# Configuring ePBR L3

This chapter describes how to configure Enhanced Policy-based Redirect (ePBR) on Cisco NX-OS devices.



# Information About ePBR L3

Enhanced Policy-based Redirect (ePBR) in Elastic Services Re-direction (ESR) provides traffic redirection and service chaining across the NX-OS and fabric topologies by leveraging policy-based redirect solution and achieves service chaining without adding extra headers, and avoids latency in using extra headers.

ePBR enables application-based routing and provides a flexible, device-agnostic policy-based redirect solution without impacting application performance. The ePBR service flow includes the following tasks:

## Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide* and the *Cisco NX-OS Licensing Options Guide* .

## Configuring ePBR Service and Policy

You must first create an ePBR service which defines the attributes of service end points. Service end points are the service appliances such as firewall, IPS, etc., that can be associated with switches. You can also define probes to monitor the health of the service end points and can define the forward and reverse interfaces where the traffic policies are applied. ePBR also supports load balancing along with service chaining. ePBR allows you to configure multiple service end points as a part of the service configuration.

In events of service endpoint failures, traffic that was redirecting to the failed service endpoints will redirect to other reachable service endpoints, configured in the ePBR service. Resilient hashing is supported during endpoint failures for an ePBR service that has multiple service endpoints. Traffic that was always being redirected to a specific service endpoint continues to redirect to the same device in events of failures of other service endpoints that are part of the same service.

Beginning with Cisco NX-OS Release 10.2(1)F, the VRF of every service in a chain may either be unique or may be exactly identical. The service endpoints and interfaces defined for a service, should pertain to the VRF defined for the service.

Service end-point interfaces having an existing IPv4 PBR policy cannot be used inside an IPv4 ePBR service. Similarly service end-point interfaces having an existing ipv6 PBR policy cannot be used inside an IPv6 ePBR service.

After creating the ePBR service, you must create an ePBR policy. The ePBR policy allows you to define traffic selection, redirection of traffic to the service end point and various fail-action mechanisms on the end point health failure. You may use IP access-list end points with permit access control entries (ACE) to define the traffic of interest to match and take the appropriate action.

The ePBR policy supports multiple ACL match definitions. A match can have multiple services in a chain which can be sequenced by a sequence number. This allows flexibility to add, insert, and modify elements in a chain in a single service policy. In every service sequence, you can define the fail action method such as drop, forward, and bypass. The ePBR policy allows you to specify source or destination-based load balancing and bucket counts in order to have granular load balancing of traffic.

# Applying ePBR to an Interface

After creating the ePBR policy you need to apply the policy on an interface. This allows you to define the interface at which the traffic ingresses into the NX-OS or Nexus fabric. You can also apply the policy in both the forward and reverse directions. There may only be two IPv4/IPv6 policies applied to the interface, one in the forward and one in the reverse direction.

Beginning with Cisco NX-OS Release 10.2(1)F, ePBR supports policy application on layer-3 port-channel sub-interfaces

Beginning with Cisco NX-OS Release 10.2(1)F, the interface on which the ePBR policy is applied may be on a different VRF than the VRF of the services in the chain.

ePBR IPv4 policies cannot be applied to an interface on which an IPv4 PBR policy is already applied. ePBR IPv6 policies cannot be applied to an interface on which an IPv6 PBR policy is already applied.

# Creating Bucket and Load Balancing

ePBR computes the number of traffic buckets based on the service that has maximum number of service-end-points in the chain. If you configure the load balance buckets, your configuration will have the precedence. ePBR supports load balancing methods of source IP and destination IP but does not support L4-based source or destination load balancing methods.

# ePBR Service Endpoint Out-of-Service

The ePBR service endpoint Out-of-Service feature provides the option to temporarily remove the endpoint from service. The following two methods can be used to move an endpoint Out-of-Service:

1. **Administrative Out-of-Service**: This method is used during maintenance or upgrades to temporarily move the service-endpoints to operationally down state and avoid sending traffic to the node, while still retaining the service-endpoints as a valid endpoint device in service.

The user would also require the ability to bring the service endpoint back in service on the Cisco NX-OS switch after the maintenance procedure has been completed. This is a standard feature provided used by load-balancers in the industry today.

2. **Auto Out-of-Service**: This method is used during recovery of endpoints after failures, ePBR detects the reachability of the endpoints getting re-established and attempts to redirect subsets of flows back to the node.

   Also, when certain networks may be tolerant to a rare endpoint failure and recovery but may require detecting endpoints that are losing and re-establishing connectivity constantly, each event disrupting end-to-end connections twice. It may be desirable to put such nodes Out-of-Service.

# ePBR Object Tracking, Health Monitoring, and Fail-Action

ePBR creates SLA and Track objects based on the probe types configured in the service and supports various probes and timers such as ICMP, TCP, UDP, DNS, and HTTP. ePBR also supports user defined tracks, which allows you to create tracks with various parameters including milli second probes in associating with ePBR.

ePBR monitors the health of the end points by provisioning IP SLA probes and object tracks to track the IP SLA reachability when you apply the ePBR probe configuration.

You can configure the ePBR probe options for a service or for each of the forward or reverse end points. You can also configure frequency, timeout, retry up and down counts, and source loopback interface so that they can be used for source IP of an IP SLA session. The retry-up and down counts are used as multipliers for the frequency to determine **delay-up** and **delay-down** intervals. Once the service endpoint is initially detected as failed or recovered, the system will act on these events after the expiry of these intervals. You can define any type of tracks and associate them with the forward or the reverse end points. The same track objects is re-used for all policies using the same ePBR service.

You can define tracks separately and assign the track ID to each service-end point in ePBR. If you do not assign any user-defined track to an endpoint, ePBR will create a track using probe method for the end point. If no probe method is defined at the end point level, the probe method configured for the service level will be used.

ePBR supports the following fail-action mechanisms for its service chain sequences:

- Bypass
- Drop on Fail
- Forward

Bypass of a service sequence indicates that the traffic must be redirected to the next service sequence when there is a failure of the current sequence.

Drop on fail of a service sequence indicates that the traffic must be dropped when all the service-end-points of the service become unreachable.

Forward is the default option and indicates that upon failure of the current service, traffic should use the regular routing tables. This is the default fail-action mechanism.

**Note** Symmetry is maintained when fail-action bypass is configured for all the services in the service chain. In other fail-action scenarios, when there are one or more failed services, symmetry is not maintained in the forward and the reverse flow.

# ePBR Session-based Configuration

ePBR sessions allow addition, deletion or modification of the following aspects of in-service services or policies. The in-service refers to a service that is associated with a policy that has been applied to an active interface or a policy that is being modified and currently configured on an active interface.

- Service endpoints with their interfaces and probes
- Reverse endpoints and probes
- Matches under policies
- Load-balance methods for matches
- Match sequences and fail-action

**Note** In ePBR Sessions, you cannot move interfaces from one service to another service in the same session. To move interfaces from one service to another service, perform the following steps:

1. Use a session operation to first remove it from the existing service.
2. Use a second session operation to add it to the existing service.

# ePBR Multi-Site

Beginning with Cisco NX-OS Release 10.2(1)F, service-chaining in a VXLAN multisite fabric can be achieved by using the following configuration and topology guidelines.

- Endpoints in a service or services in the chain may be distributed across different leaf switches, in the same or different site.
- Every service should be in its unique VRF, which is different from the tenant VRF context in which the ePBR policy is applied.
- To segregate traffic for different tenant VRFs, the VLANs used for the services would be required to be segregated and new services and policies would need to be defined.
- Tenant VRF routes should be leaked to each of the service VRFs on every leaf switch hosting the services, to allow traffic to be routed back at the end of the service chain to its destination, in the tenant VRF.
- VNIs should be symmetrically allocated across different leaf switches and sites.
- The ePBR policy should be enabled on all layer-3 VNIs of the service VRFs being used, on all leaf switches hosting services and on the border leaf or border gateway switches, if it is acting as transit for multi-site.

- The service chain may be isolated to one site entirely, with traffic arriving from different sites. Although this scenario doesn't involve multi-site distribution of service devices, the layer-3 VNIs of the service VRFs on the border gateways or border leafs should only be treated as multi-site transit and the ePBR policy should be applied on them. The ePBR policy should be also applied on the host or tenant facing interfaces in the remote sites where the traffic is arriving from.

## ACL Refresh

ePBR session ACL refresh allows you to update the policy generated ACLs, when the user-provided ACL gets modified or added or deleted with ACEs. On the refresh trigger, ePBR will identify the policies that are impacted by this change and create or delete or modify the buckets' generated ACLs for those policies.

For ePBR scale values, see Cisco Nexus 9000 Series NX-OS Verified Scalability Guide.

# Guidelines and Limitations for ePBR L3

ePBR has the following guidelines and limitations:

- The L3 ePBR feature requires sufficient ing-racl TCAM to function properly. To verify the current TCAM carving, use the **show hardware access-list tcam region** command. If the appropriate TCAM size is not allocated, use the **hardware access-list tcam region ing-racl** *size multiple of 256* command to allocate the appropriate TCAM.
- Beginning with Cisco Nexus NX-OS Release 10.1(2), ePBR with IPv4 and IPv6 is supported on N9K-C93108TC-FX3P switch.
- Beginning with Cisco NX-OS Release 10.1(1) each match statement under ePBR policy can support three action types - redirect, drop, and exclude. There can be only one drop and/or exclude match statement per policy. The ACE rules for the traffic, which needs to be excluded or dropped in the forward as well as the reverse directions, should be manually added to the match access-list that is used with the action of exclude or drop. The statistics for the exclude and drop match access-list may display traffic hit counters for both directions.
- ePBR policies require at least one match with redirect action.
- Beginning with Cisco NX-OS Release 10.1(1), ePBR with IPv4, IPv6 and ePBR over VXLAN are supported on below platform switches: N9K-C9316D-GX, N9K-C93600CD-GX, N9K-C9364C-GX, N9K-C93180YC-FX3S, N9K-C93360YC-FX3, and N9K-C93108TC-FX3P.
- When fail-action is specified in any match statement, probe is mandatory in the configuration.
- Whenever there is OTM track changes ePBR statistics is reset due to RPM reprogramming.
- Do not share the same user defined ACL across multiple match statements in the ePBR configuration.
- Symmetry in traffic is maintained only when fail-action bypass is configured for ePBR Service. For the other fail-actions such as forward/drop in the service chain, symmetry is not maintained for the forward and reverse flow of traffic.
- Unique layer-4 source and destination port parameters should be specified for the match filters if traffic is required to match any source and any destination IP as per the match access-list definition, and is required to be redirected to devices distributed in a VXLAN environment in both forward and reverse directions or service-chained through one-arm devices.

- Feature ePBR and feature ITD cannot co-exist with the same ingress interface.
- With scaled ePBR configuration, it is recommended to remove the policies before you use the **no feature epbr** command.
- It is recommended that you classify probe traffic in a separate CoPP class. Otherwise, probe traffic will go in the default CoPP class and might be dropped causing IP SLA bouncing for probe traffic. For information on CoPP configuration for IP SLA, see Configuring CoPP for IP SLA Packets.
- ePBR is supported on the Cisco Nexus 9500 and Cisco Nexus 9300 platform switches with EX, FX, and FX2 line cards.
- Beginning with Cisco NX-OS Release 9.3(5) Catena feature is deprecated.
- If you want to remove the ePBR service endpoint which is configured to a port-channel that is removed from the system, perform the following steps:
  1. Delete the existing ePBR policy.
  2. Delete the existing ePBR service.
  3. Reconfigure the ePBR service endpoint to the required port-channel.
- Please do not modify the dynamically created access-list entries of ePBR that begin with the name “epbr_”. These access-lists are reserved for ePBR internal use.

**Note** Modifying these prefix strings can cause the ePBR to not function properly and would impact ISSU.

- Router ACLs may be enabled alongside layer-3 ePBR policies on supported layer-3 interfaces, only when statistics is not enabled for either ePBR policies or the router ACLs. See **Guidelines and Limitations for Policy-Based Routing** in the Policy-based routing chapter of *Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide* for details on this limitation.
- On Cisco Nexus N9K-C9316D-GX, N9K-C93600CD-GX, and N9K-C9364C-GX switches, before performing ISSD from Cisco NX-OS, Release 10.2 and later releases to Release 10.1 disable ePBR policies and proceed with downgrade.
- ePBR policy definition can be applied to a maximum of 32 interfaces of supported interface types across forward and reverse directions.
- Beginning with Cisco NX-OS Release 10.4(1)F, ePBR supports IPv4 and IPv6 policies on GRE and IP-IP tunnel interfaces for load-balancing and redirection on Cisco Nexus 9300-FX2/FX3/GX/GX2 platform switches:
- Beginning with Cisco NX-OS Release 10.4(1)F, ePBR supports redirection or load-balancing to Layer-3 endpoints reachable over IP-IP and GRE tunnel interfaces on Cisco Nexus 9300-FX2/FX3/GX/GX2 platform switches.

**Note**

- ePBR IPv6 policies are not supported on IP-IP tunnel interfaces.
- Currently ePBR does not support service-chaining to devices reachable over IP-IP and GRE tunnels.

- Configuration rollback and configuration replace are supported only when the ePBR policy is not associated with any interfaces and the ePBR service definitions are not used in any active ePBR policy in both the source and target configurations. However, configuration rollback and configuration replace do not support policy to interface association and disassociation.
- Disabling the atomic update may allow more TCAM resources to be made available for the ePBR policies, but it may cause possible disruption in traffic during configuration changes to the policies or during fail-over and recovery of service endpoints. For further details, see **Atomic ACL Updates** section of *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.
- Unique policies are generated for every interface that is configured with an ePBR policy. Additionally unique policies are also generated for every service interface that needs to steer the traffic to the next service function inside a service-chain configured for a match inside an ePBR policy. The scale of supported EPBR policies may vary with the available ACL labels in the system for PBR policies. For further details on ACL labels sizes, see **Maximum Label Sizes Supported for ACL Types** section of *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.
- The ePBR service or endpoint hold-down timers used should be compatible with the probe (track and IP SLA) frequencies and timeouts in use, so that the failures can be detected in time.
- Endpoint states of the forward and reverse arms of dual-arm devices are not synchronized automatically. If this is needed, identical probe track configuration on the forward and reverse arms should be used. Probe tracks configured for endpoints may be shared between the forward and reverse arms of the same endpoints, but not across endpoints in the same or different services.
- Beginning with Cisco NX-OS Release 10.5(1)F, it is no longer necessary to explicitly configure the reverse IP address for one-arm service devices. If a service endpoint is not assigned a reverse IP address, it will be treated as a one-arm device, and traffic will be redirected to the same IP address in both the forward and reverse directions.
- If the IP address of a loopback interface associated with a service probe is modified, you need to remove and reapply any policies and contracts that reference the service.
- Beginning with Cisco NX-OS Release 10.5(2)F, ePBR will support the **set-vrf** command to redirect packets through a specified VRF instance on Cisco Nexus 9300-FX2, FX3, GX, GX2, H2R and H1 Series switches with the following limitations:
  - The **source-vrf** and **destination-vrf** cannot be modified or deleted through an ePBR session.
  - The **set-vrf** is not supported for ePBR on VXLAN.
  - The **set-vrf** doesn't switch VRFs for drop and exclude traffic.

The following guidelines and limitations apply to ePBR over VXLAN feature:

- In VXLAN fabric, service chaining cannot be done to devices within same VLAN. All devices must be present in separate VLANs.

- When every service in the chain is in the same VRF, ePBR is only supported at a single site in a VXLAN multisite fabric.
- When every service in the chain is in the same VRF:
  - Active/Standby chain is supported with two service nodes with no restrictions.
  - Active/Standby chain with three or more service nodes in chain requires no two nodes of different type behind same service leaf.
  - In VXLAN fabric you cannot stitch traffic from one service in a leaf and come back later to the same leaf.

**Note** These restrictions are not applicable if every service in the chain is in a different VRF context.

- When service endpoints are distributed in a VXLAN environment or on VPC peers, the service endpoints must be configured in an identical order on all switches.
- For service-endpoints distributed in a VXLAN environment, you must configure source loopback interfaces for the probe, so that a unique source IP may be used for IP SLA sessions.
- The ePBR policy should always be originally applied on host or tenant facing interfaces. The ePBR policy should be applied on Layer-3 VNI interfaces pertaining to the tenant or service VRFs only as the transit interfaces.

  Only traffic arriving for the endpoints in the specific VRF will be redirected by the policies applied on the layer-3 VNI interfaces pertaining to that VRF. The statistics for the traffic matching the policy on the layer-3 VNI interface will not be visible via ePBR statistics command.
- Beginning with Cisco NX-OS Release 10.3(3)F, you can apply an ePBR Layer 3 policy on a new L3VNI interface on Cisco Nexus 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 platform switches.

The following guidelines and limitations apply to the match ACL feature:

- Only ACEs with the permit method are supported in the ACL. ACEs with any other method (such as deny or remark) are ignored.
- A maximum of 256 permit ACEs are supported in one ACL.
- ACEs with object-groups specified as address-groups or port-groups in either source or destination parameters are not supported.
- Beginning with Cisco NX-OS Release 10.4(1)F, the Layer-4 port ranges and other port operations (such as 'not equal to', 'greater than', 'lesser than') in the match access-list rules will be honored and used for filtering traffic in the bucket access-lists.
- The configuration **hardware access-list lou resource threshold** must be used for optimal utilization of TCAM ACEs, while using layer-4 port operators in access-lists. For more information on the command, see **Configuring IP ACLs** section of *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.

The following guidelines and limitations apply to inter-VRF service chaining:

- Beginning with Cisco NX-OS 10.2(1)F Release, every service in a chain should either exist in the same VRF or completely unique VRFs.

- In version 10.2(1)F, fail-action bypass mechanism is not supported when every service in the chain exists in a unique VRF.
- From Cisco NX-OS 10.2(2)F Release, fail-action bypass is supported when the services in the chain are in unique VRFs.
- If the services are in a different VRF than the VRF context of the interface on which the ePBR policy is applied, the user should ensure that the tenant routes are leaked to every service VRF, in order to ensure that the traffic is able to route back to the tenant VRF, at the end of the service chain.
- From Cisco NX-OS Release 10.2(2)F, PBR allows multiple backup next-hops related to different VRFs to be configured for a route-map sequence. This allows ePBR to enable fail-action bypass from service pertaining to one VRF to another effectively.
- Beginning with Cisco NX-OS Release 10.2(3)F, to minimize traffic disruptions during session operations of endpoint additions, service sequence additions, deletions, and modifications, it is recommended to have load-balance buckets configured ahead and avoid modification to the load-balance configuration. Ensure that the configured buckets for load-balance are greater than the number of endpoints configured in services for every sequence in the chain.

The following guidelines and limitations applies if you have configured ePBR using source IP-based load balancing:

- The prefix length in the source IPv4 of the ACE cannot be /32
- The prefix length in the source IPv6 address of the ACE cannot be /128
- The subnet for the source address must be compatible with the buckets configured.

The following guidelines and limitations applies if you have configured ePBR using destination IP-based load balancing:

- The prefix length in the destination IPv4 of the ACE cannot be /32
- The prefix length in the destination IPv6 address of the ACE cannot be /128
- The subnet for the destination address must be compatible with the buckets configured.

The following guidelines and limitations applies if you have configured ePBR service endpoint Out-of-Service feature:

- ePBR service endpoint Out-of-Service feature is supported for Layer-3 services on Cisco Nexus 9300-FX/FX2/FX3/GX/GX2 and Cisco Nexus 9500 switches with X97160YC-EX, 9700-FX/GX line cards.
- ePBR out-of-service (shut or hold-down) requires endpoint to be configured with probes, either at a endpoint level or a service level.
- When the service is being used by a policy that is active, ePBR out-of-service (shut or hold-down) must be configured using **epbr sessions** only.

The following guidelines and limitations applies if you are using source IP-based load balancing and load-balancing traffic to more than 1 endpoint:

- The source IPv4 subnet mask of the ACE inside the match access-list cannot be /32, or the subnet mask of the source IPv6 address inside the match access-list cannot be /128.

- The destination IPv4 subnet mask of the ACE inside the match access-list cannot be /32, or the subnet mask of the source IPv6 address inside the match access-list cannot be /128.
- The subnet masks for the source address or destination address inside the match access-list, based on the load-balance method, must be compatible with the buckets configured for the match or must be compatible with the number of buckets required, based on the number of endpoints in the services being used for the match.

# Configuring ePBR L3

**Before you begin**

Make sure you have configured IP SLA and PBR features before configuring the ePBR feature.

## Configuring ePBR Service, Policy, and Associating to an Interface

The following section provides information about configuring the ePBR Service, ePBR Policy, and associating the policy on to an interface.

### SUMMARY STEPS

1. **configure terminal**
2. **epbr service** *service-name*
3. [**no**] **probe** {**icmp** | *l4-proto port-number* [**control** *status*] | **http get** [*url-name* [**version** *ver*] | **dns host***host-name* **ctp**} [**frequency** *freq-num* | **timeout** *seconds* | **retry-down-count** *down-count* | **retry-up-count** *up-count* | **source-interface** *src-intf* | **reverse** *rev-src-intf*]
4. **vrf** *vrf-name*
5. **service-endpoint** {**ip** *ipv4 address* | **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]
6. **probe track** *track ID*
7. **reverse ip** *ip address* **interface** *interface-name interface-number*
8. **exit**
9. **epbr policy** *policy-name*
10. **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] } [**redirect** | **drop** | **exclude**]
11. [**no**] **load-balance** [ **method** { **src-ip** | **dst-ip**}] [ **buckets** *sequence-number*] [**mask-position** *position-value*]
12. *sequence-number* **set service** *service-name* [ **fail-action** { **bypass** | **drop** | **forward**}]
13. **interface** *interface-name interface-number*
14. **epbr** { **ip** | **ipv6**} **policy** *policy-name* [**reverse**]
15. **exit**

## DETAILED STEPS

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters configuration mode. |
| **Step 2** | **epbr service** *service-name*<br>**Example:**<br>`switch(config)# epbr service firewall` | Creates a new ePBR service. |
| **Step 3** | [**no**] **probe** {**icmp** \| *l4-proto port-number* [**control** *status*] \| **http get** [*url-name* [**version** *ver*] \| **dns host***host-name* **ctp**} [**frequency** *freq-num* \| **timeout** *seconds* \| **retry-down-count** *down-count* \| **retry-up-count** *up-count* \| **source-interface** *src-intf* \| **reverse** *rev-src-intf*]<br>**Example:**<br>`switch(config)# probe icmp` | Configures the probe for the ePBR service. The probe types supported are ICMP, TCP, UDP, DNS, and HTTP, CTP.<br>The options are as follows:<br>• frequency—Specifies the frequency of the probe in seconds. The range is from 1 to 604800.<br>• retry-down-count —Specifies the number of recounts undertaken by the probe when the node goes down. The range is from 1 to 5.<br>• retry-up-count —Specifies the number of recounts undertaken by the probe when the node comes back up. The range is from 1 to 5.<br>• timeout —Specifies the length of the timeout period in seconds. The range is from 1 to 604800. |
| **Step 4** | **vrf** *vrf-name*<br>**Example:**<br>`switch(config)# vrf tenant_A` | Specifies the VRF for the ePBR service. |
| **Step 5** | **service-endpoint** {**ip** *ipv4 address* \| **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]<br>**Example:**<br>`switch(config-vrf)# service-endpoint ip 172.16.1.200 interface VLAN100` | Configures service endpoint for the ePBR service.<br>You can repeat steps 2 to 5 to configure another ePBR service. |
| **Step 6** | **probe track** *track ID*<br>**Example:**<br>`switch(config-vrf)# probe track 30` | Defines a track separately and assign an existing track ID to each service-endpoint in ePBR.<br>You can assign track ID to each endpoint. |
| **Step 7** | **reverse ip** *ip address* **interface** *interface-name interface-number* | Defines the reverse IP and interfaces where the traffic policies are applied. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`switch(config-vrf)# reverse ip 172.16.30.200 interface VLAN201` | **Note**<br>Beginning with Cisco NX-OS Release 10.5(1)F, it is no longer necessary to explicitly configure the reverse IP address for one-arm service devices. If a service endpoint is not assigned a reverse IP address, it will be treated as a one-arm device, and traffic will be redirected to the same IP address in both the forward and reverse directions. |
| **Step 8** | **exit**<br>**Example:**<br>`switch(config-vrf)# exit` | Exits VRF configuration mode and enters global configuration mode. |
| **Step 9** | **epbr policy** *policy-name*<br>**Example:**<br>`switch(config)# epbr policy Tenant_A-Redirect` | Configures the ePBR policy. |
| **Step 10** | **match** { [**ip address** *ipv4 acl-name*] \| [**ipv6 address** *ipv6 acl-name*] } [**redirect** \| **drop** \| **exclude**]<br>**Example:**<br>`switch(config)# match ip address WEB` | Matches an IPv4 or IPv6 address against an IP or IPv6 ACLs. Redirect is the default action for a match traffic. Drop is used when the traffic needs to be dropped on the incoming interface. Exclude option is used to exclude certain traffic from service-chaining on the incoming interface.<br>You can repeat this step to match multiple ACLs based on the requirement. |
| **Step 11** | [**no**] **load-balance** [ **method** { **src-ip** \| **dst-ip**} ] [ **buckets** *sequence-number*] [**mask-position** *position-value*]<br>**Example:**<br>`switch(config)# load-balance method src-ip mask-position 3` | Computes the load balance method and the number of buckets to be used by the ePBR service.<br>Beginning with Cisco NX-OS Release 10.3(3)F, the **mask-position** option is provided to choose the bits used for load-balancing in user-defined ACL. Default value is 0.<br>If mask-position is configured, the load-balance bits start from configured mask-position. Based on number of buckets needed, more bits are taken to generate load-balancing buckets, toward the most-significant bit.<br>**Note**<br>For any ACE in user-defined ACLs, if bits used to generate load-balancing buckets overlap with the user-defined subnet, the mask position for the ACE will be reset internally to 0. |
| **Step 12** | *sequence-number* **set service** *service-name* [ **fail-action** { **bypass** \| **drop** \| **forward**} ]<br>**Example:**<br>`switch(config)# set service firewall fail-action drop` | Computes the fail-action mechanism. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 13** | **interface** *interface-name interface-number*<br>**Example:**<br>`switch(config)# interface vlan 2010`<br>`switch(config)# interface vni500001` | Configures an interface and enters interface configuration mode.<br>**Note**<br>Beginning with Cisco NX-OS Release 10.3(3)F, you can apply an ePBR L3 policy on a new L3VNI interface. |
| **Step 14** | **epbr** { **ip** \| **ipv6**} **policy** *policy-name* [**reverse**]<br>**Example:**<br>`switch(config-if)# epbr ip policy Tenant_A-Redirect` | An interface may be associated at any time with one or more of the following:<br>• an IPV4 policy in the forward direction<br>• an IPv4 policy in the reverse direction<br>• an IPv6 policy in the forward direction<br>• an IPv6 policy in the reverse direction |
| **Step 15** | **exit**<br>**Example:**<br>`switch(config-if)# end` | Exits interface configuration mode and returns to global configuration mode. |

# Modifying a Service Using ePBR Session

The following steps explain how to modify a service using ePBR session.

### SUMMARY STEPS

1. **epbr session**
2. **epbr service** *service-name*
3. [**no**] **service-endpoint** {**ip** *ipv4 address* | **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]
4. **service-endpoint** {**ip** *ipv4 address* | **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]
5. **reverse ip** *ip address* **interface** *interface-name interface-number*
6. **commit**
7. **abort**

### DETAILED STEPS

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **epbr session**<br>**Example:**<br>`switch(config)# epbr session` | Enters ePBR session mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **epbr service** *service-name*<br>**Example:**<br>`switch(config-epbr-sess)# epbr service TCP_OPTIMIZER` | Specifies the configured ePBR service in the ePBR session mode. |
| **Step 3** | [**no**] **service-endpoint** {**ip** *ipv4 address* \| **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]<br>**Example:**<br>`switch(config-epbr-sess-svc)# no service-end-point ip 172.16.20.200 interface VLAN200` | Disables the configured service endpoint for the ePBR service. |
| **Step 4** | **service-endpoint** {**ip** *ipv4 address* \| **ipv6** *ipv6 address*} [**interface** *interface-name interface-number*]<br>**Example:**<br>`switch(config-epbr-sess-svc)#service-end-point ip 172.16.25.200 interface VLAN200` | Modifies the service endpoint and replaces the IP for the ePBR service. |
| **Step 5** | **reverse ip** *ip address* **interface** *interface-name interface-number*<br>**Example:**<br>`switch(config-epbr-sess-svc-ep)# reverse ip 172.16.30.200 interface VLAN201` | Defines the reverse IP and interfaces where the traffic policies are applied. |
| **Step 6** | **commit**<br>**Example:**<br>`switch(config-epbr-sess)# commit` | Completes the modification of the ePBR service using the ePBR session.<br>**Note**<br>Restart the ePBR session after you complete this step. |
| **Step 7** | **abort**<br>**Example:**<br>`switch(config-epbr-sess)# abort` | Aborts the session and clears or resets the current configuration under the session. Use this command to abandon the current session configuration in case of errors or unsupported configuration identified during commits.<br>**Note**<br>Restart a new ePBR session after this with the rectified configuration. |

# Modifying a Policy Using ePBR Session

The following steps explain how to modify a policy using ePBR Session.

### SUMMARY STEPS

1. **epbr session**
2. **epbr policy** *policy-name*
3. [**no**] **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] [**l2 address** *ipv6 acl-name*]} **vlan** {**vlan** | **vlan range** | **all**} [**redirect** | **drop** | **exclude**] }

4. **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] [**l2 address** *ipv6 acl-name*]} **vlan** {**vlan** | **vlan range** | **all**} [**redirect** | **drop** | **exclude**] }
5. *sequence-number* **set service** *service-name* [ **fail-action** { **bypass** | **drop** | **forward**}]
6. [**no**] **load-balance** [ **method** { **src-ip** | **dst-ip**}] [ **buckets** *sequence-number*] [**mask-position** *position-value*]
7. **commit**
8. **end**

## DETAILED STEPS

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **epbr session**<br>**Example:**<br>`switch(config)# epbr session` | Enters ePBR session mode. |
| **Step 2** | **epbr policy** *policy-name*<br>**Example:**<br>`switch(config-epbr-sess)# epbr policy Tenant_A-Redirect` | Specifies the configured ePBR policy in the ePBR session mode. |
| **Step 3** | [**no**] **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] [**l2 address** *ipv6 acl-name*]} **vlan** {**vlan** | **vlan range** | **all**} [**redirect** | **drop** | **exclude**] }<br>**Example:**<br>`switch(config-epbr-sess-pol)# no match ip address WEB` | Disables the IP address matching against the IP or IPv6 ACLs. |
| **Step 4** | **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] [**l2 address** *ipv6 acl-name*]} **vlan** {**vlan** | **vlan range** | **all**} [**redirect** | **drop** | **exclude**] }<br>**Example:**<br>`switch(config-epbr-sess-pol)# match ip address HR` | Modifies the IP address matching against the IP or IPv6 ACLs. |
| **Step 5** | *sequence-number* **set service** *service-name* [ **fail-action** { **bypass** | **drop** | **forward**}]<br>**Example:**<br>`switch(config-epbr-sess-pol-match)# set service firewall fail-action drop` | Adds, modifies, or deletes sequences for a match, or modifies the fail-action for an existing sequence. |
| **Step 6** | [**no**] **load-balance** [ **method** { **src-ip** | **dst-ip**}] [ **buckets** *sequence-number*] [**mask-position** *position-value*]<br>**Example:**<br>`switch(config-epbr-sess-pol-match)# load-balance method src-ip mask-position 3` | Computes the load balance method and the number of buckets to be used by the ePBR service.<br>**Note**<br>On omitting this configuration in the session context while modifying the service-chain for an existing match, the |

| | Command or Action | Purpose |
|---|---|---|
| | | load-balance configuration for the match will be reset to default.<br><br>Beginning with Cisco NX-OS Release 10.3(3)F, the **mask-position** option is provided to choose the bits used for load-balancing in user-defined ACL. Default value is 0.<br><br>If mask-position is configured, the load-balance bits start from configured mask-position. Based on number of buckets needed, more bits are taken to generate load-balancing buckets, toward the most-significant bit.<br><br>**Note**<br>For any ACE in user-defined ACLs, if bits used to generate load-balancing buckets overlap with the user-defined subnet, the mask position for the ACE will be reset internally to 0. |
| **Step 7** | **commit**<br>**Example:**<br>`switch(config-epbr-sess)#commit` | Completes the modification of the ePBR policy using the ePBR session. |
| **Step 8** | **end**<br>**Example:**<br>`switch(config-epbr-sess)#end` | Exits the ePBR session mode. |

# Updating the Access-list Used by ePBR Policies

The following steps explain how to update the access-list used by ePBR policies:

### SUMMARY STEPS

1. **epbr session access-list** *acl-name* **refresh**
2. **end**

### DETAILED STEPS

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **epbr session access-list** *acl-name* **refresh**<br>**Example:**<br>`switch(config)# epbr session access-list WEB refresh` | Updates or refreshes the policy generated ACLs. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **end**<br>**Example:**<br>`switch(config)# end` | Exits the global configuration mode. |

# Configuring ePBR Service Endpoint Out-of-Service

The following section provides information about configuring the ePBR Service Endpoint Out-of-Service.

### SUMMARY STEPS

1. **configure terminal**
2. **epbr service** *service-name*
3. [**no**] **shut**
4. **service-endpoint** [**interface** *interface-name interface-number*]
5. [**no**] **hold-down threshold count** *threshold count* **time** *threshold time*

### DETAILED STEPS

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters configuration mode. |
| **Step 2** | **epbr service** *service-name*<br>**Example:**<br>`switch(config)# epbr service s1` | Enters to the configured service. |
| **Step 3** | [**no**] **shut**<br>**Example:**<br>`switch(config)# shut` | Shuts the endpoint to move it out of service<br>The **no** form of the command shuts the node to bring an endpoint back into service. |
| **Step 4** | **service-endpoint** [**interface** *interface-name interface-number*]<br>**Example:**<br>`switch(config-epbr-svc)# service-end-point ip 1.1.1.1` | Configures service endpoint for the ePBR service.<br>You can repeat steps 2 to 5 to configure another ePBR service. |
| **Step 5** | [**no**] **hold-down threshold count** *threshold count* **time** *threshold time*<br>**Example:** | configures the threshold timers and failure counts at the endpoint level and/or the service level, with the endpoint level parameters overriding the service level parameters. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config)# hold-down threshold count 2 time 5` | For threshold counts greater than 1, timer is mandatory. For threshold count of 1, timer is ignored or disallowed. |

# Configuring ePBR Set-VRF for an ePBR Policy

Beginning with Cisco NX-OS Release 10.5(2)F, ePBR will support the **set-vrf** command for ePBR L3 policies. This enhancement eliminates the need for route-leaking from host VRFs to service VRFs in ePBR inter-VRF deployments.

The **set-vrf** feature allows the traffic from last hop being routed in the host VRF context without route-leaking.

You can configure the **set-vrf** command either at the ePBR policy level or the match level. The match level takes the priority if both are configured.

To configure set-vrf, follow these steps:

### Before you begin

- You must configure one dedicated port-channel interface, and one port-channel sub-interface per host VRF context before applying ePBR policy to an interface.

  The following example illustrates how to create port-channel and port-channel sub-interface for both source-vrf (vrf551) and destination vrf (vrf555):

```
int port-channel 1
   no shut
 int e1/1
   channel-group 1
   link loopback
   no shut
int port-channel 1.1
   encapsulation dot1q 10
   vrf member vrf551
   ip forward
   ipv6 address use-link-local-only
   ipv6 nd dad attempts 0
   ipv6 nd prefix default no-advertise
   ipv6 nd suppress-ra
   mtu 9216
   no shut
int port-channel 1.2
   encapsulation dot1q 11
   vrf member vrf555
   ip forward
   ipv6 address use-link-local-only
   ipv6 nd dad attempts 0
   ipv6 nd prefix default no-advertise
   ipv6 nd suppress-ra
   mtu 9216
   no shut
```

- You must also associate equivalent RPM configuration under the VRF context before applying an ePBR policy.

  The following example illustrates how to create VRF context configuration:

```
vrf context vrf551
    pbr set-vrf recirc interface port-channel1.1
```

```
vrf context vrf555
    pbr set-vrf recirc interface port-channel1.2
```

## SUMMARY STEPS

1. **configure terminal**
2. **epbr policy** *policy-name-IPv4* | *policy-name-IPv6*
3. (Optional) **source-vrf** *source-vrf-name* **destination-vrf** *destination-vrf-name*
4. (Optional) **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] } **source-vrf** *source-vrf-name* **destination-vrf** *destination-vrf-name*

## DETAILED STEPS

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters configuration mode. |
| **Step 2** | **epbr policy** *policy-name-IPv4* \| *policy-name-IPv6*<br>**Example:**<br>For IPV4:<br>`switch(config)# epbr policy p_v4`<br>`switch(config-epbr-policy)#`<br>For IPV6:<br>`switch(config-epbr-policy)# epbr policy p_v6` | Configures an ePBR policy and enters ePBR policy configuration mode. |
| **Step 3** | (Optional) **source-vrf** *source-vrf-name* **destination-vrf** *destination-vrf-name*<br>**Example:**<br>`switch(config-epbr-policy)# source-vrf vrf551 destination-vrf vrf555` | Sets the `destination-vrf` for the forward direction and `source-vrf` for the reverse direction. |
| **Step 4** | (Optional) **match** { [**ip address** *ipv4 acl-name*] \| [**ipv6 address** *ipv6 acl-name*] } **source-vrf** *source-vrf-name* **destination-vrf** *destination-vrf-name*<br>**Example:**<br>For IPv4:<br>`switch(config-epbr-policy)# match ip address acl1 source-vrf vrf551 destination-vrf vrf555`<br>For IPV6:<br>`switch(config-epbr-policy)# match ipv6 address acl1 source-vrf vrf551 destination-vrf vrf555` | Matches an IPv4 or IPv6 ACLs for the specified source and destination VRFs. |

# ePBR Show Commands

The following list provides the show commands associated with ePBR.

### SUMMARY STEPS

1. **show epbr policy** *policy-name* [**reverse**]
2. **show epbr statistics** *policy-name* [**reverse**]
3. **show tech-support epbr**
4. **show running-config epbr**
5. **show startup-config epbr**

### DETAILED STEPS

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **show epbr policy** *policy-name* [**reverse**]<br>**Example:**<br>`switch# show epbr policy Tenant_A-Redirect` | Displays information on the ePBR policy applied in forward or reverse direction. |
| **Step 2** | **show epbr statistics** *policy-name* [**reverse**]<br>**Example:**<br>`switch# show ePBR statistics policy pol2` | Displays the ePBR policy statistics. |
| **Step 3** | **show tech-support epbr**<br>**Example:**<br>`switch# show tech-support epbr` | Displays the technical support information for ePBR. |
| **Step 4** | **show running-config epbr**<br>**Example:**<br>`switch# show running-config epbr` | Displays the running configuration for ePBR. |
| **Step 5** | **show startup-config epbr**<br>**Example:**<br>`switch# show startup-config epbr` | Displays the startup configuration for ePBR |

# Verifying ePBR Configuration

To verify the ePBR configuration, use the following commands:

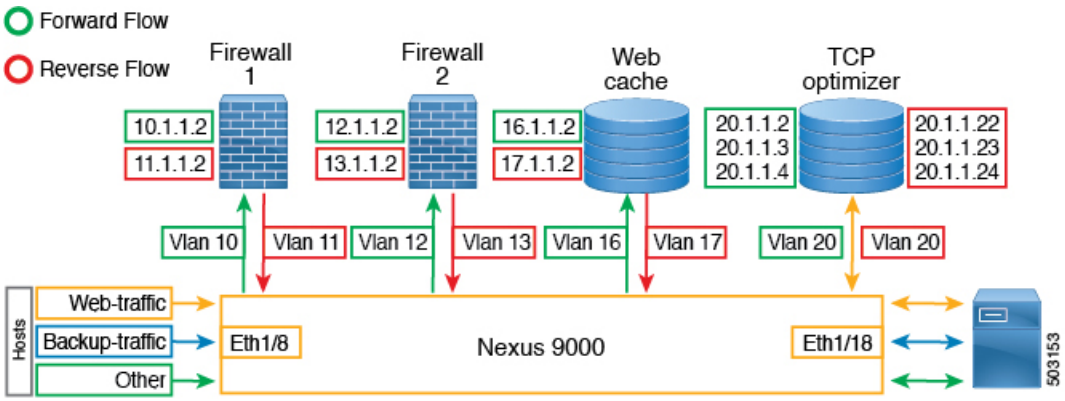
| Command | Purpose |
|---|---|
| **show ip/ipv6 policy vrf <context>** | Displays the IPv4/IPv6 route-map policies created for the Layer-3 ePBR policy, at the interfaces at which service chain is applied and the relevant end-point interfaces of the service-chain. |
| **show route-map dynamic <route-map name>** | Displays the next-hops configured for traffic re-direction for specific bucket access-lists, used for forwarding traffic at every point in the service chain. |
| **show ip/ipv6 access-list <access-list name> dynamic** | Displays the traffic match criteria for a bucket access-list. |
| **show ip sla configuration dynamic** | Displays the IP SLA configuration generated by ePBR, for the service-end-points in the chain, when probes are enabled. |
| **show track dynamic** | Displays the tracks generated by ePBR, for the service-end-points in the chain, when probes are enabled. |

# Configuration Examples for ePBR L3

### Example: ePBR NX-OS Configuration

The following topology illustrates ePBR NX-OS configuration.

***Figure 1: ePBR NX-OS Configuration***




### Example: Use-Case: Create a Service Chain for Web Traffic in Forward Direction Only

The following configuration example shows how to create a service chain for web traffic in forward direction only.

```
IP access list web_traffic
        10 permit tcp any any eq www

ePBR service FW1
  service-end-point ip 10.1.1.2 interface Vlan10
```

```
    reverse interface Vlan11

ePBR service FW2
  service-end-point ip 12.1.1.2 interface Vlan12
    reverse interface Vlan13

ePBR service Web_cache
  service-end-point ip 16.1.1.2 interface Vlan16
    reverse interface Vlan17

ePBR policy tenant_1
  match ip address web-traffic
    10 set service FW1
    20 set service FW2
    30 set service Web_cache

interface Eth1/8
  ePBR ip policy tenant_1
```

The following example shows how to verify the configuration of service chain creation for web traffic in forward direction.

```
switch# show ePBR policy tenant_1

Policy-map : tenant_1
  Match clause:
    ip address (access-lists): web-traffic
  Service chain:
    service FW1, sequence 10, fail-action No fail-action
      IP 10.1.1.2
    service FW2, sequence 20, fail-action No fail-action
      IP 12.1.1.2
    service Web_cache, sequence 30, fail-action No fail-action
      IP 16.1.1.2
  Policy Interfaces:
    Eth1/8
```

**Example: Use-Case : Load Balance TCP Traffic Using ePBR in Forward Direction Only**

The following configuration example shows how to load balance TCP traffic using ePBR in forward direction only.

```
IP access list tcp_traffic
        10 permit tcp any any

ePBR service TCP_Optimizer
  service-interface Vlan20
  service-end-point ip 20.1.1.2
  service-end-point ip 20.1.1.3
  service-end-point ip 20.1.1.4

ePBR policy tenant_1
  match ip address tcp_traffic
    10 set service TCP_Optimizer

interface Eth1/8
  ePBR ip policy tenant_1
```

The following example shows how to verify the configuration of load balance TCP traffic using EPBR in forward direction.

```
switch# show ePBR policy tenant_1

Policy-map : tenant_1
  Match clause:
```

```
    ip address (access-lists): tcp_traffic
  Service chain:
    service TCP_Optimizer, sequence 10, fail-action No fail-action
      IP 20.1.1.2
      IP 20.1.1.3
      IP 20.1.1.4
  Policy Interfaces:
    Eth1/8
```

### Example: Use-Case: Create a Service Chain for Web Traffic in Both Directions

The following configuration example shows how to create a service chain for web traffic in both forward and reverse directions.

```
IP access list web_traffic
        10 permit tcp any any eq www


ePBR service FW1
  service-end-point ip 10.1.1.2 interface Vlan10
    reverse ip 11.1.1.2 interface Vlan11

ePBR service FW2
  service-end-point ip 12.1.1.2 interface Vlan12
    reverse ip 13.1.1.2 interface Vlan13

ePBR service Web_cache
  service-end-point ip 16.1.1.2 interface Vlan16
    reverse ip 17.1.1.2 interface Vlan17

ePBR policy tenant_1
  match ip address web-traffic
    10 set service FW1
    20 set service FW2
    30 set service Web_cache

interface Eth1/8
  ePBR ip policy tenant_1

interface Eth1/18
  ePBR ip policy tenant_1 reverse
```

The following example shows how to verify the configuration of service chain creation for web traffic in both forward and reverse directions.

```
switch# show ePBR policy tenant_1

Policy-map : tenant_1
  Match clause:
    ip address (access-lists): web-traffic
  Service chain:
    service FW1, sequence 10, fail-action No fail-action
      IP 10.1.1.2
    service FW2, sequence 20, fail-action No fail-action
      IP 12.1.1.2
    service Web_cache, sequence 30, fail-action No fail-action
      IP 16.1.1.2
  Policy Interfaces:
    Eth1/8

switch# show ePBR policy tenant_1 reverse

Policy-map : tenant_1
  Match clause:
    ip address (access-lists): web-traffic
```

```
  Service chain:
    service Web_cache, sequence 30, fail-action No fail-action
      IP 17.1.1.2
    service FW2, sequence 20, fail-action No fail-action
      IP 13.1.1.2
    service FW1, sequence 10, fail-action No fail-action
      IP 11.1.1.2
  Policy Interfaces:
    Eth1/18
```

### Example: Use-Case: Load Balance TCP Traffic Using ePBR in Both Directions

The following configuration example shows how to load balance TCP traffic using ePBR in both forward and reverse directions.

```
ePBR service TCP_Optimizer
  service-interface Vlan20
  service-end-point ip 20.1.1.2
    reverse ip 20.1.1.22
  service-end-point ip 20.1.1.3
    reverse ip 20.1.1.23
  service-end-point ip 20.1.1.4
    reverse ip 20.1.1.24

ePBR policy tenant_1
  match ip address tcp_traffic
    10 set service TCP_Optimizer

interface Eth1/8
  ePBR ip policy tenant_1

interface Eth1/18
  ePBR ip policy tenant_1 reverse
```

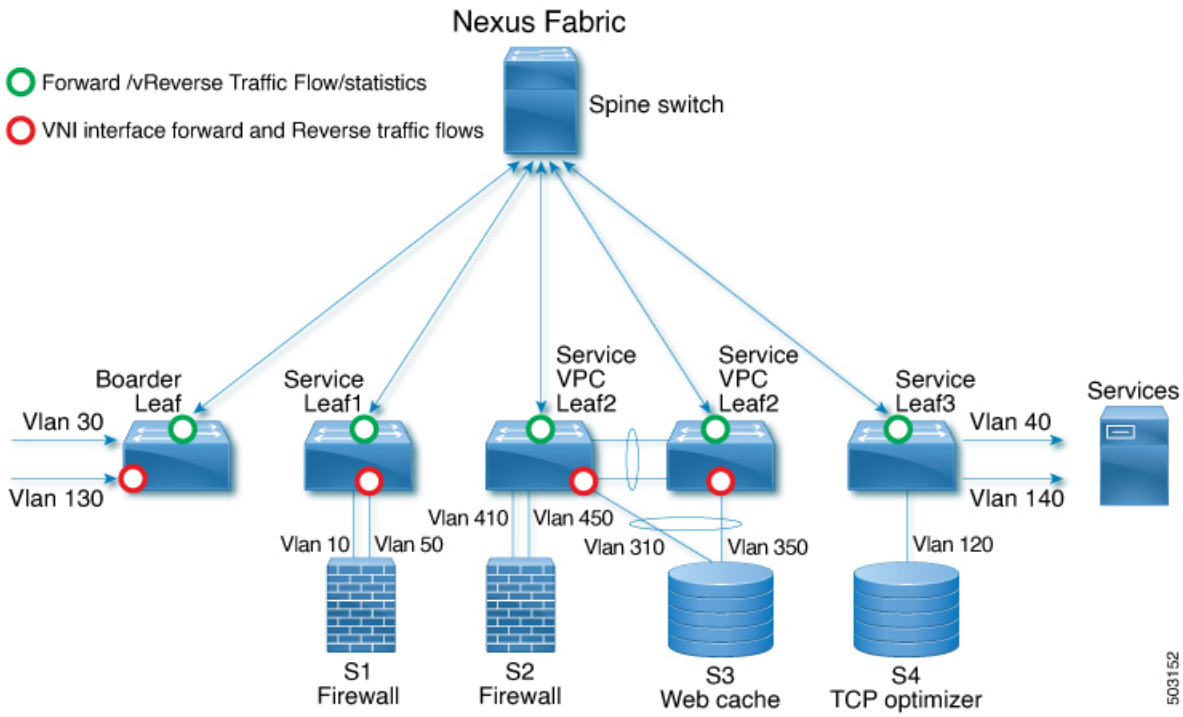
The following example shows how to verify the configuration of load balance TCP traffic using ePBR in both directions.

```
switch# show ePBR policy tenant_1

Policy-map : tenant_1
  Match clause:
    ip address (access-lists): tcp_traffic
  Service chain:
    service TCP_Optimizer, sequence 10, fail-action No fail-action
      IP 20.1.1.2
      IP 20.1.1.3
      IP 20.1.1.4
  Policy Interfaces:
    Eth1/8

switch# show ePBR policy tenant_1 reverse

Policy-map : tenant_1
  Match clause:
    ip address (access-lists): tcp_traffic
  Service chain:
    service TCP_Optimizer, sequence 10, fail-action No fail-action
      IP 20.1.1.22
      IP 20.1.1.23
      IP 20.1.1.24
  Policy Interfaces:
    Eth1/18
```

### Example: ePBR Policy Creation with VXLAN Fabric

The following example/topology shows how to configure ePBR over VXLAN fabric.

***Figure 2: Configuring ePBR over VXLAN Fabric***




```
ip access-list acl1
        10 permit ip 30.1.1.0/25 40.1.1.0/25
        20 permit ip 30.1.1.128/25 40.1.1.128/25
ip access-list acl2
        10 permit ip 130.1.1.0/25 140.1.1.0/25
        20 permit ip 130.1.1.128/25 140.1.1.128/25

epbr service s1
  vrf vrf_s1
  service-end-point ip 10.1.1.2 interface Vlan10
     probe icmp frequency 4 retry-down-count 1 retry-up-count 1 timeout 2 source-interface
 loopback9
    reverse ip 50.1.1.2 interface Vlan50

      probe icmp frequency 4 retry-down-count 1 retry-up-count 1 timeout 2 source-interface
 loopback10

epbr service s2
  vrf vrf_s2
  service-end-point ip 41.1.1.2 interface Vlan410
     probe icmp source-interface loopback11
    reverse ip 45.1.1.2 interface Vlan450

       probe icmp source-interface loopback12

 epbr service s3
  vrf vrf_s3
  service-end-point ip 31.1.1.2 interface Vlan310
     probe http get index.html source-interface loopback13
    reverse ip 35.1.1.2 interface Vlan350
```

```
     probe http get index.html source-interface loopback14

epbr service s4
  service-interface Vlan120
  vrf vrf_s4
   probe udp 6900 control enable source-interface loopback15
  service-end-point ip 120.1.1.2

    reverse ip 120.1.1.2

epbr policy p1
 statistics
  match ip address acl1
    load-balance buckets 16 method src-ip
    10 set service s1 fail-action drop
    20 set service s2 fail-action drop
    30 set service s4 fail-action bypass
  match ip address acl2
    load-balance buckets 8 method dst-ip
    10 set service s1 fail-action drop
    20 set service s3 fail-action forward
    30 set service s4 fail-action bypass

! VXLAN L3 VNI interface for vrf_s1, vrf_s2, vrf_s3, vrf_s4 to which the policy is applied
 on all service leafs
interface vlan 100
epbr ip policy p1
epbr ip policy p1 reverse

interface vlan 101
epbr ip policy p1
epbr ip policy p1 reverse

interface vlan 102
epbr ip policy p1
epbr ip policy p1 reverse

interface vlan 103
epbr ip policy p1
epbr ip policy p1 reverse

Apply forward policy on ingress interface in border leaf where traffic coming in needs to
be service-chained:

interface Vlan 30 - Traffic matching acl1
  epbr ip policy p1
  int vlan 130  -  Traffic matching acl2
  epbr ip policy p1

Apply the reverse policy On leaf connected to server if reverse traffic flow needs to be
enabled:

int vlan 40 - Traffic matching reverse flow for acl1
epbr ip policy p1 rev
int vlan 140 - Traffic matching reverse flow for acl1
epbr ip policy p1 rev
```

### Example: Configuring ePBR Service

The following example shows how to configure ePBR service.

```
epbr service FIREWALL
  probe icmp
  vrf TENANT_A
```

```
  service-endpoint ip 172.16.1.200 interface VLAN100
        reverse ip 172.16.2.200 interface VLAN101
service-endpoint ip 172.16.1.201 interface VLAN100
        reverse ip 172.16.2.201 interface VLAN101

epbr service TCP_Optimizer
   probe icmp
   vrf TENANT_A
service-endpoint ip 172.16.20.200 interface VLAN200
       reverse ip 172.16.30.200 interface VLAN201
```

### Example: Configuring ePBR Policy

The following example shows how to configure ePBR Policy.

```
epbr service FIREWALL
  probe icmp
  service-end-point ip 1.1.1.1 interface Ethernet1/1
     reverse ip 1.1.1.2 interface Ethernet1/2
epbr service TCP_Optimizer
 probe icmp
 service-end-point ip 1.1.1.1 interface Ethernet1/3
     reverse ip 1.1.1.4 interface Ethernet1/4
epbr policy Tenant_A-Redirect
 match ip address WEB
  load-balance method src-ip
  10 set service FIREWALL fail-action drop
  20 set service TCP_Optimizer fail-action bypass
match ip address APP
  10 set service FIREWALL fail-action drop
match ip address exclude_acl exclude
match ip address drop_acl drop
```

The following example shows the output of show ePBR Policy command with fail-action drop information.

```
switch(config-if)# show epbr policy Tenant_A-Redirect

Policy-map : Tenant_A-Redirect
  Match clause:
    ip address (access-lists): WEB
action:Redirect
   service FIREWALL, sequence 10, fail-action Drop
     IP 1.1.1.1 track 1 [INACTIVE]
   service TCP_Optimizer, sequence 20, fail-action Bypass
     IP 1.1.1.1 track 2 [INACTIVE]
Match clause:
     ip address (access-lists): APP
action:Redirect
   service FIREWALL, sequence 10, fail-action Drop
     IP 1.1.1.1 track 1 [INACTIVE]
Match clause:
    ip address (access-lists): exclude_acl
action:Deny
Match clause:
   ip address (access-lists): drop_acl
action:Drop
 Policy Interfaces:
  Eth1/4
```

### Example: Associating an Interface with ePBR Policy

The following example shows how to configure ePBR Policy.

```
interface vlan 2010
  epbr ip policy Tenant_A-Redirect

interface vlan 2011
  epbr ip policy Tenant_A-Redirect reverse
```

### Example: ePBR Policy applied in forward direction

The following example shows the sample Output for policy applied in forward direction.

```
show epbr policy Tenant_A-Redirect
policy-map Tenant_A-Redirect
 Match clause:
  ip address (access-lists): WEB
 Service chain:
  service FIREWALL  , sequence 10 , fail-action drop
   ip 172.16.1.200 track 10  [ UP ]
   ip 172.16.1.201 track 11 [ DOWN ]
                    service TCP_Optimizer, sequence 20 , fail-action bypass
   ip 172.16.20.200 track 12  [ UP] ]

 Match clause:
  ip address (access-lists): APP
 Service chain:
  service FIREWALL  , sequence 10 , fail-action drop
   ip 172.16.1.200 track 10  [ UP ]
   ip 172.16.1.201 track 11 [ DOWN ]

 Policy Interfaces:
     Vlan 2010
```

### Example: ePBR Policy applied in reverse direction

The following example shows the sample Output for policy applied in reverse direction.

```
show epbr policy Tenant_A-Redirect reverse
policy-map Tenant_A-Redirect
 Match clause:
  ip address (access-lists): WEB

 Service chain:
  service TCP_Optimizer, sequence 20 , fail-action bypass
   ip 172.16.30.200 track 15  [ UP] ]

  service FIREWALL  , sequence 10 , fail-action drop
   ip 172.16.2.200 track 13  [ UP ]
   ip 172.16.2.201 track 14 [ DOWN ]

 Match clause:
  ip address (access-lists): APP

 Service chain:

  service FIREWALL  , sequence 10 , fail-action drop
   ip 172.16.2.200 track 13  [ UP ]
   ip 172.16.2.201 track 14 [ DOWN ]

 Policy Interfaces:
     Vlan 2011
```

### Example: User-defined Track

The following example shows to assign track ID to each end point.

```
epbr service FIREWALL
  probe icmp
```

```
  service-end-point ip 1.1.1.2 interface Ethernet1/21
probe track 30
reverse ip 1.1.1.3 interface Ethernet1/22
  probe track 40
 service-end-point ip 1.1.1.4 interface Ethernet1/23
    reverse ip 1.1.1.5 interface Ethernet1/24
```

### Example: Modifying ePBR Service Using ePBR Session

The following example shows to replace the IP of ePBR service and add another service end point.

```
switch(config)#epbr session
switch(config-epbr-sess)#epbr service TCP_OPTIMIZER
switch(config-epbr-sess-svc)# no service-end-point ip 172.16.20.200 interface VLAN200
switch(config-epbr-sess-svc)#service-end-point ip 172.16.25.200 interface VLAN200
switch(config-epbr-sess-svc-ep)# reverse ip 172.16.30.200 interface VLAN201
switch(config-epbr-sess)#commit
```

### Example: Modifying ePBR Policy Using EPBR Session

The following example shows to replace the IP of ePBR policy and add a service chain for the modified policy traffic.

```
switch(config)#epbr session
switch(config-epbr-sess)#epbr policy Tenant_A-Redirect
switch(config-epbr-sess-pol)# no match ip address WEB
switch(config-epbr-sess-pol)#match ip address WEB
switch(config-epbr-sess-pol-match)# 10  set service Web-FW fail-action drop load-balance
method src-ip
switch(config-epbr-sess-pol-match)# 20 set service TCP_Optimizer fail-action bypass
switch(config-epbr-sess-pol)#match ip address HR
switch(config-epbr-sess-pol-match)# 10  set service Web-FW
switch(config-epbr-sess-pol-match)# 20 set service TCP_Optimizer
switch(config-epbr-sess)#commit
```

### Example: Displaying ePBR Statistics Policy

The following example shows the display of ePBR statistics policy.

```
switch# show epbr statistics policy pol2

Policy-map pol2, match testv6acl

    Bucket count: 2

      traffic match : epbr_pol2_1_fwd_bucket_1
        two : 0
      traffic match : epbr_pol2_1_fwd_bucket_2
        two : 0
```

### Example: Displaying how mask-position is used

The following example shows the sample of how mask-position is used:

```
IP access list acl1
        10 permit tcp 10.0.0.0/24 any
epbr policy l3_Pol
  statistics match ip address acl1
  load-balance buckets 4 mask-position 5
10 set service s1_l3
switch# show ip access-list dynamic
IP access list epbr_l3_Pol_1_fwd_bucket_1
 10 permit tcp 10.0.0.0 0.0.0.159 any
IP access list epbr_l3_Pol_1_fwd_bucket_2
 10 permit tcp 10.0.0.32 0.0.0.159 any
IP access list epbr_l3_Pol_1_fwd_bucket_3
 10 permit tcp 10.0.0.64 0.0.0.159 any
```

```
IP access list epbr_l3_Pol_1_fwd_bucket_4
 10 permit tcp 10.0.0.96 0.0.0.159 any
```

# Additional References

For additional information related to configuring ePBR, see the following sections:

## Related Documents

| Related Topic | Document Title |
|---|---|
| Configuring CoPP for IP SLA Packets | *Cisco Nexus 9000 Series NX-OS IP SLAs Configuration Guid 9.3(x)* |
| ePBR Licensing | *Cisco NX-OS Licensing Guide* |
| ePBR Scale Values | *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide* |

## Standards

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