R2, and the result is suboptimal routing.

One-Way Multipoint Redistribution Issue

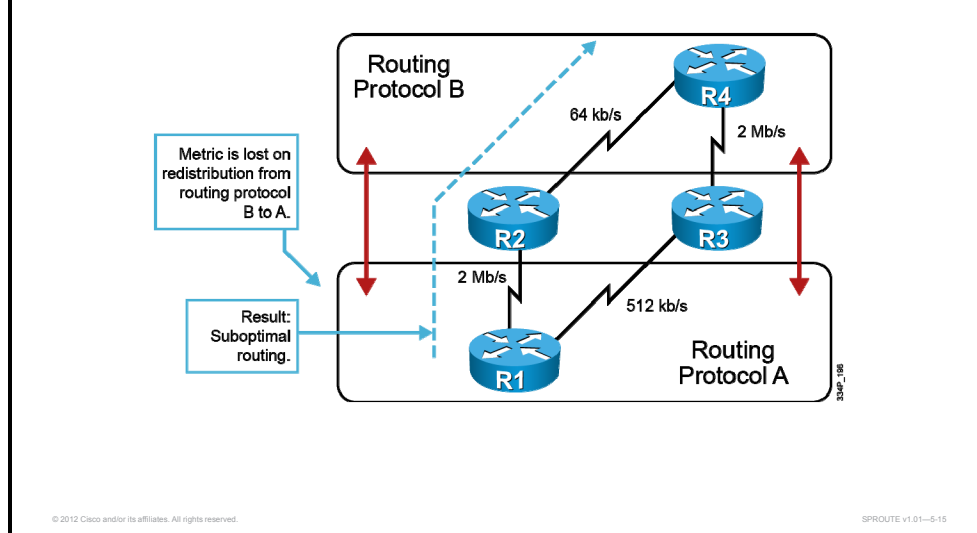


© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-14

In the figure, R1 from EIGRP is announcing routes by sending routing updates to R2 and R3. Both neighboring routers are running two routing protocols, and redistribution between EIGRP and OSPF takes place. Because multipoint redistribution is configured, R2 and R3 receive routing update information for the same route directly from R1 and via OSPF (R2 from R3 and R3 from R2). The routing update information that they receive via OSPF comes in the form of redistributed routes that they receive in OSPF updates. If the administrative distance of OSPF (110) is lower than the administrative distance of EIGRP (170), then R2 also selects the wrong route. Instead of sending packets directly to R1, R2 prefers the path via R3, and the result is suboptimal routing. It can also result in route flapping, because R2 is also redistributing the same EIGRP route in OSPF. R3 also prefers the OSPF path and sends packets via R2. The packet may be bouncing back and forth between R2 and R3 in the OSPF network until its Time to Live (TTL) value reaches 0, before being discarded. Network 10.0.0.0 is then unreachable.

Two-Way Multipoint Redistribution Issue



Generic multipoint two-way redistribution is highly problematic and requires careful design and configuration.

Routing protocols have incompatible metrics, and during redistribution the metric information can be lost. To avoid problems that are associated with incompatible metrics, statically assigned metrics can be used in redistribution points.

A number of problems can occur during multipoint two-way redistribution:

- Suboptimal routing (only part of the total cost is considered in routing decisions)
- Self-sustained routing loops upon route loss

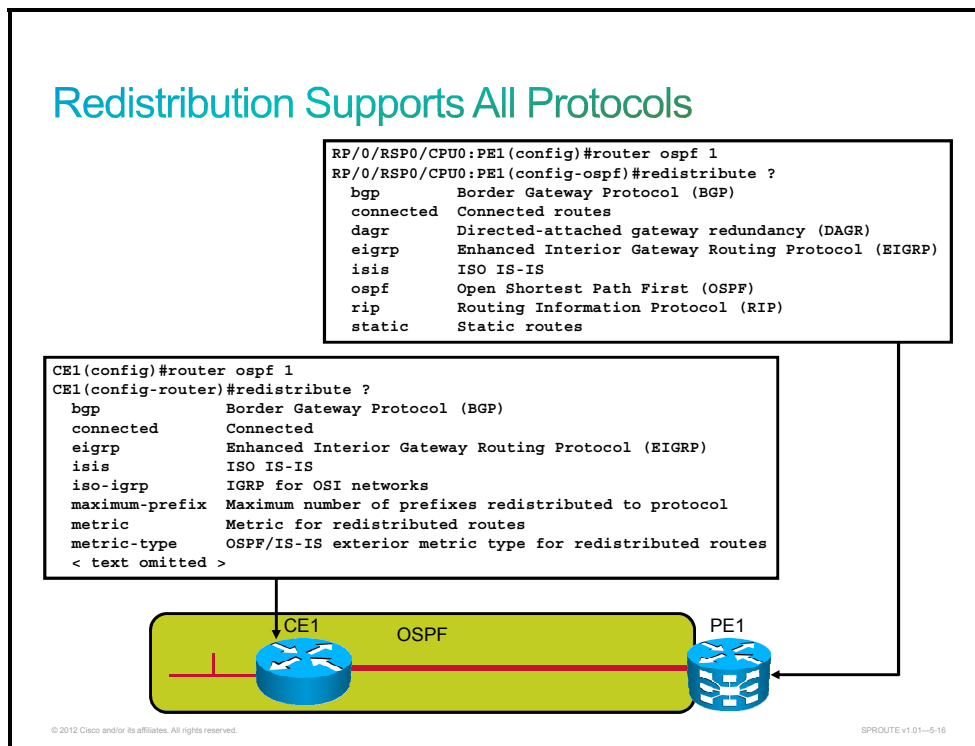
To prevent routing loops in multipoint redistribution scenarios, you should take into account the following recommendations during configuration:

- Insert only internal routes from routing protocol A to B and vice versa.
- Tag routes in redistribution points, and filter based on tags when doing redistribution.
- Propagate metrics from routing protocol A to routing protocol B properly (though this is not sufficient to prevent loops).
- Use default routes to avoid two-way redistribution.

The figure shows a two-way multipoint redistribution issue in which the cost of the internal links in routing protocol A is completely different from the cost of the links in routing protocol B. It is obvious that the best path between R1 and R4 is via R3, but during redistribution from routing protocol B to routing protocol A, the metric is lost and R1 sends the packets toward R4 via R2. The result is suboptimal routing.

Implement Redistribution

This topic describes the procedures that are necessary to configure route redistribution.



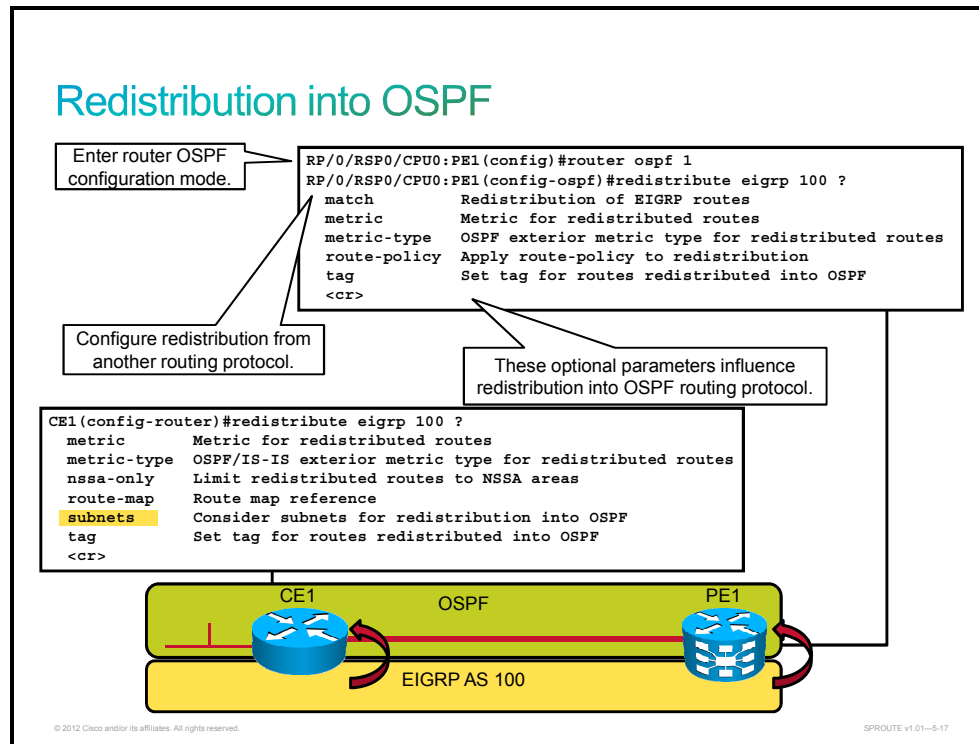
As shown in the figure, redistribution supports all routing protocols (IPv4 and IPv6). Additionally, static and connected routes can be redistributed to allow the routing protocol to advertise the routes without using a network statement for them.

Routes are redistributed into a routing protocol, so the **redistribute** command is entered under the routing process that is to receive the routes. Before implementing redistribution, consider these points:

- Only protocols that support the same protocol stack are redistributed. For example, you can redistribute between IP RIP and OSPF, because they both support the TCP/IP stack or you can redistribute RIP next generation (RIPng) and OSPF version 3 (OSPFv3) for IPv6 routes.
- The method that is used to configure redistribution varies slightly among different routing protocols and combinations of routing protocols. Some routing protocols require a metric to be configured during redistribution, but others do not.

Redistribution into OSPF

This topic describes how to redistribute routes into OSPF.



The figure shows how to configure for redistribution from EIGRP AS 100 into an OSPF routing process.

The first step is to enter router OSPF configuration mode. The Cisco IOS/IOS XE/IOS XR **router ospf 1** command is used to access the OSPF routing process into which the routes need to be redistributed. In this case, it is OSPF process 1.

The next step is to use the Cisco IOS/IOS XE/IOS XR **redistribute** command to specify the routing protocol that is to be redistributed into OSPF. You can use optional keywords to change the way distribution is performed. For example, you might modify the default metric or route filtering using **route-policy** or **route-map**. The default metric is 20, and the default metric type is 2. In the Cisco IOS/IOS XE, subnets are not redistributed by default.

In the OSPFv3, use the Cisco IOS/IOS XE/IOS XR **redistribute** command to redistribute from other routing protocols. This example shows Cisco IOS XR Software output:

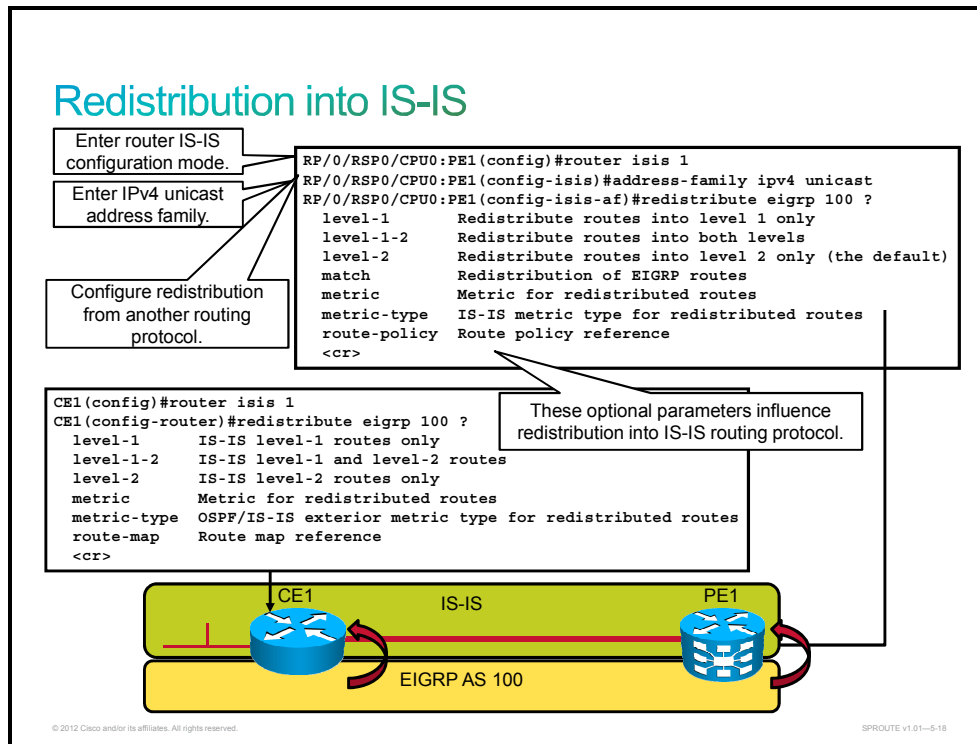
```
RP/0/RSP0/CPU0:PE1(config)#router ospfv3 1
RP/0/RSP0/CPU0:PE1(config-ospfv3)#redistribute ?
  bgp          BGP routes
  connected    Connected routes
  eigrp        Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis         IS-IS routes
  ospfv3       OSPFv3 routes
  static       Static routes
```

This example shows Cisco IOS/IOS XE Software output:

```
CE1(config)#ipv6 router ospf 1
CE1(config-rtr)#redistribute ?
  bgp          Border Gateway Protocol (BGP)
  connected    Connected Routes
  eigrp        Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis         ISO IS-IS
  nd           Neighbor Discovery
  nemo         Network Mobility (NEMO)
  ospf         Open Shortest Path First (OSPF)
  rip          IPv6 Routing Information Protocol (RIPv6)
  static       Static Routes
```

Redistribution into IS-IS

This topic describes how to redistribute routes into IS-IS.

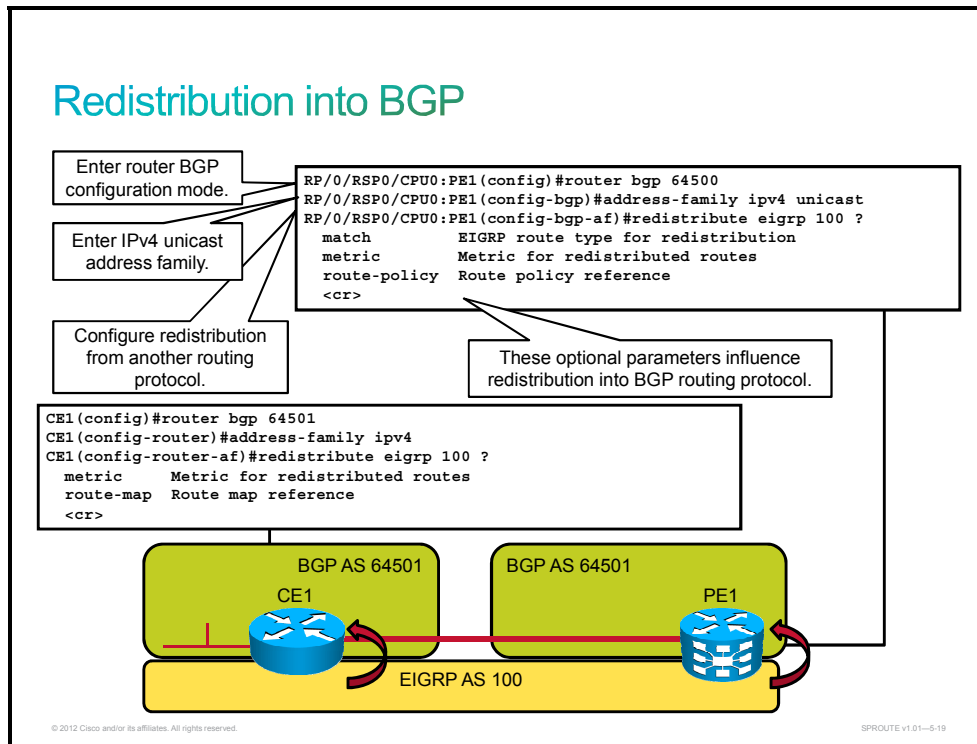


The figure shows how to configure for redistribution from EIGRP AS 100 into an IS-IS routing process. The first step is to enter router IS-IS configuration mode. The Cisco IOS/IOS XE/IOS XR **router isis 1** command is used to access the IS-IS routing process into which the routes will be redistributed. On the Cisco IOS XR use the **address-family ipv4 unicast** router IS-IS command to enter address family for IPv4 unicast. On the Cisco IOS/IOS XE/IOS XR, use the **address-family ipv6 unicast** router IS-IS command to enter address family for IPv6 unicast.

The next step is to use the Cisco IOS/IOS XE/IOS XR **redistribute** command to specify the routing protocol that is to be redistributed into IS-IS. You can use optional keywords to change the way distribution is performed. For example, you might modify the default metric or route filtering by using **route-policy** or **route-map**.

Redistribution into BGP

This topic describes how to redistribute routes into BGP.



The figure shows how to configure for redistribution from EIGRP AS 100 into a BGP routing process. The first step is to enter router BGP configuration mode. The Cisco IOS/IOS XE/IOS XR **router bgp as_number** command is used to access the BGP routing process into which the routes will be redistributed. Use the Cisco IOS/IOS XE/IOS XR **address-family ipv4|ipv6 unicast** router BGP command to enter address family for IPv4 or IPv6 unicast.

The next step is to use the Cisco IOS/IOS XE/IOS XR **redistribute** command to specify the routing protocol that is to be redistributed into BGP. You can use optional keywords to change the way distribution is performed. For example, you might modify the default metric or route filtering using **route-policy** (Cisco IOS XR) or **route-map** (Cisco IOS/IOS XE).

Administrative Distance

This topic describes administrative distance.

Administrative Distance

- Administrative distance is a way of ranking the trustworthiness of routing information.
- A lower administrative distance means a route is more trustworthy.

Route source	Default AD
Connected interface	0
Static route	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
OSPF	110
IS-IS	115
RIPv1, RIPv2	120
External EIGRP	170
Internal BGP	200
Unknown	255

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-20

Most routing protocols have metric structures and algorithms that are not compatible with other protocols. It is critical for a network that is using multiple routing protocols to have a seamless exchange of route information, and the ability to select the best path across multiple protocols.

Cisco routers use a value called administrative distance to select the best path when they learn two or more routes to the same destination from different routing protocols. Administrative distance is a way of ranking the trustworthiness of a routing protocol. Cisco has assigned a default administrative distance value to each routing protocol that is supported on its routers.

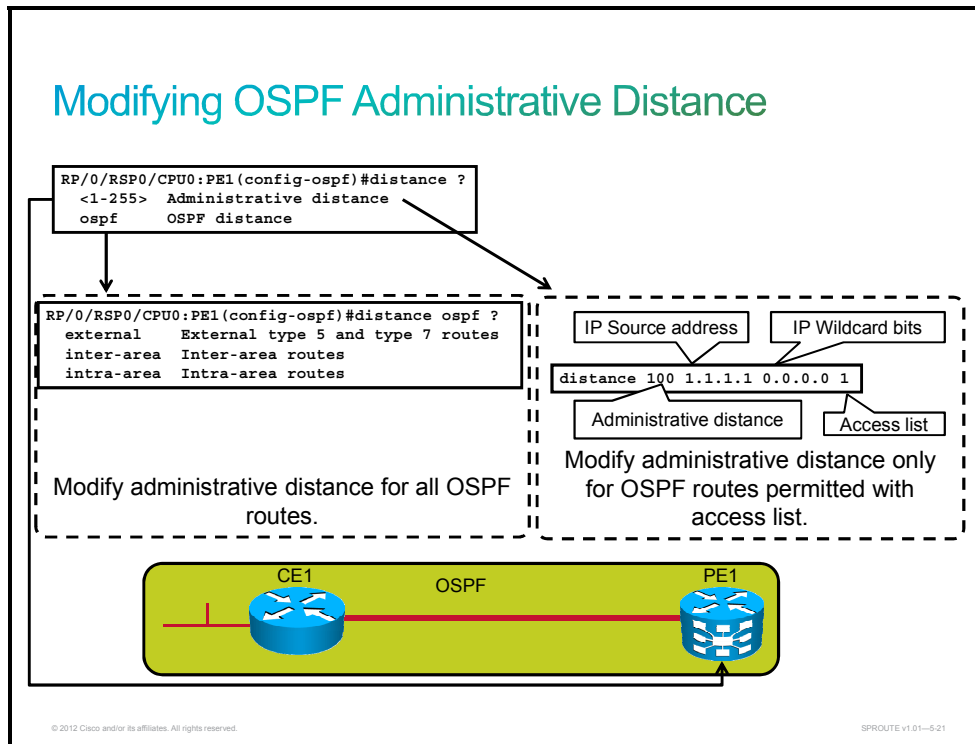
Each routing protocol is prioritized from most believable to least believable. Some examples of prioritization are as follows:

- Prefer manually configured routes (static routes) to dynamically learned routes
- Prefer protocols with sophisticated metrics to protocols with more deterministic metrics
- Prefer External Border Gateway Protocol (EBGP) to most other dynamic protocols

The table in the figure lists the default administrative distances of the protocols that are supported by Cisco routers. The administrative distance is a value between 0 and 255. The lower the administrative distance value, the higher the reliability of the protocol.

Modifying OSPF Administrative Distance

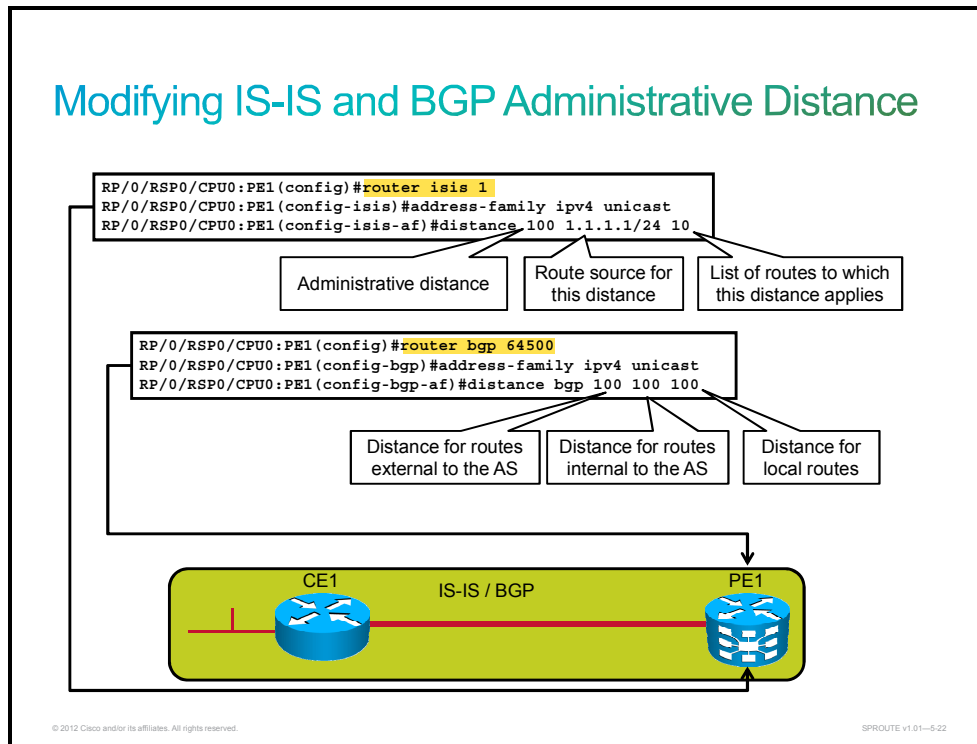
This topic describes how to modify the administrative distance for OSPF routes.



The Cisco IOS/IOS XE/IOS XR **distance** router OSPF command modifies the administrative distance for OSPF routes. OSPF assigns different administrative distance values to external, interarea, and intra-area routes (left example). The default value is 110 for each of them. OSPF allows for the modification of administrative distance only to routes that are matched with access lists and/or those generated by a specific OSPF source (right example).

Modifying IS-IS and BGP Administrative Distance

This topic describes how to modify the administrative distance for IS-IS and BGP routes.



The Cisco IOS/IOS XE/IOS XR **distance** router IS-IS command modifies the administrative distance for IS-IS routes. In the Cisco IOS XR Software, IS-IS distance for IPv4 unicast routes needs to be applied in the IPv4 unicast address family. The administrative distance can be modified for all IS-IS routes or for individual routes from individual sources. In the Cisco IOS/IOS XE Software, IS-IS distance for IPv4 unicast routes is applied in router IS-IS configuration mode, as shown in the example:

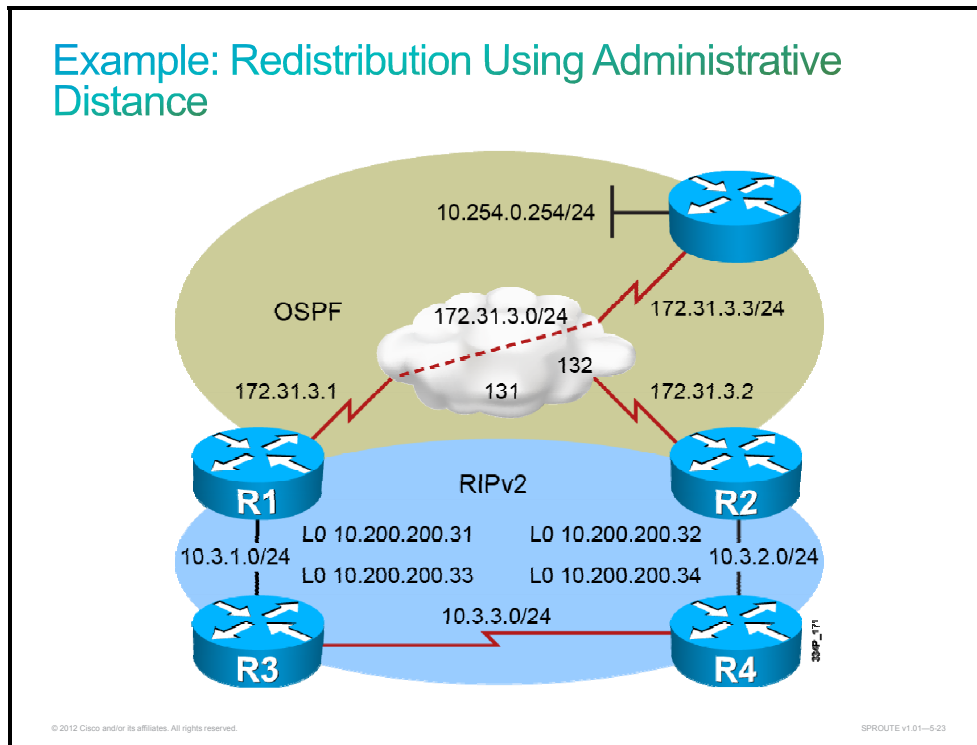
```
CE1(config)#router isis 1
CE1(config-router)#distance 100 1.1.1.1 0.0.0.0 10
```

For BGP, use the Cisco IOS/IOS XE/IOS XR **distance bgp** command. BGP assigns different administrative distance values to routes that are learned through EBGp, routes that are learned through IBGP, and local routes. The default values that are used are 20, 200, and 200, respectively. The BGP distance for IPv4 unicast routes needs to be applied in the IPv4 unicast address family. The Cisco IOS/IOS XE Software also allows you to change administrative distance only for specific BGP routes that are received from a specified BGP source, as shown in the example:

```
CE1(config)#router bgp 64501
CE1(config-router)#address-family ipv4 unicast
CE1(config-router-af)#distance 100 1.1.1.1 0.0.0.0 10
```

Example: Redistribution Using Administrative Distance

This topic provides an example of using the administrative distance to modify route redistribution.



This example describes the before-and-after results of using the administrative distance to modify route redistribution. The figure shows a network that is using multiple routing protocols. There are several ways to correct path selection problems in a redistribution environment. These examples show how problems occur, where they occur, and one possible way to resolve them. Redistribution using administrative distance is presented. The figure illustrates a network with RIP and OSPF routing domains. Recall that OSPF is more believable than RIP, because OSPF has an administrative distance of 110 and RIP has an administrative distance of 120.

If, for example, a boundary router (R1 or R2) learns about network 10.3.3.0 via RIP version 2 (RIPv2) and also via OSPF, the OSPF route will be used and inserted into the routing table, because OSPF has a lower administrative distance than RIPv2, even though the path via OSPF might be the longer (less desirable) path.

Example: Redistribution Using Administrative Distance (Cont.)

R1#

```
router ospf 1
redistribute rip metric 10000 metric-type 1 subnets
network 172.31.0.0 0.0.255.255 area 0
!
router rip
version 2
redistribute ospf 1 metric 5
network 10.0.0.0
no auto-summary
```

R2#

```
router ospf 1
redistribute rip metric 10000 metric-type 1 subnets
network 172.31.3.2 0.0.0.0 area 0
!
router rip
version 2
redistribute ospf 1 metric 5
network 10.0.0.0
no auto-summary
```

3549_173

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-524

The figure illustrates the configurations for R1 and R2. These configurations redistribute RIP into OSPF, and OSPF into RIP on both routers.

The redistribution into OSPF sets a default OSPF metric of 10000, to make these routes less preferred than native OSPF routes and to protect against route feedback. The **redistribute** statement also sets the metric type to E1, so that the route metrics continue to accrue, and the router redistributes subnet information.

The redistribution into RIP sets a default RIP metric of 5, also to protect against route feedback.

Example: Redistribution Using Administrative Distance (Cont.)

With OSPF and RIP running



```
R2#show ip route
<Output Omitted>

Gateway of last resort is not set

  172.31.0.0/24 is subnetted, 7 subnets
    O   172.31.55.0 [110/2343] via 172.31.3.3, 00:09:46, Serial0/0/0
    C   172.31.3.0 is directly connected, Serial0/0/0
    O   172.31.2.0 [110/1562] via 172.31.3.3, 00:09:46, Serial0/0/0
  10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
    O E1 10.3.1.0/24 [110/10781] via 172.31.3.1, 00:09:47, Serial0/0/0
    O E1 10.3.3.0/24 [110/10781] via 172.31.3.1, 00:04:51, Serial0/0/0
    C   10.3.2.0/24 is directly connected, fastethernet0/0
    O E1 10.200.200.31/32 [110/10781] via 172.31.3.1, 00:09:48, Serial0/0/0
    O E1 10.200.200.34/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
    C   10.200.200.32/32 is directly connected, Loopback0
    O E1 10.200.200.33/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
    O E2 10.254.0.0/24 [110/50] via 172.31.3.3, 00:09:48, Serial0/0/0
```

- R2 includes suboptimal paths and loops.

The figure displays the routing table on R2 after redistribution has occurred. R2 learned RIP and OSPF routes, but it lists only OSPF routes in the routing table.

The first edge router to set up redistribution has a normal routing table and retains the RIP routes. The second edge router chooses the OSPF routes over its RIP routes. The paths to the internal RIP routes are shown as going through the core because of the dual mutual redistribution points.

OSPF is informed about the RIP routes via redistribution. OSPF then advertises the RIP routes via OSPF routes to its neighboring router. The neighbor router is also informed about the same routes via RIP. However, OSPF has a better administrative distance than RIP, so the RIP routes are not put into the routing table.

OSPF was configured on R1 first, then R2 received information about the internal (native RIP) routes from both OSPF and RIP. It prefers the OSPF routes, because OSPF has a lower administrative distance. Therefore, none of the RIP routes appear in the table.

Refer back to the topology diagram to trace some of the routes. The redistribution has resulted in suboptimal paths to many of the networks.

For instance, 10.200.200.34 is a loopback interface on R4. R4 is directly attached to R2. However, the OSPF path to that loopback interface goes through R1, then R3, and then R4 before it reaches its destination. The OSPF path that is taken is actually a longer (less desirable) path than the more direct RIP path.

Example: Redistribution Using Administrative Distance (Cont.)

R1#

```
router ospf 1
 redistribute rip metric 10000 metric-type 1
 subnets
 network 172.31.0.0 0.0.255.255 area 0
 distance 125 0.0.0.0 255.255.255.255 64
 !
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
 !
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

R2#

```
router ospf 1
 redistribute rip metric 10000 metric-type 1
 subnets
 network 172.31.3.2 0.0.0.0 area 0
 distance 125 0.0.0.0 255.255.255.255 64
 !
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
 !
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

394/283

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-28

One of the boundary routers (R2 in this example) selected the poor paths, because OSPF has a better administrative distance than RIP. You can change the administrative distance of the redistributed RIP routes to ensure that the boundary routers select the native RIP routes, as shown in the figure.

The **distance** command modifies the administrative distance of the OSPF routes to the networks that match ACL 64.

ACL 64 is used to match all the native RIP routes. The **access-list 64 permit 10.3.1.0** command configures a standard access control list (ACL) to permit the 10.3.1.0 network. Other similar access list statements permit the other internal native RIP networks.

In the figure, both of the redistributing routers are configured to assign an administrative distance of 125 to OSPF routes that are advertised for the networks that are listed in ACL 64.

ACL 64 has permit statements for the internal native RIP networks of 10.3.1.0, 10.3.2.0, and 10.3.3.0, as well as the loopback networks of 10.200.200.31, 10.200.200.32, 10.200.200.33, and 10.200.200.34.

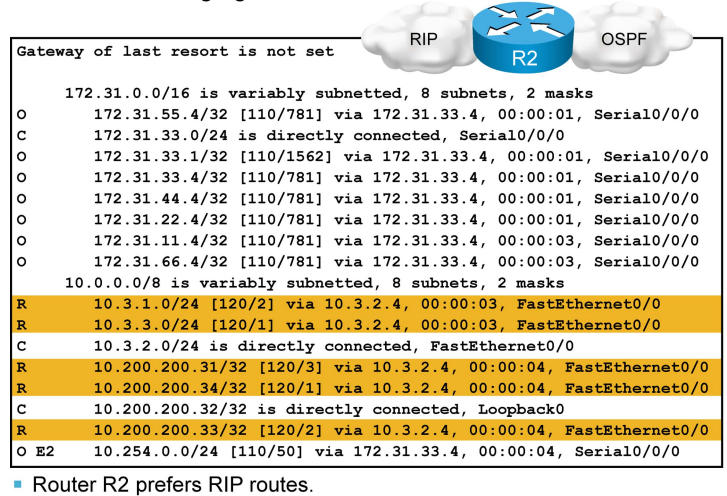
When either one of the redistributing routers learns about these networks from RIP, it selects the routes that are learned from RIP (with a lower administrative distance of 120) over the same routes that are learned from OSPF (with an administrative distance of 125), and puts only the RIP routes in the routing table.

Note that the **distance** command is part of the OSPF routing process configuration, because the administrative distance should be changed for these routes when they are advertised by OSPF, not by RIP.

You must configure the **distance** command on both redistributing routers, because either one of them can have suboptimal routes, depending on which redistributing router first sends the OSPF updates about the RIP networks to the other redistributing router.

Example: Redistribution Using Administrative Distance (Cont.)

With OSPF changing administrative distance



The figure shows that R2 now retains the more direct paths to the internal networks by learning them from RIP. However, some routing information is lost with this configuration. For example, depending on the actual bandwidths, the OSPF path may have been better for the 10.3.1.0 network. It may have made sense not to include 10.3.1.0 in the ACL.

This example illustrates the importance of knowing your network before you implement redistribution, and of closely examining the results of the redistribution to determine which routes are now being selected.

Pay particular attention to routers that can select from a number of possible redundant paths to a network, because they are more likely to select suboptimal paths.

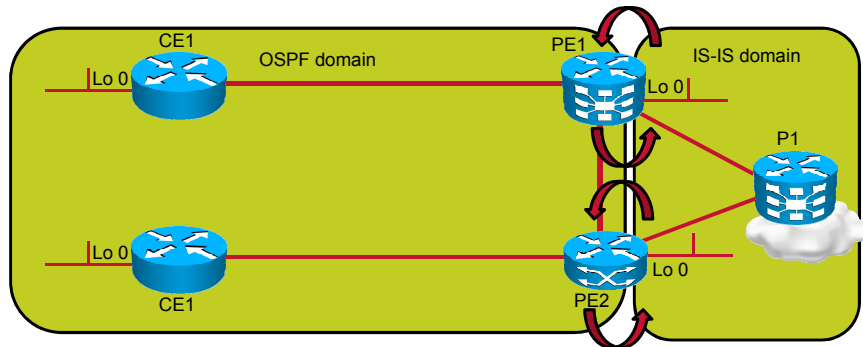
The most important benefit of using administrative distance to control route preference is that no path information is lost; the OSPF information is still in the OSPF database. If the primary path is lost, the OSPF path can reassert itself, and the router will maintain connectivity with those networks.

Redistribution to Prevent Routing Loops

This topic describes how to prevent routing loops with multipoint two-way redistribution.

Redistribution to Prevent Routing Loops

- Simple redistribution results in a loop after a route is lost.
- To prevent routing loops, use RPLs or route maps:
 - Redistribution of internal routes only
 - Route tagging

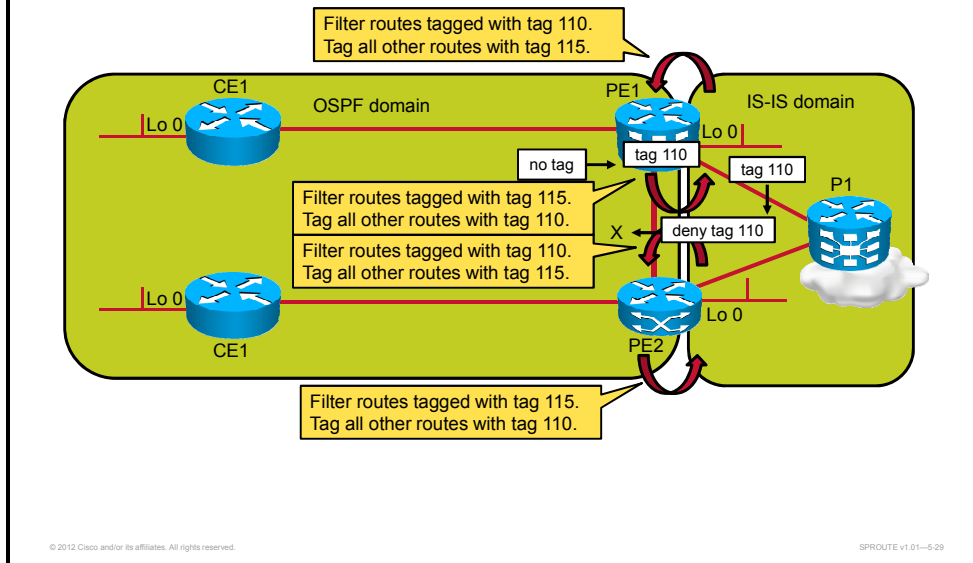


The figure represents multipoint two-way redistribution. With this kind of redistribution, it is very likely that routing loops exist. Suppose that PE1 and PE2 are both running IS-IS and OSPF routing protocols.

As networks are learned from the other IS-IS routers, they will also be redistributed into OSPF. Similarly, OSPF routes are redistributed into IS-IS. With this type of multipoint two-way redistribution, routing loops, or suboptimal routing, may occur. You can use RPLs or route maps to prevent routing loops:

- Redistributing internal routes only
- Route tagging

Solution With Route Tagging



To avoid routing loops, several solutions exist, which often have to be combined. Among these solutions, you can use either RPLs or route maps to redistribute internal routes, or you can use route tagging. You can also manipulate the metric during the redistribution process.

The solution that uses RPLs or route maps and tagging is shown in the figure. IS-IS routes that are redistributed into OSPF are tagged with the value 115 during the redistribution process. In the opposite direction, routes that are redistributed into IS-IS are filtering tagged updates. Updates with a tag of 115 are not allowed to go back to the IS-IS routing protocol. This will effectively break the routing loop.

In the example, OSPF routes that are redistributed into IS-IS are tagged with the value 110 during the redistribution process. In the opposite direction, routes that are redistributed into OSPF are filtering tagged updates. Updates with a tag of 110 are not allowed to go back to the OSPF routing protocol.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- In networks with more than one routing protocol, redistribution is needed to exchange routing updates.
- Redistribution supports all routing protocols and connected and static routes.
- The seed metric is used as initial metric of a redistributed route.
- Routing protocols have different seed metrics by default.
- One-point redistribution defines only one redistribution point between two routing protocols.
- Multipoint redistribution is redistribution between two routing protocols that takes place on two or more separate devices that are running both routing protocols.
- Before implementing redistribution, you must first determine which routing protocol is the core routing protocol, and which ones are edge routing protocols.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-30

Summary (Cont.)

- Be aware that routing loops can be introduced when you use multipoint two-way redistribution.
- Redistribution is configured under router configuration mode using the redistribute command.
- The default seed metric when redistributing into OSPF is 20.
- The default seed metric when redistributing into IS-IS is 0.
- The default seed metric when redistributing into BGP is the same as IGP metric.
- Administrative distance is used to rank the trustworthiness of a routing protocol.
- Administrative distance of OSPF is 110 by default and can be changed using the **distance** command.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-31

Summary (Cont.)

- Administrative distance of IS-IS is 115 by default and can be changed using the distance command.
- You can manipulate administrative distance to prevent suboptimal routing when performing multipoint redistribution.
- You can use route tagging to prevent routing loops when performing multipoint redistribution.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6.32

Large, complex networks must be adjusted based on factors like political borders, geographical borders, and mergers with other companies. In such complex scenarios, running multiple IP routing protocols is common. Using a routing protocol to advertise routes that are learned by some other means, such as by another routing protocol, static routes, or directly connected routes, is called redistribution.

When a router advertises a link that is directly connected to one of its interfaces, the initial, or seed, metric that is used is derived from the characteristics of that interface. During redistribution the default metric is used. Redistribution can be one-point (one-way or two-way) or multipoint (one-way or two-way). One-point redistribution, whether one-way or two-way, is always safe, because it represents the only exit from one routing protocol to another. In contrast, multipoint redistribution is likely to introduce routing loops, so careful design and configuration must be applied.

The most important issues that arise when you are using redistribution are routing feedback (routing loops), incompatible routing information, and inconsistent convergence time. Solutions to these issues include correct design in terms of the administrative distance, manipulation of metrics, and filtering using RPLs or route maps and distribution lists.

Influencing BGP Route Selection

Overview

Routes that are learned by routers, via Border Gateway Protocol (BGP), have properties associated with them that aid a router in determining the best route to a destination when there are multiple paths to that destination. These properties are referred to as BGP attributes. This lesson introduces the role of BGP attributes, and explains how their presence influences route selection in BGP. Understanding how BGP attributes influence route selection is required for the design of robust networks.

This lesson provides advanced information on how to connect customers to service providers, and to connect service providers to upstream service providers. It includes an in-depth description of the BGP attributes that are used in route selection, including weight, local preference, autonomous system (AS) path prepending, multi-exit discriminator (MED), and BGP communities.

Objectives

Upon completing this lesson, you will be able to complete the correct BGP configuration to successfully connect the service provider network to the customer or upstream service provider. This ability includes being able to meet these objectives:

- Describe the use of BGP weights to influence the BGP route selection process
- Describe how to configure per-neighbor weights
- Describe how to change BGP weights using RPLs or Route Maps
- Describe the order of operation in setting BGP weights
- Describe how the BGP local preference attribute influences BGP route selection
- Describe how to change the local preference
- Describe how to monitor the local preference values
- Describe the function of AS path prepending and how you can use it to facilitate proper return path selection
- Describe design considerations for implementing AS path prepending
- Describe how to configure AS path prepending

- Describe how to monitor AS path prepending
- Describe how AS path prepending can impact AS path filtering
- Describe how MED can be used to facilitate proper return path selection
- Describe how to change the MED
- Describe how to monitor MED values
- Describe how BGP communities facilitate proper return path selection
- Describe how to configure BGP Communities
- Describe BGP named community lists
- Describe the use of sequenced entries in extended community lists
- Describe how to set attributes based on community values
- Describe how to monitor BGP community values
- Show examples of using BG communities

BGP Weight

This topic describes the use of BGP weights to influence the BGP route selection process.

Influencing BGP Route Selection Outbound Traffic

- BGP routing policy can be specified by using:
 - **Weight:** provides local routing policy (within a router)
 - **Local preference:** provides AS-wide routing policy
- BGP weights are specified per neighbor.
 - Default weight
 - AS path-based weight
 - Complex criteria with RPLs or route maps

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5.4

When connections to multiple providers are required, it is important that Border Gateway Protocol (BGP) select the optimum route for traffic to use. The optimum, or best, route may not be what the network designer intended, based on design criteria, administrative policies, or corporate mandate. Fortunately, BGP provides many tools for administrators to use to influence route selection. One of these tools is the weight attribute.

BGP route selection criteria take the weight parameter into consideration first. If a router has two alternative paths to the same destination, and their weight values are different, BGP selects the route with the highest weight value as the best. Only when the two alternatives have equal weight is the next criterion, local preference, checked.

A high local preference value is preferred over a low value. Only when the two alternatives have an equal local preference is the next criterion checked.

The weight attribute is local to a single router only. The weight value is never propagated by the BGP protocol, and this value constitutes a routing policy local to the router.

Local preference is assigned to a route as an attribute. This attribute is carried with the route on all internal BGP sessions. In this situation, all other BGP-speaking routers within the autonomous system (AS) receive the same information. Normally, a router assigns a local preference to a route that is received on an external BGP session before it is accepted and entered in the BGP table of the border router. Routers propagate the local preference attribute on internal BGP sessions only. This policy constitutes a routing policy for the entire AS.

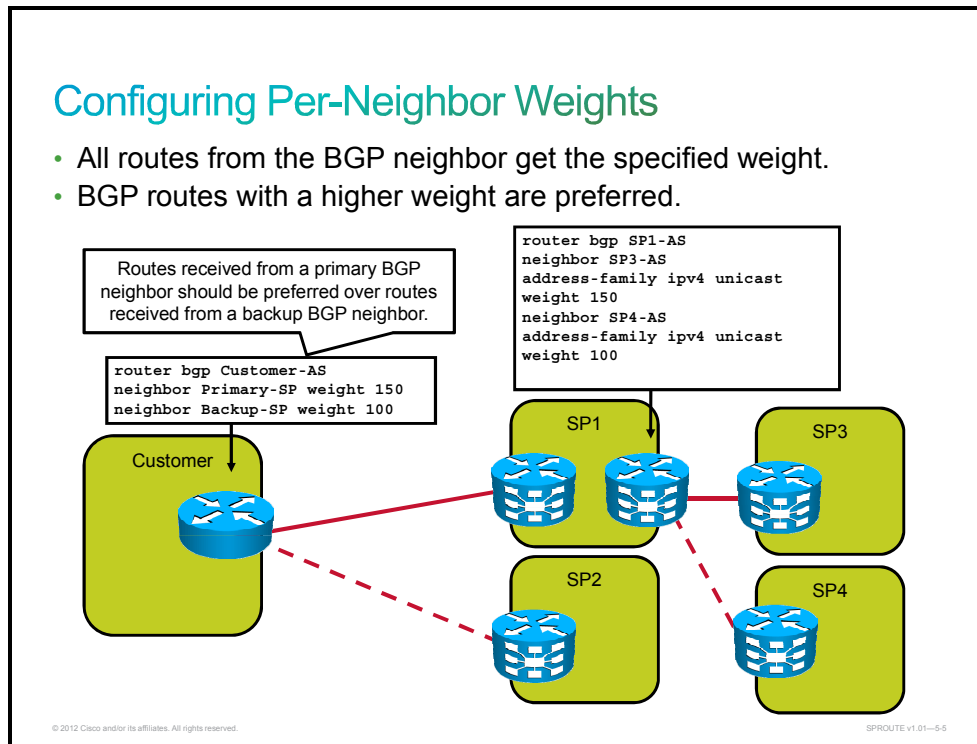
The router can assign the weight attribute to a route in two ways:

- All routes that are received from a specific neighbor can be assigned a default weight value. This weight value indicates that the neighbor is preferred over the other neighbors.
- A routing policy language (RPL) command or route map that is applied on incoming routes from a neighbor can be used to select some routes and assign them weight values.

If configured, the default weight assignment on routes that are received from a neighbor is applied first. All routes that are received from the neighbor are assigned a weight value as defined by the default weight.

Configuring Per-Neighbor Weights

This topic describes how to configure per-neighbor weights.



All routes that are received from the neighbor after the configuration line is in place are assigned the weight value. If no weight value is specified, the default value of 0 is applied.

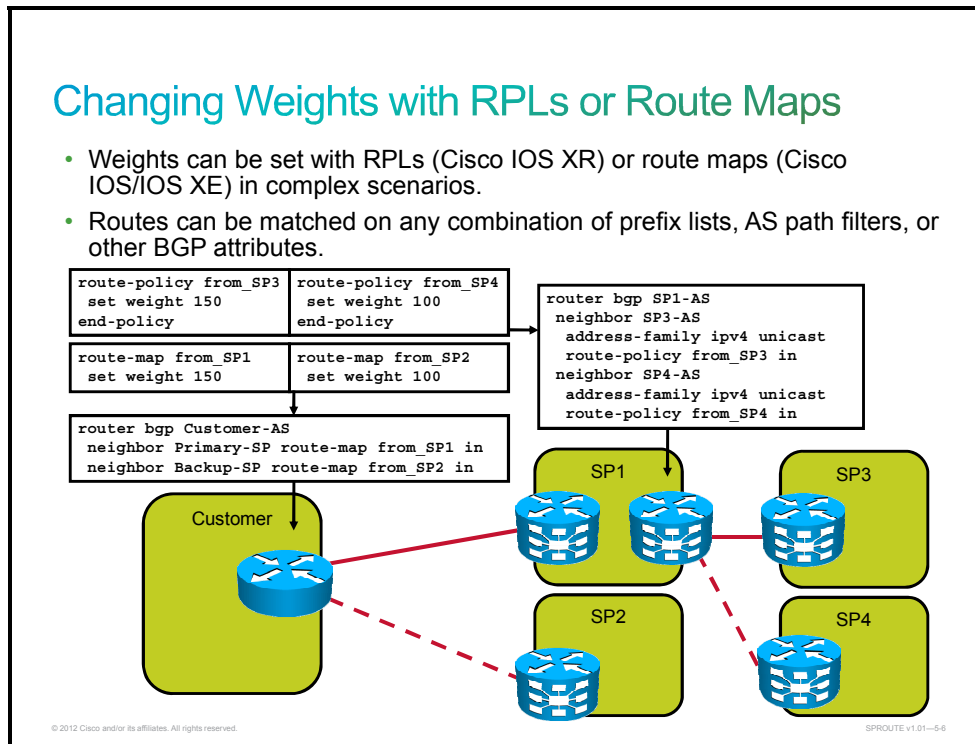
The weight is configured on both BGP sessions, giving a higher weight to the routes that are received from the primary BGP neighbor, compared to those that are received from the backup BGP neighbor.

Verify BGP weights for routes received from a neighbor by using Cisco IOS/IOS XE **show ip bgp** or Cisco IOS XR **show bgp** commands.

Weight is applied only to new incoming updates. To enforce new weights, reestablish BGP sessions with your neighbors by using the Cisco IOS/IOS XE **clear ip bgp** or Cisco IOS XR **clear bgp** commands.

Changing Weights with RPLs or Route Maps

This topic describes how to change BGP weights using RPLs or Route Maps.



The RPL and route map are powerful tools to select and alter routing information. The RPL is used in the Cisco IOS XR Software while route map is used in the Cisco IOS/IOS XE Software. When an RPL or route map is applied to incoming information from a BGP neighbor, each received update is examined as it passes through the RPL or route map.

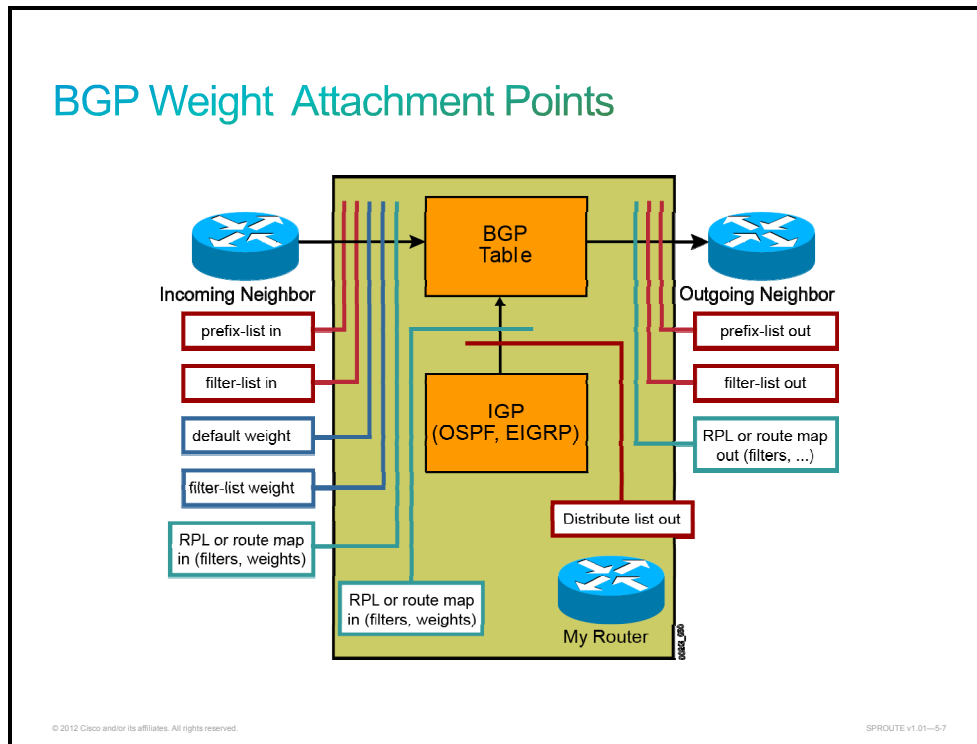
These examples show the RPLs and route maps used in the figure:

```
route-policy from_SP3
set weight 150
!
route-map from_SP4
set weight 100

route-map from_SP1 permit
set weight 150
!
route-map from_SP2 permit
set weight 100
```

BGP Weight Attachment Points

This topic describes the order of operation in setting BGP weights.

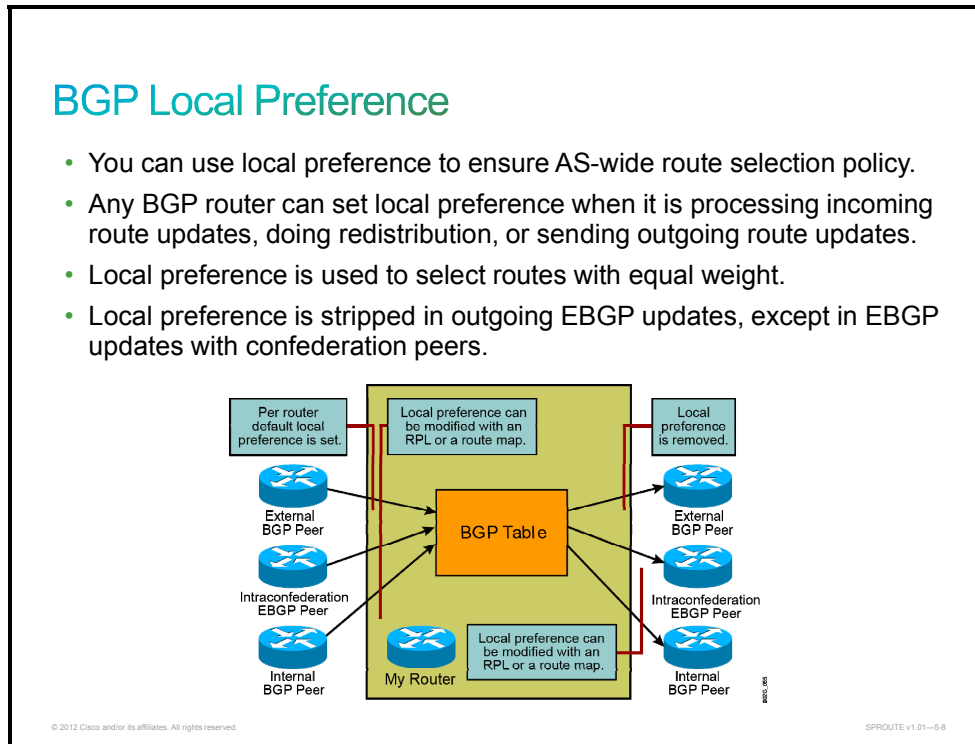


The figure shows all the possible applications of prefix lists, filter lists, weights, RPLs, and route maps. They are applied in the order indicated.

Prefix lists and filter lists, both in and out, filter out routes and discard those that are not permitted. Weight setting is applicable only on incoming routes because a router never propagates the weight attribute to its neighbors. RPLs or route maps can be filters that discard routes, but they can also be used to modify and set various attributes on both incoming and outgoing routes.

BGP Local Preference

This topic discusses how to influence BGP route selection by setting the BGP local preference attribute of incoming BGP routes. Local preference is similar to the weight attribute, except that the weight attribute is local to the specific router on which it is configured. Two methods that are used to set the local preference attribute, default local preference and RPLs or route maps, are discussed in this topic.



Local preference is similar to weight; because it is an attribute, you can set it once and then view it on neighboring routers without having to reset it. This attribute has a default value of 100, which the router will apply to locally originated routes and updates that come in from external neighbors. Updates that come from internal neighbors already have the local preference attribute.

Local preference is the second highest attribute in the BGP route selection sequence. If there are two or more paths available for the same network, a router will first compare weight, and if the weights are equal for all paths, the router will then compare the local preference attributes. The path with the highest local preference value will be preferred.

The local preference attribute is automatically stripped out of outgoing updates to External Border Gateway Protocol (EBGP) sessions. This practice means that you can use this attribute only within a single AS to influence the route selection process.

BGP Local Preference (Cont.)

- Local preference is the second highest attribute in the BGP route selection sequence.
- Remember the BGP route selection rules:
 - Highest weight preferred (local to router)
 - Highest local preference preferred (global within AS)
 - Other BGP route selection rules
- Weights configured on a router override local preference settings.
- To ensure consistent AS-wide route selection:
 - Do not change local preference within the AS.
 - Do not use BGP weights.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-5-8

Local preference is the second highest attribute in the BGP route selection sequence. Remember the route selection rules:

1. Prefer the highest weight (local to router).
2. Prefer the highest local preference (global within AS).
3. Process all remaining BGP route selection rules.

Because network administrators can use both weight and local preference to manipulate the route selection process, they must decide which one to use. If local preference is used, the weight should be the same for all paths.

Network administrators can use weight on an individual router to override local preference settings that are used in the rest of the AS.

In most cases, it is enough to change the default local preference on updates coming from external neighbors. Network administrators should avoid changing the local preference attribute on internal sessions to prevent unnecessary complexity and unpredictable behavior.

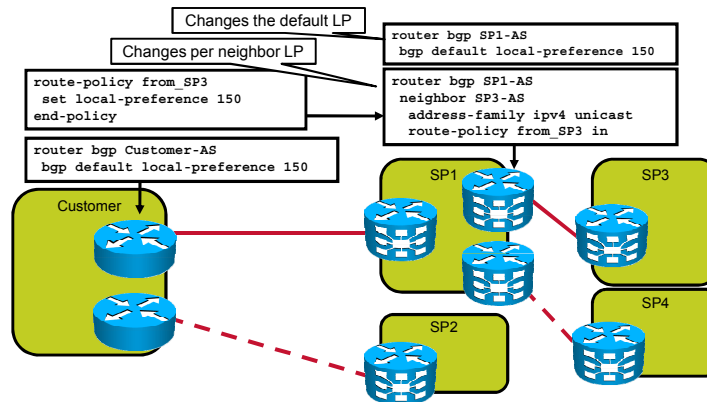
The BGP weight is local to the router, while most commonly used attributes in the BGP route selection are BGP local preference, AS path, and MED.

Changing Local Preference

This topic describes how to change the local preference.

Changing Local Preference

- A default local preference value is applied to all routes that do not have local preference set (EBGP routes).
- The default value of local preference is 100, allowing you to specify more desirable or less desirable routers.



Network administrators can apply local preference in the following ways:

- Use an RPL or route map with the Cisco IOS/IOS XE/IOS XR **set local-preference** command. You can use the RPL or route map on incoming updates from all neighbors, or on outgoing updates to internal neighbors (not recommended).
- Use the Cisco IOS/IOS XE/IOS XR **bgp default local-preference** command to change the default local preference value that is applied to all updates that come from external neighbors, or that originate locally.

Setting a value lower than the default of 100 will result in the router preferring internal paths to external paths (normally a router would prefer external routes). Setting a value higher than 100 will result in external paths being preferred to all internal paths (also those with a shorter AS path).

Monitoring Local Preference

This topic describes how to monitor the local preference values.

Monitoring Local Preference

```
RP/0/RSP0/CPU0:PE1#show bgp
< text omitted >
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop        Metric LocPrf Weight Path
*> 10.1.1.1/32      0.0.0.0          0         32768 i
*> 10.1.10.1/32     192.168.101.11   0         0 6450
*>i10.2.1.1/32      10.2.1.1         0        100
*>i10.2.10.1/32     10.2.1.1         0        100 0 64502 i
< text omitted >
```

Nondefault LP is displayed.

LP from the IBGP peer is displayed.

```
RP/0/RSP0/CPU0:PE1#show bgp 10.1.10.1/32
< text omitted >
Paths: (1 available, best #1)
  Advertised to peers (in unique update groups):
    10.0.1.1
  Path #1: Received by speaker 0
  Advertised to peers (in unique update groups):
    10.0.1.1
    64501
    192.168.101.11 from 192.168.101.11 (10.1.10.1)
      Origin IGP, metric 0, localpref 100, valid, external, best, group-best
      Received Path ID 0, Local Path ID 1, version 9
```

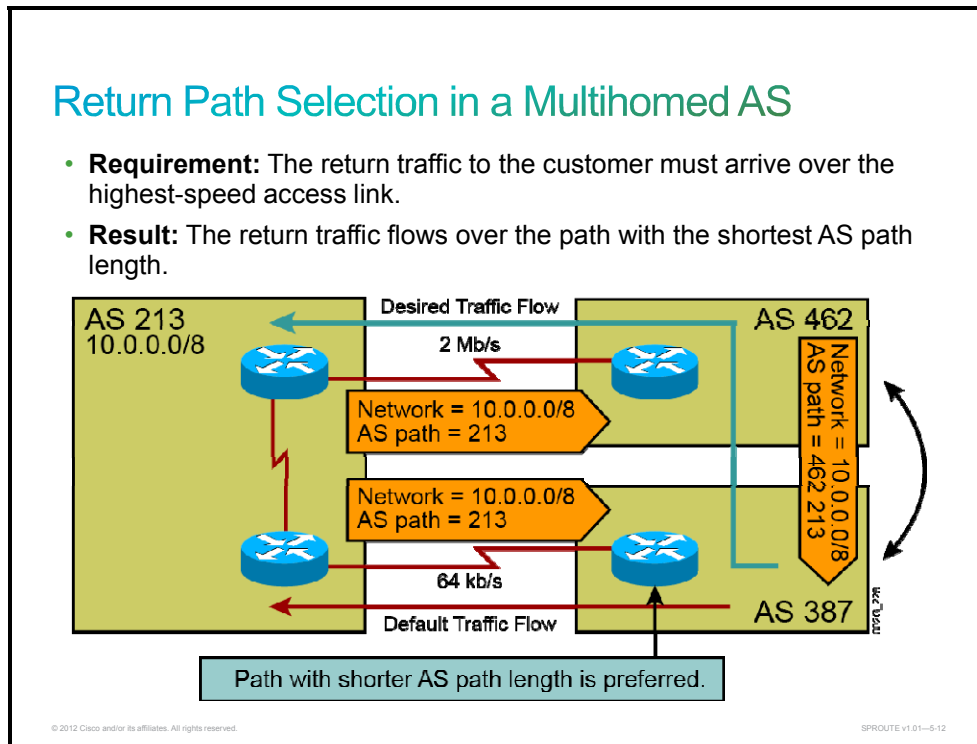
LP is always displayed.

© 2012 Cisco and/or its affiliates. All rights reserved. SPRUTE v1.01-5-11

Although local preference is not a mandatory attribute, it is applied to every route. When you are using the Cisco IOS/IOS XE **show ip bgp** or Cisco IOS XR **show bgp** commands, a locally applied default value is not shown. All other values are displayed. You should use the Cisco IOS/IOS XE **show ip bgp prefix** or Cisco IOS XR **show bgp prefix** command to also display the locally applied value.

AS Path Prepending

Problems can arise when administrative policies mandate that a specific return path be used for traffic that is returning to the AS, but AS path prepending potentially allows the customer to influence the route selection of its service providers. This topic describes AS path prepending and the Cisco IOS/IOS XE/IOS XR commands that are required to properly configure and monitor AS path configurations; it also describes the filtering requirements for influencing route selection using AS path prepending.



It is fairly easy for an AS to select the appropriate path for outgoing traffic. It is much more complicated to influence other autonomous systems to select the appropriate path for traffic that is returning to a specific AS.

Configuring the preferred path for outgoing traffic only (and not for incoming or return traffic) is likely to result in an asymmetrical traffic flow, as well as suboptimal performance of the return traffic. In the figure, outgoing traffic is directed to the high-speed line (2 Mb/s) as a result of configuring local preference or weight. However, the return traffic from AS 387 would take the default path over the low-speed line (64 kb/s). The low-speed line would be a limiting factor in the overall performance of that network.

In this example, AS 213 requests AS 387 to send packets toward network 10.0.0.0/8 via AS 462. The reason for this request is to improve network performance and minimize delay (assuming, of course, that the connectivity between AS 387 and AS 462 is better than the direct 64-kbps link between AS 387 and AS 213).

If no BGP path selection tools are configured on the route to influence the traffic flow, AS 387 will use the shortest AS path. This action will result in unwanted behavior, because the return traffic to AS 213 will be sent over the low-speed WAN link.

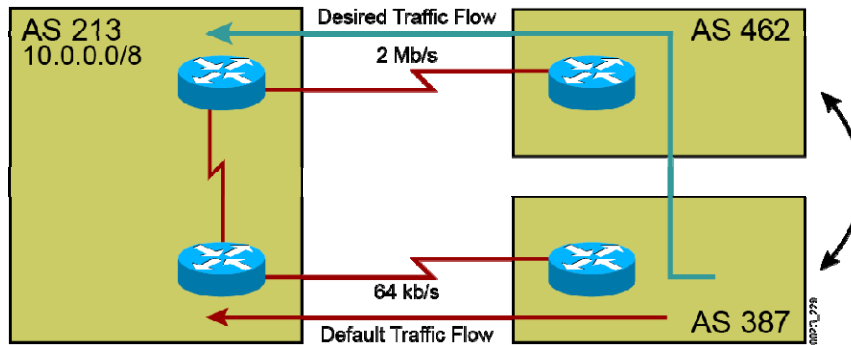
Proper Return Path Selection

Q: How do you select the proper return path from AS 387?

A: Use local preference in AS 387.

Q: Will the administrator of AS 387 configure it?

A: Unlikely.



Remember that the incoming traffic flow (from the perspective of AS 213) will be a result of the route selection for outgoing traffic in AS 387. The traffic that is going out from AS 387 will end up as incoming traffic in AS 213.

If AS 387 configures some changes that cause the route selection process for outgoing traffic to prefer to reach network 10.0.0.0/8 via AS 462, the changes would result in behavior matching the desired administrative policy for AS 213, which specifies that incoming traffic to the AS should be received over the high-speed link.

One way to accomplish the desired administrative policy in AS 213 is to configure the router in AS 387, which is receiving EBGp updates directly from AS 213, to assign a local preference value less than the default value (100) to all routes that are received from AS 213. The router in AS 387 is also configured specifically not to set local preference on EBGp routes that are received from AS 462. This configuration results in assignment of the default value of 100 to all routes received from AS 462. When the route selection process in AS 387 selects the best route to reach network 10.0.0.0/8, the difference in local preference values causes AS 387 routers to select the path via AS 462 as the best.

However, all the configuration work to complete this process must be performed in AS 387. The network administrators of AS 387 would be required to modify the router configurations in AS 387 to satisfy the administrative policy requirements of AS 213. All changes must be documented and maintained according to the rules and procedures that have been adopted by AS 387.

If AS 387 is a major Internet service provider (ISP), the network administrators most likely are too busy doing other things to tailor router configurations that are based on the demand of a single leaf (nontransit) AS that lacks bandwidth, on a redundant connection.

AS Path Prepending

- BGP route selection uses these criteria:
 - Prefer largest weight.
 - Prefer largest local preference.
 - Prefer routes that the router originated.
 - **Prefer shorter AS paths.**
 - Use other route selection rules.
- Manipulating the outgoing AS path length (called **AS path prepending**) could result in proper return path selection.
- The AS path should be extended with multiple copies of the **AS number of the sender.**
- AS path prepending is used to achieve these goals:
 - Ensure proper return path selection.
 - Distribute the return traffic load for multihomed customers.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5-14

Recall that BGP route selection uses the following criteria:

- Prefer the largest weight.
- Prefer the largest local preference.
- Prefer routes that the router originated.
- Prefer shorter AS paths.
- Then, prefer all other route selection criteria.

It is unlikely that the operator of an AS can request changes in router configurations in another AS. This limitation makes it virtually impossible to influence another AS to select the desired path, based on the weight and local preference attributes, because both options would require configuration changes in the neighboring AS.

But if both the weight and the local preference parameters are left at their default settings, they will not indicate a difference. This configuration causes the route selection process to continue down the list of selection criteria. The third criterion for selection will not influence route selection in this scenario, because none of the routes originated at the router that is performing the route selection. The fourth criterion will apply, however, because the AS paths have different lengths.

If the AS path is not manually manipulated by some administrative means, the path going over the fewest number of autonomous systems is selected by the router regardless of available bandwidth. However, if the AS that is attempting to influence the incoming traffic flow is sending out EBGp updates with a manipulated AS path attribute over that undesired path, the receiver of this update is less likely to select it as the best because the AS path now appears to be longer.

The benefit of manipulating AS paths to influence the route selection is that the configuration that is needed is done in the AS that is requesting a desired return path.

You can manipulate AS paths by prepending AS numbers to existing AS paths. Normally, you perform AS path prepending on outgoing EGP updates over the undesired return path. Because the AS paths sent out over the undesired link become longer than the AS path sent out over the preferred path, the undesired link is now less likely to be used as the return path.

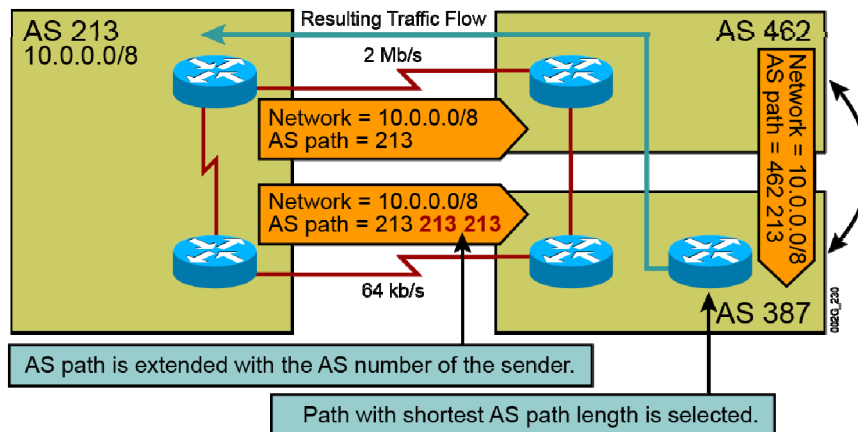
The length of the AS path is extended because additional copies of the AS number of the sender are prepended to (added to the beginning of) the AS path attribute. To avoid clashes with BGP loop prevention mechanisms, no other AS number, except that of the sending AS, should be prepended to the AS path attribute.

If another AS number is prepended in the AS path, the routers in the AS that has been prepended will reject the update because of BGP loop prevention mechanisms.

You can configure prepending on a router for all routing updates that you send to a neighbor or only on a subset of them.

AS Path Prepending (Cont.)

- **Result:** The return traffic flows over the desired return path.



As long as the high-speed link between AS 213 and AS 462 is available, all traffic should flow toward AS 213 using the high-speed link.

To accomplish this goal, you can configure the router in AS 213 that sends EBGp updates to AS 387 by prepending the AS path with two copies of the AS number 213. AS 387 receives two alternative routes to reach network 10.0.0.0/8: the update that it has received directly from AS 213 (that has a manipulated AS path with a length of three) and the update that it has received via AS 462 (that was not manually manipulated and therefore contains an AS path length of two).

When AS 387 starts the route selection process to determine which route to use to reach network 10.0.0.0/8, it checks the AS path length after the weight and local preference parameters. In this case, neither weight nor local preference has been configured, so the length of the AS path will be the deciding factor in the route selection process. Consequently, AS 387 prefers the shortest AS path and thus forwards packets toward network 10.0.0.0/8 via AS 462. The desired administrative policy has been met, and AS 213 will receive incoming traffic over the high-speed link.

If the forwarding path from AS 387 via AS 462 to AS 213 and network 10.0.0.0/8 is later broken, the BGP update to reach network 10.0.0.0/8 is revoked. In case of such a network failure, AS 387 will have only one remaining path to reach network 10.0.0.0/8. The route selection process now has only one choice, the route directly to AS 213 over the low-speed WAN link. The low-speed link will therefore serve as backup to the high-speed WAN link.

AS Path Prepending Design Considerations

This topic describes design considerations for implementing AS path prepending.

AS Path Prepending Design Considerations

- There is no exact mechanism to calculate the required prepended AS path length.
- If a primary and backup scenario is desired, consider this strategy:
 - Use a long prepended AS path over the backup link to ensure that the primary AS path will always be shorter.
 - A long backup AS path consumes memory on every Internet router.
 - Experiment with various AS path lengths until the backup link is idle.
 - Add a few more AS numbers for additional security (unexpected changes in the Internet).
- If traffic load distribution is desired, consider this strategy:
 - Start with a short prepended AS path, monitor the link use, and extend the prepended path length as needed.
 - Continuously monitor the link use and change the prepended AS path length if required.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-18

How many copies of the AS number of the sender should you prepend to the AS path? The answer depends on the goals of the administrative policy. In the general case, it is not easy to determine the exact number of required AS numbers to prepend. The sending AS does not know what alternative paths are available to other autonomous systems.

The following are two typical cases in which you can use AS path prepending for return path selection:

Establishing a primary link and a backup link: As an announced backup (prepended) route propagates through the Internet, all the routers along the way that receive the route need to store it together with its AS path attribute. If this information is long, it will consume extra memory in these routers. However, because routers forward only the routes that are selected as best, an AS that receives multiple alternatives to a destination will select the route with the shortest AS path and forward only that route.

If both the primary and the secondary link are up, the neighboring AS will receive two routes to the same destination that differ only in the AS path length. The route with the shorter AS path will be subsequently advertised through the Internet.

If the primary link fails, the route with the longer AS path is the only remaining route. As a result, the primary route is withdrawn, and the prepended route is advertised through the Internet. In this case, extra memory will be consumed in each Internet router because of the storage of the prepended (longer) AS path.

The longer the AS path that is announced to the EBGP neighbor on the other side of the backup link, the less likely it is that incoming traffic will be received from that neighbor. The network administrator can make a clever guess about how many copies of the AS number to prepend. After the prepending is implemented, the network administrator has to examine the result. If the expected result is not achieved, the configuration can be changed and a few more copies of the AS number can be prepended.

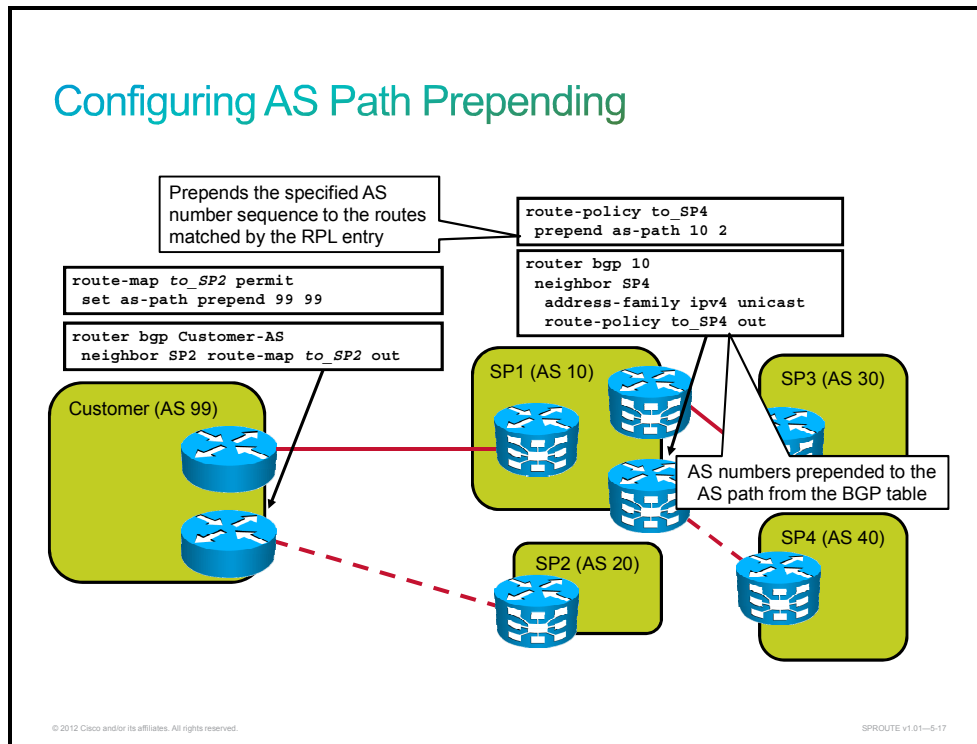
After AS path prepending has generated the desired results, the network administrator may take the precaution of prepending a few more copies of the AS number to the AS path. This action protects the customer from packets being routed over the backup link at a possible later stage when the topology between remote autonomous systems has unexpectedly changed, yielding a longer AS path to reach the primary link.

Distributing the load of return traffic: In a multihomed scenario, there is no way to predetermine the exact volume of traffic that will be received over a particular link. The traffic load on different links will change, depending on where the senders are located (in which autonomous systems they belong.). The network topology and the way that different remote autonomous systems are interconnected may also change with time, changing the load distribution. Only constant monitoring and fine-tuning will ensure that the desired results are achieved.

In a first attempt at load distribution, the network administrator can configure a router that is connected to an overused link to prepend only a few extra copies of the local AS number. After the network has been given time to converge, the network administrator must check the change in load distribution. The monitoring of the load must be done for a period long enough to be statistically significant (several days or more). If enough volume of traffic has not moved from the overused link to the underused link, the administrator must prepend more copies of the local AS number, and the process of resending local routes and monitoring the results starts all over again.

Configuring AS Path Prepending

This topic describes how to configure AS path prepending.



You can configure manual manipulation of the AS path attribute (prepending) using an RPL or route map with the Cisco IOS/IOS XE route map **set as-path prepend** command or the Cisco IOS XR RPL **prepend as-path** command. The RPL or route map is used to prepend the specified AS numbers to outgoing EBGp route updates that are matched with the **match** condition. AS path prepending is completed first, and then the route is subject to the normal AS path modification procedures when it is sent over an EBGp session.

Monitoring AS Path Prepending

This topic describes how to monitor AS path prepending.

Monitoring AS Path Prepending

- AS path prepending cannot be monitored or debugged on the sending router.
 - **debug bgp (debug ip bgp)** updates displays the BGP entry prior to RPL or route map processing.
 - **show policy-map (show route-map)** does not display how many routes have matched a RPL or route map entry.
- The results of AS path prepending can be observed on the receiving router.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--6-18

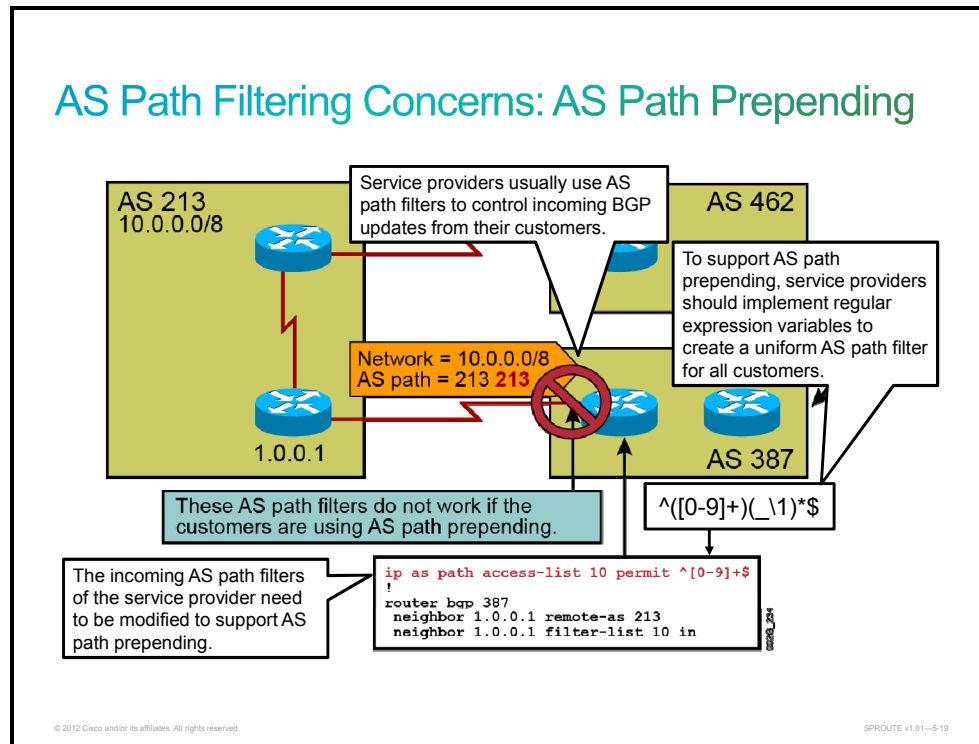
When you are monitoring AS path prepending, the router doing the prepending is not the proper point to observe the results of the AS path prepend operation. For instance, output from the Cisco IOS/IOS XE **debug ip bgp updates** or the Cisco IOS XR **debug bgp update** command does not display the prepended paths, because the RPL or route map doing the prepending is applied afterward.

The Cisco IOS/IOS XE **show route-map** or the Cisco IOS XR **show policy-map** command displays the configuration details of an RPL or route map. The matching criteria and AS path manipulation are displayed as output of the command. However, there is no indication of how many routes have been matched by an RPL or route map statement, and have therefore had their AS paths manipulated.

A better place for observing AS path prepending is on the router receiving the BGP update that contains the prepended AS path. At that point, you can use the pattern of AS number sequences in the received AS path attribute of received routes to find the routes that have a prepended AS path.

AS Path Filtering Concerns: AS Path Prepending

This topic describes how AS path prepending can impact AS path filtering.



Service providers normally expect their customers to send routes that originate only in the AS of the customer. However, because customers might not do so, proactive thinking and care for the rest of the Internet cause the service provider to implement AS path filters on incoming updates that are received from their customers.

The network administrator of the service provider in the figure could configure individual filters for each neighbor. However, a single AS path access list permitting only AS paths with a length of exactly one AS number would be a better solution because the service provider can uniformly apply it to all incoming routes from all customers.

In the figure, the service provider (AS 387) has configured a filter list, which allows only AS paths that have a length of one AS number. When the customer changes its router configuration and starts to announce network 10.0.0.0/8 with a prepended AS path, the filter-list for incoming routes to AS 387 in the service provider router will filter those routes out. This filtering results in a situation where the network 10.0.0.0/8 is not reachable over the link between AS 213 and AS 387. Therefore, the backup function is not available.

Network 10.0.0.0/8 is, however, still reachable via the path going through AS 462. This situation means that AS 387 can send packets to network 10.0.0.0/8 but not over the direct link to AS 213. This failure may be difficult to detect because, during normal conditions, all autonomous systems in the figure can exchange traffic.

After AS 387 loses the route to network 10.0.0.0/8 via AS 462, possibly because the primary link between AS 213 and AS 462 is gone, the problem will be obvious. AS 387 can now no longer reach network 10.0.0.0/8 at all, although the physical link between AS 213 and AS 387 is available. Because the AS of the service provider will receive customer routes with prepended AS paths that have a length greater than one AS number, the provider must modify its incoming filters.

The service provider needs to create a new inbound regular expression filter, using regular expression variables and parentheses for recall.

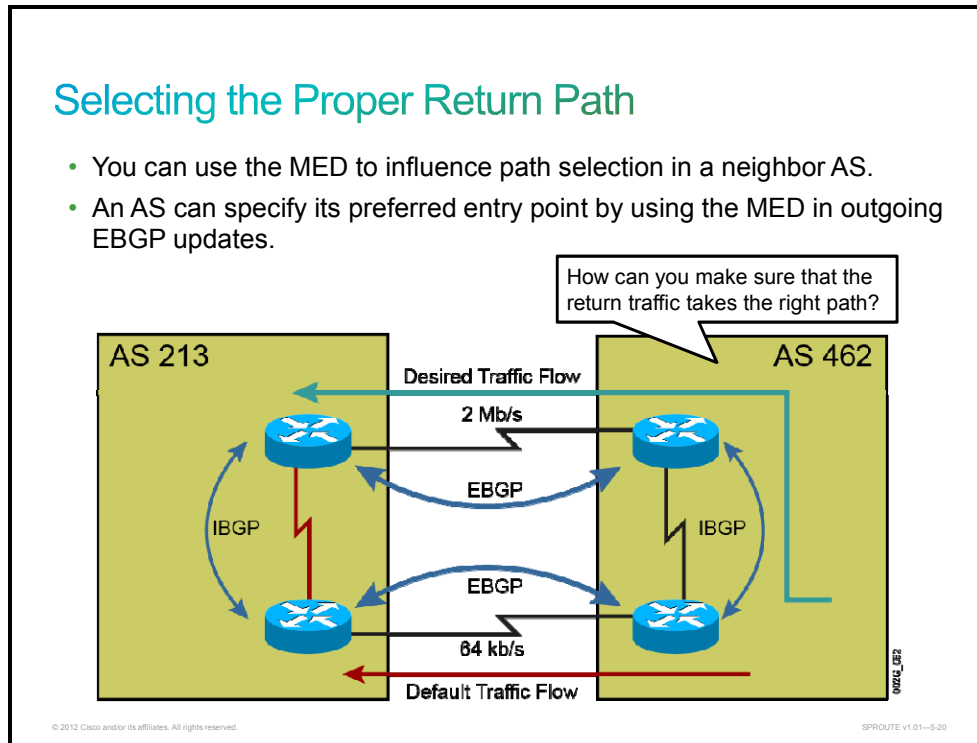
What is needed is a filter that will allow any AS path containing one or multiple copies of the same AS number. An example of such a filter is as follows:

```
^([0-9]+)(_\1)*$
```

This filter matches any AS path beginning with any AS number and continues with no or multiple repetitions of that same AS number (the variable “\1” repeats the value in the brackets). The regular expression would therefore match AS paths 99 99 99, 2 2 2, or 100, but it would not match AS path 100 99.

BGP Multi-Exit Discriminators

BGP provides a tool for administrators to influence route selection, the multi-exit discriminator (MED) attribute. This topic discusses how to influence BGP route selection by setting the BGP MED attribute of outgoing BGP routes. Two methods that are used to set the MED attribute, the default MED and RPLs or route maps, are discussed in this topic.



When multiple connections between providers are required, BGP attributes such as weight and local preference solve only half the problem: how to choose the right path out of the AS. Here we will focus on the second, more complex half of the problem: how to influence neighboring autonomous systems to choose the correct return path back into the AS.

The MED attribute is a hint to external neighbors about the preferred path into an AS when multiple entry points exist. You can apply the MED attribute on outgoing updates to a neighboring AS to influence the route selection process in that AS. The MED attribute is useful only when there are multiple entry points into an AS.

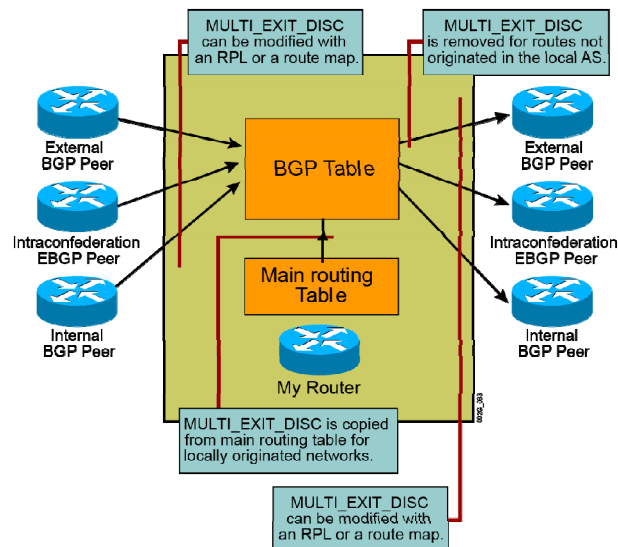
The MED attribute, which is sent to an external neighbor, will be seen only within that AS. An AS that receives a route that contains the MED attribute will not advertise that MED beyond its local AS.

The default value of the MED attribute is 0. A lower value of the MED attribute indicates a more preferred path.

The MED attribute is considered a “weak” metric. In contrast with weight and local preference, a router will prefer a path with the smallest MED value, but only if the weight, local preference, AS path, and origin code attributes are equal. Using the MED may not yield the expected result if the neighboring AS modifies any of the stronger BGP route selection mechanisms.

In Cisco IOS/IOS XE/IOS XR Software, **metric** is the term that is used for MED; this also applies to the Cisco IOS/IOS XE Software **set** command that is used in route maps, and in all **show** and **debug** commands. In the Cisco IOS XR Software, the term **med** is used in the RPL, but in all **show** and **debug** commands the term **metric** is used.

MED Propagation in a BGP Network



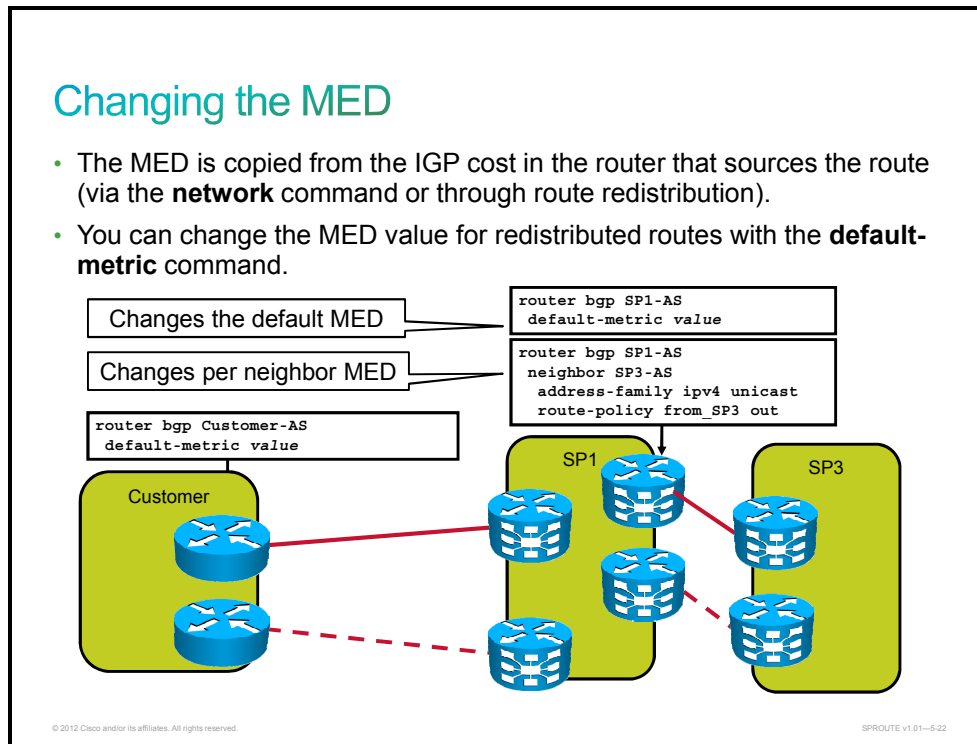
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5.21

The figure shows how the value of the MED attribute is assigned, depending upon the routing information source. An RPL or route map must be configured on a router to manually assign a value to the MED attribute. For the networks that are also present in the BGP table, the router assigns a default value from the metric in the routing table and copies it into the MED attribute. The MED attribute is automatically removed on external sessions if the attribute did not originate in the local AS.

Changing the MED

This topic describes how to change the MED.



The MED is not a mandatory attribute, and there is no MED attribute that is attached to a route by default. The only exception is if the router is originating networks that have an exact match in the routing table (through the Cisco IOS/IOS XE/IOS XR **network** command or through redistribution). If that is the case, the router uses the metric in the routing table as the MED attribute value.

Using the Cisco IOS/IOS XE/IOS XR **default-metric** command in BGP configuration mode causes all redistributed networks to have the specified MED value.

You can use an RPL or route map to set the MED on incoming or outgoing updates. Use the Cisco IOS/IOS XE **set metric** command within route map or Cisco IOS XR **set med** command within RPL configuration mode to set the MED attribute.

Advanced MED Configuration

	Cisco IOS/IOS XE	Cisco IOS XR
By default, the MED is considered only during the selection of routes from the same AS. The MED is also considered for routes coming from a different AS.	<code>bgp always-compare-med</code>	<code>bgp bestpath med always</code>
If the MED is not attached to a BGP route, it is interpreted as value 0, and thus as the best metric. A missing MED is interpreted as infinity (worst).	<code>bgp bestpath med missing-med-worst</code>	<code>bgp bestpath med missing-as-worst</code>
By default, the MED is considered only during the selection of routes from the same AS, which does not include intraconfederation autonomous systems. Allow routers to compare paths learned from confederation peers.	<code>bgp bestpath med confed</code>	<code>bgp bestpath med confed</code>
Changes the BGP route selection procedure to a deterministic but slower one.	<code>bgp deterministic-med</code>	<code>default</code>

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5.23

There are several rules about when and how you should use the MED attribute:

- You should use the MED in the route selection process only if both (all) paths come from the same AS. Use the Cisco IOS/IOS XE **bgp always-compare-med** or Cisco IOS XR **bgp bestpath med always** command to force the router to compare the MED even if the paths come from different autonomous systems. You need to enable this option in the entire AS; otherwise, routing loops can occur.
- According to a BGP standard describing MED, you should regard a missing MED attribute as an infinite value. Cisco IOS/IOS XE/IOS XR Software, however, regards a missing MED attribute as having a value of 0. Use the Cisco IOS/IOS XE **bgp bestpath med missing-med-worst** or Cisco IOS XR **bgp bestpath med missing-as-worst** command when combining equipment from different vendors. An even better solution is to make sure that every update carries a MED attribute.
- You must use the Cisco IOS/IOS XE/IOS XR **bgp bestpath med confed** command when you use the MED within a confederation to influence the route selection process. A router will compare MED values for the routes that originate in the confederation.
- When you enable a deterministic MED comparison, you allow a router to compare MED values before it considers BGP route type (external or internal) and Interior Gateway Protocol (IGP) metric to the next-hop address. The router will compare MED values immediately after the AS path length.

Monitoring the MED

This topic describes how to monitor MED values.

Monitoring the MED

```
RP/0/RSP0/CPU0:PE1#show bgp
< text omitted >
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          Metric LocPrf Weight Path
*> 10.1.1.1/32      0.0.0.0            0         32768 i
*> 10.1.10.1/32     192.168.101.11     0         0 64501 i
*>i10.2.1.1/32     10.2.1.1           0        100  0 i
*>i10.2.10.1/32    10.2.1.1           0        100  0 64502 i
< text omitted >
```

MED is displayed as metric.

```
RP/0/RSP0/CPU0:PE1#show bgp 10.1.10.1/32
< text omitted >
Paths: (1 available, best #1)
  Advertised to peers (in unique update groups):
    10.0.1.1
  Path #1: Received by speaker 0
  Advertised to peers (in unique update groups):
    10.0.1.1
    64501
    192.168.101.11 from 192.168.101.11 (10.1.10.1)
      Origin IGP, metric 0, localpref 100, valid, external, best, group-best
      Received Path ID 0, Local Path ID 1, version 9
```

MED is displayed only for those routes that contain a MED attribute.



Both the original and the modified routes are displayed when inbound soft reconfiguration is configured.

© 2012 Cisco and/or its affiliates. All rights reserved. SPROUTE v1.01-524

All BGP-related **show** and **debug** commands display the value of the MED attribute. If the inbound soft reconfiguration feature is enabled on the router, the original MED attribute that is received by the router is also displayed. To see the original MED, you need to enable soft reconfiguration on the router. The Cisco IOS/IOS XE **show ip bgp neighbor address received-routes** or the Cisco IOS XR **show bgp neighbor address received routes** command displays the original updates before any filters or RPL or route maps have filtered or changed them.

If soft reconfiguration is enabled, the original updates to the MED attribute are available by using the Cisco IOS/IOS XE **show ip bgp prefix** or the Cisco IOS XR **show bgp prefix** command. The original versions are marked with the **received-only** keyword and follow the version that is in the global BGP table.

BGP Communities

This topic discusses how to influence BGP route selection by setting the BGP community attribute on outgoing BGP routes, and describes BGP communities and their use to facilitate proper return path selection. The configuration details of BGP communities and the use of community lists and RPLs or route maps to influence route selection are also discussed.

BGP Communities Overview

- BGP communities are a means of tagging routes to ensure a consistent filtering or route selection policy.
- The community attribute is a transitive optional attribute. Its value is a 32-bit number (range 0 to 4,294,967,200).
- The standards define several filtering-oriented communities:
 - **no-advertise**: Do not advertise routes to any peer.
 - **no-export**: Do not advertise routes to real EBGp peers.
 - **local-as**: Do not advertise routes to any EBGp peers.
 - **internet**: Advertise this route to the Internet community.
- A 32-bit community value is split into two parts:
 - High-order 16 bits contain the AS number of the AS that defines the community meaning.
 - Low-order 16 bits have local significance.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5-25

BGP communities are attributes that are used to group and filter routes. Communities are designed to give the network operator the ability to apply policies to large numbers of routes by using **match** and **set** clauses in the configuration of RPLs or route maps. Community lists are used in this process to identify and filter routes by their common attributes.

A community is an attribute that is used to tag BGP routes. A router can apply it to any BGP route by using an RPL or route map. Other routers can then perform any action, based on the tag (community) that is attached to the route. There can be more than one BGP community that is attached to a single route, but the routers, by default, remove communities in outgoing BGP updates.

The community attribute is a 32-bit transitive optional BGP attribute that is designed to group destinations and apply routing decisions (accept, prefer, redistribute, and so on) according to communities, to allow the easy application of administrative policies. BGP communities provide a mechanism to reduce BGP configuration complexity on a router that is controlling the distribution of routing information.

A set of community values has been predefined. When a router receives a route that has been marked with a predefined community, the router will perform a specific, predefined action that is based on that community setting:

- **no-advertise:** If a router receives an update carrying this community, it will not forward that update to any neighbor.
- **no-export:** If a router receives an update carrying this community, it will not propagate that update to any external neighbors, except intraconfederation external neighbors. This is the most widely used predefined community attribute.
- **local-as:** This community has a similar meaning to **no-export**, but it keeps a route within the local AS (or member AS within the confederation). The route is not sent to external BGP neighbors or to intraconfederation external neighbors.
- **internet:** Advertise this route to the Internet community. All routers belong to it.

Routers that do not support the community attribute will pass the attribute to other neighbors because it is a transitive attribute.

Community attributes are usually used between neighboring autonomous systems. For the BGP communities to be globally unique, a public AS number should be part of the community value. For this reason, you can enter the community value as two 16-bit numbers that are separated by a colon. The first number (high-order 16 bits) should be the AS number of the AS that defines the community value, and the second number should be a value that is assigned a certain meaning (that is, translation of a community value into local preference in the neighboring AS).

Communities can also be used internally, within an AS (to ensure AS-wide routing policy); in this case, the first 16 bits should contain the AS number of the local AS.

Using Communities

- Define administrative policy goals:
 - Solve asymmetrical customer routing problems.
- Design filters and route selection policy to achieve administrative goals:
 - Set local preference of customer routes to 50 for customers using the backup service provider.
- Define communities that signal individual goals:
 - Community 387:17 is used to indicate that the local preference of the route should be lowered to 50.
- Configure route tagging on entry points, or let BGP neighbors tag the routes.
- Configure community distribution.
- Configure route filters and route selection parameters, based on communities.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-28

Designing a BGP solution around BGP communities generally requires the following steps:

- Step 1** Define the administrative policy goals that you need to implement.
- Step 2** Define the filters and route selection policy that will achieve the required goals.
- Step 3** Assign a community value to each goal. This table lists the goals and the community values. All customers of the service provider should know this list so that they can use the BGP communities without having to discuss their use with the service provider.

Goal	Community Value
Set local preference of 50.	387:17
Set local preference of 150.	387:18
Prepend the AS path once when sending the network to external neighbors.	387:21
Prepend the AS path twice when sending the network to external neighbors.	387:22
Prepend the AS path three times when sending the network to external neighbors.	387:23

- Step 4** Apply communities on incoming updates from neighboring autonomous systems or tell the neighbors to set the communities themselves.
- Step 5** Enable community distribution throughout your AS to allow community propagation.
- Step 6** Match communities with route maps and route filters, change BGP attributes, or influence the route selection process, based on the communities that are attached to the BGP routes.

Configuring BGP Communities

This topic describes how to configure BGP Communities.

Configuring BGP Communities

Use these procedures to configure BGP communities:

- Configure route tagging with BGP communities.
- Configure BGP community propagation.
- Define BGP community access lists (community lists) to match BGP communities.
- Configure RPLs or route maps that match on community lists and filter routes, or set other BGP attributes.
- Apply RPLs or route maps to incoming or outgoing updates.

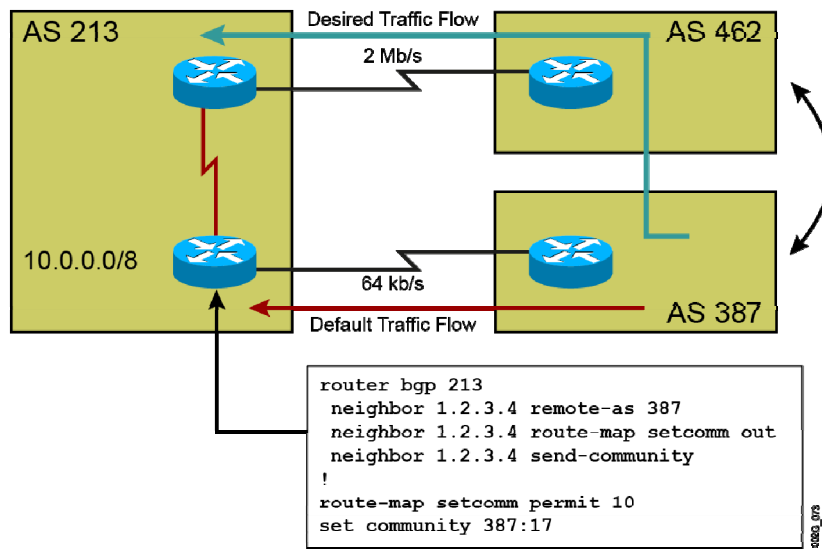
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01-527

Use these procedures to configure BGP communities:

- Set communities, which requires an RPL or route map.
- Enable community propagation per neighbor for all internal neighbors. If communities are sent to external neighbors, you must enable community propagation for external neighbors.
- Create community lists to be used within RPLs or route maps to match on community values.
- Create RPLs or route maps where community lists are used, to match on community values. You can then use RPLs or route maps to filter, based on community values, or to set other parameters or attributes (for example, local preference, MED, or AS path prepending).
- Apply RPLs or route maps to incoming or outgoing updates.

Configuring Route Tagging with BGP Communities

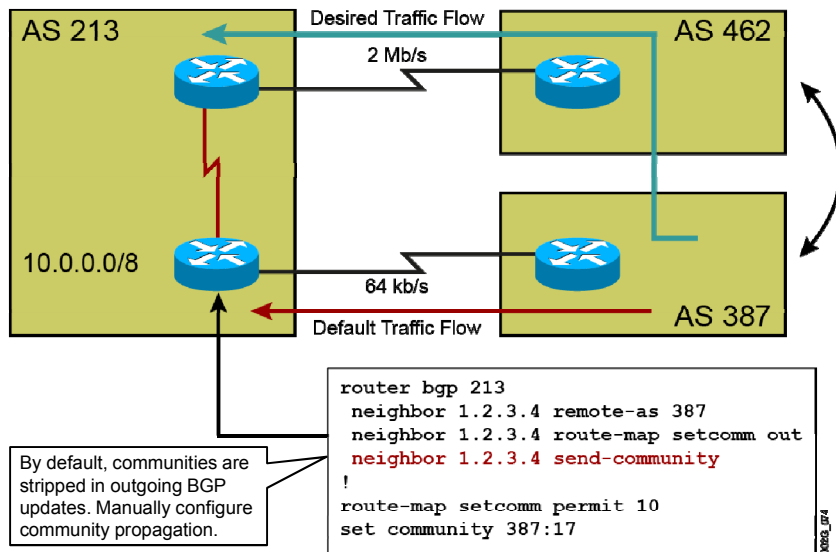


In an RPL or route map configuration mode, you should use the Cisco IOS/IOS XE/IOS XR **set community** command to attach a community attribute (or a set of communities) to a route. If the keyword **additive** is used, the original communities are preserved and the router simply appends the new communities to the route. Omitting the **additive** keyword results in the overwriting of any original community attributes.

You can apply an RPL or route map to incoming or outgoing updates. You can also use it with redistribution from another routing protocol.

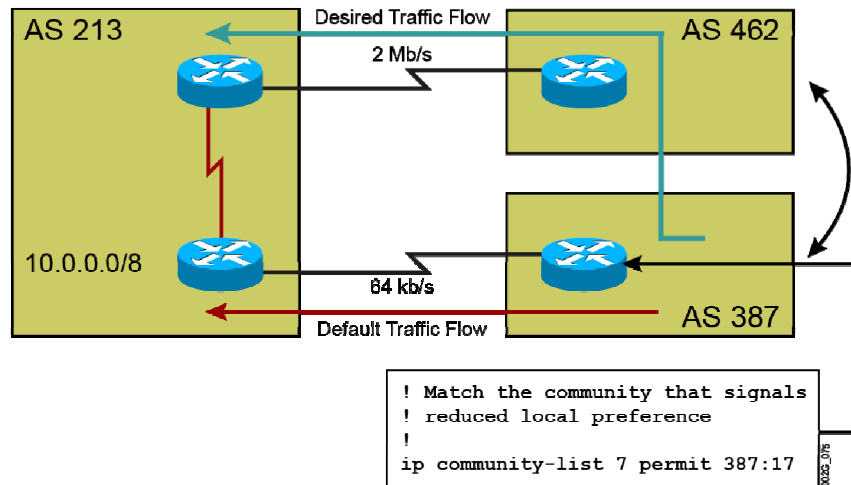
In this example, a border router in AS 213 applies a community value of 387:17 to all networks that are sent to neighboring AS 387.

Configuring Community Propagation: Cisco IOS/IOS XE



A command that is commonly forgotten by network administrators when configuring BGP communities is the Cisco IOS/IOS XE **neighbor *ip-address* send-community** command. This command is needed to propagate community attributes to BGP neighbors. Even if you use an outgoing route map to set communities, by default, the router will strip out any community values that are attached to outgoing BGP updates if you have not configured this command for the specific BGP neighbor.

Defining BGP Community Lists: Cisco IOS/IOS XE



You can use a standard (Cisco IOS/IOS XE) community access list to find community attributes in routing updates. A standard community list is defined by its assigned list number, which can range from 1 to 99 (Cisco IOS/IOS XE).

Community lists are similar to standard IP access lists in the following ways:

- The router evaluates the lines in the community list sequentially.
- If no line matches communities that are attached to a BGP route, the route is implicitly denied.

Standard community lists are different from standard IP access lists in the following ways:

- The keyword **internet** should be used to permit any community value.
- If more values are listed in a single line, they all have to be in an update to produce a match.

An expanded (Cisco IOS/IOS XE) community list is defined by its assigned list number, which can range from 100 to 199. Regular expressions are used to match community attributes. When a router processes a list of communities that are attached to a network update, they are converted into an ordered string of characters.

The access list has been configured to match communities that were previously set by the router in AS 213.

BGP Named Community Lists

This topic describes BGP named community lists.

BGP Named Community Lists

- Naming allows the network operator to assign meaningful names to community lists, and increases the number of community lists that can be configured.
- Named community lists can be configured with regular expressions and with numbered community lists.
- There is no limitation on the number of community attributes that can be configured for a named community list.
- The number of community lists that can be configured by a network operator increases, because there is no limitation on the number of named community lists that can be configured.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTER v1.01-531

The BGP named community lists feature (Cisco IOS/IOS XE/IOS XR) introduces a new type of community list, the named community list. A named community list can be configured with regular expressions and with numbered community lists. The BGP named community lists feature allows the network operator to assign meaningful names to community lists. All rules of numbered communities apply to named community lists except that there is no limitation on the number of community attributes that can be configured for a named community list. Although both standard and expanded community lists (Cisco IOS/IOS XE) have a limitation of 100 community groups that can be configured within each type of list, a named community list does not have this limitation.

BGP Support for Sequenced Entries in Extended Community Lists

This topic describes the use of sequenced entries in extended community lists.

BGP Support for Sequenced Entries in Extended Community Lists

- Allows automatic sequencing of individual entries in BGP extended community lists
- Provides the ability to remove or resequence extended community list entries without deleting the entire existing extended community list
- Configures sequence numbers for extended community list entries
- Resequences the existing sequence numbers for extended community list entries
- Configures an extended community list to use default values

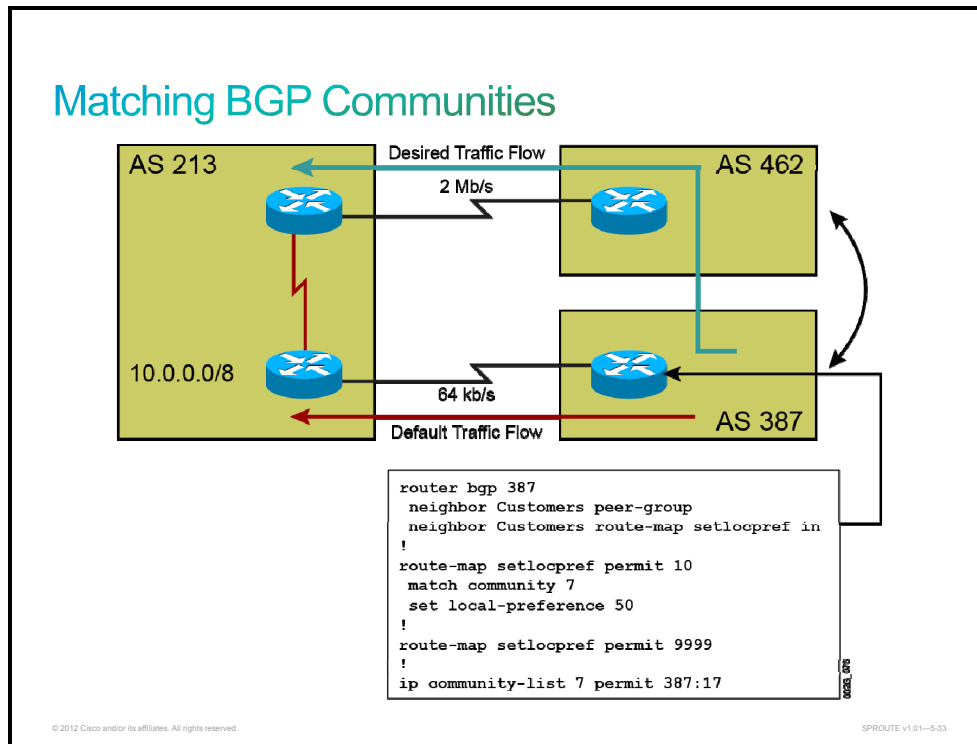
© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--5.32

This feature allows automatic sequencing of individual entries in BGP extended community lists. This feature also provides the ability to remove or resequence extended community list entries without deleting the entire existing extended community list. Both named and numbered extended community lists can be configured in IP extended community list configuration mode.

Matching BGP Communities

This topic describes how to set attributes based on community values.



Network administrators use RPLs or route maps to match networks that carry a subset of communities that are permitted by the community list. Other parameters or attributes can then be set, based on community values. If you use the keyword **exact**, all communities that are attached to a BGP route have to be matched by the community list.

You can use an RPL or route map to filter or modify BGP routing updates. Any BGP-related **set** commands can be used to set BGP parameters and attributes (that is, weight, local preference, and MED).

In the example, all updates that are received from neighboring AS 213 are processed by the route map, which uses a community list to find community 387:17. If the community list matches one of the community attributes, the **set** command is executed and the route is permitted. If the route does not contain the right community, the route is simply permitted by route map statement 9999 without changing anything in the update. The result is that AS 387 prefers other paths to AS 213 because they have a default local preference of 100.

Monitoring BGP Communities

This topic describes how to monitor BGP community values.

Monitoring Communities

- Communities are displayed in a **show bgp *prefix*** printout.
- Communities are not displayed in debugging outputs.
- Routes in the BGP table that are tagged with a set of communities, or routes matching a community list, can be displayed.

```
RP/0/RSP0/CPU0:PE5#show bgp 10.5.10.1/32
< text omitted >
Paths: (1 available, best #1)
  Advertised to update-groups (with more than one peer):
    0.1
  Path #1: Received by speaker 0
  Advertised to update-groups (with more than one peer):
    0.1
64505
192.168.105.51 from 192.168.105.51 (10.5.100.1)
  Origin IGP, metric 0, localpref 100, valid, external, best, group-best
  Received Path ID 0, Local Path ID 1, version 66
  Community: 1:100
```

```
RP/0/RSP0/CPU0:PE5# show bgp route-policy com1
< text omitted >
      Network          Next Hop           Metric LocPrf Weight Path
*> 10.5.10.1/32        192.168.105.51           0             0 64505 i
* 10.5.100.1/32       192.168.105.51           0             0 64505 i
```

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTe v1.01--634

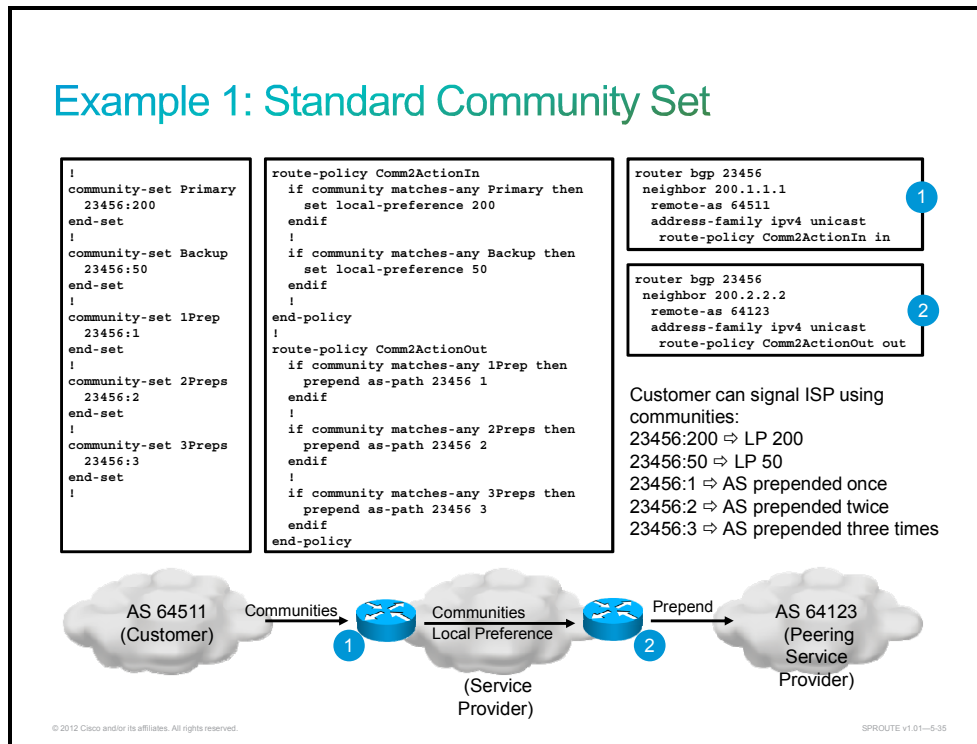
Because a community is an attribute that can appear more than once in a single update, the Cisco IOS/IOS XE **show ip bgp** or Cisco IOS XR **show bgp** command does not show it. You can view communities only if you use the Cisco IOS/IOS XE **show ip bgp *prefix*** or Cisco IOS XR **show bgp *prefix*** command.

If you use the Cisco IOS/IOS XE **show ip bgp community-list** command, all networks that are permitted by the community list are listed. In the Cisco IOS XR you have to attach community list to the RPL as shown in the following example, to display BGP routes with specified communities:

```
community-set 1
  1:100
end-set
!
route-policy com1
  if community matches-any 1 then
    pass
  endif
end-policy
```

BGP Community Examples

This topic shows examples of using BG communities.

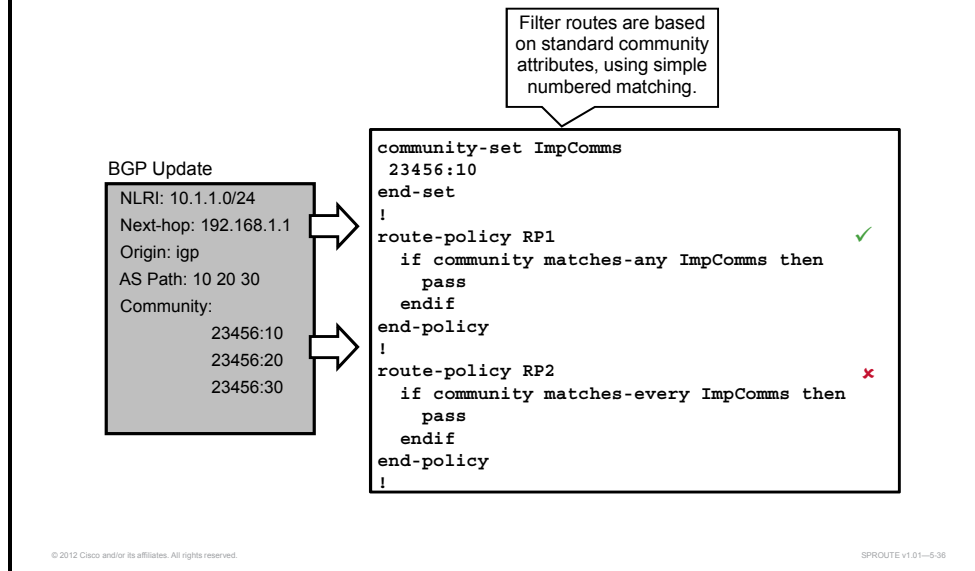


The sample configuration illustrates an AS-wide implementation of a policy:

- Allow customers to signal preference using BGP Communities, which are translated into appropriate local preference values.
- Additionally, the egress routers perform prepending on behalf of customers if they have tagged the routes with appropriate BGP communities.

In this example, five community sets are used to match on BGP community attributes that are coming from external neighbors (i.e., a customer). A customer can, for example, signal that he wants to use this service provider for a backup connection and this customer may choose to attach two BGP communities to achieve the desired goal – 23456:50 and 23456:3. The route policy **Comm2ActionIn**, used in the inbound direction on AS edges, will apply some action, based on the matched communities. The second **if** statement will match the first community and set the local preference to 50, thus making it less desirable than some other paths that will have the default local preference, 100. The **Comm2ActionOut**, used in the outbound direction on AS edges, will match the second community in the third **if** statement and prepend the AS path attribute three times, using its own AS number.

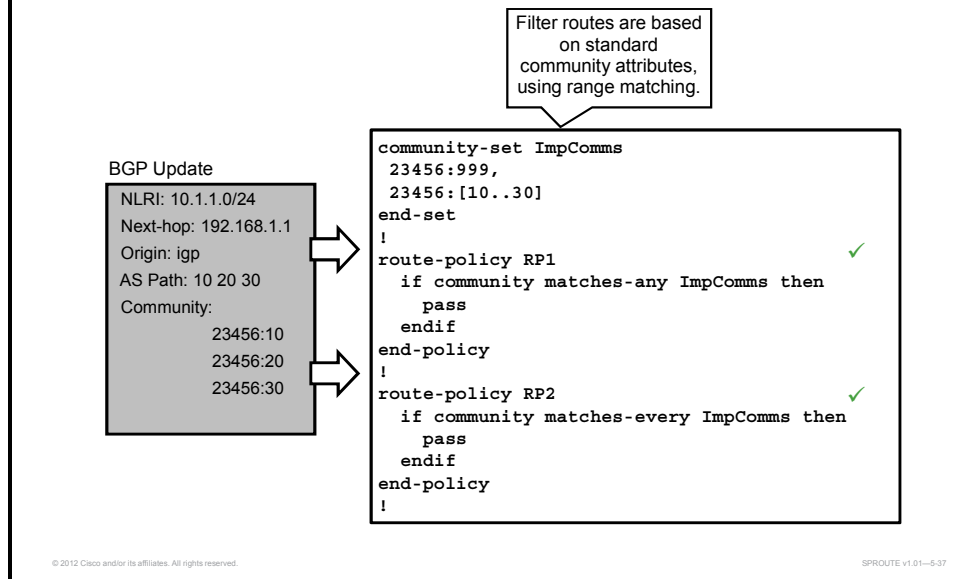
Example 2: Standard Community Set



The second sample configuration illustrates the difference between the **community matches-any** command and the **community matches-every** command:

- RP1: the route with the three community values will match the community set ImpComms, because it contains the 23456:10 community.
- RP2: the route with the three community values will not match the community set ImpComms because it does not match for two community values (23456:20 and 23456:30)

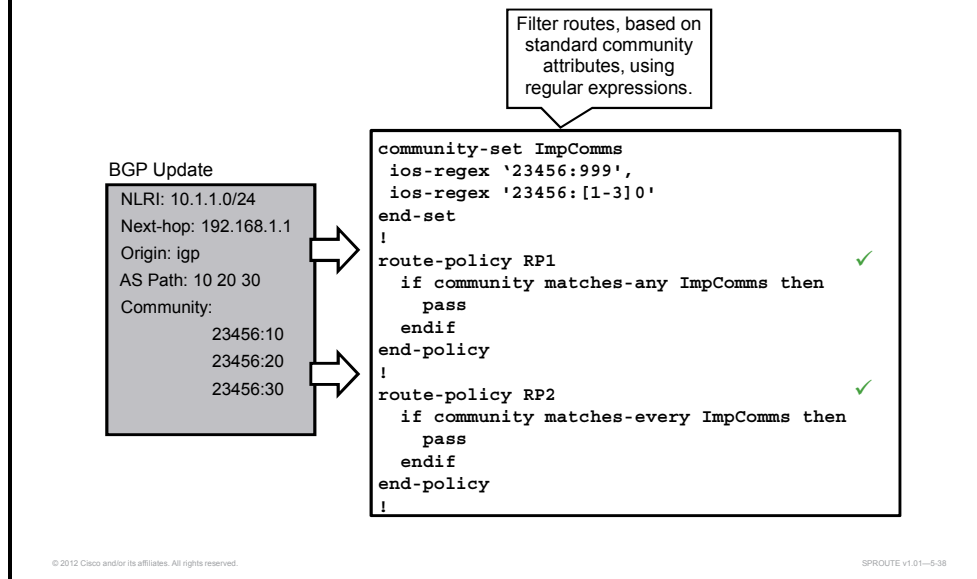
Example 3: Standard Community Set



The third example shows that the route with three community values will match in both policies, because all three community values are matched by the modified community set, ImpComms, which now contains range-based matching for 23456:10-23456:30.

The community set in the example uses number-based matching.

Example 4: Standard Community Set

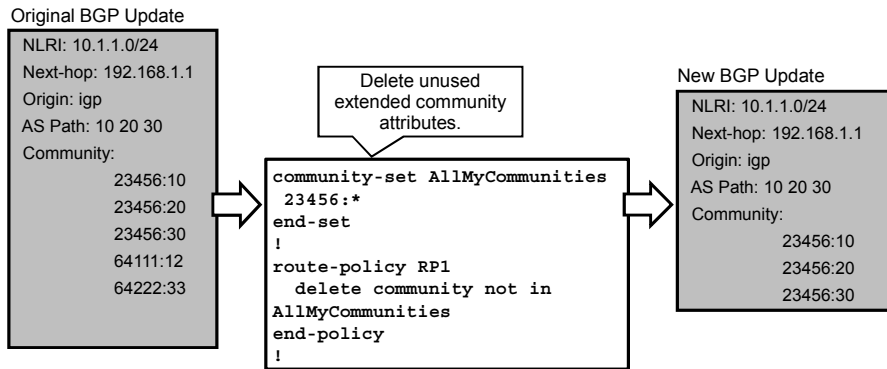


The fourth example shows the same result as the previous one, except that this example uses regular expression-based matching.

The regular expression 23456:[1-3]0 will match 23456:10 and 23456:20 and 23456:30.

Example 5: Standard Community Set

- On incoming updates, delete all communities that have no meaning in your AS 23456.



© 2012 Cisco and/or its affiliates. All rights reserved.

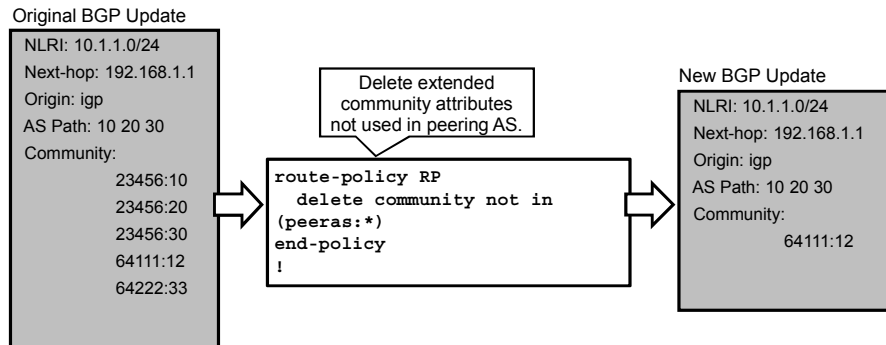
SPROUTE v1.01--5-39

The configuration in the fifth example shows how to delete communities from incoming updates that are outside the desired range (only keep communities that have meaning in the local AS).

This would be a common filter that a service provider would use to strip the updates of any BGP communities that have no meaning in its AS. The numbered matching specifies the AS number 23456 of the service provider, and matches any community value for this AS, using the wildcard ("*"). The route policy then deletes all but those communities that are in this range.

Example 6: Standard Community Set

- On outgoing updates, delete all communities that have no meaning in peering AS 64111.

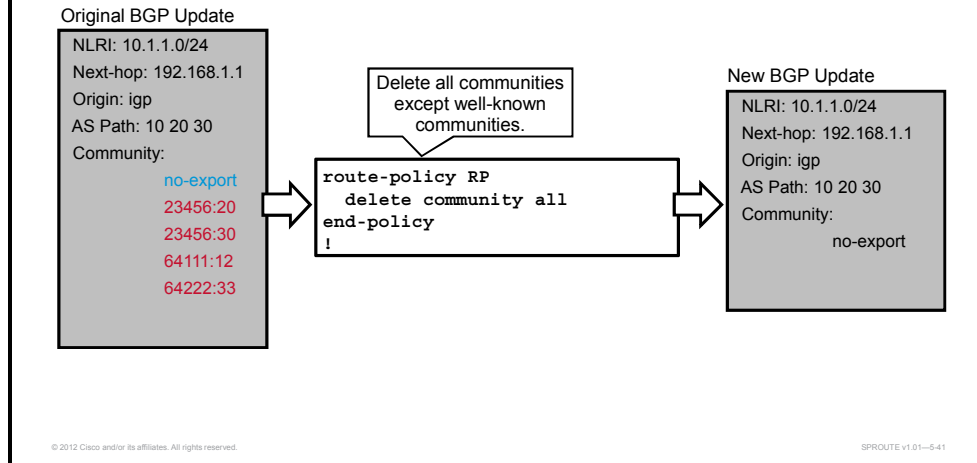


The configuration in example six shows how to delete, from outgoing updates, the communities that are outside the desired range (only keep communities that have meaning in the neighboring AS).

Like the previous example, a service provider can strip out any BGP communities that have no meaning in the neighboring AS. The built-in **peeras** keyword can be used to automatically match on the neighbor AS number, and the wildcard, to match any subsequent value. Instead of using a named community set, the example uses an in-line community set defined within parentheses.

Example 7: Standard Community Set

- Delete all communities except well-known communities (e.g. no-export, no-advertise, local-as)



The configuration in example seven shows how to delete all communities, by using the **all** keyword in place of the community set.

Note This command does not remove the well-known communities (for example, **no-export**) which have a predefined action and must be explicitly deleted if required.

As shown in this example, all communities (except the well-known community **no-export**) have been removed from the update.

Summary

This topic summarizes the key points that were discussed in this lesson.

Summary

- BGP weights can be used to influence the BGP route selection process.
- Weight can be configured on a BGP session and is applied to incoming BGP updates.
- Weight can be changed using route maps or RPL.
- Weight setting is applicable only on incoming routes because a router never propagates the weight attribute to its neighbors.
- Local preference is similar to the weight attribute in that you can use both to influence BGP path selection, but it differs from the BGP weight attribute in that weight is local to the specific router on which it is configured.
- Local preference is set to 100 by default and can be changed using route maps and RPL.
- You can determine local preference of a route by examining the BGP table.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-42

Summary (Cont.)

- You can use AS path prepending to influence incoming path selection.
- AS path prepending is performed on outgoing EBGP updates over the undesired return path, or the path where the traffic load should be reduced.
- You can configure AS path prepending using route maps or RPL.
- When you are monitoring AS path prepending, the router doing the prepending is not the proper point to observe the results of the AS path prepend operation.
- Service providers should take into account possible AS path prepending done by customers when designing AS path filters.
- The MED is a “weak” parameter in the route selection process; it is used only if weight, local preference, AS path, and origin code are equal. By default, the MED is compared only for paths that were received from the same AS.
- You can use the RPL or a route map to set the MED on incoming or outgoing updates.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-43

Summary (Cont.)

- You can determine MED of a route by examining the BGP table.
- BGP communities are a means of tagging routes to ensure consistent filtering or routing policy.
- You can use the BGP community attribute to create an AS-wide routing policy or to provide services to neighboring autonomous systems.
- The BGP named community lists feature allows the network operator to assign meaningful names to community lists.
- BGP support for sequenced entries allows automatic sequencing of individual entries in BGP extended community lists.
- You can use the RPL or route maps to match routes that carry specific BGP communities.
- You can use the **show bgp prefix** command to examine BGP communities that the route is tagged with.
- You can use BGP communities to allow customers to signal preference of a specific path to the SP. SP then performs route manipulation based on received communities.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-44

Module Summary

This topic summarizes the key points that were discussed in this module.

Module Summary

- Tools for routing protocol manipulation include prefix-based filtering for security and AS path-based filtering for flexibility.
- You should use RPLs or route maps to implement modular and complex routing policies.
- Routes are redistributed into a routing protocol by using the **redistribute** command under the routing process. Extreme care has to be taken, when you are configuring multipoint redistribution, to prevent suboptimal routing and routing loops.
- Weight, local preference, AS path prepending, MED, and BGP communities can be used to manipulate BGP path selection and enforce routing policies.

© 2012 Cisco and/or its affiliates. All rights reserved.

SPROUTE v1.01--5-1

This module identified mechanisms that are available, in combination with routing protocols, to filter routing information or to implement desired routing policies in service provider networks. Also covered in this module were IP route redistribution and the control of redistributed routing updates. The use of attributes to influence BGP route selection was discussed and was illustrated with examples.

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

- Q1) Which statement about BGP policies that influence route selection in a multihomed BGP network is true? (Source: Introducing Routing Protocol Tools)
- A) The default BGP route selection does not always result in optimum routing.
 - B) The default BGP route selection always results in optimum routing.
 - C) After the route selection behavior has been set, it cannot be changed.
 - D) The customer receives all routes from both service providers, giving redundancy; therefore, BGP policies are not necessary.
- Q2) Which two potential multihomed network issues can be prevented with IP prefix filters? (Choose two.) (Source: Introducing Routing Protocol Tools)
- A) the propagation of private AS numbers
 - B) the propagation of private addresses that are used in the network
 - C) the propagation of unreachable next-hop addresses
 - D) the propagation of more specific prefixes from an address range
- Q3) Which three goals represent appropriate reasons to apply AS path filters? (Choose three.) (Source: Introducing Routing Protocol Tools)
- A) to ensure that only locally originated routes are announced
 - B) to limit routes that are advertised from IBGP neighbors
 - C) to select a subset of all routes, based on their originating AS
 - D) to limit neighbor route updates to specific AS-originated routes
 - E) to ensure that all destination autonomous systems should be received from a specified neighbor
 - F) to change the weight or local preference attributes for all destination autonomous systems
- Q4) Which AS path is matched by the regular expression “72\$”? (Source: Introducing Routing Protocol Tools)
- A) 213 72 218 31 727
 - B) 27 317 271 50 72
 - C) 315 27 723 19 91
 - D) 72 591 368 20 87
- Q5) What is the difference between the regular expressions “_100_” and “_100\$”? (Source: Introducing Routing Protocol Tools)
- A) The first expression refers to routes that have the substring “100” in their AS paths; the second expression refers only to routes that are directly connected to AS 100.
 - B) The first expression refers to routes that have the substring “100” in their AS paths; the second expression refers only to routes that originated in AS 100.
 - C) The first expression refers to routes that go through AS 100; the second expression refers to routes that originated in AS 100.
 - D) The first expression refers to routes that are directly connected to AS 100; the second expression refers to routes that originated in AS 100.

- Q6) How do you implement a **permit all** statement when you are using route maps?
(Source: Introducing Route Maps and Routing Policy Language)
- A) By default, a route map has an **implicit permit any** statement if no match is found.
 - B) You must configure a route map with a **permit** parameter and no match clause.
 - C) You must configure a route map with a **deny** parameter and a **deny none** clause.
 - D) You must configure a route map with a **permit any** match clause.
- Q7) What happens to incoming BGP updates that do not match any route map match clauses? (Source: Introducing Route Maps and Routing Policy Language)
- A) They are entered into the BGP table.
 - B) They are entered into the BGP table and marked with a weight of 32768.
 - C) They are not accepted by the router or entered into the BGP table.
 - D) They are entered into the BGP table if a matching route exists in the IP routing table.
- Q8) What are two reasons for using route map sequence numbers? (Choose two.) (Source: Introducing Route Maps and Routing Policy Language)
- A) to allow insertion or deletion of route map entries
 - B) to order the execution sequence of route map match clauses
 - C) to provide an ordered execution sequence for the route map
 - D) to map between prefix list statements and route map match clauses
- Q9) Route maps provide better modularity than RPLs. (Source: Introducing Route Maps and Routing Policy Language)
- A) true
 - B) false
- Q10) Redistribution supports BGP, EIGRP, IS-IS, OSPF, RIP, static, and connected routes. (Source: Implementing Route Redistribution)
- A) true
 - B) false
- Q11) What does the **redistribute eigrp 100 subnets metric-type 1** command do when used in router configuration mode under OSPF process 1? (Source: Implementing Route Redistribution)
- A) redistributes from EIGRP AS 100 into OSPF and changes the metric type from type 2 to type 1
 - B) redistributes from EIGRP AS 100 into OSPF process 1 and changes the metric value of each subnet
 - C) redistributes only from EIGRP AS 100 subnets of metric type 1
 - D) redistributes from EIGRP and processes the first 100 subnets of metric type 1
- Q12) Administrative distance is a way of ranking the trustworthiness of routing information. A higher administrative distance means that a route is more trustworthy. (Source: Implementing Route Redistribution)
- A) true
 - B) false

- Q13) Which routing protocol has the lowest administrative distance? (Source: Implementing Route Redistribution)
- A) OSPF
 - B) external EIGRP
 - C) RIP
 - D) internal EIGRP
- Q14) In which two ways can you avoid routing loops that might result from two-way redistribution? (Choose two.) (Source: Implementing Route Redistribution)
- A) Use the **passive-interface** command to suppress routing updates.
 - B) Use RPLs or route maps to redistribute internal routes only.
 - C) Use RPLs or route maps to implement route tagging.
 - D) Use the **no-loop** keyword when redistributing routes.
- Q15) What is the difference between local preference and weight? (Source: Influencing BGP Route Selection)
- A) Local preference has a higher priority in BGP path selection.
 - B) Local preference is used AS-wide while weight is local to a single router.
 - C) Local preference is local only to a specific BGP-speaking router.
 - D) Local preference is used to influence incoming path selection.
- Q16) What is the default weight for routes that are received from a BGP neighbor? (Source: Influencing BGP Route Selection)
- A) 0
 - B) 100
 - C) 32768
 - D) depends on the Cisco IOS release
- Q17) Which two statements about the influence of local preference on BGP route selection are true? (Choose two.) (Source: Influencing BGP Route Selection)
- A) When you set local preference, you can view it on neighboring routers, but you must reset it.
 - B) You can use local preference to ensure AS-wide route selection policy.
 - C) Local preference is used to select routes with unequal weights.
 - D) Local preference is the second-strongest criterion in the route selection process.
- Q18) Which option describes AS path prepending? (Source: Influencing BGP Route Selection)
- A) when a router, sending a BGP update, adds the AS number of the router from which it received the route to the AS path attribute
 - B) when a router, sending a BGP update, adds the AS number of the router to which it is sending the route to the AS path attribute
 - C) when a router, sending a BGP update, adds its AS number to the AS path attribute multiple times
 - D) when a router uses the AS path attribute in route selection
- Q19) What are three BGP attributes that are compared before the MED is compared? (Choose three.) (Source: Influencing BGP Route Selection)
- A) largest weight
 - B) originated routes
 - C) AS path length
 - D) lowest IP address

- Q20) What are two reasons why it is not feasible to use the MED to influence return path selection, when multiple autonomous systems are involved? (Choose two.) (Source: Influencing BGP Route Selection)
- A) The MED attribute is designed to influence outbound path selection only.
 - B) The AS path attribute would be used for path selection regardless of any configured MED value.
 - C) The weight attribute will always be used, because it is first in the BGP route selection process.
 - D) The MED cannot be propagated across several autonomous systems.
- Q21) Does the community attribute have any influence on BGP path selection? (Source: Influencing BGP Route Selection)
- A) No, communities are simply tags that are applied to BGP routes.
 - B) No, communities are nontransitive attributes.
 - C) Yes, BGP paths are selected based on the value in the community tag.
 - D) Yes, the community attribute is part of the BGP route selection process.
- Q22) What is the result of tagging a route with the no-export community? (Source: Influencing BGP Route Selection)
- A) The route will not be advertised within the local AS.
 - B) The upstream AS will not be allowed to export the route.
 - C) The route cannot be exported to another routing protocol.
 - D) The router will not propagate the route to any external neighbors, except to intraconfederation external neighbors.

Module Self-Check Answer Key

Q1)	A
Q2)	B, D
Q3)	A, C, D
Q4)	B
Q5)	C
Q6)	B
Q7)	C
Q8)	A, C
Q9)	B
Q10)	A
Q11)	A
Q12)	B
Q13)	D
Q14)	B, C
Q15)	B
Q16)	A
Q17)	B, D
Q18)	C
Q19)	A, B, C
Q20)	B, D
Q21)	A
Q22)	D

