BGP Flowspec

This chapter details BGP Flowspec, a routing feature enabling dynamic traffic filtering, policing, and automated threat mitigation, especially against DDoS attacks. It covers configuring client and server roles, verifying operations, and implementing advanced traffic redirection from global VRF to L3VPN or segment routing policies.

BGP flowspec

The BGP flowspec is a routing feature that

- dynamically distributes traffic filtering and policing rules
- enables granular control over network traffic, and
- automates threat mitigation across BGP-speaking routers.

You use BGP flowspec primarily to quickly and automatically respond to network threats, especially Distributed Denial-of-Service (DDoS) attacks. This feature allows you to deploy filtering rules rapidly across many routers, stopping attack traffic closer to its source. It provides granular control over traffic, letting you define precise matching criteria and actions. By automating rule deployment through BGP updates, you reduce manual configuration effort and ensure consistent policy enforcement across your network.

Table 1: Feature History Table

Feature Name	Release Information	Feature Description

BGP Flowspec	Release 24.4.1	Introduced in this release on: Fixed Systems (8700) (select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*) You can now rapidly deploy and propagate filtering and policing functionality across many BGP peer routers, which helps to mitigate the effects of a distributed denial-of-service (DDoS) attack on your network. This feature allows you to create detailed instructions for matching specific traffic flows based on various parameters (for example, IP addresses, ports, and packet specifics) and to define actions (such as dropping, policing, or redirecting the traffic) through BGP updates. This helps in effectively managing and mitigating unwanted traffic. *This functionality is now supported on: • 8712-MOD-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM • 8212-48FH-M • 8711-32FH-M

Table 2: Feature History Table

Feature Name	Release Information	Feature Description
Scaling BGP Flowspec to 6000 Rules	Release 7.5.2	You can now assign 6000 BGP Flowspec rules for Cisco 8800 series routers and 3000 BGP Flowspec rules for Cisco 8100 and 8200 series routers. This feature thus provide enhanced mitigation against Distributed Denial-of-Service (DDoS) attacks. In earlier releases, you could assign 2000 BGP Flowspec rules. These are one dimensional scale numbers; the numbers vary based on other intersecting features like AccessList (ACL), Quality of Service (QoS), and Local Path Transport Switching (LPTS).

How BGP flowspec client-server controller models work

The BGP flowspec model operates with a controller (often a server) and clients (BGP speakers). The controller originates and injects the flowspec NLRI entries, while the clients receive these entries and program their hardware accordingly.

The BGP flowspec model operates with a controller (often a server) and clients (BGP speakers). The controller originates and injects the Flowspec NLRI entries, while the clients receive these entries and program their

hardware accordingly. This model is often visualized with the controller on the left-hand side (refer to the figure: BGP flowspec Controller) and the client on the right-hand side (refer to the figure: BGP flowspec Client), illustrating the flow of information.

Summary

The BGP Flowspec process describes how a controller defines and distributes traffic management rules to client routers, which then enforce these rules on network traffic. This ensures dynamic and granular control over data flows.

Workflow

Figure 1: BGP flowspec client

Figure 2: BGP flowspec controller

These stages describe the BGP flowspec client-server controller model:

1. Rule construction:
    - Actor: The controller
    - Action: The controller defines specific traffic flow rules. These rules include both detailed matching criteria (what traffic to identify) and the actions the network takes (what to do with that traffic). The controller is typically configured using commands to provide these entries for NLRI injection.
2. BGP propagation:
    - Actor: The controller
    - Action: The controller encodes these rules. It uses BGP NLRI for the matching criteria and BGP extended communities for the actions.
3.
    - Actor: The BGP flowspec-enabled controller
    - Action: The controller originates these rules and advertises them to its BGP peers, which function as clients or speakers.
4. Local enforcement:
    - Actor: The receiving BGP Flowspec Client routers
    - Action: The client routers install these rules into their local forwarding plane. The client receives the information, sends it to its internal flowspec manager, and configures the enhanced Policy-based Routing (ePBR) infrastructure, which in turn programs the underlying hardware.
5. Traffic processing:
    - Actor: The client router's active Layer 3 interfaces.
    - Action: Once installed and programmed in the applicable line cards, the interfaces start processing ingress traffic. They apply the specified actions to any traffic flows that match the defined rules.

Result

This process results in network devices actively identifying and applying predefined actions (such as filtering, policing, or redirection) to specific traffic flows in real-time, based on the rules distributed via BGP.

Restrictions for BGP flowspec

These are the specific restrictions for configuring of BGP flowspec. You should be aware of them when deploying and managing the BGP flowspec:

- Flowspec statistics are supported only when a policer rate limit is configured.
- The policer action scale is limited to a maximum of 128 per slice.
- Statistics for the Redirect action are supported only if a policer is attached; statistics are not supported for Redirect action alone.
- Redirects from a VRF to the default VRF are not supported.

Understanding these limitations helps you design and operate Flowspec policies effectively and avoid unsupported configurations.

Configure BGP flowspec on the client

This section provides configuration examples for a scenario where a BGP flowspec controller (server) with IP address 10.2.3.4 sends flowspec NLRI to a client with IP address 10.2.3.3. The NLRI contains matching criteria, and the Client processes traffic based on these criteria. Traffic is then dropped or accepted based on the configured criteria.

Before you begin

You must enable and configure the Border Gateway Protocol (BGP) routing process on both the client and server routers.

This task describes how you configure a BGP Flowspec client to receive and process flowspec NLRI from a BGP flowspec server.

Procedure

Step 1 Define a Virtual Routing and Forwarding (VRF) instance named vrf1 and set up import and export route targets for different address families.

Example:

```
Router(config)# router bgp 140
Router(config-bgp)# vrf vrf1
Router(config-bgp-vrf)# address-family ipv4 unicast
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit
```

```
Router(config-bgp-vrf)# address-family ipv4 flowspec
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit

Router(config-bgp-vrf)# address-family ipv6 unicast
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit

Router(config-bgp-vrf)# address-family ipv6 flowspec
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit

Router(config-bgp-vrf)# address-family vpnv4 flowspec
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit

Router(config-bgp-vrf)# address-family vpnv6 flowspec
Router(config-bgp-vrf-af)# import route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# export route-target
Router(config-bgp-vrf-af)# 101:2000
Router(config-bgp-vrf-af)# 201:2000
Router(config-bgp-vrf-af)# exit
```

Step 2 Configure BGP flowspec for various address families to allow local installation of rules on all interfaces.

Example:

```
Router(config)# flowspec
Router(config-flowspec)# address-family ipv4
Router(config-flowspec-af)# local-install interface-all
Router(config-flowspec-af)# exit
Router(config-flowspec)# address-family ipv6
Router(config-flowspec-af)# local-install interface-all
Router(config-flowspec-af)# exit
Router(config-flowspec)# address-family vpnv4
Router(config-flowspec-af)# local-install interface-all
Router(config-flowspec-af)# exit
Router(config-flowspec)# address-family vpnv6
Router(config-flowspec-af)# local-install interface-all
Router(config-flowspec-af)# exit
```

Step 3 Configure route policies to accept all routes (pass-all) and to reject all routes (drop-all).

Example:

```
Router(config)# route-policy pass-all
Router(config)# pass
Router(config)# end-policy

Router(config)# route-policy drop-all
Router(config)# drop
Router(config)# end-policy
```

Step 4 Configure the BGP process and define the neighbor relationship with the Flowspec server (10.2.3.4), applying the defined route policies for inbound and outbound Flowspec NLRI.

Example:

```
Router(config)# router bgp 1
Router(config-bgp)# nsr
Router(config-bgp)# bgp router-id 10.2.3.3
Router(config-bgp)# address-family ipv4 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# address-family ipv6 flowspec
Router(config-bgp-af)# exit

Router(config-bgp)# address-family vpnv4 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# address-family vpnv6 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# neighbor 10.2.3.4
Router(config-bgp-nbr)# remote-as 1
Router(config-bgp-nbr)# address-family ipv4 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy drop-all out
Router(config-bgp-af)# exit
Router(config-bgp-nbr)# address-family ipv6 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy drop-all out
Router(config-bgp-nbr-af)# exit

Router(config-bgp-nbr)# address-family vpnv4 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy drop-all out
Router(config-bgp-af)# exit
Router(config-bgp-nbr)# address-family vpnv6 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy drop-all out
Router(config-bgp-af)# exit

Router(config-bgp-nbr)# update-source Loopback0
```

Step 5 Disable BGP flowspec.

Example:

```
Router(config)# interface bundle-ether 3.1
Router(config-subif)# ipv4 flowspec disable
Router(config-subif)# ipv6 flowspec disable
```

This configuration establishes a VRF instance named `vrf1` within BGP AS 140, defining import and export route targets for IPv4, IPv6, and VPN address families, enabling its participation in an MPLS VPN environment. Additionally, it configures the router as a BGP flowspec client that locally installs received flowspec rules on

all interfaces, accepting rules from neighbor 10.2.3.4 (a flowspec server) while preventing the advertisement of its own flowspec rules.

Configure BGP flowspec on the server

This task describes how you configure a BGP Flowspec server to define and advertise Flowspec NLRI to a BGP flowspec client.

-

Before you begin

-

Procedure

Step 1 Configure route policies to accept all routes (pass-all) and to reject all routes (drop-all).

Example:

```
Router(config)# route-policy pass-all
Router(config)# pass
Router(config)# end-policy

Router(config)# route-policy drop-all
Router(config)# drop
Router(config)# end-policy
```

Step 2 Configure the BGP process and define the neighbor relationship with the flowspec client (10.2.3.3), applying the defined route policies for inbound and outbound flowspec NLRI.

Example:

```
Router(config)# router bgp 1
Router(config-bgp)# nsr
Router(config-bgp)# bgp router-id 10.2.3.4
Router(config-bgp)# address-family ipv4 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# address-family ipv6 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# address-family vpnv4 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# address-family vpnv6 flowspec
Router(config-bgp-af)# exit
Router(config-bgp)# neighbor 10.2.3.3
Router(config-bgp-nbr)# remote-as 1
Router(config-bgp-nbr)# address-family ipv4 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# exit
Router(config-bgp-nbr)# address-family ipv6 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# exit
Router(config-bgp-nbr)# address-family vpnv4 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# exit
Router(config-bgp-nbr)# address-family vpnv6 flowspec
Router(config-bgp-nbr-af)# route-policy pass-all in
```

```
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# exit
Router(config-bgp-nbr)# update-source Loopback0
```

Step 3 Define IPv4 traffic classes and associate them with a policy-map to specify actions.

Example:

```
Router(config)# class-map type traffic match-all ipv4_fragment
Router(config-cmap)# match destination-address ipv4 10.2.1.1 255.255.255.255
Router(config-cmap)# match source-address ipv4 172.16.0.1 255.255.255.255
Router(config-cmap)# end-class-map
Router(config)# class-map type traffic match-all ipv4_icmp
Router(config-cmap)# match destination-address ipv4 10.2.1.1 255.255.255.255
Router(config-cmap)# match source-address ipv4 172.16.0.1 255.255.255.255
Router(config-cmap)# end-class-map

Router(config)# policy-map type pbr scale_ipv4
Router(config-pmap)# class type traffic ipv4_fragment
Router(config-pmap-c)# drop
Router(config-pmap-c)# exit
Router(config-pmap)# class type traffic ipv4_icmp
Router(config-pmap-c)# exit
Router(config-pmap)# class type traffic class-default
Router(config-pmap-c)# end-policy-map
Router(config)# flowspec
Router(config)# address-family ipv4
Router(config-af)# service-policy type pbr scale_ipv4
Router(config-af)# exit
Router(config)# vrf vpn1
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# service-policy type pbr scale_ipv4
Router(config-vrf-af)# exit
Router(config-vrf)# exit
```

Step 4 Define IPv6 traffic classes and associate them with a policy-map for service policies.

Example:

```
Router(config)# flowspec
Router(config)# address-family ipv6
Router(config-af)# service-policy type pbr scale_ipv6
Router(config-af)# exit
Router(config)# vrf vpn1
Router(config-vrf)# address-family ipv6
Router(config-vrf-af)# service-policy type pbr scale_ipv6
Router(config-vrf-af)# exit
Router(config-vrf)# exit

Router(config)# class-map type traffic match-all ipv6_tcp
Router(config-cmap)# match destination-address ipv6 70:1:1::5a/128
Router(config-cmap)# match source-address ipv4 ipv6 80:1:1::5a/128
Router(config-cmap)# match destination-port 22
Router(config-cmap)# match source-port 4000
Router(config-cmap)# end-class-map
Router(config)# class-map type traffic match-all ipv6_icmp
Router(config-cmap)# match destination-address ipv6 70:2:1::1/128
Router(config-cmap)# match source-address ipv4 ipv6 80:2:1::1/128
Router(config-cmap)# end-class-map

Router(config)# policy-map type pbr scale_ipv6
Router(config-pmap)# class type traffic ipv6_tcp
Router(config-pmap-c)# exit
Router(config-pmap)# class type traffic ipv6_icmp
Router(config-pmap-c)# exit
```

```
Router(config-pmap)# class type traffic class-default
Router(config-pmap-c)# end-policy-map
Router(config)# flowspec
Router(config)# address-family ipv6
Router(config-af)# service-policy type pbr scale_ipv6
```

Step 5 Define a class map to match traffic based on DSCP values and a policy map to redirect IPv4 traffic to a next-hop with a specific police rate.

Example:

```
Router(config)# class-map type traffic match-all class_dscp_5
Router(config-cmap)# match destination-address ipv4 192.0.2.254 255.255.255.0
Router(config-cmap)# match dscp 10-12

Router(config-pmap)# class type traffic class_dscp_5
Router(config-pmap-c)# redirect ipv4 nexthop 10.26.245.2
Router(config-pmap-c)# police rate 5 mbps
Router(config-pmap-c)# root
```

This configuration establishes the router as a BGP flowspec server, allowing it to fully exchange flowspec rules for various traffic types with its client (10.2.3.3). It then defines specific traffic patterns for IPv4 and IPv6, applying actions such as dropping, redirecting, or rate-limiting these flows based on received rules, both globally and within a VPN.

Verify running configuration for BGP flowspec

The purpose of this running configuration is to verify a functional BGP flowspec client-server deployment. This setup enables a centralized server (Controller) to dynamically define, propagate, and enforce granular traffic filtering and policing rules across a client router, thereby providing a robust mechanism for network-wide traffic management and DDoS mitigation.

This configuration details the setup for two routers: a BGP Flowspec client (router ID 10.2.3.3) and a BGP Flowspec server (router ID 10.2.3.4), both within AS 1.

Procedure

Running configuration.

Example:

```
/* Define a Virtual Routing and Forwarding (VRF) instance named vrf1 and set up
import and export route targets for different address families. */
router bgp 140
 vrf vrf1
  address-family ipv4 unicast
   import route-target
    101:2000
    201:2000
   export route-target
    101:2000
    201:2000
   !
   address-family ipv4 flowspec
    import route-target
```

```
    101:2000
    201:2000
    export route-target
    101:2000
    201:2000
   !
   address-family ipv6 flowspec
    import route-target
    101:2000
    201:2000
    export route-target
    101:2000
    201:2000
   !
    address-family vpnv4 flowspec
    import route-target
    101:2000
    201:2000
    export route-target
    101:2000
    201:2000
   !
    address-family vpnv6 flowspec
     import route-target
     101:2000
     201:2000
     export route-target
     101:2000
     201:2000
    !

/* Configure BGP Flowspec both on the server and client side. */
flowspec
 address-family ipv4
  local-install interface-all
  !
  address-family ipv6
  local-install interface-all
  !
  address-family vpnv4
   local-install interface-all
  !
  address-family vpnv6
  local-install interface-all
  !

/* Configure the policy to accept all presented routes without modifying the routes. */
route-policy pass-all
 pass
 end-policy

/* Configure the policy to reject all presented routes without modifying the routes */
 route-policy drop-all
  drop
  end-policy

/* Configure BGP towards flowspec server */
router bgp 1
 nsr
  bgp router-id 10.2.3.3
   address-family ipv4 flowspec
   !
   address-family ipv6 flowspec
   !
```

```
   address-family vpnv4 flowspec
   !
   address-family vpnv6 flowspec
   !
    neighbor 10.2.3.4
     remote-as 1
      address-family ipv4 flowspec
       route-policy pass-all in
       route-policy drop-all out
       !
      address-family ipv6 flowspec
       route-policy pass-all in
       route-policy drop-all out
       !
      address-family vpnv4 flowspec
       route-policy pass-all in
       route-policy drop-all out
       !
       address-family vpnv6 flowspec
       route-policy pass-all in
       route-policy drop-all out
       !
       update-source Loopback0

/* Disable BGP Flowspec */
interface bundle-ether 3.1
 ipv4 flowspec disable
 ipv6 flowspec disable
/* The following section describes how you can configure BGP Flowspec on the server: */
/* Configure the policy to accept all presented routes without modifying the routes. */

route-policy pass-all
 pass
 end-policy

/* Configure the policy to reject all presented routes without modifying the routes */

route-policy drop-all
 drop
end-policy

/* Configure BGP towards flowspec client */
router bgp 1
 nsr
  bgp router-id 10.2.3.4
   address-family ipv4 flowspec
   !
   address-family ipv6 flowspec
   !
   address-family vpnv4 flowspec
   !
   address-family vpnv6 flowspec
   !
   neighbor 10.2.3.3
    remote-as 1
     address-family ipv4 flowspec
      route-policy pass-all in
      route-policy pass-all out
      !
     address-family ipv6 flowspec
      route-policy pass-all in
      route-policy pass-all out
        !
        address-family vpnv4 flowspec
```

```
         route-policy pass-all in
         route-policy pass-all out
        !
        address-family vpnv6 flowspec
         route-policy pass-all in
         route-policy pass-all out
         !
        update-source Loopback0


/* Configure IPv4 flowspec to be advertised to client. Define traffic classes. */
class-map type traffic match-all ipv4_fragment
 match destination-address ipv4 10.2.1.1 255.255.255.255
 match source-address ipv4 172.16.0.1 255.255.255.255end-class-map

class-map type traffic match-all ipv4_icmp
 match destination-address ipv4 10.2.1.1 255.255.255.255
 match source-address ipv4 172.16.0.1 255.255.255.255
 end-class-map

/* Define a policy map and associate it with traffic classes. */
policy-map type pbr scale_ipv4
 class type traffic ipv4_fragment
  drop
  !
  class type traffic ipv4_icmp
  !
  class type traffic class-default
  end-policy-map

flowspec
 address-family ipv4
  service-policy type pbr scale_ipv4
  !
  vrf vpn1
   address-family ipv4
    service-policy type pbr scale_ipv4
     !
      !

flowspec
 address-family ipv6
  service-policy type pbr scale_ipv6
  !
   vrf vpn1
    address-family ipv6
     service-policy type pbr scale_ipv6
      !
       !

/* Configure IPv6 flowspec to be advertised to client. Define traffic classes. */
class-map type traffic match-all ipv6_tcp
    match destination-address ipv6 70:1:1::5a/128
    match source-address ipv4 ipv6 80:1:1::5a/128
    match destination-port 22
    match source-port 4000
    end-class-map

class-map type traffic match-all ipv6_icmp
    match destination-address ipv6 70:2:1::1/128
    match source-address ipv4 ipv6 80:2:1::1/128
    end-class-map

/* Define a policy map and associate it with traffic classes. */
```

```
policy-map type pbr scale_ipv6
 class type traffic ipv6_tcp
  !
 class type traffic ipv6_icmp
  !
 class type traffic class-default
   end-policy-map

/* Class map configuration with DSCP. */
flowspec
 address-family ipv6
  service-policy type pbr scale_ipv6
class-map type traffic match-all class_dscp_5
 match destination-address ipv4 192.0.2.254 255.255.255.0
 match dscp 10-12

/* Policy map configuration with IPv4 Redirect and Rate Limiter */
 class type traffic class_dscp_5
  redirect ipv4 nexthop 10.26.245.2
   police rate 5 mbps
     root
```

Verify flowspec flow information, statistics, and NLRI data, and related policy maps

Verify the information about BGP Flowspec routes, including their attributes, status, and operational metrics. These outputs help users monitor and verify the distribution and application of Flowspec rules within the BGP routing environment, facilitating effective traffic filtering and mitigation of network threats.

Procedure

Step 1 Use the **show flowspec ipv4 detail** command to verify the Flowspec flow information and traffic statistics for IPv4.

Example:

```
Router# show flowspec ipv4 detail
Thu Jan 25 09:10:14.965 UTC

AFI: IPv4
  Flow           :Dest:10.0.0.1/8
    Actions      :Traffic-rate: 5000000 bps Redirect: VRF vpn1 Route-target: ASN2-1:1  (bgp.1)
    Statistics                        (packets/bytes)
      Matched             :                 200/25600
      Transmitted         :                 200/25600
      Dropped             :                   0/0
```

Step 2 Use the **show flowspec ipv6 detail** command to verify the Flowspec flow information and traffic statistics for IPv6.

Example:

```
Router# show flowspec ipv6 detail
AFI: IPv6
Flow:Dest:70:1:1::1/0-128,Source:80:1:1::1/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12

 Actions :Traffic-rate: 1000000 bps DSCP: cs1 Nexthop: 202:158:2::1 (bgp.1)
 Statistics (packets/bytes)
```

```
    Matched         : 64091597/19483845488
    Transmitted     : 33973978/10328089312
    Dropped         : 30117619/9155756176
```

Step 3 Use the **show flowspec vrf customer_1 ipv4 detail** command to verify the Flowspec flow information and statistics for IPv4 within the specified VRF.

Example:

```
show flowspec vrf customer_1 ipv4 detail
VRF: customer_1 AFI: IPv4
Flow :Dest:202.158.3.2/32,Source:202.158.1.2/32
Actions :Traffic-rate: 250000000 bps DSCP: cs6 Redirect: VRF dirty_dancing
Route-target: ASN2-4787:666 (bgp.1)
Statistics (packets/bytes)
Matched : 37260786850/4098686553500
Transmitted : 21304093027/2343450232970
Dropped : 15956693823/1755236320530
```

Step 4 Use the **show flowspec vrf customer_1 ipv6 detail** command to verify the Flowspec flow information and statistics for IPv6 within the specified VRF.

Example:

```
Router# show flowspec vrf customer_1 ipv6 detail
VRF: customer_1 AFI: IPv6
Flow
:Dest:200:158:3::2/0-128,Source:200:158:1::2/0-128,NH:=6,DPort:=22,SPort:=4000,Length:=300,DSCP:=12
Actions :Traffic-rate: 250000000 bps DSCP: cs6 Redirect: VRF dirty_dancing
Route-target: ASN2-4787:666 (bgp.1)
Statistics (packets/bytes)
Matched : 16130480136/4903665961344
Transmitted : 8490755776/2581189755904
Dropped : 7639724360/2322476205440
```

Step 5 Use the **show flowspec ipv4 nlri** command to verify the NLRI in hex format for IPv4 Flowspec routes.

Example:

```
Router# show flowspec ipv4 nlri
AFI: IPv4
NLRI (hex) :0x0120460101010381060581500691 0bb80a81c80b810a
Actions :Traffic-rate: 0 bps (bgp.1)
```

Step 6 Use the **show flowspec ipv6 nlri** command to verify the NLRI in hex format for IPv6 Flowspec routes.

Example:

```
Router# show flowspec ipv6 nlri
AFI: IPv6
NLRI (hex)
:0x018000007000010001000000000000000000010280000080000100010000000000000000000010381060581160691 0fa00981100a91012c0b810c
Actions :Traffic-rate: 1000000 bps DSCP: cs1 Nexthop: 202:158:2::1 (bgp.1)
```

Step 7 Use the **show flowspec vrf customer_1 ipv4 nlri** command to verify the NLRI in hex format for IPv4 Flowspec routes within the specified VRF.

Example:

```
Router# show flowspec vrf customer_1 ipv4 nlri
VRF: customer_1 AFI: IPv4
NLRI (hex) :0x0120ca9e03020220ca9e0102
Actions :Traffic-rate: 250000000 bps DSCP: cs6 Redirect: VRF dirty_dancing
Route-target: ASN2-4787:666 (bgp.1)
```

Step 8 Use the **show flowspec vrf customer_1 ipv6 nlri** command to verify the NLRI in hex format for IPv6 Flowspec routes within the specified VRF.

Example:

```
Router# show flowspec vrf customer_1 ipv6 nlri
VRF: customer_1 AFI: IPv6
NLRI (hex)
:0x0180000200015800030000000000000000000202800002000158000100000000000000000002038106058116069 10fa00a91012c0b810c
Actions :Traffic-rate: 250000000 bps DSCP: cs6 Redirect: VRF dirty_dancing
Route-target: ASN2-4787:666 (bgp.1)
```

Step 9 Use the **show policy-map transient type pbr** command to verify the policy-map dynamically created for Flowspec-based traffic policies.

Example:

```
Router# show policy-map transient type pbr
policy-map type pbr __bgpfs_default_IPv4
handle:0x36000004
table description: L3 IPv4 and IPv6
class handle:0x760013eb sequence 1024
match destination-address ipv4 10.1.1.1 255.255.255.255
match protocol tcp
match destination-port 80
match source-port 3000
```

The show outputs provides the user with the visibility into the Flowspec environment, facilitating validation and maintenance of security policies within the BGP routing framework.

Verify the BGP flowspec on the client

Verify the correct reception, installation, and propagation of BGP Flowspec routes for IPv4, IPv6, VPNv4, and VPNv6 address families. Confirm that Flowspec information exchanges properly with BGP peers.

Before you begin

Ensure that you have appropriate user privileges to execute privileged EXEC mode commands.

Follow these steps to verify the BGP flowspec on the client.

Procedure

Step 1 Use the **show bgp ipv4 flowspec** command to verify BGP Flowspec routes for the IPv4 address family.

Example:

```
Router# show bgp ipv4 flowspec
GP router identifier 202.158.0.1, local AS number 4787
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0 RD version: 7506
BGP main routing table version 7506
BGP NSR Initial initsync version 130 (Reached)
BGP NSR/ISSU Sync-Group versions 7506/0
BGP scan interval 60 secs
Status codes: s suppressed, d damped, h history, * valid, > best
```

```
i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
Network Next Hop Metric LocPrf Weight Path
*>iDest:10.1.1.1/32,Proto:=6,DPort:=80,SPort:=3000,Length:=200,DSCP:=10/176
0.0.0.0 10 0 ?
*>iDest:10.1.1.2/32,Proto:=6,DPort:=80,SPort:=3000,Length:=200,DSCP:=10/176
0.0.0.0 10 0 ?
*>iDest:10.1.1.3/32,Proto:=6,DPort:=80,SPort:=3000,Length:=200,DSCP:=10/176
0.0.0.0 10 0 ?
*>iDest:10.1.1.4/32,Proto:=6,DPort:=80,SPort:=3000,Length:=200,DSCP:=10/176
0.0.0.0 10 0 ?
*>iDest:10.1.1.5/32,Proto:=6,DPort:=80,SPort:=3000,Length:=200,DSCP:=10/176
0.0.0.0 10 0 ?
```

Step 2 Use the **show bgp ipv6 flowspec** command to verify BGP Flowspec routes for the IPv6 address family.

Example:

```
Router# show bgp ipv6 flowspec
BGP router identifier 202.158.0.1, local AS number 4787
...
Network Next Hop Metric LocPrf Weight Path
*>iDest:70:1:1::1/0-128,Source:80:1:1::1/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12/464
202:158:2::1 100 0 i
*>iDest:70:1:1::2/0-128,Source:80:1:1::2/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12/464
202:158:2::1 100 0 i
*>iDest:70:1:1::3/0-128,Source:80:1:1::3/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12/464
202:158:2::1 100 0 i
*>iDest:70:1:1::4/0-128,Source:80:1:1::4/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12/464
202:158:2::1 100 0 i
*>iDest:70:1:1::5/0-128,Source:80:1:1::5/0-128,NH:=6,DPort:=22,SPort:=4000,TCPFlags:=0x10,Length:=300,DSCP:=12/464
202:158:2::1 100 0 i
```

Step 3 Use the **show bgp vpnv4 flowspec** command to verify BGP Flowspec routes for VPNv4 address families.

Example:

```
Router# show bgp vpnv4 flowspec
BGP router identifier 202.158.0.1, local AS number 4787
...
Route Distinguisher: 202.158.0.1:0 (default for vrf customer_1)
*>iDest:202.158.3.2/32,Source:202.158.1.2/32/96
0.0.0.0 100 0 i
Route Distinguisher: 202.158.0.2:1
*>iDest:202.158.3.2/32,Source:202.158.1.2/32/96
0.0.0.0 100 0 i
Processed 2 prefixes, 2 paths
```

Step 4 Use the **show bgp vpnv6 flowspec** command to verify BGP Flowspec routes for VPNv6 address families.

Example:

```
Router# show bgp vpnv6 flowspec

BGP router identifier 202.158.0.1, local AS number 4787
...
Route Distinguisher: 202.158.0.1:0 (default for vrf customer_1)
*>iDest:200:158:3::2/0-128,Source:200:158:1::2/0-128,NH:=6,DPort:=22,SPort:=4000,Length:=300,DSCP:=12/440
0.0.0.0 100 0 i
Route Distinguisher: 202.158.0.2:1
*>iDest:200:158:3::2/0-128,Source:200:158:1::2/0-128,NH:=6,DPort:=22,SPort:=4000,Length:=300,DSCP:=12/440
0.0.0.0 100 0 i
Processed 2 prefixes, 2 paths
```

Step 5 Use the **show bgp ipv6 flowspec summary** command to view a summary of BGP Flowspec information for IPv6.

Example:

```
Router# show bgp ipv6 flowspec summary
BGP router identifier 202.158.0.1, local AS number 4787
...
Neighbor Spk AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down St/PfxRcd
202.158.2.1 0 4787 1548 1648 1504 0 0 1d01h 750 <-- this
many flowspecs were received from server
202.158.3.1 0 4787 1683 1644 1504 0 0 1d01h 751
202.158.4.1 0 4787 1543 1649 1504 0 0 1d01h 0
```

Step 6 Use the **show bgp vpnv4 flowspec summary** command to view a summary of BGP Flowspec information for VPNv4.

Example:

```
Router# show bgp vpnv4 flowspec summary

BGP router identifier 202.158.0.1, local AS number 4787
...
Neighbor Spk AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down St/PfxRcd
02.158.2.1 0 4787 1549 1648 5 0 0 1d01h 1 <-- this
many flowspecs were received from server
202.158.3.1 0 4787 1684 1644 5 0 0 1d01h 0
202.158.4.1 0 4787 1543 1649 5 0 0 1d01h 0
```

Step 7 Use the **show bgp vpnv6 flowspec summary** command to view a summary of BGP Flowspec information for VPNv6.

Example:

```
Router# show bgp vpnv6 flowspec summary
BGP router identifier 202.158.0.1, local AS number 4787
...
Neighbor Spk AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down St/PfxRcd
202.158.2.1 0 4787 1549 1649 5 0 0 1d01h 1 <-- this
many flowspecs were received from server
202.158.3.1 0 4787 1684 1645 5 0 0 1d01h 0
202.158.4.1 0 4787 1543 1650 5 0 0 1d01h 0
```

Step 8 Use the **show flowspec vrf all afi-all summary** command to view a summary of Flowspec routes and service policies across all VRFs and address families.

Example:

```
Router# show flowspec vrf all afi-all summary
Flowspec VRF+AFI table summary:
VRF: default
  AFI: IPv4
    Total Flows:              1
    Total Service Policies:   1
VRF: default
  AFI: IPv6
    Total Flows:              0
    Total Service Policies:   0
```

The show outputs confirm that BGP Flowspec routes for IPv4, IPv6, VPNv4, and VPNv6 address families are present and match the expected destinations and policies. Additionally, the summary commands show that Flowspec routes are being successfully received from specific BGP neighbors, verifying correct propagation and peering.

BGP flowspec redirect from global VRF to L3VPN and segment routing policy

The BGP flowspec redirect from Global VRF to L3VPN and Segment Routing policy is a BGP routting feature that

- dynamically redirect traffic to the VRF table
- enable traffic to search for the destination IP address within the L3VPN or via a segment routing policy, and
- improve routing adaptability and service continuity.

This feature also allows you to execute precise traffic actions, which optimize network performance and security.

When traffic arrives with a destination IP address not found in the global routing table but present in a VRF routing table, this feature ensure the packet is redirected correctly to the customer VRF. This redirection enhances routing flexibility and maintains service continuity by leveraging L3VPN or segment routing policies for forwarding decisions.

This capability helps you manage complex network environments by applying fine-grained traffic control and improving overall network efficiency.

Table 3: Feature History Table

Feature Name	Release Name	Description
BGP flowspec redirect from global VRF to L3VPN and segment routing policy	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.
BGP flowspec redirect from global VRF to L3VPN and segment routing policy	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only); Modular Systems (8800 [LC ASIC: P100])(select variants only*) *This feature is supported on: • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM

Feature Name	Release Name	Description
BGP flowspec redirect from global VRF to L3VPN and segment routing policy	Release 24.2.11	You can now enhance network routing efficiency by enabling BGP Flowspec to dynamically redirect traffic to the VRF table, where the traffic searches for the destination IP address either within the L3VPN or via a segment routing policy. This improvement boosts routing adaptability and service continuity. Additionally, the protocol extension equips you to execute precise traffic actions, optimizing network performance and security.

How BGP flowspec redirect from global VRF to L3VPN and segment routing policy works

This process redirects packets arriving on a global VRF interface with destination IPs found only in a customer VRF. Without this redirect, packets are dropped. The BGP Flowspec server programs and sends redirect rules via BGP NLRI to neighbors, which store and activate them. Matching packets are then forwarded to the correct VRF using the specified route target through L3VPN or Segment Routing Policy. This ensures accurate routing and precise traffic control in complex networks.

Summary

The key components involved in this process are:

- BGP Flowspec server: Programs and distributes redirect rules using BGP NLRI to neighbors.
- BGP Flowspec neighbors (clients): Store and activate the redirect rules in their databases.
- Network interfaces (for example, VRFA): Receive incoming packets that require VRF-specific routing.
- Routing tables (global VRF and customer VRF): Contain destination IP information used for packet forwarding.
- L3VPN and Segment Routing Policy (SR-Policy): Mechanisms used to forward packets to the correct VRF instance.

Workflow

Figure 3: Forwarding based on SR-Policy

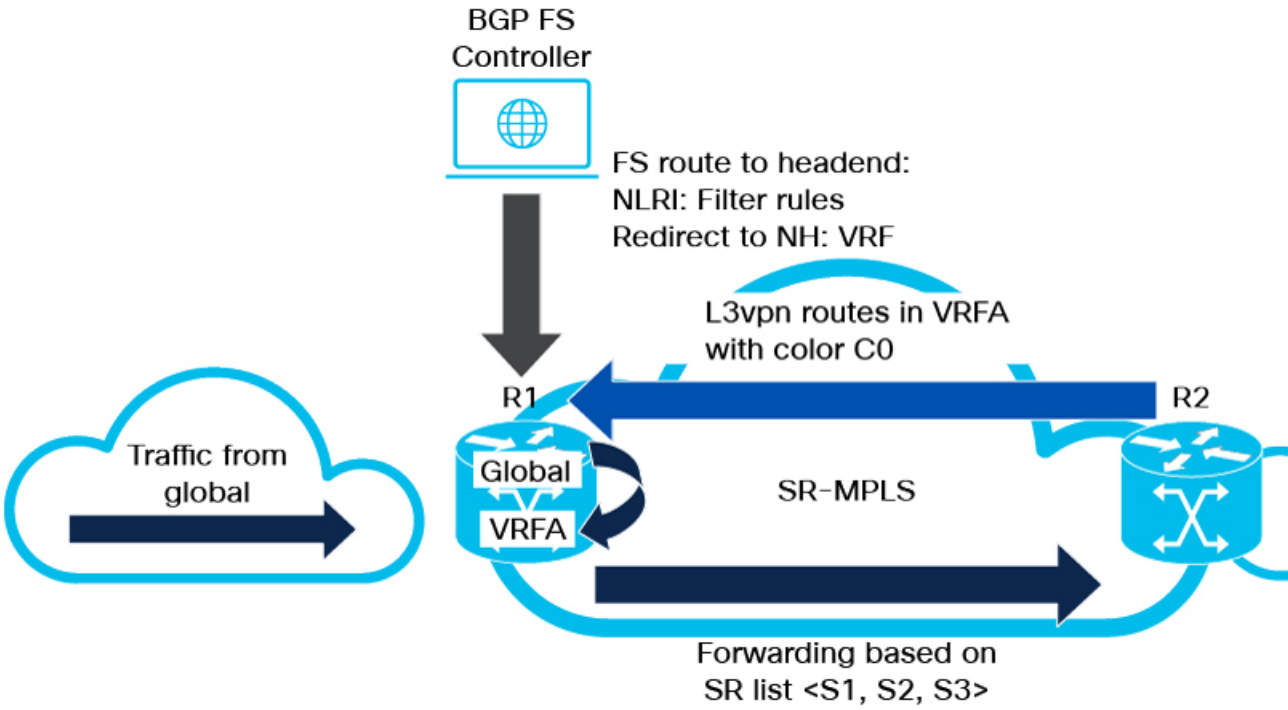
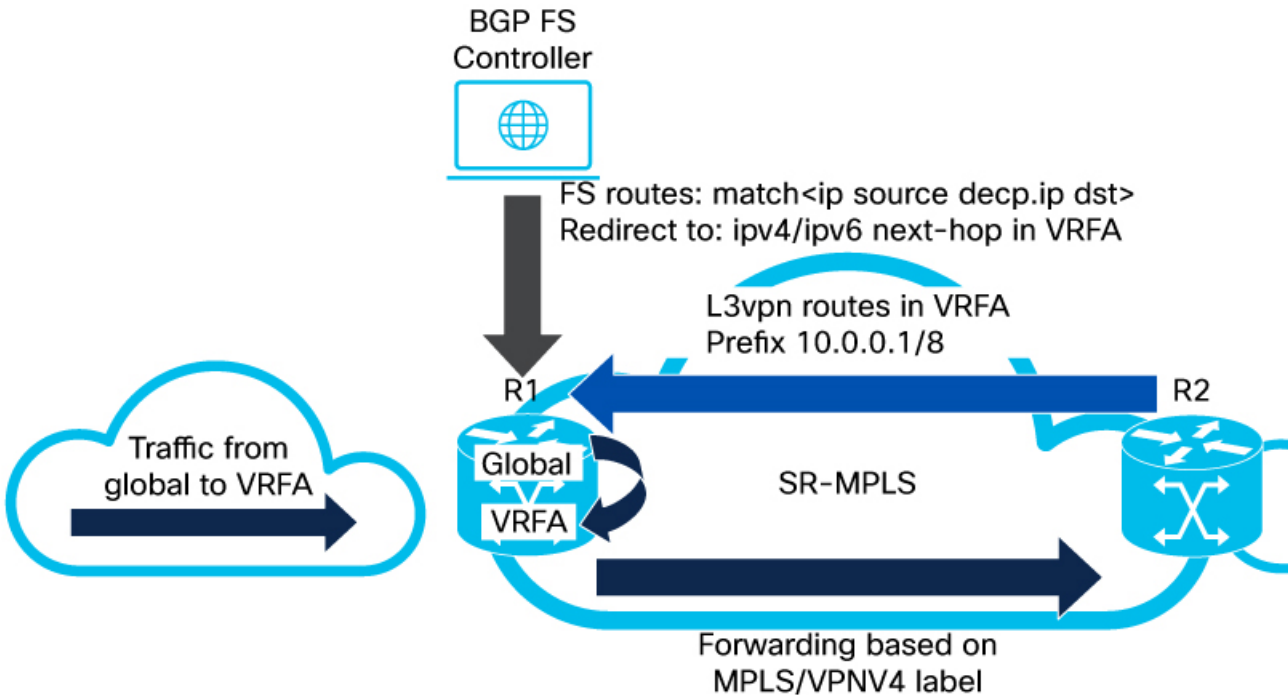


Figure 4: Forwarding based on MPLS



These stages describe How BGP flowspec redirect from global VRF to L3VPN and segment routing policy works.

1. Incoming packets arrive on a global VRF interface with destination IPs that exist only in a customer VRF routing table.
2. Without intervention, these packets are dropped due to lookup failure in the global VRF.
3. The BGP Flowspec server programs redirect rules and propagates them to BGP Flowspec neighbors via BGP NLRI.
4. Neighbors store and activate these rules in their databases.
5. Incoming packets matching the active rules are redirected to the appropriate customer VRF instance.
6. The redirect action specifies the correct route target to ensure accurate routing.
7. Forwarding occurs through L3VPN or Segment Routing Policy, enabling precise traffic control and preventing packet loss.

Result

This process ensures that packets with destination IPs only present in customer VRFs are correctly redirected and forwarded, preventing packet drops and enabling fine-grained traffic control within complex network environments.

Configure BGP flowspec redirect from global VRF

Enable BGP Flowspec redirect from the global VRF to customer VRFs by configuring class maps, policy maps, and applying service policies on the BGP Flowspec controller.

This task enables BGP Flowspec redirect from the global VRF to customer VRFs to prevent packet drops when destination IPs exist only in customer VRF routing tables. The BGP Flowspec server programs and distributes redirect rules to neighbors, which store and activate them. Matching packets are redirected to the correct VRF via the specified route target, ensuring accurate forwarding through L3VPN or Segment Routing Policy for precise traffic control in complex networks.

Before you begin

- Ensure BGP Flowspec is supported and enabled on the router.
- Confirm that the global VRF and customer VRFs are properly configured.
- Verify that L3VPN or Segment Routing Policy (SR-Policy) mechanisms are in place for forwarding.
- Have access to configure class maps, policy maps, and flowspec service policies on the BGP Flowspec controller.
- Confirm that route targets for the customer VRFs are defined and reachable.
- Ensure a policer action is configured to enable BGP Flowspec statistics.

Follow these steps to configure BGP flowspec redirect from global VRF.

Procedure

Step 1 Define traffic classes to match packets based on destination IP addresses.

Example:

```
Router# configure terminal
Router(config)# class-map type traffic match-all ipv4_CM1
Router(config-cmap)# match destination-address ipv4 10.0.0.1 255.255.255.0
Router(config-cmap)# end-class-map
Router(config)# class-map type traffic match-all ipv6_CM1
Router(config-cmap)# match destination-address ipv6 2000:0:0:1::/64
Router(config-cmap)# end-class-map
Router(config)# exit
```

Step 2 Create policy maps to specify redirect actions for the matched traffic classes.

Example:

```
Router(config)# policy-map type pbr ipv4_PM1
Router(config-pmap)# class type traffic ipv4_CM1
Router(config-pmap-c)# redirect nexthop route-target 1:1
Router(config-pmap-c)# exit
Router(config-pmap)# class type traffic class-default
Router(config-pmap)# end-policy-map
Router(config)# policy-map type pbr ipv6_PM1
Router(config-pmap)# class type traffic ipv6_CM1
Router(config-pmap-c)# redirect nexthop route-target 1:1
Router(config-pmap-c)# exit
Router(config-pmap)# class type traffic class-default
Router(config-pmap)# end-policy-map
Router(config)# exit
```

Step 3 Attach the policy maps to the respective address families under flowspec configuration.

Example:

```
Router(config)# flowspec
Router(config)# address-family ipv4
Router(config-af)# service-policy type pbr ipv4_PM1
Router(config-af)# exit
Router(config)# address-family ipv6
Router(config-af)# service-policy type pbr ipv6_PM1
Router(config-af)# exit
Router(config)# exit
```

Step 4 Install flowspec rules on all interfaces locally:

Example:

```
Router(config)# flowspec
Router(config)# local-install interface-all
Router(config)# exit
```

Step 5 Verify the running configuration.

Example:

```
Router# show running-config
class-map type traffic match-all ipv4_CM1
 match destination-address ipv4 10.0.0.1. 255.255.255.0
 end-class-map
!
```

```
class-map type traffic match-all ipv6_CM1
 match destination-address ipv6 2000:0:0:1::/64
 end-class-map
!

policy-map type pbr ipv4_PM1
 class type traffic ipv4_CM1
  redirect nexthop route-target 1:1

  !
 !
 class type traffic class-default
 !
 end-policy-map
!
policy-map type pbr ipv6_PM1
 class type traffic ipv6_CM1
  redirect nexthop route-target 1:1

  !
 !
 class type traffic class-default
 !
 end-policy-map
!

flowspec
 address-family ipv4
  service-policy type pbr ipv4_PM1
 address-family ipv6
  service-policy type pbr ipv6_PM1


flowspec config on PE1:

flowspec
 local-install interface-all
```

Step 6 Use the **show ofa objects pbr object-count location 0/RP0/CPU0** command in privileged EXEC mode to verify the number of BGP Flowspec entries in the OFA object.

Example:

```
Router# show ofa objects pbr object-count location 0/RP0/CPU0

Table [PBR] has 4200 entries in DB
Table [PBR] had 4200 as highest count @ Tue Feb  6 20:08:04 2024
```

You will see the current count of BGP Flowspec entries in the OFA object and detailed statistics for each BGP Flowspec rule, including matched, transmitted, and dropped packets and bytes.

Step 7 Use the **show flowspec ipv4 detail** command in privileged EXEC mode to verify the BGP Flowspec rules and their statistics.

Example:

```
Router# show flowspec ipv4 detail
AFI: IPv4
  Flow           :Dest:10.0.0.1/8
    Actions      :Traffic-rate: 5000000 bps Redirect: VRF vpn1 Route-target: ASN2-1:1  (bgp.1)
    Statistics                       (packets/bytes)
      Matched             :                 200/25600
      Transmitted         :                 200/25600
      Dropped             :                   0/0
  Flow           :Dest:10.0.0.2/8
```

```
Actions      :Traffic-rate: 5000000 bps Redirect: VRF vpn1 Route-target: ASN2-1:1  (bgp.1)
Statistics                          (packets/bytes)
  Matched             :                 200/25600
  Transmitted         :                 200/25600
  Dropped             :                   0/0
```

Traffic filtering actions: what you need to know about controlling traffic with BGP flowspec

You use traffic filtering actions to control how IP traffic matching specific flow rules is handled. This topic explains how to specify actions like dropping or policing traffic using BGP flowspec extended communities. Understanding these actions helps you manage network traffic effectively and maintain performance and security.

The default action accepts IP traffic that matches a flow specification rule.

You can change this behavior by applying extended community values that specify different actions.

Table 4: Traffic Filtering Actions

Type	Extended Community	PBR Action	Description
0x8006	traffic-rate 0 traffic-rate <rate>	Drop Police	The traffic-rate extended community is a non-transitive extended community across the autonomous-system boundary and uses following extended community encoding: The first two octets carry the 2-octet id, which can be assigned from a 2-byte AS number. When a 4-byte AS number is locally present, the 2 least significant bytes of such an AS number can be used. This value is informational. The remaining 4 octets carry the rate information in IEEE floating point [IEEE.754.1985] format, bytes per second. A traffic-rate of 0 should result on all traffic for the particular flow to be discarded. Command syntax **police rate < > | drop**
0x8009	traffic-marking	Set DSCP	The traffic marking extended community instructs a system to modify the differentiated service code point (DSCP) bits of a transiting IP packet to the corresponding value. This extended community is encoded as a sequence of 5 zero bytes followed by the DSCP value encoded in the 6 least significant bits of 6th byte. Command syntax **set dscp** *<6 bit value>*

Type	Extended Community	PBR Action	Description
0x0800	Redirect IP NH	Redirect IPv4 or IPv6 Nexthop	Announces the reachability of one or more flowspec NLRI. When a BGP speaker receives an UPDATE message with the redirect-to- IP extended community it is expected to create a traffic filtering rule for every flow-spec NLRI in the message that has this path as its best path. The filter entry matches the IP packets described in the NLRI field and redirects them or copies them towards the IPv4 or IPv6 address specified in the Network Address of Next-Hop field of the associated MP_REACH_NLRI. **Note** The redirect-to-IP extended community is valid with any other set of flow-spec extended communities except if that set includes a redirect-to-VRF extended community (type 0x8008) and in that case the redirect-to-IP extended community should be ignored. **Note** Redirect IP NH is supported only in default VRF. Command syntax **redirect {ipv4 | ipv6} next-hop {ipv4-address | ipv6-address}**

Note

- You cannot use the BGP flowspec actions *rate limit* and *redirect*together.
- The *redirect* action works only with nexthop IPv4 and IPv6 addresses, not with nexthop VRF IPv4 or IPv6.