Chapter 20

Filtering IP Routes

Introduction

This chapter describes the switch's functions for filtering IP routes. IP route filtering enables you to control your routing tables, for example, to meet the terms of business relationships you have with the networks to which you are connected.

If you are a network provider, you can filter the routing information that your switches receive from the networks they connect to, and that they advertise to those networks. This gives you control over the path of any traffic originating from or traversing your network. Usually, one or more of your switches form peer relationships with switches at other ISPs with which you have entered into data transporting agreements. The process of filtering is, in effect, the process of specifying the routes that your switches send or receive from each of their peers.

The switch provides several different mechanisms for filtering routes. Some of the functionality of these mechanisms overlaps, so sometimes you can achieve a given filtering effect in several ways. This chapter discusses all the different mechanisms and places them in context within the overall picture of how you can filter routes.

In general, configuring any filter involves the following steps, which this chapter describes:

1. Select the desired filter type, as described in **Types of Filters**.
2. Create the filter, as described in **Creating Filters**.
3. Apply it, as described in **Applying Filters**.

When to use filters

You can use route filtering to select the routes that:

- the switch copies from a routing protocol into its Routing Information Base (RIB). This determines which routes the switch uses to send traffic (**Applying Filters when Writing to the RIB**).
- the switch copies from its RIB into a routing protocol. This determines which routes the protocol has available for advertising to neighbouring devices (**Applying Filters when Redistributing from the RIB**).
- routing protocols actually advertise to neighbouring devices (**Applying Filters before Advertising Routes**).

The RIB is another term for the switch's main IP route table, which is described in "The Routing Table" on page 13-22 of Chapter 13, Internet Protocol (IP).

Types of routes you can filter

As explained above, this chapter first divides the information about filtering into sections about each type of filter, rather than each type of routing protocol. Then it summarises the available filters for each routing protocol, in the following sections:

- **Open Shortest Path First (OSPF)**
- **Routing Information Protocol (RIP)**
- **Statically-Configured Routes**

Types of Filters

The type of filter to use depends on the route source and the point at which you want to filter. This section describes the available filters and summarises the circumstances in which you use them:

- **Prefix Lists**
- **IP Route Filters**

Prefix Lists

Description

A prefix list is a list of entries, each of which specifies:

- an IPv4 prefix, and a mask length or range of mask lengths
- whether those prefixes explicitly match or explicitly do not match the prefix list

When to use

Prefix lists offer detailed control over which routes you import, export or advertise.

"Applying Filters" on page 20-5 describes in detail how to use prefix lists, but this section summarises the uses.

IP Route Filters

Description

Route filters are simple filters that examine a number of aspects of each route. When you apply filters to routing information that the switch receives, the filter determines whether each route is added to the RIB. When you apply filters to routing information that the switch transmits, the filter determines whether each route is advertised.

When to use

"Applying Filters" on page 20-5 describes in detail how to use IP route filters, but this section summarises the uses.

The main uses of IP route filters are to select:

- RIP routes when adding routes to the RIB
- RIP routes when determining which routes to advertise
- OSPF, static or interface routes when determining which routes to redistribute from the RIB into RIP
- OSPF summary routes when determining which routes to redistribute from the RIB into the OSPF LSA database

You can also use IP route filters to select:

- OSPF routes to add to the RIB
- RIP or static routes to redistribute from the RIB into the OSPF LSA database

IP route filters affect the interaction between the routing module and the RIB, but IP route filters do not filter receipt of routing protocol messages by the routing module and do not directly filter messages sent from the routing protocol. Messages sent from the routing protocol are affected if and only if they are derived from the RIB, which is true in most situations, including RIP, OSPF-ext messages, and OSPF summary Link State Advertisements (LSAs). Note that the design of OSPF prevents route filters from filtering some types of OSPF LSAs (see "Limitations of route filtering on OSPF" on page 20-9).

Creating Filters

This section describes the commands, options and procedures for creating each of the different types of filter. It contains the following subsections:

- Creating Prefix Lists
- Creating IP Route Filters

Creating Prefix Lists

To create a prefix list and add entries to it, use the command:

```
add ip prefixlist=name entry=1..65535
    [action={match|nomatch}] [masklength=range] [prefix=ipadd]
```

The **masklength** parameter specifies the range of prefix mask lengths matched by this entry in the prefix list. The *range* is either a single CIDR mask from 0 to 32, or two masks separated by a hyphen. These options are valid for setting the mask length:

- As a mask length range (**masklength=*a*-*b*)**.

  For a route to match against this entry, its prefix mask length must be between *a* and *b* inclusive. *a* must be less than *b*.

- As a single mask length (**masklength=*a***).

  For a route to match against this entry, its prefix mask length must be exactly *a*.

- As an implicit mask length, by not specifying **masklength** (for example, **prefix=192.168.0.0**).

  For a route to match against this entry, its prefix mask length must correspond exactly to the mask for the class of the given address—in this example, 24.

Creating IP Route Filters

To create a route filter, use the command:

```
add ip route filter[=filter-id] ip=ipadd mask=ipadd
    action={include|exclude} [direction={receive|send|both}]
    [interface=interface] [nexthop=ipadd] [policy=0..7]
    [protocol={any|ospf|rip}]
```

The **protocol** parameter specifies the routing protocol to which the filter applies. When **direction** is **receive**, then **protocol** specifies the routing protocol that receives the route information. If **direction** is **send**, **protocol** specifies the routing protocol that advertises the routes.

When the routing protocol receives or transmits a route, it searches the list of route filters for a match to the route. The **ip**, **mask**, **interface**, **nexthop**, and **policy** parameters define a pattern to match against. The **action** parameter determines whether routes matching the pattern are used or discarded.

The switch checks each route against each filter, starting with the lowest-numbered filter, until it finds a match. Then it applies that filter and stops processing the list of filters.

When you create a list of filters—even a list of only one filter—the switch ends the list with an implicit filter to *exclude* all routes. So if you want the switch to include all routes that do not match your filters, end your filter list with a filter that matches all routes and includes them, such as:

```
add ip route filter=100 ip=*.*.*.* mask=*.*.*.*
   action=include
```

Applying Filters

This section describes how to apply the filters you have created, to achieve the following results:

- **Applying Filters when Writing to the RIB**
- **Applying Filters when Redistributing from the RIB**
- **Applying Filters before Advertising Routes**

Applying Filters when Writing to the RIB

When the switch receives information about a route, it normally adds that route to its RIB. This makes the route available for the switch to use. You can use route filters to stop the switch from adding certain routes—or routes with certain characteristics—into the RIB. This gives you control over the routes packets take when they leave the switch.

Filtering RIP Routes

To filter RIP routes before adding them to the RIB, simply create a filter or series of filters, using the command:

```
add ip route filter[=filter-id] ip=ipadd mask=ipadd
   action={include|exclude} protocol=rip direction=receive
   [other-options]
```

The switch automatically applies the filter when importing RIP routes, because **protocol=rip**.

The immediate effect of a route filter with **direction** set to **receive** and **action** set to **exclude** is that route advertisements received matching the filter do not result in a new entry in the local RIB. However, routes already in the RIB are not deleted even when they match the route filter. Therefore, if you dynamically add a route filter at the manager prompt, you may also need to manually delete unwanted routes from the RIB.

Invalid Filtering

You cannot filter in a way that excludes statically-configured or interface routes from the RIB.

Applying Filters when Redistributing from the RIB

The switch can import routes from the RIB into OSPF or RIP, even if it learned them from a different routing protocol or source. When you import routes from some route sources, you can also filter in order to block certain routes.

When Copying Routes to OSPF

OSPF can import the following:

- RIP routes
- non-interface routes
- statically-configured routes

When Copying Routes to RIP

RIP can import static and OSPF routes. It also automatically imports interface routes. The following table shows how to filter routes.

From	How to Filter
OSPF	1. Turn on exporting of OSPF routes into RIP, by using the command: **set ospf** rip=export [*other-options*] 2. Create IP route filters to determine which OSPF routes are copied into the LSA database, by using the command: **add ip route filter**[=*filter-id*] ip=*ipadd* mask=*ipadd* action={include|exclude} protocol=rip direction=send [*other-options*] The switch automatically applies the filter when importing routes into RIP, because **protocol=rip**.
Static	1. By default, RIP imports and advertises static routes. If this has been turned off, turn it on for the required interfaces by using the command: **set ip rip interface**=*interface* staticexport=yes [*other-options*] 2. Create IP route filters to determine which static routes are imported, by using the command: **add ip route filter**[=*filter-id*] ip=*ipadd* mask=*ipadd* action={include|exclude} protocol=rip direction=send [*other-options*] The switch automatically applies the filter when importing routes into RIP, because **protocol=rip**.
Interface	RIP automatically imports interface routes. Create IP route filters to determine which interface routes are imported, by using the command: **add ip route filter**[=*filter-id*] ip=*ipadd* mask=*ipadd* action={include|exclude} protocol=rip direction=send [*other-options*] The switch automatically applies the filter when importing routes into RIP, because **protocol=rip**.

Applying Filters before Advertising Routes

Routing protocols send their neighbours or peers information about the routes in the switch's RIB. You can use route filters to stop the switch from advertising certain routes or routes with certain characteristics. This gives you control over the routes that packets take through your network and when leaving your network.

When using OSPF

The design of the OSPF protocol does not allow you to filter LSAs before advertising them. This is because OSPF shares LSAs between all the routers in an area. The protocol assumes that all the routers in the area have shared all the advertisements among each other, and that all agree on the state of the complete link state database for the area. If some routers in the area are learning, but not advertising, that breaks the OSPF model.

Therefore, once a route is in the LSA database, you have no control over whether it is advertised.

When using RIP

To filter routes before advertising them with RIP, create a filter or series of filters, using the command:

```
add ip route filter[=filter-id] ip=ipadd mask=ipadd
   action={include|exclude} protocol=rip direction=send
   [other-options]
```

The switch automatically applies the filter when advertising routes to RIP neighbours, because **protocol=rip**.

When Advertising Static and Interface Routes

Statically-configured and interface routes do not have mechanisms to advertise routes. Only the routing protocols (OSPF and RIP) advertise routes. To advertise static and interface routes, with or without filtering, import the routes into the required routing protocol.

Filters for Route Sources

The previous sections describe each type of filter. This section has a series of diagrams that summarise the following filters that are available for each route source:

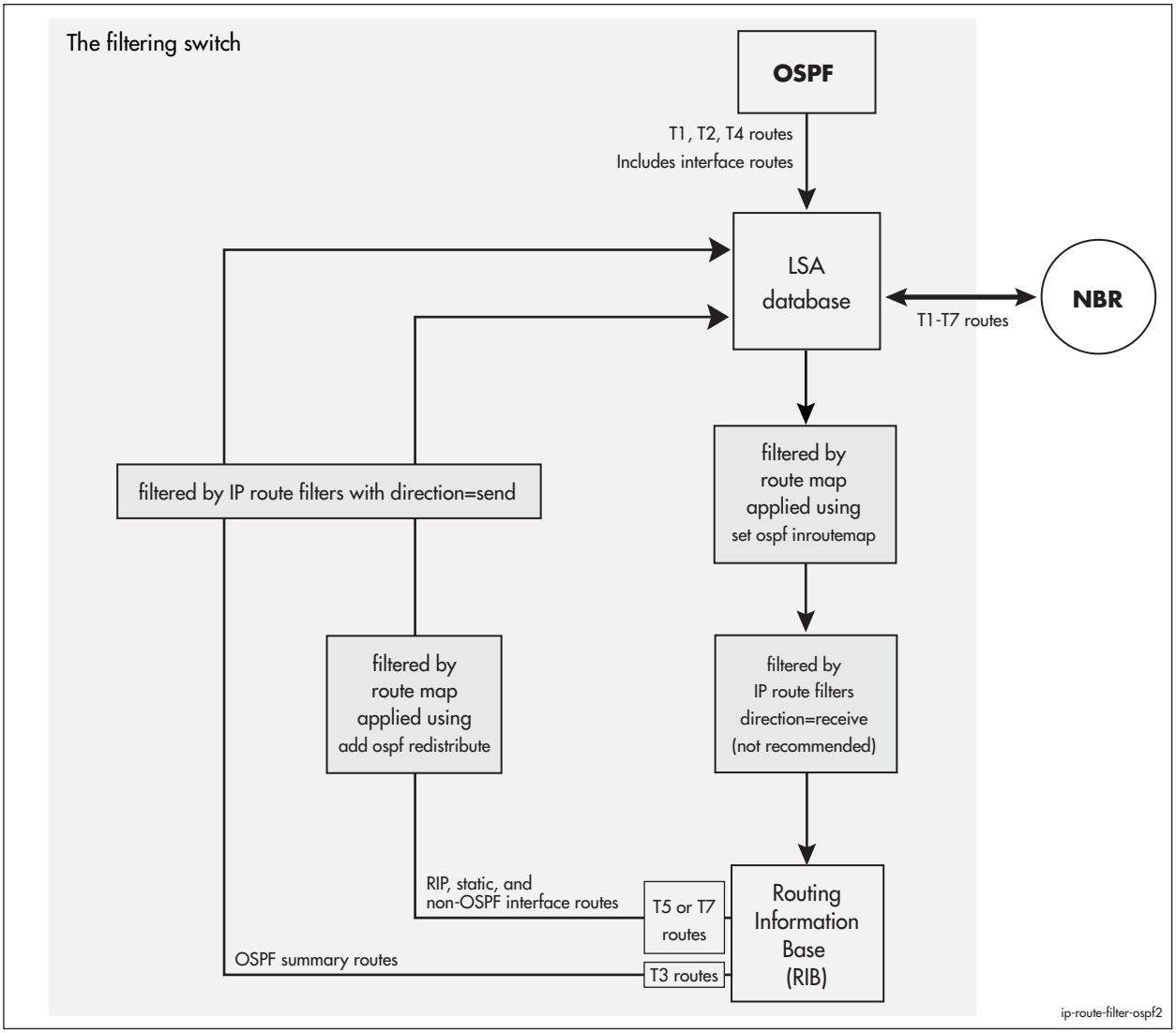
- **Open Shortest Path First (OSPF)**
- **Routing Information Protocol (RIP)**
- **Interface Routes**
- **Statically-Configured Routes**

Open Shortest Path First (OSPF)

When the switch runs OSPF, it receives routing information from neighbouring routers and advertises routing information to neighbouring routers. This routing information is contained in Link State Advertisements (LSAs). OSPF also generates LSAs internally. The following table describes types of LSA.

LSA Name	LSA describes	LSA is created
Type-1 Router-LSA	the state and cost of each of the switch's interfaces to the area	by OSPF, on every router in the area
Type-2 Network-LSA	all routers attached to the network	by OSPF, on the network's Designated Router
Type-3 Summary-LSA	inter-area destinations, when the destination is an IP network	from the RIB, by Area Border Routers
Type-4 Summary-LSA	inter-area destinations, when the destination is an Autonomous System (AS) boundary router	by OSPF, by Area Border Routers
Type-5 AS-external-LSA	a destination outside the AS	from the RIB, by AS boundary routers
Type-7 AS-external-LSA	a destination outside the AS. Used in not-so-stubby areas	from the RIB, by AS boundary routers

You can filter routing information at the processing points shown in the following figure. The figure also indicates the type of LSA at each processing point.



Limitations of route filtering on OSPF

As the previous diagram shows, the OSPF LSA database is a completely separate entity to the switch's RIB. The OSPF design does not allow you to filter the contents of the database before advertising routes to neighbouring routers. This is because OSPF shares LSAs between all the routers in an area. The protocol assumes that all the routers in the area have shared all the advertisements among each other, and that all agree on the state of the complete link state database for the area. If some routers in the area are learning, but not advertising, that breaks the OSPF model.

These limitations mean you can only filter to control the following:

- which routes learned by OSPF can be imported by the switch from the LSA database into the RIB
- which static or RIP routes can be exported from the RIB into the LSA database

- which summary routes can be exported from the RIB into the LSA database for advertising as summary LSAs

  You can use IP route filters to filter this (see "When Copying Routes to OSPF" on page 20-6)

  Another way to filter summary LSAs is to define a "do not advertise" OSPF range on an Area Border Router. This stops OSPF from advertising inter-area routes into another area. To do this, use the command:

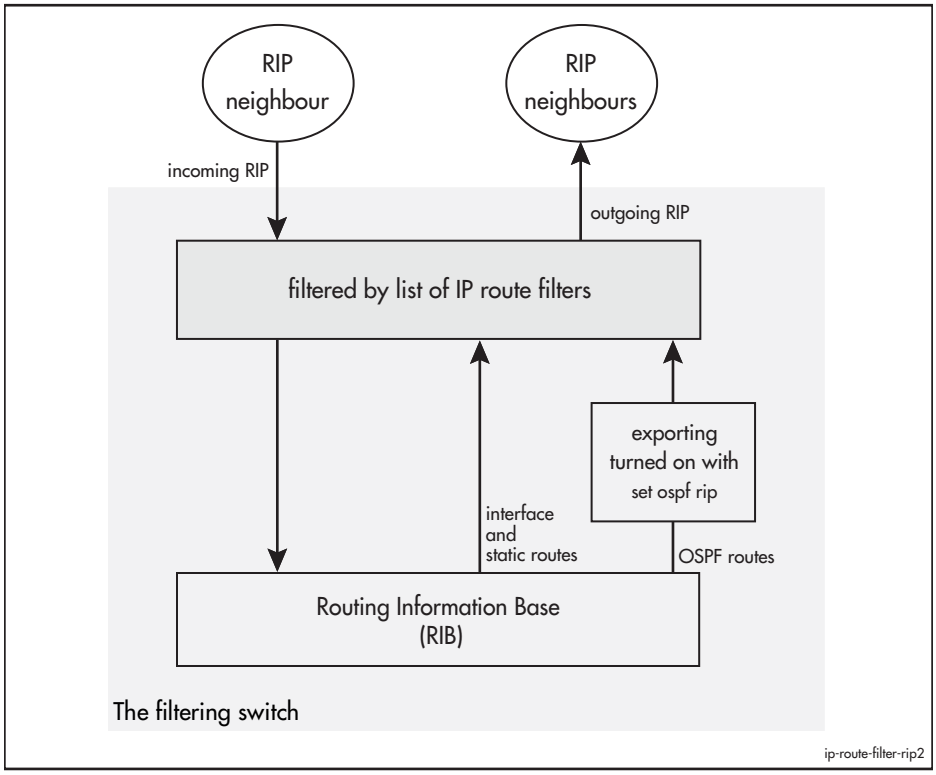
  ```
  set ospf range=ipadd effect=donotadvertise [other-options]
  ```

Note that filtering **cannot** do the following:

- remove an entry from the LSA database once the entry has been added
- prevent the switch from advertising an entry to interfaces in the same area that the entry is relevant to
- prevent updates that OSPF learns from being put into the database
- change the properties of an entry in the database

Routing Information Protocol (RIP)

When the switch runs RIP, it receives routing information from neighbouring routers, and can advertise RIP, statically-configured and interface routes to neighbouring routers. You can filter routing information at the processing points shown in the following figure.



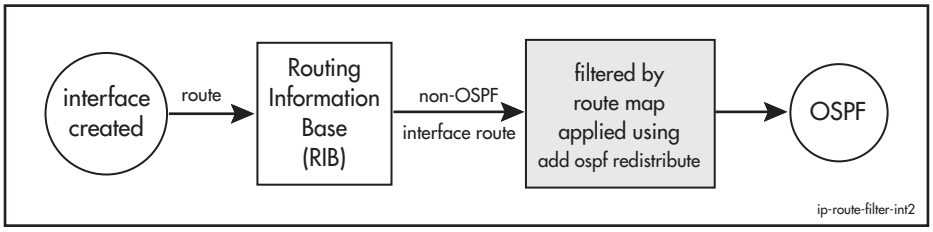
Interface Routes

When you create an interface on the switch, it automatically creates an interface route. This route tells the switch to send packets over that interface when the packets are addressed to the interface's subnet.

Filtering these routes before placing them in the RIB would be meaningless, so there is no mechanism to do so.

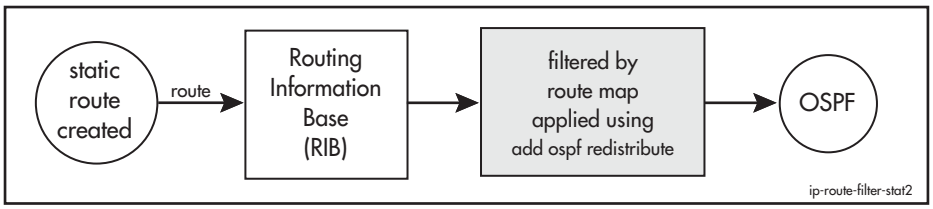
However, RIP and OSPF also use the interface routes. RIP uses all interface routes, and does not allow you to filter them. OSPF uses all routes that belong to OSPF interfaces, and does not allow you to filter them. OSPF does not automatically import non-OSPF interface routes, but you can choose to redistribute these into OSPF, with or without filtering.

The following diagram summarises how to filter interface routes.



Statically-Configured Routes

You can manually enter routing information into the switch, which creates static routes. Dynamic routing protocols import these routes. For OSPF, you can filter static routes when the protocol imports them, as shown in the following figure.



Command Reference

This section describes the commands available on the switch to configure IP route filtering.

The shortest valid command is denoted by capital letters in the Syntax section. See "Conventions" on page xxxviii of About this Software Reference in the front of this manual for details of the conventions used to describe command syntax. See Appendix A, Messages for a complete list of messages and their meanings.

add ip prefixlist

Syntax

```
ADD IP PREFIXList=name ENTry=1..65535
   [ACTion={MATch|NOMatch}] [MASklength=range]
   [PREfix=ipadd]
```

Description

This command adds a numbered *entry* to a prefix list. If the prefix list does not already exist, this command first creates it. You can create up to 400 prefix lists, with up to 1000 entries in each list.

Parameter	Description
PREFIXList	A name to identify the prefix list. A string 1 to 15 characters long. Valid characters are uppercase letters (A-Z), lowercase letters (a-z), digits (0-9) and the underscore. If *name* contains spaces, it must be in double quotes. Default: no default
ENTry	An integer to specify the position of the new entry in the prefix list. When the switch uses a prefix list, it checks the entries in order, starting with the lowest, until it finds a match. Therefore, give more specific entries lower numbers than general entries. If you leave gaps between entry numbers, you can add future entries between existing entries. Each prefix list has an implicit final entry that matches all addresses, with an action of **nomatch**. Default: no default
ACTion	Whether matching prefixes are included or excluded by the process that is using the prefix list. You can use multiple entries in a prefix list with actions of **match** and **nomatch** to build up a list of prefixes. Prefixes with **action=match** are included in the list. Default: **match**

MATch	The prefix list includes the prefix.
NOMatch	The prefix list excludes the prefix.

Parameter	Description (Continued)
MASklength	The range of prefix mask lengths matched by this entry in the prefix list. The *range* is either a single CIDR mask from 0 to 32, or two masks separated by a hyphen. These options are valid for setting the mask length: as a mask length range (**masklength=*a*-*b*)**. For a route to match against this entry, its prefix mask length must be between *a* and *b* inclusive. *a* must be less than *b*. as a single mask length (**masklength=*a***). For a route to match against this entry, its prefix mask length must be exactly *a*. as an implicit mask length, by not specifying **masklength** (for example, **prefix=192.168.0.0**). For a route to match against this entry, its prefix mask length must correspond exactly to the mask for the class of the given address; in this example, 24. Default: The natural mask for the prefix, based on whether it is a class A, B, or C network
PREfix	The network address matched by this entry in the prefix list, specified in dotted decimal notation. If you do not specify a prefix, the switch sets it to 0.0.0.0. This is correct if you are matching all routes or the default route. Default: 0.0.0.0

Examples

To match only routes from the 192.168.0.0/16 network, use the command:

```
add ip prefixlist=sample1 entry=1 action=match
   prefix=192.168.0.0 masklength=16
```

To match all routes in all 192.168.0.0 networks, except those in the 192.168.7.0 network, use the commands:

```
add ip prefixlist=sample2 entry=1 action=nomatch
   prefix=192.168.7.0 masklength=24-32

add ip prefixlist=sample2 entry=2 action=match
   prefix=192.168.0.0 masklength=16-32
```

To exclude the default route, use the command:

```
add ip prefixlist=sample3 entry=1 action=nomatch masklength=0
```

To include all routes, use the command:

```
add ip prefixlist=sample4 entry=1 action=match
   masklength=0-32
```

Related Commands

delete ip prefixlist
set ip prefixlist
show ip prefixlist

add ip route filter

Syntax

```
ADD IP ROUte FILter[=filter-id] IP=ipadd MASK=ipadd
   ACtion={INCLude|EXCLude|SWItch}
   [DIrection={RECeive|SENd|BOTH}] [INTerface=interface]
   [NEXThop=ipadd] [POLIcy=0..7] [PROTocol={ANY|OSPF|RIP}]
```

Description This command creates a route filter. A route filter can control which routes RIP receives and advertises, and which external routes OSPF copies into its LSA database.

Parameter	Description	
FILter	The ID number of the filter, from 1 to 100. Default: no default (the filter is added to the end of the list of currently-defined filters)	
IP	The network address to match. You can use the wildcard character ("*") to match a network range. For example, 192.168.*.* matches all destination networks that start with 192.168. The wildcard character can only replace a complete number. For example, 192.168.*.* is valid but 192.16*.*.* is not. Default: no default	
MASK	The network mask of the network to match. You can use the wildcard character ("*") to match a network mask range. For example, 255.255.*.* matches all destination network masks that start with 255.255. The wildcard character can only replace a complete number. For example, 255.255.*.* is valid but 255.25*.*.* is not. Default: no default	
ACtion	What the switch does with routes that match the filter. Default: no default	
	INCLude	The switch includes matching routes in its RIB or the advertisement.
	EXCLude	The switch excludes matching routes from its RIB or the advertisement.
	SWItch	The switch learns matching routes and adds them to the special default IP route table in hardware. The default IP route table can contain up to 16 summary routes.

Parameter	Description (Continued)	
DIrection	Whether the switch applies this filter to routes that the routing protocol receives or routes that it advertises. The routing protocol is specified using the **protocol** parameter. Default: **both**	
	RECeive	The switch applies this filter to routes that the routing protocol receives, to determine whether to write those routes into the RIB. If **protocol=ospf**, a route filter with **direction=receive** filters routes when copying them from the LSA database to the RIB. If a filter excludes a matching route from the RIB, OSPF does not advertise a summary LSA for that route because summary LSA messages are derived from the filtered RIB. This means that incorrect filters can prevent Area Border Routers from advertising routes to other areas. Plan your filters carefully.
	SENd	The switch applies this filter to routes, to determine whether the routing protocol will advertise the routes. If **protocol=ospf**, a route filter with **direction=send** only matches AS external routes (RIP and static routes) and summary routes.
	BOTH	The switch applies this filter to determine which routes to write into the RIB and which routes to advertise.
INTerface	The interface to which the filter applies. The switch only uses this filter on routes that are received on this interface, or that will be advertised out this interface. Valid interfaces are: VLAN (such as vlan1, vlan1-1) To see a list of interfaces currently available, use the **show ip interface** command on page 13-154 of Chapter 13, Internet Protocol (IP). If a logical interface is specified, the route filter is only applied to the specified logical interface. If a logical interface is not specified, 0 is assumed and the route filter is only applied to logical interface 0. If **protocol=ospf**, this parameter has no effect. The switch always applies the filter on all interfaces. Default: no default (the switch applies this filter to routes on all interfaces)	
NEXThop	The IP address of the next hop router. If you specify this, the switch applies this filter to routes that specify this next hop. Default: no default	
POLIcy	The value of the route's Type of Service, from 0 to 7. The filter matches routes with this TOS setting. Default: no default	
PROTocol	The routing protocol to which the filter applies. If **direction** is **receive**, then **protocol** specifies the routing protocol that receives the route information. If **direction** is **send**, then **protocol** specifies the routing protocol that advertises the routes. Default: **any**	
	OSPF	Open Shortest Path First
	RIP	Routing Information Protocol
	ANY	Both RIP and OSPF

Examples To add a route filter that includes RIP-derived routes from all sources, use the command:

```
add ip rou fil=1 prot=rip ac=incl di=both ip=*.*.*.*
   mask=*.*.*.*
```

To exclude all routes received from the 10.0.0.0 network from the route table, but include all other received routes in the route table, use the commands:

```
add ip rou fil=1 ip=10.0.0.0 mask=255.0.0.0 ac=excl di=rec
add ip rou fil=2 ip=*.*.*.* mask=*.*.*.* ac=incl
```

The second filter is necessary to override the effect of the implicit "exclude all" following the last entry in a filter list.

Related Commands **delete ip route filter**
set ip route filter
show ip route filter

delete ip prefixlist

Syntax `DELete IP PREFIXList[=name] [ENTry=1..65535]`

Description This command deletes one of the following:

- an entry from a particular prefix list if you specify a name in the **prefixlist** parameter and an **entry** number
- a prefix list if you specify a name in the **prefixlist** parameter but do not specify an **entry** number
- all prefix lists if you do not specify a name in the **prefixlist** parameter or an **entry** number

Examples To delete entry 2 from the prefix list "office", use the command:

```
del ip prefixl=office entry=2
```

Related Commands add ip prefixlist
show ip prefixlist

delete ip route filter

Syntax `DELete IP ROUte FILter=1..100`

Description This command deletes a route filter. A route filter controls which routes are sent and received by the routing protocols.

The **filter** parameter specifies the index in the filter list of the filter to delete. The specified entry must exist.

Examples To delete route filter 3, use the command:

```
del rou fil=3
```

Related Commands add ip route filter
set ip route filter
show ip route filter

set ip prefixlist

Syntax

```
SET IP PREFIXList=name ENTry=1..65535
   [ACTion={MATch|NOMatch}] [MASklength=range]
   [PREfix=ipadd]
```

Description

This command modifies an existing entry in a prefix list.

Parameter	Description
PREFIXList	A name that identifies the prefix list. Default: no default
ENTry	An integer that specifies the position of the entry in the prefix list. Default: no default
ACTion	Whether matching prefixes are included or excluded by the process that is using the prefix list. You can use multiple entries in a prefix list with actions of **match** and **nomatch** to build up a list of prefixes. Prefixes with **action=match** are included in the list. Default: **match**
MATch	The prefix list includes the prefix.
NOMatch	The prefix list excludes the prefix.
MASklength	The range of prefix mask lengths matched by this entry in the prefix list. The *range* is either a single CIDR mask from 0 to 32, or two masks separated by a hyphen. These options are valid for setting the mask length: as a mask length range (**masklength=*a*-*b*)**. For a route to match against this entry, its prefix mask length must be between *a* and *b* inclusive. *a* must be less than *b*. as a single mask length (**masklength=*a***). For a route to match against this entry, its prefix mask length must be exactly *a*. as an implicit mask length, by not specifying **masklength** (for example, **prefix=192.168.0.0**). For a route to match against this entry, its prefix mask length must correspond exactly to the mask for the class of the given address; in this example, 24. Default: The natural mask for the prefix, based on whether it is a class A, B, or C network
PREfix	The network address matched by this entry in the prefix list, specified in dotted decimal notation. If you do not specify a prefix, the switch assumes the prefix is unchanged, and this entry's previously given prefix is used.

Examples

To modify entry 1 in prefix list sample1 so that it matches only routes from the 192.168.0.0/16 network, use the command:

```
set ip prefixlist=sample1 entry=1 action=match
   prefix=192.168.0.0 masklength=16
```

Related Commands

add ip prefixlist
delete ip prefixlist
show ip prefixlist

set ip route filter

Syntax

```
SET IP ROUte FILter=filter-id [IP=ipadd] [MASK=ipadd]
   [ACtion={INCLude|EXCLude|SWItch}]
   [DIrection={RECeive|SENd|BOTH}] [INTerface=interface]
   [NEXThop=ipadd] [POLIcy=0..7] [PROTocol={ANY|OSPF|RIP}]
```

Description

This command modifies a route filter. A route filter controls which routes are sent and received by the routing protocols. Route filters do not apply to static or interface routes.

Parameter	Description	
FILter	The ID number of the filter, from 1 to 100. Default: no default	
IP	The network address to match. You can use the wildcard character ("*") to match a network range. For example, 192.168.*.* matches all destination networks that start with 192.168. The wildcard character can only replace a complete number. For example, 192.168.*.* is valid but 192.16*.*.* is not. Default: no default	
MASK	The network mask of the network to match. You can use the wildcard character ("*") to match a network mask range. For example, 255.255.*.* matches all destination network masks that start with 255.255. The wildcard character can only replace a complete number. For example, 255.255.*.* is valid but 255.25*.*.* is not. Default: no default	
ACtion	What the switch does with routes that match the filter. Default: no default	
	INCLude	The switch includes matching routes in its RIB or the advertisement.
	EXCLude	The switch excludes matching routes from its RIB or the advertisement.
	SWItch	The switch learns matching routes and adds them to the special default IP route table in hardware. The default IP route table can contain up to 16 summary routes.
DIrection	Whether the switch applies this filter to routes that the routing protocol receives or routes that it advertises. The routing protocol is specified using the **protocol** parameter. Default: **both**	
	RECeive	The switch applies this filter to routes that the routing protocol receives, to determine whether to write those routes into the RIB.
	SENd	The switch applies this filter to routes, to determine whether the routing protocol will advertise the routes. Note that the nature of the OSPF protocol affects how route filtering works on OSPF Link State Advertisement (LSA). A route filter with **direction=send** only matches Autonomous System (AS) external routes. Also, the switch ignores the **interface** parameter, so it applies the filter on all interfaces.
	BOTH	The switch applies this filter to determine which routes to write into the RIB and which routes to advertise.

Parameter	Description (Continued)
INTerface	The interface to which the filter applies. The switch only uses this filter on routes that are received on this interface, or that will be advertised out this interface. Valid interfaces are: VLAN (such as vlan1, vlan1-1) To see a list of interfaces currently available, use the **show ip interface** command on page 13-154 of Chapter 13, Internet Protocol (IP). If a logical interface is specified, the route filter is only applied to the specified logical interface. If a logical interface is not specified, 0 is assumed and the route filter is only applied to logical interface 0. If **protocol=ospf**, the switch ignores this setting when filtering routes to advertise. Default: no default (the switch applies this filter to routes on all interfaces)
NEXThop	The IP address of the next hop router. If you specify this, the switch applies this filter to routes that specify this next hop. Default: no default
POLIcy	The value of the route's Type of Service, from 0 to 7. The filter matches routes with this TOS setting. Default: no default
PROTocol	The routing protocol to which the filter applies. If **direction** is **receive**, then **protocol** specifies the routing protocol that receives the route information. If **direction** is **send**, then **protocol** specifies the routing protocol that advertises the routes. Default: **any**
OSPF	Open Shortest Path First
RIP	Routing Information Protocol
ANY	Both RIP and OSPF

Examples

To modify route filter 1 to include only OSPF-derived routes, use the command:

```
set ip rou fil=1 prot=ospf
```

Related Commands

add ip route filter
delete ip route filter
show ip route filter

show ip prefixlist

Syntax `SHow IP PREFIXList[=name]`

Description This command displays information about prefix lists on the switch. If you specify a prefix list name, detailed information about that prefix list and its entries is displayed (Figure 20-2, Table 20-2). Otherwise, summary information about all existing prefix lists is displayed (Figure 20-1, Table 20-1).

Figure 20-1: Example summary output from the **show ip prefixlist** command

```
IP Prefix Lists
Name              Entries
---------------------------
Sample                 11
Test                    3
---------------------------
```

Table 20-1: Parameters in output of the **show ip prefixlist** command

Parameter	Meaning
Name	The name of the prefix list.
Entries	The number of entries in the prefix list.

Figure 20-2: Example detailed output from the **show ip prefixlist** command

```
IP Prefix List
-------------------------------------------------------------------------
 Name ................... Sample
 Entries:
   Number       Action         Prefix            Length Range
   ---------------------------------------------------------------------
        1       Match          192.168.0.0                 16
        3       No Match       0.0.0.0                  25-30
       10       No Match       10.10.10.0               24-30
   ---------------------------------------------------------------------
```

Table 20-2: Parameters in detailed output of the **show ip prefixlist** command

Parameter	Meaning
Name	Name of the prefix list.
Number	The entry number of the prefix list entry. The switch checks entries in order, starting with the lowest entry number.
Action	Whether the prefix list includes ("match") or excludes ("nomatch") any prefix that is within the entry's prefix range.
Prefix	IP network address for the entry to match on.
Length Range	Range of CIDR mask lengths that the entry can match on.

Examples To see the entries in prefix list "office", use the command:

```
sh ip prefixl=office
```

Related Commands **add ip prefixlist**
delete ip prefixlist

show ip route filter

Syntax `SHow IP ROUte FILter`

Description This command displays information about configured IP route filters (Figure 20-3, Table 20-3).

Figure 20-3: Example output from the **show ip route filter** command

```
IP Route Filters
-------------------------------------------------------------------------------
Ent.    IP Address        Mask              Nexthop           Policy      Matched
        Protocol          Direction         Interface         Action
-------------------------------------------------------------------------------
  1     0.0.0.0           0.0.0.0           Any               0                 0
        RIP               Both              -                 Include

        Request: 1               Passes: 1               Fails: 0
-------------------------------------------------------------------------------
```

Table 20-3: Parameters in output of the **show ip route filter** command

Parameter	Meaning
Ent	Filter number.
IP Address	IP address of the network that is filtered.
Mask	Network mask for the network address.
Nexthop	Next hop to which the filter applies.
Policy	Policy or type of service to which the filter applies.
Matched	Number of times this pattern has been matched.
Protocol	Routing protocol to which the filter applies.
Direction	Whether the filter applies to routes the switch receives, advertises, or both.
Interface	Interface to which the filter applies.
Action	Whether matching routes are included or excluded.

Related Commands

add ip route filter
delete ip route filter
set ip route filter