



Configuring Route Policy Manager

This chapter contains the following sections:

- [Finding Feature Information, on page 1](#)
- [Information About Route Policy Manager, on page 1](#)
- [Licensing Requirements for Route Policy Manager , on page 9](#)
- [Prerequisites for Route Policy Manager , on page 9](#)
- [Guidelines and Limitations, on page 9](#)
- [Default Settings for Route Policy Manager Parameters, on page 9](#)
- [Configuring Route Policy Manager, on page 10](#)
- [Configuration Examples for Route Policy Manager, on page 20](#)
- [Related Documents for Route Policy Manager, on page 20](#)
- [Standards for Route Policy Manager, on page 20](#)
- [Feature History for Route Policy Manager, on page 20](#)

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at <https://tools.cisco.com/bugsearch/> and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information" chapter or the Feature History table in this chapter.

Information About Route Policy Manager

Route Policy Manager supports route maps and IP prefix lists. These features are used for route redistribution. A prefix list contains one or more IPv4 network prefixes and the associated prefix length values. You can use a prefix list by itself in features such as Border Gateway Protocol (BGP) templates, route filtering, or redistribution of routes that are exchanged between routing domains.

Route maps can apply to both routes and IP packets. Route filtering and redistribution pass a route through a route map.

Prefix Lists

You can use prefix lists to permit or deny an address or range of addresses. Filtering by a prefix list involves matching the prefixes of routes or packets with the prefixes listed in the prefix list. An implicit deny is assumed if a given prefix does not match any entries in a prefix list.

You can configure multiple entries in a prefix list and permit or deny the prefixes that match the entry. Each entry has an associated sequence number that you can configure. If you do not configure a sequence number, Cisco NX-OS assigns a sequence number automatically. Cisco NX-OS evaluates prefix lists starting with the lowest sequence number. Cisco NX-OS processes the first successful match for a given prefix. Once a match occurs, Cisco NX-OS processes the permit or deny statement and does not evaluate the rest of the prefix list.

Prefix Lists in Cisco NX-OS support only one of the following addresses at a time:

- source
- destination
- group address



Note An empty prefix list permits all routes.

MAC Lists

You can use MAC lists to permit or deny a MAC address or range of addresses. A MAC list consists of a list of MAC addresses and optional MAC masks. A MAC mask is a wild-card mask that is logically AND-ed with the MAC address when the route map matches on the MAC list entry. Filtering by a MAC list involves matching the MAC address of packets with the MAC addresses listed in the MAC list. An implicit deny is assumed if a given MAC address does not match any entries in a MAC list.

You can configure multiple entries in a MAC list and permit or deny the MAC addresses that match the entry. Each entry has an associated sequence number that you can configure. If you do not configure a sequence number, Cisco NX-OS assigns a sequence number automatically. Cisco NX-OS evaluates MAC lists starting with the lowest sequence number. Cisco NX-OS processes the first successful match for a given MAC address. Once a match occurs, Cisco NX-OS processes the permit or deny statement and does not evaluate the rest of the MAC list.

Route Maps

You can use route maps for route redistribution or policy-based routing. Route map entries consist of a list of match and set criteria. The match criteria specify match conditions for incoming routes or packets, and the set criteria specify the action taken if the match criteria are met.

You can configure multiple entries in the same route map. These entries contain the same route map name and are differentiated by a sequence number.

You create a route map with one or more route map entries arranged by the sequence number under a unique route map name. The route map entry has the following parameters:

- Sequence number
- Permission—permit or deny

- Match criteria
- Set changes

The IPv4 and the IPv6-based matches on the same route map sequence number is not supported in Cisco Nexus 7000 Series.

By default, a route map processes routes or IP packets in a linear fashion, that is, starting from the lowest sequence number. You can configure the route map to process in a different order using the **continue** statement, which allows you to determine which route map entry to process next.

Match Criteria

You can use a variety of criteria to match a route or IP packet in a route map. Some criteria, such as BGP community lists, are applicable only to a specific routing protocol, while other criteria, such as the IP source or the destination address, can be used for any route or IP packet.

When Cisco NX-OS processes a route or packet through a route map, it compares the route or packet to each of the match statements configured. If the route or packet matches the configured criteria, Cisco NX-OS processes it based on the permit or deny configuration for that match entry in the route map and any set criteria configured.

The match categories and parameters are as follows:

- IP access lists—(For policy-based routing only). Match based on source or destination IP address, protocol, or QoS parameters.
- BGP parameters—Match based on AS numbers, AS-path, community attributes, or extended community attributes.
- Prefix lists—Match based on an address or range of addresses.
- Multicast parameters—Match based on rendezvous point, groups, or sources.
- Other parameters—Match based on IP next-hop address or packet length.

Set Changes

Once a route or packet matches an entry in a route map, the route or packet can be changed based on one or more configured set statements.

The set changes are as follows:

- BGP parameters—Change the AS-path, tag, community, extended community, dampening, local preference, origin, or weight attributes.
- Metrics—Change the route-metric, the route-tag, or the route-type.
- Policy-based routing only—Change the interface or the default next-hop address.
- Other parameters—Change the forwarding address or the IP next-hop address.

Access Lists

IP access lists can match the packet to a number of IP packet fields such as the following:

- Source or destination IPv4 or IPv6 address

- Protocol
- Precedence
- ToS

You can use ACLs in a route map for policy-based routing only. See the *Cisco Nexus 7000 Series NX-OS Security Configuration Guide* for more information on ACLs.

AS Numbers for BGP

You can configure a list of AS numbers to match against BGP peers. If a BGP peer matches an AS number in the list and matches the other BGP peer configuration, BGP creates a session. If the BGP peer does not match an AS number in the list, BGP ignores the peer. You can configure the AS numbers as a list, a range of AS numbers, or you can use an AS-path list to compare the AS numbers against a regular expression.

AS-path Lists for BGP

You can configure an AS-path list to filter inbound or outbound BGP route updates. If the route update contains an AS-path attribute that matches an entry in the AS-path list, the router processes the route based on the permit or deny condition configured. You can configure AS-path lists within a route map.

You can configure multiple AS-path entries in an AS-path list by using the same AS-path list name. The router processes the first entry that matches.

Community Lists for BGP

You can filter BGP route updates based on the BGP community attribute by using community lists in a route map. You can match the community attribute based on a community list, and you can set the community attribute using a route map.

A community list contains one or more community attributes. If you configure more than one community attribute in the same community list entry, the BGP route must match all community attributes listed to be considered a match.

You can also configure multiple community attributes as individual entries in the community list by using the same community list name. In this case, the router processes the first community attribute that matches the BGP route, using the permit or deny configuration for that entry.

You can configure community attributes in the community list in one of the following formats:

- A named community attribute, such as **internet** or **no-export**.
- In *aa:nn* format, where the first two bytes represent the two-byte AS number and the last two bytes represent a user-defined network number.
- A regular expression.

See the *Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference* for more information on regular expressions.

Extended Community Lists for BGP

Extended community lists support 4-byte AS numbers. You can configure community attributes in the extended community list in one of the following formats:

- In *aa4.nn* format, where the first four bytes represent the four-byte AS number and the last two bytes represent a user-defined network number.
- A regular expression.

See the *Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference* for more information on regular expressions.

Cisco NX-OS supports generic specific extended community lists, which provide similar functionality to regular community lists for four-byte AS numbers. You can configure generic specific extended community lists with the following properties:

- Transitive—BGP propagates the community attributes across autonomous systems.
- Nontransitive—BGP removes community attributes before propagating the route to another autonomous system.

Route Redistribution and Route Maps

You can use route maps to control the redistribution of routes between routing domains. Route maps match on the attributes of the routes to redistribute only those routes that pass the match criteria. The route map can also modify the route attributes during this redistribution using the set changes.

The router matches redistributed routes against each route map entry. If there are multiple match statements, the route must pass all of the match criteria. If a route passes the match criteria defined in a route map entry, the actions defined in the entry are executed. If the route does not match the criteria, the router compares the route against subsequent route map entries. Route processing continues until a match is made or the route is processed by all entries in the route map with no match.

Each ACL ends with an implicit deny statement, by design convention; there is no similar convention for route-maps. If the end of a route-map is reached during matching attempts, the result depends on the specific application of the route-map. Fortunately, route-maps that are applied to redistribution behave the same way as ACLs: if the route does not match any clause in a route-map, then the route redistribution is denied, as if the route-map contained a deny statement at the end.



Note When you redistribute BGP to IGP, iBGP is redistributed as well. To override this behavior, you must insert an additional deny statement into the route map.

Route Map Support Matrix for Routing Protocols

The following tables include the configurable match and set statements for routing protocols on Cisco Nexus 70xx and 77xx Series switches running the latest shipping release. For specific release information, see the [Feature History for Route Policy Manager, on page 20](#).

The following legend applies to the tables:

- Yes—The statement is supported for the protocol.
- No—The statement is not supported for the protocol.
- If a statement does not apply for the protocol, there is an em dash (—) in the column next to the statement.

- Where clarification is required, information is added in the appropriate row/column.

Table 1: SET Route Map Statements by Protocol

SET Route Map Statement	OSPF Redistribution	EIGRP Redistribution	ISIS Redistribution	RIP Redistribution	BGP Redistribution
Forwarding-address	Yes	—	—	—	—
Standard/Extended Community	—	—	—	—	Standard community only
Site of Origin (SOO)	—	—	—	—	No
Routing Protocol Metric	Yes	Yes	Yes	Yes	Yes
Routing Protocol Metric Type	Yes	—	No	—	—
Route Tag	Yes	Yes	No	Yes	—
NSSA Only	Yes	—	—	—	—
Origin	—	—	—	—	Yes
Level	—	—	Yes	—	—
Weight	—	—	—	—	Yes

Table 2: SET Route Map Statements by Protocol

SET Route Map Statement	BGP Neighbor	BGP Table Map	OSPF Table Map	EIGRP Table Map	ISIS Table Map	EIGRP Distribute List
Standard/Extended Community	Yes	No	—	—	—	—
Standard/Extended Community-List Deletion	Yes	No	—	—	—	—
Site of Origin (SOO)	No	—	—	—	—	—
Routing Protocol Metric	No	No	—	—	—	Yes
Routing Protocol Metric Type	Yes	No	—	—	—	—
IPv4 Next Hop	Yes	—	—	—	—	—
IPv6 Next Hop	Yes	—	—	—	—	—

SET Route Map Statement	BGP Neighbor	BGP Table Map	OSPF Table Map	EIGRP Table Map	ISIS Table Map	EIGRP Distribute List
IPv4 Prefix list	Yes	—	—	—	—	—
IPv6 Prefix list	Yes	—	—	—	—	—
Interface	No	—	—	—	—	—
Route Tag	—	—	—	—	—	Yes
AS PATH	Yes	No	—	—	—	—
Orgin	Yes	No	—	—	—	—
All Path Advertisement	Yes	No	—	—	—	—
Distance	—	Yes	Yes	Yes	Yes	—
Dampening	No	No	—	—	—	—
Level	No	No	—	—	—	—
Weight	Yes	Yes	No	—	—	—

Table 3: MATCH Route Map Statements by Protocol

MATCH Route Map Statement	OSPF Redistribution	EIGRP Redistribution	ISIS Redistribution	RIP Redistribution	BGP Redistribution
Community List	OSPFv2 only	Yes	yes	Yes	—
Ext Community List	OSPFv2 only	Yes	—	Yes	—
Interface	Yes	Yes	Yes	Yes	Yes
IPv4 Next Hop	Yes	Yes	Yes	Yes	Yes
IPv6 Next Hop	Yes	Yes	Yes	No	Yes
Metric	Yes	Yes	Yes	Yes	Yes
Route Type	Yes	Yes	Yes	Yes	Yes
Tag	Yes	Yes	Yes	Yes	Yes
IPv6 Prefix List	Yes	Yes	Yes	No	Yes
IPv4 Prefix list	Yes	Yes	Yes	Yes	Yes
IP ACL	No	No	No	No	No
Source Protocol	Yes	Yes	Yes	Yes	—

MATCH Route Map Statement	OSPF Redistribution	EIGRP Redistribution	ISIS Redistribution	RIP Redistribution	BGP Redistribution
AS Path	No	No	No	No	—
AS Number	No	No	No	No	—

Table 4: MATCH Route Map Statements by Protocol

MATCH Route Map Statement	BGP Neighbor	BGP Table Map	OSPF Table Map	EIGRP Table Map	ISIS Table Map	EIGRP Distribute List
Community List	Yes	Yes	—	—	—	—
Ext Community List	Yes	Yes	—	—	—	—
Interface	—	No	Yes	Yes	Yes	—
IPv4 Next Hop	Yes	Yes	Yes	Yes	Yes	Yes
IPv6 Next Hop	Yes	Yes	Yes	Yes	Yes	Yes
Metric	Yes	Yes	Yes	No	No	No
Route Type	Yes	Yes	Yes	Yes	Yes	No
Tag	—	Yes	Yes	Yes	No	Yes
IPv6 Prefix List	Yes	Yes	Yes	Yes	Yes	Yes
IPv4 Prefix list	Yes	Yes	Yes	Yes	Yes	Yes
IP ACL	No	No	No	No	No	No
AS Path	Yes	Yes	—	—	—	—
AS Number	Yes	No	—	—	—	—
IPv4 Route Source	—	—	Yes	—	—	—

Policy-Based Routing

You can use policy-based routing to forward a packet to a specified next-hop address based on the source of the packet or other fields in the packet header.

Licensing Requirements for Route Policy Manager

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites for Route Policy Manager

If you configure VDCs, install the appropriate license and enter the desired VDC (see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for configuration information and the *Cisco NX-OS Licensing Guide* for licensing information).

Guidelines and Limitations

- An empty route map denies all the routes.
- An empty prefix list permits all the routes.
- Without any match statement in a route-map entry, the permission (permit or deny) of the route-map entry decides the result for all the routes or packets.
- If referred policies (for example, prefix lists) within a match statement of a route-map entry return either a no-match or a deny-match, Cisco NX-OS fails the match statement and processes the next route-map entry.
- When you change a route map, Cisco NX-OS holds all the changes until you exit from the route-map configuration submode. Cisco NX-OS then sends all the changes to the protocol clients to take effect.
- Because you can use a route map before you define it, verify that all your route maps exist when you finish a configuration change.
- You can view the route-map usage for redistribution and filtering. Each individual routing protocol provides a way to display these statistics.
- When you redistribute BGP to IGP, iBGP is redistributed as well. To override this behavior, you must insert an additional deny statement into the route map.
- If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings for Route Policy Manager Parameters

Default Route Policy Manager Parameters

Parameters	Default
Route Policy Manager	Enabled

Parameters	Default
Administrative distance	115

Configuring Route Policy Manager

Configuring IP Prefix Lists

IP prefix lists match the IP packet or route against a list of prefixes and prefix lengths. You can create an IP prefix list for IPv4 and create an IPv6 prefix list for IPv6.

You can configure the prefix list entry to match the prefix length exactly, or to match any prefix with a length that matches the configured range of prefix lengths.

Use the **ge** and **le** keywords to create a range of possible prefix lengths. The incoming packet or route matches the prefix list if the prefix matches and if the prefix length is greater than or equal to the **ge** keyword value (if configured) and less than or equal to the **le** keyword value (if configured).

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	(Optional) switch(config)# {ip ipv6} prefix-list name description string	Adds an information string about the prefix list.
Step 3	switch(config)# ip prefix-list name [seq number] [{permit deny} prefix {[eq prefix-length] [ge prefix-length] [le prefix-length]}]	Creates an IPv4 prefix list or adds a prefix to an existing prefix list. The prefix length is matched as follows: <ul style="list-style-type: none"> • eq—Matches the exact <i>prefix length</i>. • ge—Matches a prefix length that is equal to or greater than the configured <i>prefix length</i>. • le—Matches a prefix length that is equal to or less than the configured <i>prefix length</i>.
Step 4	switch(config)# ipv6 prefix-list name [seq number] [{permit deny} prefix {[eq prefix-length] [ge prefix-length] [le prefix-length]}]	Creates an IPv6 prefix list or adds a prefix to an existing prefix list. The prefix length is matched as follows: <ul style="list-style-type: none"> • eq—Matches the exact <i>prefix length</i>. • ge—Matches a prefix length that is equal to or greater than the configured <i>prefix length</i>. • le—Matches a prefix length that is equal to or less than the configured <i>prefix length</i>.

	Command or Action	Purpose
Step 5	(Optional) switch(config)# show {ip ipv6} prefix-list name	Displays information about prefix lists.
Step 6	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create an IPv4 prefix list with two entries and apply the prefix list to a BGP neighbor:

```
switch# configure terminal
switch(config)# ip prefix-list allowprefix seq 10 permit 192.0.2.0/24 eq 24
switch(config)# ip prefix-list allowprefix seq 20 permit 209.165.201.0/27 eq 27
switch(config)# router bgp 65536:20
switch(config-router)# neighbor 192.0.2.1/16 remote-as 65535:20
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# prefix-list allowprefix in
```

Configuring MAC Lists

You can configure a MAC list to permit or deny a range of MAC addresses.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# mac-list name [seq number] {permit deny} mac-address [mac-mask]	Creates a MAC list or adds a MAC address to an existing MAC list. The <i>seq</i> range is from 1 to 4294967294. The <i>mac-mask</i> specifies the portion of the MAC address to match against and is in MAC address format.
Step 3	(Optional) switch(config)# show mac-list name	Displays information about MAC lists.
Step 4	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create a MAC list and copy the running configuration to the startup configuration:

```
switch# configure terminal
switch(config)# mac-list AllowMac seq 1 permit 0022.5579.a4c1 ffff.ffff.0000
switch# copy running-config startup-config
```

Configuring AS-path Lists

You can specify an AS-path list filter on both inbound and outbound BGP routes. Each filter is an access list based on regular expressions. If the regular expression matches the representation of the AS-path attribute of the route as an ASCII string, the permit or deny condition applies.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# ip as-path access-list <i>name</i> {deny permit} <i>expression</i>	Creates a BGP AS-path list using a regular expression.
Step 3	(Optional) switch(config)# show {ip ipv6} as-path-access-list <i>name</i>	Displays information about as-path access lists.
Step 4	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create an AS-path list with two entries and apply the AS path list to a BGP neighbor:

```
switch# configure terminal
switch(config)# ip as-path access-list AllowAS permit 64510
switch(config)# ip as-path access-list AllowAS permit 64496
switch(config)# copy running-config startup-config
switch(config)# router bgp 65536:20
switch(config-router)# neighbor 192.0.2.1/16 remote-as 65535:20
switch(config-router-neighbor)# address-family ipv4 unicast
switch(config-router-neighbor-af)# filter-list AllowAS in
```

Configuring Community Lists

You can use community lists to filter BGP routes based on the community attribute. The community number consists of a 4-byte value in the *aa.nn* format. The first two bytes represent the autonomous system number, and the last two bytes represent a user-defined network number.

When you configure multiple values in the same community list statement, all community values must match to satisfy the community list filter. When you configure multiple values in separate community list statements, the first list that matches a condition is processed.

Use community lists in a match statement to filter BGP routes based on the community attribute.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# ip community-list standard <i>list-name</i> {deny permit} [<i>community-list</i>] [internet] [local-AS] [no-advertise] [no-export]	Creates a standard BGP community list. The <i>list-name</i> can be any case-sensitive, alphanumeric string up to 63 characters. The <i>community-list</i> can be one or more communities in the <i>aa:nn</i> format. Do not perform this step if you need to create an expanded BGP community list.
Step 3	switch(config)# ip community-list expanded <i>list-name</i> {deny permit} <i>expression</i> Example:	Creates an expanded BGP community list using a regular expression. Do not perform this step if you need to create a standard BGP community list.
Step 4	(Optional) switch(config)# show ip community list <i>name</i>	Displays information about community lists.
Step 5	(Optional) switch# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create a standard community list with two entries:

```
switch# configure terminal
switch(config)# ip community-list standard BGPCommunity permit no-advertise 65536:20
switch(config)# ip community-list standard BGPCommunity permit local-AS no-export
switch(config)# copy running-config startup-config
```

Configuring Extended Community Lists

You can use extended community lists to filter BGP routes based on the community attribute. The community number consists of a 6-byte value in the *aa4:nn* format. The first four bytes represent the autonomous system number, and the last two bytes represent a user-defined network number.

When you configure multiple values in the same extended community list statement, all extended community values must match to satisfy the extended community list filter. When you configure multiple values in separate extended community list statements, the first list that matches a condition is processed.

Use extended community lists in a match statement to filter BGP routes based on the extended community attribute.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# ip extcommunity-list standard <i>list-name</i> {deny permit} 4bytegeneric {transitive nontransitive} <i>community1</i> [<i>community2...</i>]	Creates a standard BGP extended community list. The <i>community</i> can be one or more extended communities in the <i>aa4:nn</i> format. Do not perform this step if you need to create an expanded BGP extended community list.
Step 3	switch(config)# ip extcommunity-list expanded <i>list-name</i> {deny permit} <i>expression</i>	Creates an expanded BGP extended community list using a regular expression. Do not perform this step if you need to create a standard BGP extended community list.
Step 4	(Optional) switch(config)# show ip extcommunity list <i>name</i>	Displays information about community lists.
Step 5	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create a generic specific extended community list:

```
switch# configure terminal
switch(config)# ip extcommunity-list standard test1 permit 4bytegeneric transitive 65536:40
65536:60
switch(config)# copy running-config startup-config
```

Optional Match Parameters for Route Maps

You can configure the following optional match parameters for route maps in route-map configuration mode:



Note The **default-information originate** command ignores **match** statements in the optional route map.

Command	Purpose
switch(config-route-map)# match as-path <i>name</i> [<i>name...</i>]	Matches against one or more AS-path lists. Create the AS-path list with the ip as-path access-list command.
switch(config-route-map)# match as-number { <i>number</i> [, <i>number...</i>] as-path-list <i>name</i> [<i>name...</i>]}	Matches against one or more AS numbers or AS-path lists. Create the AS-path list with the ip as-path access-list command. The number range is from 1 to 65535. The AS-path list name can be any case-sensitive, alphanumeric string up to 63 characters.

Command	Purpose
switch(config-route-map)# match community <i>name</i> [<i>name...</i>][exact-match]	Matches against one or more community lists. Create the community list with the ip community-list command.
switch(config-route-map)# match extcommunity <i>name</i> [<i>name...</i>][exact-match]	Matches against one or more extended community lists. Create the community list with the ip extcommunity-list command.
switch(config-route-map)# match interface <i>interface-type number</i> [<i>interface-type number...</i>]	Matches any routes that have their next hop out one of the configured interfaces. Use ? to find a list of supported interface types.
switch(config-route-map)# match ip address prefix-list <i>name</i> [<i>name...</i>]	Matches against one or more IPv4 prefix lists. Use the ip prefix-list command to create the prefix list.
switch(config-route-map)# match ipv6 address prefix-list <i>name</i> [<i>name...</i>]	Matches against one or more IPv6 prefix lists. Use the ipv6 prefix-list command to create the prefix list.
switch(config-route-map)# match ip multicast [source <i>ipsource</i>] [[group <i>ipgroup</i>] [rp <i>iprp</i>]]	Matches an IPv4 multicast packet based on the multicast source, group, or rendezvous point.
switch(config-route-map)# match ipv6 multicast [source <i>ipsource</i>] [[group <i>ipgroup</i>] [rp <i>iprp</i>]]	Matches an IPv6 multicast packet based on the multicast source, group, or rendezvous point.
switch(config-route-map)# match ip next-hop prefix-list <i>name</i> [<i>name...</i>]	Matches the IPv4 next-hop address of a route to one or more IP prefix lists. Use the ip prefix-list command to create the prefix list.
switch(config-route-map)# match ipv6 next-hop prefix-list <i>name</i> [<i>name...</i>]	Matches the IPv6 next-hop address of a route to one or more IP prefix lists. Use the ipv6 prefix-list command to create the prefix list.
switch(config-route-map)# match ip route-source prefix-list <i>name</i> [<i>name...</i>]	Matches the IPv4 route source address of a route to one or more IP prefix lists. Use the ip prefix-list command to create the prefix list.
switch(config-route-map)# match ipv6 route-source prefix-list <i>name</i> [<i>name...</i>]	Matches the IPv6 route-source address of a route to one or more IP prefix lists. Use the ipv6 prefix-list command to create the prefix list.
switch(config-route-map)# match mac-list <i>name</i> [<i>name...</i>]	Matches against one or more MAC lists. Use the mac-list command to create the MAC list.
switch(config-route-map)# match metric <i>value</i> [+ <i>deviation</i> .] [<i>value..</i>]	Matches the route metric against one or more metric values or value ranges. Use +- deviation argument to set a metric range. The route map matches any route metric that falls the range: <i>value - deviation</i> to <i>value + deviation</i> .

Command	Purpose
switch(config-route-map)# match route-type <i>route-type</i>	Matches against a type of route. The <i>route-type</i> can be one or more of the following: <ul style="list-style-type: none"> • external • inter-area • internal • intra-area • level-1 • level-2 • local • nssa-external • type-1 • type-2
switch(config-route-map)# match tag <i>tagid</i> [<i>tagid...</i>]	Matches a route against one or more tags for filtering or redistribution.
switch(config-route-map)# match vlan <i>vlan-id</i> [<i>vlan-range</i>]	Matches against a VLAN.

Optional Set Parameters for Route Maps

You can configure the following optional set parameters for route maps in route-map configuration mode:

Command	Purpose
switch(config-route-map)# set as-path { tag prepend { last-as <i>number</i> <i>as-1</i> [<i>as-2...</i>]} }	Modifies an AS-path attribute for a BGP route. You can prepend the configured number of last AS numbers or a string of particular AS-path values (<i>as-1 as-2...as-n</i>).
switch(config-route-map)# set comm-list <i>name</i> delete	Removes communities from the community attribute of an inbound or outbound BGP route update. Use the ip community-list command to create the community list.

Command	Purpose
<pre>switch(config-route-map)# set community {none additive local-AS no-advertise no-export community-1 [community-2...]}</pre>	<p>Sets the community attribute for a BGP route update.</p> <p>Note When you use both the set community and set comm-list delete commands in the same sequence of a route map attribute, the deletion operation is performed before the set operation.</p> <p>Note Use the send-community command in BGP neighbor address family configuration mode to propagate BGP community attributes to BGP peers.</p>
<pre>switch(config-route-map)# set dampening halflife reuse suppress duration</pre>	<p>Sets the following BGP route dampening parameters:</p> <ul style="list-style-type: none"> • <i>halflife</i>—The range is from 1 to 45 minutes. The default is 15. • <i>reuse</i>—The range is from is 1 to 20000 seconds. The default is 750. • <i>suppress</i>—The range is from is 1 to 20000. The default is 2000. • <i>duration</i>—The range is from is 1 to 255 minutes. The default is 60.
<pre>switch(config-route-map)# set distance value</pre>	<p>Sets the administrative distance of routes for OSPFv2 or OSPFv3. The range is from 1 to 255.</p>
<pre>switch(config-route-map)# set extcomm-list name delete</pre>	<p>Removes communities from the extended community attribute of an inbound or outbound BGP route update. Use the ip extcommunity-list command to create the extended community list.</p>
<pre>switch(config-route-map)# set extcommunity 4byteas-generic {transitive nontransitive} {none additive} community-1 [community-2...]}</pre>	<p>Sets the extended community attribute for a BGP route update.</p> <p>Note When you use both the set extcommunity and set extcomm-list delete commands in the same sequence of a route map attribute, the deletion operation is performed before the set operation.</p> <p>Note Use the send-community command in BGP neighbor address family configuration mode to propagate BGP extended community attributes to BGP peers.</p>

Command	Purpose
switch(config-route-map)# set extcommunity cost <i>community-id1 cost [igp pre-bestpath] [community-id2...]</i>	<p>Sets the cost community attribute for a BGP route update. This attribute allows you to customize the BGP best path selection process for a local autonomous system or confederation. The <i>community-id</i> range is from 0 to 255. The <i>cost</i> range is from 0 to 4294967295. The path with the lowest cost is preferred. For paths with equal cost, the path with the lowest community ID is preferred.</p> <p>The igp keyword compares the cost after the IGP cost comparison. The pre-bestpath keyword compares before all other steps in the bestpath algorithm.</p>
switch(config-route-map)# set extcommunity rt <i>community-1 [additive] [community-2...]</i>	<p>Sets the extended community route target attribute for a BGP route update. The <i>community</i> value can be a 2-byte AS number:4-byte network number, a 4-byte AS number:2-byte network number, or an IP address:2-byte network number.</p> <p>Use the additive keyword to add a route target to an existing extended community route target attribute.</p>
switch(config-route-map)# set forwarding-address	Sets the forwarding address for OSPF.
switch(config-route-map)# set level { backbone level-1 level-1-2 level-2 }	Sets what area to import routes to for IS-IS. The options for IS-IS are level-1, level-1-2, or level-2. The default is level-1.
switch(config-route-map)# set local-preference <i>value</i>	Sets the BGP local preference value. The range is from 0 to 4294967295.
switch(config-route-map)# set metric [+ -] <i>bandwidth-metric</i>	Adds or subtracts from the existing metric value. The metric is in Kb/s. The range is from 0 to 4294967295.
switch(config-route-map)# set metric <i>bandwidth [delay reliability load mtu]</i>	<p>Sets the route metric values. Metrics are as follows:</p> <ul style="list-style-type: none"> • <i>metric0</i>—Bandwidth in Kb/s. The range is from 0 to 4294967295. • <i>metric1</i>—Delay in 10-microsecond units. • <i>metric2</i>—Reliability. The range is from 0 to 255 (100 percent reliable). • <i>metric3</i>—Loading. The range is from 1 to 200 (100 percent loaded). • <i>metric4</i>—MTU of the path. The range is from 1 to 4294967295.

Command	Purpose
switch(config-route-map)# set metric-type { external internal type-1 type-2 }	<p>Sets the metric type for the destination routing protocol. The options are as follows:</p> <ul style="list-style-type: none"> • external—IS-IS external metric • internal— IGP metric as the MED for BGP • type-1—OSPF external type 1 metric • type-2—OSPF external type 2 metric <p>The set metric-type internal command affects an outgoing policy and an eBGP neighbor only. If you configure both the metric and metric-type internal commands in the same BGP peer outgoing policy, then Cisco NX-OS ignores the metric-type internal command.</p>
switch(config-route-map)# set nssa-only	Sets Type-7 LSA generated on ASBR with no P bit set. This prevents Type-7 to Type-5 LSA translation in OSPF.
switch(config-route-map)# set origin { egp as-number igp incomplete }	Sets the BGP origin attribute. The EGP <i>as-number</i> range is from 0 to 65535.
switch(config-route-map)# set tag name	Sets the tag value for the destination routing protocol. The <i>name</i> parameter is an unsigned integer.
switch(config-route-map)# set weight count	Sets the weight for the BGP route. The range is from 0 to 65535.

Verifying the Route Policy Manager Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show ip community-list [<i>name</i>]	Displays information about a community list.
show ip extcommunity-list [<i>name</i>]	Displays information about an extended community list.
show [ip ipv6] prefix-list [<i>name</i>]	Displays information about an IPv4 or IPv6 prefix list.
show route-map [<i>name</i>]	Displays information about a route map.

Configuration Examples for Route Policy Manager

This example shows how to use an address family to configure Route Policy Manager so that any unicast and multicast routes from neighbor 209.0.2.1 are accepted if they match prefix-list AllowPrefix:

```
router bgp 64496

neighbor 209.0.2.1 remote-as 64497
  address-family ipv4 unicast
    route-map filterBGP in

route-map filterBGP
  match ip address prefix-list AllowPrefix

ip prefix-list AllowPrefix 10 permit 192.0.2.0/24
ip prefix-list AllowPrefix 20 permit 209.165.201.0/27
```

Related Documents for Route Policy Manager

Related Topic	Document Title
Route Policy Manager CLI commands	<i>Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference</i>
VDCs and VRFs	<i>Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide</i>

Standards for Route Policy Manager

No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.

Feature History for Route Policy Manager

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Feature Name	Releases	Feature Information
Multiple match statements under table-map	6.2(14)	Added support for multiple match statements under table-map.
Route map support matrix	6.2(2)	Added the route map support matrix for routing protocols.

Feature Name	Releases	Feature Information
Match interfaces	6.2(2)	<p>Added support for null interfaces to the match interface command.</p> <p>Added support for the following set and match statements in a route map for the EIGRP distribute list:</p> <ul style="list-style-type: none"> • Set routing protocol metric • Set route tag • Match tag
Route policy manager	6.1(1)	Added support for the set distance command and for the inter-area and intra-area options for the match route-type command.
MPLS set clauses	5.2(1)	Added support for set extcommunity cost , set extcommunity rt , and set nssa-only commands.
MAC lists , metric, and VLANs	5.0(2)	Added support for the match mac-list , match metric , and match vlan commands.
Extended community lists	4.2(1)	Added support for generic specific extended community lists.
Match interfaces	4.1(2)	Added support to match a list of interfaces in a route map.
Match AS numbers	4.1(2)	Added support to match a range of AS numbers in a route map.
Route policy manager	4.0(1)	This feature was introduced.

