# Contents

# 1 Configuring Routing Policies

## 1.1 Introduction

### 1.1.1 Overview

Routing policies are a policy set for changing the packet forwarding path or routing information and are often implemented by a filter list and a route map. Routing policies provide services for different routing protocols, and completes the policy-based control of routing information based on a specified policy set. Policy-based routing (PBR) refers to policy-based routing and forwarding. It can forward packets based on the source address, destination address, packet length, and port, and these matching policies can be provided by routing policies.

Routing policies are flexibly and widely applied in the following ways:

- A filter list is used in a routing protocol to filter or modify routing information.
- A route map is used in a routing protocol to filter or modify routing information. The route map can further use a filter list.
- A route map is used in PBR to control packet forwarding or modify packet fields.

Note

In this chapter, the term "router" refers to any network device that supports the routing function. These network devices can be Layer-3 switches, routers, or firewalls.

### 1.1.2 Principles

Filter list: It is a group of lists defined based on a routing attribute and can be used by a routing protocol for route filtering.

Route map: It defines a policy that "if certain conditions are matched, processing actions are performed accordingly."

#### 1. Filter List

A filter list is a group of lists defined based on a routing attribute and is a filtering tool of routing policies. To achieve the effect of route filtering, filter lists must be used in a routing protocol or route map. Five types of filter lists are available currently: access control list (ACL), prefix list, autonomous system (AS) path list, community list, and extended community (extcommunity) list.

- ACL

ACLs include IPv4 and IPv6 ACLs. When defining an ACL, you can specify IPv4/IPv6 addresses and masks to match the destination network segment or next-hop addresses of routing information.

- Prefix list

  Similar to ACLs, prefix lists, including IPv4 prefix lists and IPv6 prefix lists, are used to match destination network segments or next hops of routing information.

  When configuring a prefix list as the filter list, you can define the prefix address, mask length, and mask range. The prefix address and mask length are used to specify the prefix of the routing information. The mask range specifies the mask range of the routing information to be filtered. That is, a prefix list can be defined to directly filter the subnet segment under a specified prefix network segment without calculating the prefix of the subnet segment in advance. Therefore, prefix lists are more flexible than ACLs.

- AS-path list

  AS-path lists are applicable only to Border Gateway Protocol (BGP). They define filtering rules based on the AS path information. When configuring an AS-path list, you can specify an AS path by using a regular expression to match the AS path.

- Community list

Community lists are applicable only to BGP. They define filtering rules based on the community attributes of routes. When configuring a community list, you can specify a community ID or a regular expression to match the community attribute.

- Extcommunity list

Extcommunity lists are applicable only to BGP. They define filtering rules based on the extended community attributes of routes. When configuring an extcommunity list, you can specify an extended community ID or a regular expression to match the extended community attribute.

### 2. Route Map

A policy in a route map is a match and set statement, which indicates that if certain conditions are matched, some processing actions will be performed accordingly.

- Main application scenarios of route maps
  - Route filtering: A filter list is used in a routing protocol to filter the routing information sent or received by the protocol.
  - Route redistribution: A route map is used in a routing protocol to filter or modify routing information and redistribute Routing Information Protocol (RIP) routes to Open Shortest Path First (OSPF). Only RIP routes with the hop count of 4 can be redistributed.
  - PBR: A route map is used in PBR to control packet forwarding or modify packet fields and specify optimum outbound interfaces for packets from different subnets.
- Policy execution

  A route map contains multiple policies. Each policy has a corresponding sequence number. A smaller sequence number means a higher priority. You can execute policies by sequence number. Once the matching

condition of a policy is met, the processing action of this policy is performed and the route map exits. If no matching condition of any policy is met, no processing action is performed.

- Working modes of policies

  Policies support two working modes:

  - **permit**: When the matching condition of a policy is met, the processing action for this policy is performed and the system exits the route map.
  - **deny**: When the matching condition of a policy is met, the processing action for this policy is not performed and the system exits the route map.

- Matching conditions of policies

  The matching condition of a policy may contain zero, one, or more match rules.

  - If the matching condition contain zero match rules, all packets match the policy.
  - If the matching condition contains one or more match rules, the matching condition is met only when all the rules are matched.

- Processing action for a policy

  The processing action of a policy may contain zero, one, or more set rules.

  - If the processing action contains zero set rules, no processing action is performed and the system directly exits the route map.
  - If the processing action contains one or more set rules, all processing actions are performed and then the system exits the route map.

## 1.2 Configuration Task Summary

The configuration of a routing policy includes the following tasks:

(1) (Optional) Configuring a Filter List

- Defining AS Path Filtering Rules
- Defining a Community List
- Defining an Extcommunity List
- Creating a Prefix List
- Adding a Text Description for a Prefix List
- Displaying Sequence Numbers in a Prefix List
- Creating an IPv6 Prefix List
- Adding a Text Description for an IPv6 Prefix List
- Displaying Sequence Numbers in an IPv6 Prefix List

(2) Configuring a Route Map

a Creating a Policy

b Configuring the Matching Conditions of a Rule

c Configuring the Processing Actions of a Policy

# 1.3 Configuring a Filter List

## 1.3.1 Overview

This section describes how to define a set of route filtering rules for a routing protocol.

## 1.3.2 Restrictions and Guidelines

A configured filter list can take effect only after it is associated with a routing protocol.

## 1.3.3 Configuration Tasks

The filter list configuration includes the following tasks. All the configuration tasks below are optional. Select the configuration tasks as required.

- Defining AS Path Filtering Rules
- Defining a Community List
- Defining an Extcommunity List
- Creating a Prefix List
- Adding a Text Description for a Prefix List
- Displaying Sequence Numbers in a Prefix List
- Creating an IPv6 Prefix List
- Adding a Text Description for an IPv6 Prefix List
- Displaying Sequence Numbers in an IPv6 Prefix List

## 1.3.4 Defining AS Path Filtering Rules

### 1. Overview

An AS path filter list defines AS path filtering rules by using regular expressions. AS-path is a BGP attribute. Therefore, defined AS path filtering rules are applied to a BGP network.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Define AS path filtering rules.

**ip as-path access-list** *path-list-num* { **permit** | **deny** } *regular-expression*

By default, no AS path filtering rule is configured.

### 1.3.5 Defining a Community List

#### 1. Overview

A community is a set of routes with the same characteristics. Community attributes are BGP attributes and can help reduce the difficulties in BGP route management and maintenance. Similar to an ACL, you can define multiple entries in a community list, and entries are matched based on their sequence numbers. As long as routes match one entry, the routes are permitted by the community list and no further matching is performed.

#### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Define a standard community list.

**ip community-list** { *community-list-number* | **standard** *community-list-name* } { **permit** | **deny** } [ { *community-list-number* | **internet** | **local-AS** | **no-advertise** | **no-export** | **gshut** } ]

By default, no standard community list is defined.

(4) Define an expanded community list.

**ip community-list** { *community-list-number* | **expanded** *community-list-name* } { **permit** | **deny** } [ *regular-expression* ]

By default, no expanded community list is defined.

### 1.3.6 Defining an Extcommunity List

#### 1. Overview

An extended community is a set of routes with the same characteristics. Extended community attributes are BGP attributes and can help reduce the difficulties in BGP route management and maintenance. Similar to an ACL, you can define multiple entries in an extcommunity list, and entries are matched based on their sequence numbers. As long as routes match one entry, the routes are permitted by the extcommunity list and no further matching is performed.

#### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Define a standard extcommunity list.

**ip extcommunity-list** { *standard-list* | **standard** *list-name* } { **permit** | **deny** } [ **rt** *rt-value* | **soo** *soo-value* ]

By default, no standard extcommunity list is defined.

(4) Define an expanded extcommunity list.

**ip extcommunity-list** { *expanded-list* | **expanded** *list-name* } { **permit** | **deny** } [ *regular-expression* ]

By default, no expanded extcommunity list is defined.

## 1.3.7 Creating a Prefix List

### 1. Overview

Prefix lists can be directly used by protocols, or used together with a route map as a matching condition of the route map. Similar to an ACL, you can define multiple entries in a prefix list, and entries are matched based on their sequence numbers. As long as routes match one entry, the routes are permitted by the prefix list and no further matching is performed. If routes do not match any entry, it is considered that the routes are denied by the prefix list.

### 2. Restrictions and Guidelines

This command is not a routing policy command, and will not be applied to any routing protocol.

### 3. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Create a prefix list.

**ip prefix-list** *prefix-list-name* [ **seq** *seq-number* ] { **deny** | **permit** } *ipv4-prefix* [ **ge** *minimum-prefix-length* ] [ **le** *maximum-prefix-length* ]

By default, no prefix list is configured.

## 1.3.8 Adding a Text Description for a Prefix List

### 1. Overview

You can add a text description for a prefix list, for example, to describe the purpose of the prefix list. Adding text descriptions for prefix lists can facilitate maintenance and tracking.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Add a text description for a prefix list.

**ip prefix-list** *prefix-list-name* **description** *description-text*

By default, no text description is configured for a prefix list.

## 1.3.9 Displaying Sequence Numbers in a Prefix List

### 1. Overview

When you run the **show ip prefix-list** command, the sequence numbers of entries in a prefix list are not displayed by default. If these sequence numbers are displayed, you can conveniently add or remove an entry to or from the prefix list.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Enable the function of displaying sequence numbers in a prefix list.

**ip prefix-list sequence-number**

By default, the function of displaying sequence numbers in a prefix list is disabled.

## 1.3.10 Creating an IPv6 Prefix List

### 1. Overview

Prefix lists can be directly used by protocols, or used together with a route map as a matching condition of the route map. Similar to an ACL, you can define multiple entries in a prefix list and entries are matched based on their sequence numbers. As long as routes match one entry, the routes are permitted by the prefix list and no further matching is performed. If routes do not match any entry, it is considered that the routes are denied by the prefix list.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Create an IPv6 prefix list.

**ipv6 prefix-list** *prefix-list-name* [ **seq** *seq-number* ] { **deny** | **permit** } *ipv6-prefix* [ **ge** *minimum-prefix-length* ] [ **le** *maximum-prefix-length* ]

By default, no IPv6 prefix list is configured.

## 1.3.11 Adding a Text Description for an IPv6 Prefix List

### 1. Overview

You can add a text description for a prefix list, for example, to describe the purpose of the prefix list. Adding text descriptions for prefix lists can facilitate maintenance and tracking.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Add a text description for an IPv6 prefix list.

**ipv6 prefix-list** *prefix-list-name* **description** *description-text*

By default, no text description is configured for an IPv6 prefix list.

## 1.3.12 Displaying Sequence Numbers in an IPv6 Prefix List

### 1. Overview

When you run the **show ipv6 prefix-list** command, the sequence numbers of entries in a prefix list are not displayed by default. If these sequence numbers are displayed, you can conveniently add or remove an entry to or from the prefix list.

### 2. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Enable the function of displaying sequence numbers in an IPv6 prefix list.

**ipv6 prefix-list sequence-number**

By default, the function of displaying sequence numbers in an IPv6 prefix list is displayed.

# 1.4 Configuring a Route Map

## 1.4.1 Overview

This section describes how to define a set of routing policies to be used by routing protocols or PBR.

## 1.4.2 Configuration Tasks

The route map configuration includes the following tasks:

(1) Creating a Policy

(2) Configuring the Matching Conditions of a Rule

(3) Configuring the Processing Actions of a Policy

## 1.4.3 Creating a Policy

### 1. Overview

You can create a route map to control route attributes.

### 2. Restrictions and Guidelines

- If the route map is unavailable, this command will create a route map and add a policy to the route map.
- If the route map is available, this command will add a policy to the route map.

### 3. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Create a route map and enter the route map configuration mode.

**route-map** *route-map-name* [ { **permit** | **deny** } *sequence* ]

## 1.4.4 Configuring the Matching Conditions of a Rule

### 1. Overview

In a route map, you can configure match rules based on the route attributes. If multiple match rules are configured, all the rules must be matched.

### 2. Restrictions and Guidelines

- If you use an ACL in a **match** command to define packet matching conditions, you must configure this ACL.
- The following table lists the **match** commands that cannot be configured at the same time:

**Table 1-1 Match Commands That Cannot Be Configured at the Same Time**

| match Command | Exclusive match Command |
|---|---|
| **match ip address** | **match ip prefix-list** |
| **match ipv6 address** | **match ipv6 prefix-list** |
| **match ip next-hop** | **match ip next-hop prefix-list** |
| **match ipv6 next-hop** | **match ipv6 next-hop prefix-list** |
| **match ip route-source** | **match ip route-source prefix-list** |
| **match ipv6 route-source** | **match ipv6 route-source prefix-list** |

### 3. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Enter the route map configuration mode.

**route-map** *route-map-name* [ { **permit** | **deny** } *sequence* ]

(4) Configure a rule to match the next-hop outbound interface.

**match interface** { *interface-type interface-number* }&<1-4>

By default, traffic is not matched with an interface that is used as the next-hop outbound interface.

Use an IPv4 filter list in the route map to match attributes. The following configurations are optional. You can configure multiple match rules, all of which must be matched.

- Configure an AS-path list for packet matching.

  **match as-path** *as-path-acl-list-number*&<1-10>

  By default, no AS-path list is configured for packet matching.

- Configure a community list as a match rule.

  **match community** { *community-list-number* | *community-list-name* } [ **exact-match** | *community-list-number* | *community-list-name* ]&<1-6>

  By default, no community list is configured for packet matching.

- Configure an extcommunity list for packet matching.

  **match extcommunity** { *standard-list-number* | *standard-list-name* | *expanded-list-num* | *expanded-list-name* }&<1-6>

By default, no extcommunity list is configured for packet matching.

- Configure a rule to match the target network routes that are permitted in the ACL or prefix list.

  **match ip address** { { *acl-number* | *acl-name* } &<1-6> | **prefix-list** *prefix-list-name*&<1-6> }

  By default, no ACL or prefix list used for packet matching is configured.

- Configure a rule to match the target network routes whose next-hop IPv4 addresses meet rules in the ACL or prefix list.

  **match ip next-hop** { { *acl-number* | *acl-name* } &<1-6> | **prefix-list** *prefix-list-name*&<1-6> }

  By default, no ACL or prefix list used for next-hop IP address matching is configured.

- Configure a rule to match the target network routes of a specified L3 authentication traffic diversion domain type.

  **match ip policy** { *acl-number* | *acl-name* }&<1-6> **class** *class-id*

- Configure a rule to match the target network routes whose source IP addresses meet rules in the ACL or prefix list.

  **match ip route-source** { { *acl-number* | *acl-name* } &<1-6> | **prefix-list** *prefix-list-name*&<1-6> }

  By default, no ACL or prefix list is configured to match the source IP addresses of routes.

(5) Use an IPv6 filter list in the route map to match attributes.

- Configure a rule to match the target IPv6 network routes that are permitted in the ACL or prefix list.

  **match ipv6 address** { *acl-name* | **prefix-list** *prefix-list-name* }

  By default, no IPv6 ACL or IPv6 prefix list is configured for packet matching.

- Configure a rule to match the target network routes whose next-hop IPv6 addresses meet rules in the ACL or prefix list.

  **match ipv6 address** { *acl-name* | **prefix-list** *prefix-list-name* }

  By default, no IPv6 ACL or IPv6 prefix list is configured for next-hop IP address matching.

- Configure a rule to match the target network routes whose source IPv6 addresses meet rules in the ACL or prefix list.

  **match ipv6 route-source** { *acl-name* | **prefix-list** *prefix-list-name* }

  By default, no IPv6 ACL or IPv6 prefix list is configured to match the source IP addresses of routes.

(6) Configure a rule to match the routes with a fixed metric.

**match metric** *metric*

By default, the metric values of routes are not matched.

(7) Configure a rule to match the route type.

- Configure a rule to match the BGP route source.

  **match origin** { **egp** | **igp** | **incomplete** }

  By default, no rule is configured to match any source routes.

- Configure a rule to match the route type.

  **match route-type** { **static** | **connect** | **rip** | **local** | **internal** | **external** [ **type-1** | **type-2** ] | **level-1** | **level-2** | **evpn-type-1** | **evpn-type-2** | **evpn-type-3** | **evpn-type-4** | **evpn-type-5** }

  By default, no rule is configured to match any route type.

(8) Configure a rule to match the routes with a specified tag.

**match tag** *tag*&<1-4>

### 1.4.5 Configuring the Processing Actions of a Policy

#### 1. Overview

Processing actions are configured for traffic that matches the rules. The device provides abundant **set** commands for setting flexible processing actions.

#### 2. Restrictions and Guidelines

- If no set rule is configured, no processing action will be performed.
- If multiple set rules are configured, all set rules must be executed. If the set rules have different priorities, only the set rule with the highest priority takes effect.
- The following table lists the **set** commands that cannot be configured at the same time.

**Table 1-1 set Commands That Cannot Be Configured at the Same Time**

| set Command | Exclusive set Command |
|---|---|
| **set ip next-hop** | **set ip next-hop verify-availability** |
| **set ip dscp** | **set ip tos** |
| **set ip dscp** | **set ip precedence** |

#### 3. Procedure

(1) Enter the privileged EXEC mode.

**enable**

(2) Enter the global configuration mode.

**configure terminal**

(3) Enter the route map configuration mode.

**route-map** *route-map-name* [ { **permit** | **deny** } *sequence* ]

(4) Configure a rule to adjust the BGP attributes.

- Configure the AS number for BGP.

**set aggregator as** *as-number ipv4-address*

By default, no AS number of the aggregator is specified.

- Configure the Accumulated Interior Gateway Protocol (AIGP) metric attribute for Exterior Gateway Protocol (EGP).

  **set aigp-metric** { *metric-number* | **igp-metric** }

  By default, no AIGP metric attribute is configured for routes.

- Configure a rule to add the specified AS_PATH values.

  **set as-path prepend** *as-number*&<1-10>

  By default, no rule is configured to add AS_PATH values for routes.

- Configure a rule to replace the AS_PATH attributes of routes.

  **set as-path replace** *as-number*&<1-10>

  By default, the AS_PATH values of matched routes will not be replaced.

- Configure the atomic-aggregate attribute for routes.

  **set atomic-aggregate**

  By default, no atomic-aggregate attribute is configured for routes.

- Configure a rule to delete all the community values from BGP routes matching the rules according to the community list.

  **set comm-list** { *community-list-number* | *community-list-name* } **delete**

  By default, no rule is configured to delete all the community values from routes matching the rules according to the community list.

- Configure the community value for BGP routes.

  **set community** { **none** | { *community-number* | **internet** | **local-AS** | **no-advertise** | **no-export** }&<1-32> [ **additive** ] }

  By default, no community value is specified for a route.

- Configure the flapping parameters for BGP routes.

  **set dampening** *half-life reuse suppress max-suppress-time*

  By default, when the half-life period is 15 minutes, and the penalty value of a route is lower than 750, route suppression is canceled. When the penalty value of a route exceeds 2000, the route is suppressed. A route can be suppressed for a maximum of 60 minutes.

- Configure a rule to delete the extended community attribute of BGP routes.

  **set extcomm-list** { *extcommunity-list-number* | *extcommunity-list-name* } **delete**

- Configure the extended community attribute of BGP routes.

  **set extcommunity** { **rt** *extend-community-value* | **soo** *extend-community-value* }

(5) Configure the backup outbound interface and backup next hop for fast reroute (FRR).

- Configure the backup outbound interface and backup next hop for FRR.

  **set fast-reroute backup-interface** *interface-type interface-number* **backup-nexthop** *ipv4-address*

  By default, no backup outbound interface and backup next hop of FRR are specified for matched routes.

(6) Configure the next hop.

- Configure the default next-hop IPv4 address for routes.

  **set ip default next-hop** { *ipv4-address* [ *weight* ] }&<1-32>

  By default, no default next-hop IPv4 address is specified for matched routes.

- Configure the next-hop IPv4 address for IPv4 packets.

  **set ip next-hop** { *ipv4-address* [ *weight* ] }&<1-32>

  By default, no next-hop IPv4 address is specified for matched packets.

- Configure the next-hop IP address for routes.

  **set next-hop** *ipv4-address*

  By default, no next-hop IP address is specified for matched routes.

- Configure the default next-hop IPv6 address for an IPv6 packet.

  **set ipv6 default next-hop** { *ipv6-address* [ *weight* ] }&<1-32>

  By default, no default next-hop IPv6 address is specified for matched IPv6 routes.

- Configure the next-hop IPv6 address for an IPv6 packet.

  **set ipv6 next-hop** { *ipv6-address* [ *weight* ] }&<1-32>

- Configure the recursive next-hop IP address for packets.

  **set ip next-hop recursive** *ipv4-address*

  By default, no recursive next-hop IPv4 address is specified for matched packets.

- Set the next hops of routes to the device itself.

  **set ip next-hop self**

  By default, the next hop is not set to the device itself for matched packets.

- Set the next hop of a route to keep unchanged.

  **set ip next-hop unchanged**

  By default, the next hop is not set to keep unchanged for matched packets.

- Set the next hop of a route to the device itself.

  **set next-hop self**

  By default, the next hop is not set to the device itself for matched routes.

- Set the next hop of the specified route to keep unchanged.

  **set next-hop unchanged**

  By default, the next hop is not set to keep unchanged for matched routes.

- Configure the recursive next-hop IPv6 address.

  **set ipv6 next-hop recursive** *ipv6-address*

  By default, no recursive next-hop IPv6 address is specified for matched IPv6 packets.

- Set the next hop of the specified IPv6 route to the device itself.

  **set ipv6 next-hop self**

  By default, the next hop address is not set to the device itself for matched IPv6 packets.

- Set the next hop of the specified IPv6 route to keep unchanged.

  **set ipv6 next-hop unchanged**

  By default, the next hop is not set to keep unchanged for matched IPv6 packets.

- Set the L3 VPN next hop to the local virtual routing and forwarding (VRF) instance.

  **set l3vpn nexthop local-vrf**

By default, the L3 VPN next hop is not set to the local VRF instance for routes matching the rules.

- Verify availability of the next-hop IPv4 address.

  **set ip next-hop verify-availability** *ipv4-address* [ *weight* ] { **track** *track-obj-number* | **bfd** *interface-type interface-number gateway* }

  By default, availability of the next-hop IPv4 address is not verified for matched packets.

- Verify availability of the next-hop IPv6 address.

  **set ipv6 next-hop verify-availability** *ipv6-address* [ *weight* ] **bfd** *interface-type interface-number gateway*

  By default, availability of the next-hop IPv6 address is not verified for matched packets.

(7) Configure the flow control parameters.

- Configure the IPv4 header priority.

  **set ip precedence** { *precedence* | **critical** | **flash** | **flash-override** | **immediate** | **internet** | **network** | **priority** | **routine** }

  By default, no IPv4 header priority is configured for matched packets.

- Configure the IPv6 header priority.

  **set ipv6 precedence** { *precedence-number* | **critical** | **flash** | **flash-override** | **immediate** | **internet** | **network** | **priority** | **routine** }

  By default, no IPv6 header priority is configured for matched packets.

- Configure the type of service (ToS) for the IPv4 header of a packet.

  **set ip tos** { *tos-number* | **max-reliability** | **max-throughput** | **min-delay** | **min-monetary-cost** | **normal** }

  By default, no ToS is configured for the IP header of a matched packet.

- Configure the LOCAL_PREFERENCE parameter.

  **set local-preference** *precedence-number*

By default, no LOCAL_PREFERENCE value is configured for matched routes.

- Configure the quality of service (QoS) ID for a route.

  **set qos-id** *qos-id*

  By default, no QoS ID is specified for matched routes.

- Configure the Differentiated Services Code Point (DSCP) in an IPv4 packet.

  **set ip dscp** *dscp_value*

  By default, no DSCP is configured in an IPv4 packet for matched routes.

(8) Configure redistribution-related information.

- Configure the type of the destination area to which redistributed routes are to be advertised.

  **set level** { **level-1** | **level-1-2** | **level-2** | **stub-area** | **backbone** }

  By default, the type of the destination area is not specified for matched routes.

- Configure the redistributed route tag.

  **set tag** *tag*

  By default, no route tag is configured for matched routes.

(9) Configure the weight, metric, and metric type.

- Configure the metric for routes.

  **set metric** { **+** *metric-value* | **-** *metric-value* | *metric-value* }

  By default, the metric of a matched route is not modified.

- Configure the metric type for routes.

  **set metric-type** { **external** | **internal** | **type-1** | **type-2** }

- Configure the weight for BGP routes.

  **set weight** *weight-number*

  By default, no weight is configured for matched routes.

(10) Configure the route source information.

- Configure the route source attribute.

  **set origin** { **egp** | **igp** | **incomplete** }

  By default, no route source is specified for matched routes.

- Configure the route originator address.

  **set originator-id** *ipv4-address*

  By default, no route originator address is configured for matched routes.

(11) Configure the route management distance.

**set distance** *distance-number*

By default, no management distance is modified for matched routes.

## 1.5 Monitoring

Run the **show** command to check the running status of a configured function to verify the configuration effect.

**Table 1-1 Routing Policy Monitoring**

| Command | Purpose |
|---|---|
| **show route-map** [ *route-map-name* ] | Displays the configurations of a route map. |
| **show access-lists** [ *acl-id* \| *name* ] | Displays the configurations of an ACL. |
| **show ip prefix-list** [ *prefix-name* ] | Displays the configurations of an IPv4 prefix list. |
| **show ip protocols** [ **vrf** *vrf-name* ] [ **bgp** \| **isis** \| **ospf** \| **rip** ] | Displays the status of the IPv4 routing protocols that are currently running. |
| **show ipv6 prefix-list** [ *prefix-name* ] | Displays the configurations of an IPv6 prefix list. |
| **show ip as-path-access-list** [ *as-path-access-list-num* ] | Displays the configurations of an AS-path list. |
| **show ip community-list** [ *community-list-number* \| *community-list-name* ] | Displays the configurations of a community list. |
| **show ip extcommunity-list** [ *extcommunity-list-num* \| *extcommunity-list-name* ] | Displays the configurations of an extcommunity list. |

## 1.6 Configuration Examples

### 1.6.1 Applying Rules Configured in a Route Map During Route Redistribution

#### 1. Requirements

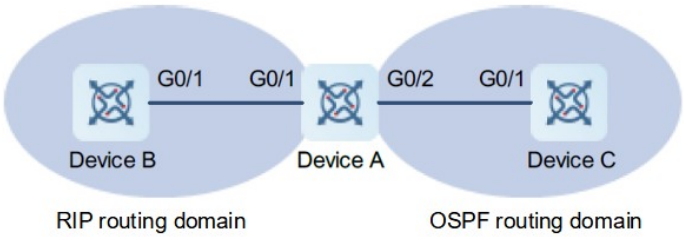
The RIP routing domain needs to exchange routes with the OSPF routing domain to realize the network-wide interworking.

When RIP routes are redistributed to OSPF, the routes with too many hops are considered unreliable. Therefore, it is required that only RIP routes with the hop count of 4 be redistributed. In the OSPF routing domain, if a route is a Type-1 external route, the tag value of the route is set to 40.

When OSPF routes are redistributed to RIP, only OSPF routes with the tag value of 10 are redistributed. In the RIP routing domain, the initial metric of the routes redistributed to RIP is set to 10.

## 2. Topology

**Figure 1-1 Topology of Route Map Application During Route Redistribution**




## 3. Notes

- Configure the route map **redrip** as follows: Configure a rule to match routes with the hop count of 4, set the initial metric of the routes to **40**, set the route type to the external route **type-1**, and set the tag of the routes to **40**.
- Configure the route map **redospf** as follows: Configure a rule to match routes with the tag of **10** and set the initial metric of the routes to **10**.
- Apply the route map **redrip** when RIP routes are redistributed to OSPF.
- Apply the route map **redospf** when OSPF routes are redistributed to RIP.

## 4. Procedure

(1) Configure IP addresses of the interfaces, and complete the basic OSPF and RIP configuration. (Omitted)

(2) Configure a route redistribution policy for OSPF. In the policy, configure a rule to match routes with the hop count of 4, set the route type to **Type-1** and the tag to **40**.

```
Device A> enable
Device A# configure terminal
Device A(config)# route-map redrip permit 10
Device A(config-route-map)# match metric 4
Device A(config-route-map)# set metric-type type-1
Device A(config-route-map)# set tag 40
Device A(config-route-map)# exit
```

(3) Configure a route redistribution policy for RIP. In the policy, configure a rule to match routes with the route tag of **10** and set the metric to **10**.

```
Device A(config)# route-map redospf permit 10
Device A(config-route-map)# match tag 10
Device A(config-route-map)# set metric 10
Device A(config-route-map)# exit
```

(4) Configure redistribution of RIP routes to OSPF, and use the route map **redrip** for routing control.

```
Device A(config)# router ospf 1
Device A(config-router)# redistribute rip subnets route-map redrip
Device A(config-router)# exit
```

(5) Configure the redistribution of OSPF routes to RIP, and use the route map **redospf** for routing control.

```
Device A(config)# router rip
Device A(config-router)# redistribute ospf 1 route-map redospf
Device A(config-router)# exit
```

## 5. Verification

Check the route map configurations on device A to verify the policy rules.

```
Device A# show route-map
route-map redrip, permit, sequence 10
  Match clauses:
    metric 4
  Set clauses:
    metric 40
    metric-type type-1
    tag 40
route-map redospf, permit, sequence 10
  Match clauses:
    tag 10
  Set clauses:
metric 10
```

Check the OSPF routing information library on device A to verify that the routes matching the policy rules are redistributed.

```
Device A# show ip ospf database external

            OSPF Router with ID (192.100.1.9) (Process ID 1)

                AS External Link States

  LS age: 5
  Options: 0x2 (-|-|-|-|-|-|E|-)
  LS Type: AS-external-LSA
  Link State ID: 192.168.199.0 (External Network Number)
  Advertising Router: 192.100.1.9
  LS Seq Number: 80000001
  Checksum: 0x554d
  Length: 36
  Network Mask: /24
        Metric Type: 1
```

```
        TOS: 0
        Metric: 4
        Forward Address: 0.0.0.0
        External Route Tag: 40
```

**6. Configuration Files**

Device A configuration file

```
!
route-map redrip permit 10
 match metric 4
 set metric-type type-1
 set tag 40
!
route-map redospf permit 10
 match tag 10
 set metric 10
!
router ospf 1
redistribute rip route-map redrip subnets
!
router rip
redistribute ospf 1 route-map redospf
!
```

## 1.6.2 Applying a Route Map in PBR

**1. Requirements**

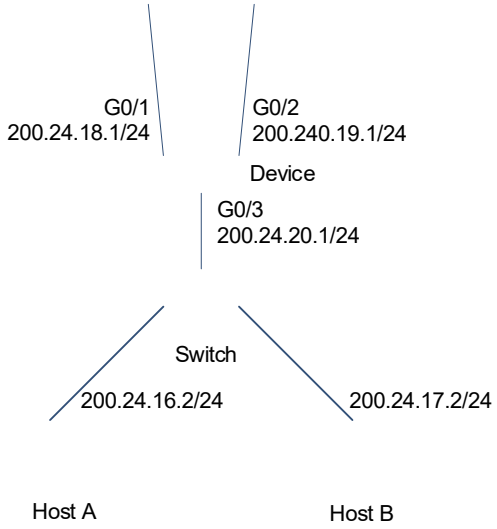
Configure PBR on the devices to achieve the following purposes:

- Packets from subnet 1 (200.24.16.0/24) are sent from GE0/1 first.
- Packets from subnet 2 (200.24.17.0/24) are sent from GE0/2 first.

The two egress links work in mutual backup mode.

## 2. Topology

**Figure 1-1 Topology of Route Map Application in PBR**




## 3. Notes

- Configure two different ACLs to match packets from subnets 1 and 2 respectively.
- Configure the route map RM_FOR_PBR: Configure policy 10 to ensure that "packets from host A are sent from GigabitEthernet 0/1 first"; configure policy 20 to ensure that "packets from host B are sent from GigabitEthernet 0/2 first".
- Configure PBR and apply the route map RM_FOR_PBR to packets received from GigabitEthernet 0/3.
- Configure a PBR to implement redundant backup among multiple next hops.

## 4. Procedure

(1) Configure the IP addresses of interfaces to implement interworking between networks. (Omitted)

(2) Configure ACLs for subnets 200.24.16.0/24 and 200.24.17.0/24, respectively.

```
Device> enable
Device# configure terminal
Device(config)# access-list 1 permit 200.24.16.0 0.0.0.255
Device(config)# access-list 2 permit 200.24.17.0 0.0.0.255
```

(3) Configure the routing map **RM_FOR_PBR**, set the next hop of 200.24.16.0 to 200.24.18.1 and its backup next hop to 200.24.19.1, and set the next hop of 200.24.17.0 to 200.24.19.1 and its backup next hop to 200.24.18.1.

```
Device(config)# route-map RM_FOR_PBR 10
Device(config-route-map)# match ip address 1
```

```
Device(config-route-map)# set ip next-hop 200.24.18.1
Device(config-route-map)# set ip next-hop 200.24.19.1
Device(config-route-map)# exit
Device(config)# route-map RM_FOR_PBR 20
Device(config-route-map)# match ip address 2
Device(config-route-map)# set ip next-hop 200.24.19.1
Device(config-route-map)# set ip next-hop 200.24.18.1
Device(config-route-map)# exit
```

(4) Apply the route map **RM_FOR_PBR** to the interface GigabitEthernet 0/3, and enable load balancing.

```
Device(config)# interface GigabitEthernet 0/3
Device(config-if-GigabitEthernet 0/3)# ip policy route-map RM_FOR_PBR
Device(config-if-GigabitEthernet 0/3)# exit
Device(config)# ip policy redundance
```

## 5. Verification

Check the PBR configurations to verify that the route map is applied to the interface.

```
Device# show ip policy
Balance mode: redundance
Interface                                Route map
GigabitEthernet 0/3                      RM_FOR_PBR
```

Check the configurations of the route map to verify the policy rules.

```
Device# show route-map
route-map RM_FOR_PBR, permit, sequence 10
  Match clauses:
    ip address 1
  Set clauses:
    ip next-hop 200.24.18.1
    ip next-hop 200.24.19.1
route-map RM_FOR_PBR, permit, sequence 20
  Match clauses:
    ip address 2
  Set clauses:
    ip next-hop 200.24.19.1
    ip next-hop 200.24.18.1
```

Check the ACL configurations to verify the packet filtering rules.

```
Device# show access-lists
ip access-list standard 1
 10 permit 200.24.16.0 0.0.0.255
 10 permit 200.24.16.0 0.0.0.255
ip access-list standard 2
```

```
 10 permit 200.24.17.0 0.0.0.255
```

### 6. Configuration Files

Device configuration file

```
!
ip policy redundance
!
access-list 1 permit 200.24.16.0 0.0.0.255
access-list 2 permit 200.24.17.0 0.0.0.255
!
route-map RM_FOR_PBR 10
 match ip address 1
 set ip next-hop 200.24.18.1
 set ip next-hop 200.24.19.1
!
route-map RM_FOR_PBR 20
 match ip address 2
 set ip next-hop 200.24.19.1
 set ip next-hop 200.24.18.1
!
interface GigabitEthernet 0/3
 ip policy route-map RM_FOR_PBR
!
```

### 7. Common Errors

- After a policy that uses ACLs and prefix-lists for matching is configured, the corresponding ACLs and prefix lists are not defined.

## 1.6.3 Configuring a Prefix List

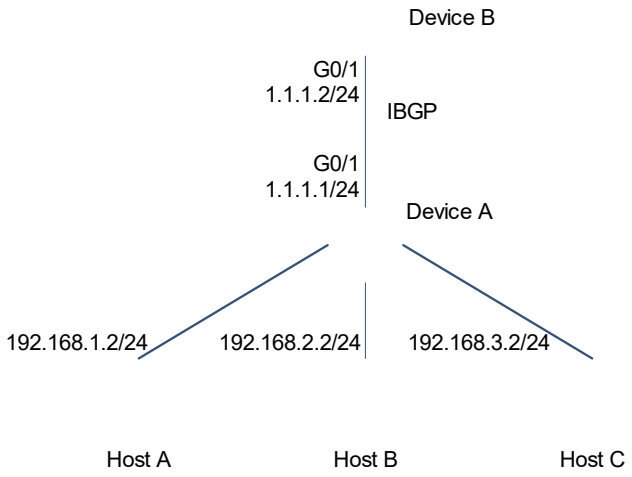
### 1. Requirements

Device A and device B are Interior Border Gateway Protocol (IBGP) neighbors.

On device A, perform routing control so that only routes in the 192.168.1.0/24 network segment are advertised to device B.

## 2. Topology

**Figure 1-1 Topology of the Prefix List Function**




## 3. Notes

- Configure device A and device B as IBGP neighbors and configure device A to advertise routes to the connected three subnets.
- Configure a prefix list.
- Associate the prefix list with device A to filter sent routes.

## 4. Procedure

(1) Configure the IP addresses of interfaces, and configure device A and device B to advertise routes and become IBGP neighbors. (Omitted)

(2) Configure a prefix list to match the traffic of 192.168.1.0/24.

```
Device A> enable
Device A# configure terminal
Device A(config)# ip prefix-list pre1 permit 192.168.1.0/24
```

(3) Specify the neighbor 1.1.1.2 in BGP, and apply the prefix list **pre1** to filter sent routes.

```
Device A(config)# router bgp 100
Device A(config-router)# neighbor 1.1.1.2 prefix-list pre1 out
```

## 5. Verification

Check the prefix list.

```
Device A> enable
Device A# show ip prefix-list
```

```
ip prefix-list pre1: 1 entries
     seq 5 permit 192.168.1.0/24

Device A# show ip bgp
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 192.168.1.0      0.0.0.0                  0                    32768     i
*> 192.168.2.0      0.0.0.0                  0                    32768     i
*> 192.168.3.0      0.0.0.0                  0                    32768     i

Total number of prefixes 3
```

Check the BGP routing table to verify that the routes are filtered correctly.

```
Device B> enable
Device B# show ip bgp
BGP table version is 4, local router ID is 1.1.1.2
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
*>i192.168.1.0      1.1.1.1             0         100           0     i

Total number of prefixes 1
```

### 6. Configuration Files

- Device A configuration file

```
!
ip prefix-list pre1 permit 192.168.1.0/24
!
router bgp 100
neighbor 1.1.1.2 prefix-list pre1 out
!
```

## 1.6.4 Configuring an AS-Path List

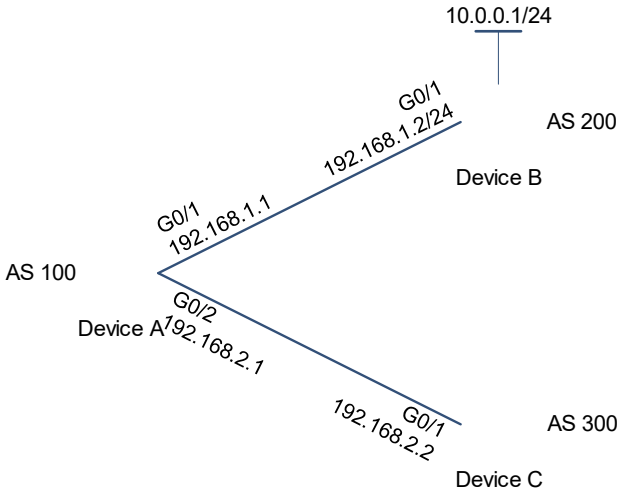
### 1. Requirements

Device A, device B, and device C are Exterior Border Gateway Protocol (EBGP) neighbors.

Device A filters the routes that are received from device B and device C and contain the AS number of 200.

## 2. Topology

**Figure 1-1 Topology of the AS-Path List Function**




## 3. Notes

- Create an AS-path filtering rule to match path information containing only the AS number of 200.
- On device A, establish the EBGP neighborship with device B and device C.
- Associate an AS-path list with device A to filter the routes received from device B and device C.

## 4. Procedure

(1) Configure IP addresses of the interfaces, and complete the basic EBGP configuration. (Omitted)

(2) Configure an AS-path list to match routes with the AS number of 200.

```
Device A(config)# ip as-path access-list 123 permit ^200$
```

(3) On device A, specify the BGP neighbors, and use the AS path list **123** to filter routes.

```
Device A(config)# router bgp 100
Device A(config-router)# neighbor 192.168.1.2 filter-list 123 in
Device A(config-router)# neighbor 192.168.2.2 filter-list 123 in
```

## 5. Verification

Check the AS-path list.

```
Device A> enable
Device A# show ip as-path-access-list
AS path access list 123
```

```
permit ^200$
```

Before associating the AS-path list, check the BGP routing table to verify that all routes are learned.

```
Device A# show ip bgp
BGP table version is 1, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 10.0.0.0/24      192.168.1.2               0                       0 200 i
*> 20.0.0.0/24      192.168.2.2               0                       0 300 i

Total number of prefixes 2
```

After associating the AS path list with device A, check the BGP routing table to verify that routes are filtered correctly.

```
Device A# show ip bgp
BGP table version is 1, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 10.0.0.0/24      192.168.1.2               0                       0 200 i
Total number of prefixes 1
```

### 6. Configuration Files

- Device A configuration file

```
!
ip as-path access-list 123 permit ^200$
!
router bgp 100
neighbor 192.168.1.2 filter-list 123 in
neighbor 192.168.2.2 filter-list 123 in
!
```

## 1.6.5 Configuring a Community List

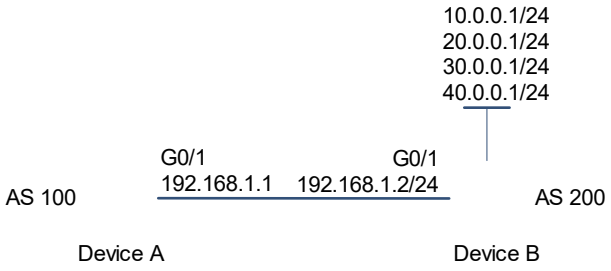
### 1. Requirements

Device A and device B are EBGP neighbors.

Device A filters routes received from device B, and receives only routes with the community attribute of **100:20**.

## 2. Topology

**Figure 1-1 Topology of the Community List Function**




## 3. Notes

- Define a standard community list to match routes with the community attribute of **100:20**.
- Establish the EBGP neighborship between device A and device B.
- Configure device B to advertise routes with the community attribute.
- Associate the community list on device A (BGP can be applied only through a route map) to filter routes received from device B.

## 4. Procedure

(1) Configure IP addresses of the interfaces, and complete the related basic configuration. (Omitted)

(2) Configure a route map on device B, and set the community attribute.

```
Device B> enable
Device B# configure terminal
Device B(config)# route-map comm1
Device B(config-route-map)# set community 100:20 200:20
Device B(config-route-map)# exit
Device B(config)# route-map comm2
Device B(config-route-map)# set community 100:20
Device B(config-route-map)# exit
Device B(config)# route-map comm3
Device B(config-route-map)# set community 200:20
Device B(config-route-map)# exit
```

(3) Configure device B to advertise routes with the community attribute.

```
Device B(config)# router bgp 200
Device B(config-router)# neighbor 192.168.1.1 send-community
Device B(config-router)# network 10.0.0.0 mask 255.255.255.0 route-map comm1
```

```
Device B(config-router)# network 20.0.0.0 mask 255.255.255.0 route-map comm2
Device B(config-router)# network 30.0.0.0 mask 255.255.255.0 route-map comm3
Device B(config-router)# network 40.0.0.0 mask 255.255.255.0
```

(4) On device A, configure a community list and route map to match routes with the community attribute of **100:20**.

```
Device A> enable
Device A# configure terminal
Device A(config)# ip community-list standard test permit 100:20
Device A(config)# route-map COM
Device A(config-route-map)# match community test
Device A(config-route-map)# exit
```

(5) On device A, specify the neighbor 192.168.1.2, and use the route map **COM** to filter routes received from the neighbor.

```
Device A(config)# router bgp 100
Device A(config-router)# neighbor 192.168.1.2 route-map COM in
```

## 5. Verification

Check the community list of device A.

```
Device A> enable
Device A# show ip community-list
Named Community standard list test
permit 100:20
```

Before associating the community list with device A, check the BGP routing table.

```
Device A# show ip bgp
BGP table version is 1, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 10.0.0.0/24      192.168.1.2               0                       0 200 i
*> 20.0.0.0/24      192.168.1.2               0                       0 200 i
*> 30.0.0.0/24      192.168.1.2               0                       0 200 i
*> 40.0.0.0/24      192.168.1.2               0                       0 200 i

Total number of prefixes 4
```

Check the BGP routes on device A.

```
Device A# show ip bgp 10.0.0.0
BGP routing table entry for 10.0.0.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
```

```
  Not advertised to any peer
  200
    192.168.1.2 from 192.168.1.2 (192.168.1.2)
      Origin IGP, metric 0, localpref 100, valid, external, best
      Community: 100:20 200:20
      Last update: Wed Nov  6 18:58:18 2013
```

After configuring the community list on device A, check the BGP routes to verify that the routes are filtered correctly.

```
Device A# show ip bgp
BGP table version is 1, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 10.0.0.0/24      192.168.1.2               0                       0 200 i
*> 20.0.0.0/24      192.168.1.2               0                       0 200 i

Total number of prefixes 2
```

## 6. Configuration Files

- Device A configuration file

```
!
ip community-list standard test permit 100:20
!
route-map COM
 match community test
!
router bgp 100
 neighbor 192.168.1.2 route-map COM in
!
```

- Device B configuration file

```
!
route-map comm1
 set community 100:20 200:20
!
route-map comm2
 set community 100:20
!
route-map comm3
```

```
 set community 200:20
!
router bgp 200
 neighbor 192.168.1.1 send-community
 network 10.0.0.0 mask 255.255.255.0 route-map comm1
 network 20.0.0.0 mask 255.255.255.0 route-map comm2
 network 30.0.0.0 mask 255.255.255.0 route-map comm3
 network 40.0.0.0 mask 255.255.255.0
!
```