



BGP Named Community Lists

The BGP Named Community Lists feature allows the network operator to assign meaningful names to community lists and increases the number of community lists that can be configured.

- [Finding Feature Information, page 1](#)
- [Information About BGP Named Community Lists, page 1](#)
- [How to Use BGP Named Community Lists, page 2](#)
- [Configuration Examples for BGP Named Community Lists, page 12](#)
- [Additional References for BGP Named Community Lists, page 12](#)
- [Feature Information for BGP Named Community Lists, page 13](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and 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 feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About BGP Named Community Lists

BGP COMMUNITIES Attribute

A BGP community is a group of routes that share a common property, regardless of their network, autonomous system, or any physical boundaries. In large networks, applying a common routing policy by using prefix lists or access lists requires individual peer statements on each networking device. Using the BGP COMMUNITIES attribute, BGP speakers with common routing policies can implement inbound or outbound route filters based on the community tag, rather than consult long lists of individual permit or deny statements. A COMMUNITIES attribute can contain multiple communities.

A route can belong to multiple communities. The network administrator defines the communities to which a route belongs. By default, all routes belong to the general Internet community.

In addition to numbered communities, there are several predefined (well-known) communities:

- no-export—Do not advertise this route to external BGP peers.
- no-advertise—Do not advertise this route to any peer.
- internet—Advertise this route to the Internet community. All BGP-speaking networking devices belong to this community.
- local-as—Do not send this route outside the local autonomous system.
- gshut—Community of routes gracefully shut down.

The COMMUNITIES attribute is optional, which means that it will not be passed on by networking devices that do not understand communities. Networking devices that understand communities must be configured to handle the communities or else the COMMUNITIES attribute will be discarded. By default, no COMMUNITIES attribute is sent to a neighbor. In order for a COMMUNITIES attribute to be sent to a neighbor, use the **neighbor send-community** command.

BGP Community Lists

A BGP community list is used to create groups of communities to use in a match clause of a route map. A community list can be used to control which routes are accepted, preferred, distributed, or advertised, for example. You can also use a community list to set, append, or modify the communities of a route.

- Standard community lists are used to specify well-known communities and community numbers.
- Expanded community lists are used to filter communities using a regular expression. Regular expressions are used to specify patterns to match COMMUNITIES attributes.

A BGP named community list allows you to assign a meaningful name to a community list. A named community list can be configured using community numbers, well-known communities, or regular expressions. All the rules of numbered community lists apply to named community lists, except that there is no limit on the number of named community lists that can be configured.

**Note**

A maximum of 100 standard community lists and 100 expanded community lists can be configured. A named community list does not have this limitation.

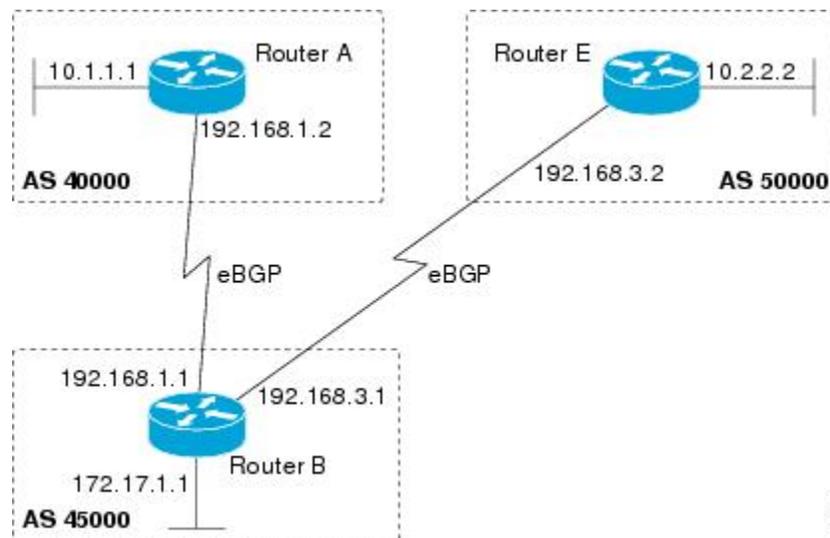
How to Use BGP Named Community Lists

Filtering Traffic Using Community Lists

Perform this task to filter traffic by creating a BGP community list, referencing the community list within a route map, and then applying the route map to a neighbor.

In this task, Router B in the figure below is configured with route maps and a community list to control incoming routes.

Figure 1: Topology for Which a Community List Is Configured



SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *autonomous-system-number*
4. **neighbor** {*ip-address* | *peer-group-name*} **remote-as** *autonomous-system-number*
5. **address-family ipv4** [**unicast** | **multicast** | **vrf** *vrf-name*]
6. **neighbor** {*ip-address* | *peer-group-name*} **route-map** *route-map-name* {**in** | **out**}
7. **exit**
8. **exit**
9. **route-map** *map-name* [**permit** | **deny**] [*sequence-number*]
10. **match community** {*standard-list-number* | *expanded-list-number* | *community-list-name* [**exact**]}
11. **set weight** *weight*
12. **exit**
13. **route-map** *map-name* [**permit** | **deny**] [*sequence-number*]
14. **match community** {*standard-list-number* | *expanded-list-number* | *community-list-name* [**exact**]}
15. **set community** *community-number*
16. **exit**
17. **ip community-list** {*standard-list-number* | **standard** *list-name* {**deny** | **permit**} [*community-number*] [*AA:NN*] [**internet**] [**local-AS**] [**no-advertise**] [**no-export**]} | {*expanded-list-number* | **expanded** *list-name* {**deny** | **permit**} *regular-expression*}
18. Repeat Step 17 to create all the required community lists.
19. **exit**
20. **show ip community-list** [*standard-list-number* | *expanded-list-number* | *community-list-name*] [**exact-match**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>router bgp <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Device(config)# router bgp 45000</pre>	Enters router configuration mode for the specified routing process.
Step 4	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} remote-as <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Device(config-router)# neighbor 192.168.3.2 remote-as 50000</pre>	Adds the IP address or peer group name of the neighbor to the specified autonomous system BGP neighbor table of the local router.
Step 5	<p>address-family ipv4 [unicast multicast vrf <i>vrf-name</i>]</p> <p>Example:</p> <pre>Device(config-router)# address-family ipv4 unicast</pre>	<p>Specifies the IPv4 address family and enters address family configuration mode.</p> <ul style="list-style-type: none"> The unicast keyword specifies the IPv4 unicast address family. By default, the router is placed in address family configuration mode for the IPv4 unicast address family if the unicast keyword is not specified with the address-family ipv4 command. The multicast keyword specifies IPv4 multicast address prefixes. The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent IPv4 address family configuration mode commands.
Step 6	<p>neighbor {<i>ip-address</i> <i>peer-group-name</i>} route-map <i>route-map-name</i> {in out}</p> <p>Example:</p> <pre>Device(config-router-af)# neighbor 192.168.3.2 route-map 2000 in</pre>	<p>Applies a route map to inbound or outbound routes.</p> <ul style="list-style-type: none"> In this example, the route map called 2000 is applied to inbound routes from the BGP peer at 192.168.3.2.
Step 7	<p>exit</p> <p>Example:</p> <pre>Device(config-router-af)# exit</pre>	Exits address family configuration mode and enters router configuration mode.
Step 8	<p>exit</p> <p>Example:</p> <pre>Device(config-router)# exit</pre>	Exits router configuration mode and enters global configuration mode.
Step 9	<p>route-map <i>map-name</i> [permit deny] [<i>sequence-number</i>]</p>	<p>Creates a route map and enters route map configuration mode.</p> <ul style="list-style-type: none"> In this example, the route map called 2000 is defined.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# route-map 2000 permit 10</pre>	
Step 10	<p>match community {<i>standard-list-number</i> <i>expanded-list-number</i> <i>community-list-name</i> [exact]}</p> <p>Example:</p> <pre>Device(config-route-map)# match community 1</pre>	<p>Matches on the communities in a BGP community list.</p> <ul style="list-style-type: none"> In this example, the route's community attribute is matched to communities in community list 1.
Step 11	<p>set weight <i>weight</i></p> <p>Example:</p> <pre>Device(config-route-map)# set weight 30</pre>	<p>Sets the weight of BGP routes that match the community list.</p> <ul style="list-style-type: none"> In this example, any route that matches community list 1 will have its weight set to 30.
Step 12	<p>exit</p> <p>Example:</p> <pre>Device(config-route-map)# exit</pre>	<p>Exits route map configuration mode and enters global configuration mode.</p>
Step 13	<p>route-map <i>map-name</i> [permit deny] [<i>sequence-number</i>]</p> <p>Example:</p> <pre>Device(config)# route-map 3000 permit 10</pre>	<p>Creates a route map and enters route map configuration mode.</p> <ul style="list-style-type: none"> In this example, the route map called 3000 is defined.
Step 14	<p>match community {<i>standard-list-number</i> <i>expanded-list-number</i> <i>community-list-name</i> [exact]}</p> <p>Example:</p> <pre>Device(config-route-map)# match community 2</pre>	<p>Matches on the communities in a BGP community list.</p> <ul style="list-style-type: none"> In this example, the route's COMMUNITIES attribute is matched to communities in community list 2.
Step 15	<p>set community <i>community-number</i></p> <p>Example:</p> <pre>Device(config-route-map)# set community 99</pre>	<p>Sets the BGP communities attribute.</p> <ul style="list-style-type: none"> In this example, any route that matches community list 2 will have the COMMUNITIES attribute set to 99.
Step 16	<p>exit</p> <p>Example:</p> <pre>Device(config-route-map)# exit</pre>	<p>Exits route map configuration mode and enters global configuration mode.</p>
Step 17	<p>ip community-list {<i>standard-list-number</i> standard <i>list-name</i> {deny permit} [<i>community-number</i>]</p>	<p>Creates a community list for BGP and controls access to it.</p>

	Command or Action	Purpose
	<p>[AA:NN] [internet] [local-AS] [no-advertise] [no-export]} {expanded-list-number expanded list-name {deny permit} regular-expression}</p> <p>Example:</p> <pre>Device(config)# ip community-list 1 permit 100</pre> <p>Example:</p> <pre>Device(config)# ip community-list 2 permit internet</pre>	<ul style="list-style-type: none"> In the first example, community list 1 permits routes with a COMMUNITIES attribute of 100. Router E routes all have a COMMUNITIES attribute of 100, so their weight will be set to 30. In the second example, community list 2 effectively permits all routes by specifying the internet community. Any routes that did not match community list 1 are checked against community list 2. All routes are permitted, but no changes are made to the route attributes. <p>Note Two examples are shown here because the task example requires both of these statements to be configured.</p>
Step 18	Repeat Step 17 to create all the required community lists.	—
Step 19	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	Exits global configuration mode and enters privileged EXEC mode.
Step 20	<p>show ip community-list [standard-list-number expanded-list-number community-list-name] [exact-match]</p> <p>Example:</p> <pre>Device# show ip community-list 1</pre>	Displays configured BGP community list entries.

Examples

The following sample output verifies that community list 1 has been created and it permits routes that have a community attribute of 100:

```
Device# show ip community-list 1
Community standard list 1
  permit 100
```

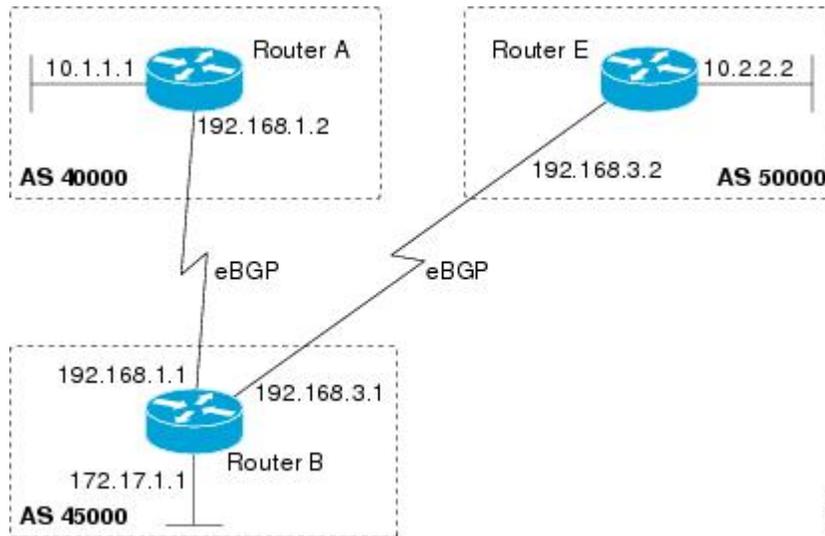
The following sample output verifies that community list 2 has been created and it effectively permits all routes by specifying the **internet** community:

```
Device# show ip community-list 2
Community standard list 2
  permit internet
```

Filtering Traffic Using Extended Community Lists

Perform this task to filter traffic by creating an extended BGP community list to control outbound routes.

Figure 2: Topology for Which a Community List Is Configured



In this task, Router B in the figure above is configured with an extended named community list to specify that the BGP peer at 192.168.1.2 is not sent advertisements about any path through or from autonomous system 50000. The IP extended community-list configuration mode is used and the ability to resequence entries is shown.



Note

A sequence number is applied to all extended community list entries by default, regardless of the configuration mode. Explicit sequencing and resequencing of extended community list entries can be configured only in IP extended community-list configuration mode, not in global configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip extcommunity-list** {*expanded-list-number* | **expanded** *list-name* | *standard-list-number* | **standard** *list-name*}
4. [*sequence-number*] {**deny** [*regular-expression*] | **exit** | **permit** [*regular-expression*]}
5. Repeat Step 4 for all the required permit or deny entries in the extended community list.
6. **resequence** [*starting-sequence*] [*sequence-increment*]
7. **exit**
8. **router bgp** *autonomous-system-number*
9. **neighbor** {*ip-address* | *peer-group-name*} **remote-as** *autonomous-system-number*
10. Repeat the prior step for all of the required BGP peers.
11. **address-family ipv4** [**unicast** | **multicast** | **vrf** *vrf-name*]
12. **network** *network-number* [**mask** *network-mask*]
13. **end**
14. **show ip extcommunity-list** [*list-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip extcommunity-list { <i>expanded-list-number</i> expanded <i>list-name</i> <i>standard-list-number</i> standard <i>list-name</i> }	Enters IP extended community-list configuration mode to create or configure an extended community list. <ul style="list-style-type: none"> • In this example, the expanded community list DENY50000 is created.
Step 4	[<i>sequence-number</i>] { deny [<i>regular-expression</i>] exit permit [<i>regular-expression</i>]}	Configures an expanded community list entry. <ul style="list-style-type: none"> • In the first example, an expanded community list entry with the sequence number 10 is configured to deny advertisements about paths from autonomous system 50000.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-extcomm-list)# 10 deny _50000_</pre> <p>Example:</p> <pre>Device(config-extcomm-list)# 20 deny ^50000 .*</pre>	<ul style="list-style-type: none"> In the second example, an expanded community list entry with the sequence number 20 is configured to deny advertisements about paths through autonomous system 50000. <p>Note Two examples are shown here because the task example requires both these statements to be configured.</p> <p>Note Only the syntax applicable to this task is used in this example. For more details, see the <i>Cisco IOS IP Routing: BGP Command Reference</i>.</p>
Step 5	Repeat Step 4 for all the required permit or deny entries in the extended community list.	—
Step 6	<p>resequence <i>[starting-sequence]</i> <i>[sequence-increment]</i></p> <p>Example:</p> <pre>Device(config-extcomm-list)# resequence 50 100</pre>	<p>Resequences expanded community list entries.</p> <ul style="list-style-type: none"> In this example, the sequence number of the first expanded community list entry is set to 50 and subsequent entries are set to increment by 100. The second expanded community list entry is therefore set to 150. <p>Note Only the syntax applicable to this task is used in this example. For more details, see the <i>Cisco IOS IP Routing: BGP Command Reference</i>.</p>
Step 7	<p>exit</p> <p>Example:</p> <pre>Device(config-extcomm-list)# exit</pre>	Exits expanded community-list configuration mode and enters global configuration mode.
Step 8	<p>router bgp <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Device(config)# router bgp 45000</pre>	Enters router configuration mode for the specified routing process.
Step 9	<p>neighbor <i>{ip-address peer-group-name}</i> remote-as <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Device(config-router)# neighbor 192.168.3.2 remote-as 50000</pre>	Adds the IP address or peer group name of the neighbor to the specified autonomous system BGP neighbor table of the local router.
Step 10	Repeat the prior step for all of the required BGP peers.	—
Step 11	address-family ipv4 [unicast multicast vrf <i>vrf-name</i>]	Specifies the IPv4 address family and enters address family configuration mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-router)# address-family ipv4 unicast</pre>	<ul style="list-style-type: none"> The unicast keyword specifies the IPv4 unicast address family. By default, the router is placed in address family configuration mode for the IPv4 unicast address family if the unicast keyword is not specified in the address-family ipv4 command. The multicast keyword specifies IPv4 multicast address prefixes. <p>Note The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent IPv4 address family configuration mode commands.</p>
Step 12	<p>network <i>network-number</i> [mask <i>network-mask</i>]</p> <p>Example:</p> <pre>Device(config-router-af)# network 172.17.1.0 mask 255.255.255.0</pre>	<p>(Optional) Specifies a network as local to this autonomous system and adds it to the BGP routing table.</p> <ul style="list-style-type: none"> For exterior protocols, the network command controls which networks are advertised. Interior protocols use the network command to determine where to send updates. <p>Note Only the syntax applicable to this task is used in this example. For more details, see the <i>Cisco IOS IP Routing: BGP Command Reference</i>.</p>
Step 13	<p>end</p> <p>Example:</p> <pre>Device(config-router-af)# end</pre>	<p>Exits address family configuration mode and enters privileged EXEC mode.</p>
Step 14	<p>show ip extcommunity-list [<i>list-name</i>]</p> <p>Example:</p> <pre>Device# show ip extcommunity-list DENY50000</pre>	<p>Displays configured BGP expanded community list entries.</p>

Examples

The following sample output verifies that the BGP expanded community list DENY50000 has been created, with the output showing that the entries to deny advertisements about autonomous system 50000 have been resequenced from 10 and 20 to 50 and 150:

```
Device# show ip extcommunity-list DENY50000

Expanded extended community-list DENY50000
 50 deny _50000_
150 deny ^50000 .*
```

Configuration Examples for BGP Named Community Lists

Example: Filtering Traffic Using COMMUNITIES Attributes

This section contains two examples of the use of BGP COMMUNITIES attributes with route maps.

The first example configures a route map named *set-community*, which is applied to the outbound updates to the neighbor 172.16.232.50. The routes that pass access list 1 are given the well-known COMMUNITIES attribute value **no-export**. The remaining routes are advertised normally. The **no-export** community value automatically prevents the advertisement of those routes by the BGP speakers in autonomous system 200.

```
router bgp 100
 neighbor 172.16.232.50 remote-as 200
 neighbor 172.16.232.50 send-community
 neighbor 172.16.232.50 route-map set-community out
!
route-map set-community permit 10
 match address 1
  set community no-export
!
route-map set-community permit 20
 match address 2
```

The second example configures a route map named *set-community*, which is applied to the outbound updates to neighbor 172.16.232.90. All the routes that originate from autonomous system 70 have the COMMUNITIES attribute values 200 200 added to their already existing communities. All other routes are advertised as normal.

```
route-map bgp 200
 neighbor 172.16.232.90 remote-as 100
 neighbor 172.16.232.90 send-community
 neighbor 172.16.232.90 route-map set-community out
!
route-map set-community permit 10
 match as-path 1
  set community 200 200 additive
!
route-map set-community permit 20
!
ip as-path access-list 1 permit 70$
ip as-path access-list 2 permit .*
```

Additional References for BGP Named Community Lists

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
BGP commands	Cisco IOS IP Routing: BGP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 1997	<i>BGP Communities Attribute</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for BGP Named Community Lists

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for BGP Named Community Lists

Feature Name	Releases	Feature Information
BGP Named Community Lists		The BGP Named Community Lists feature introduces a new type of community list called the named community list. The BGP Named Community Lists feature allows the network operator to assign meaningful names to community lists and increases the number of community lists that can be configured. A named community list can be configured with regular expressions and with numbered community lists. All rules of numbered communities apply to named community lists except that there is no limitation on the number of community attributes that can be configured for a named community list.