Configuring ePBR L2

Information About ePBR L2

Enhanced Policy-based Redirect Layer2 (ePBR) in Elastic Services Re-direction (ESR) provides transparent service redirection and service chaining of Layer1/ Layer2 service appliances by leveraging Port ACL and VLAN translation. This action helps achieve service chaining and load-balancing capabilities 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:

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.

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 L2 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 the NX-OS switch and the interface through which traffic needs to exit the switch after redirection or service-chaining. You can also apply the policy in both the forward and reverse directions into the NX-OS switch.

Enabling Production Interfaces as Access Port

If the service-chaining switch is inserted in between the two L3 routers for traffic redirection, the production interfaces are enabled as access port with the following limitations:

- You must use the VLAN of the port as part of the match configuration.
- It is limited to mac-learn disable mode.

Enabling Production Interfaces as Trunk Ports

Production interfaces may be configured as trunk ports. The VLANs of the incoming traffic that needs to be service-chained that is trunked by the interfaces must be configured as part of the match configuration.

Alternatively, using 'vlan all' in the match configuration will allow any traffic pertaining to any incoming VLANs on the interface to be matched and service chained.

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 Object Tracking, Health Monitoring, and Fail-Action

Layer-2 ePBR performs link state monitoring of the service end-points by default. The user may additionally enable CTP (Configuration Testing Protocol) if supported by the service.

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, and retry up and down counts. The same track objects is re-used for all policies using the same ePBR service.

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 forward to the egress interface. 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.

Beginning with Cisco NX-OS Release 10.4(1)F, ePBR L2 fail-action feature is optimized to modify only the ACEs that are currently affected by the failure of the node. However, the fail-action optimization will be enabled only for those service-chains where the user configures **load-balance buckets** under the ePBR match statement.

The fail-action optimization is supported on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2, C9364C, C9332C, and Cisco Nexus 9500 switches with 9700-EX/FX/GX line cards.

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.

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 L2

ePBR has the following guidelines and limitations:

- When fail-action is specified in any match statement, probe is mandatory in the configuration.
- To disable MAC learning on the switch, use the command **mac-learn disable**.
- 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.
- 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.
- ePBRv6 over VXLAN is not supported on Cisco Nexus 9500 series switches.
- 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.

- All redirection rules are programmed in ACL TCAM using ing-ifacl region. This region needs to be carved and allocated prior to the application of ePBR L2 policies.

Note For steps on how to carve TCAM region, refer to the **Configuring IP ACLs** section of *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.

- ePBR policies require at least one match with redirect action.
- ePBR L2 requires a VLAN range to be reserved for VLAN translation and Q-in-Q. It is recommended that this range does not overlap with the VLANs used for traffic match configuration.
- The ePBR 'infra' VLANs should be reserved prior to the application of the ePBR Layer-2 policies.
- For production interfaces configured as trunk ports, enable VLAN trunking only for the VLANs specified in the ePBR 'infra vlan' range.
- ePBR L2 expects the service appliance to be configured to forward the packet as is without modifying or stripping the VLAN headers.
- Each match in an ePBR L2 policy needs to have a unique match VLAN or unique VLAN range when applied on trunk interfaces. Only a single match with 'vlan all' can exist in a policy that is applied on trunk interfaces.
- ePBR L2 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 10.3(1)F, multiple matches in the same EPBR L2 policy may share the same VLAN or VLAN range or may be configured with ‘vlan all’ in a policy that is applied on trunk interfaces.

Note Ensure that the ACL filters across the configured match ACLs are unique and do not overlap when multiple match ACLs of the same address family (IPv4, ipv6, or L2) share the same VLANs in a policy.

- For a production port pair, the policy that is applied on an interface in the forward direction and on its reverse interface in the reverse direction, should consist of matches, that are individually mapped to identical match-vlans or vlan ranges.
- In order to load-balance between multiple service devices and uniquely detect failure of these devices via CTP health-checks, each service device should be defined as a unique endpoint in the ePBR service.
- Bucket-based load-balance is not supported for layer-2 matches in the ePBR policy.
- In order to service-chain or redirect IPv6 traffic such as Neighbor discovery, ICMPv6 aces with protocol types of ND-NA and ND-NS should be explicitly defined in the user-defined match access-list.
- In order to service-chain or redirect Layer-2 traffic for protocols such as ARP (0x806), VN-tag (0x8926), FCOE (0x8906), MPLS Unicast (0x8847), MPLS Multicast (0x8848), the protocol information should be explicitly added to the ACEs inside the user-defined match access-list.
- Beginning with Cisco NX-OS Release 10.4(1)F, ePBR L2 supports redirection of all control traffic that matches the ePBR policy. For more information, see Enforcing Redirection and Drop for Control Traffic, on page 13 section.
- Defaulting ePBR production and/or service interfaces while they are in use should be avoided to prevent any unintended behavior.
- Beginning with Cisco NX-OS Release 10.3(1)F, ePBR L2 supports only redirection of L2 control packets on Cisco Nexus 9300-GX platform switches. Service-chaining is not supported on Cisco Nexus 9300-GX platform switches.

- Beginning with Cisco NX-OS Release 10.4(1)F, the ePBR provides **mask-position** option to choose the bits used for load-balancing in user-defined ACL for IPv4 or IPv6 matches on Cisco Nexus 9300-EX/FX/FX2/FX3/GX/GX2, and Cisco Nexus 9500 platform switches with Nexus 9700- EX/FX/GX line cards.
- 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.

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.
- 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 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.

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. **[no] epbr infra vlans** *[vlan range]*
3. **epbr service** *service-name* **type** *l2*
4. **mode** [**full duplex** | **half duplex**]
5. **probe** {**ctp**} [**frequency** *seconds*] [**timeout** *seconds*] [**retry-down-count** *count*] **retry-up-count** *count*]
6. **service-endpoint** [**interface** *interface-name interface-number*]
7. **reverse interface** *interface-name interface-number*
8. **exit**
9. **epbr policy** *policy-name*
10. **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] | [**l2 address** *l2 acl-name*]} {**drop** | **exclude** | **redirect** | **vlan**{**vlan** | **vlan range** | **all**}}
11. [**no**] **load-balance** [**method** { **src-ip** | **dst-ip**}] [**buckets** *count*] [**mask-position** *position-value*]
12. *sequence-number* **set service** *service-name* [**fail-action** { **bypass** | **drop** | **forward**}]
13. **interface** *interface-name interface-number*
14. **epbr** {**l2**} **policy** *policy-name egress-interface interface-name* [**reverse**]
15. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters configuration mode.

Step 2	**[no] epbr infra vlans** *[vlan range]*	VLAN range is using to indicate the VLANs reserved for selective dot1q translation while redirecting to the service devices.
Step 3	**epbr service** *service-name* **type** *l2* **Example:** `switch(config)# epbr service firewall type l2`	Creates a new ePBR L2 service.
Step 4	**mode** [**full duplex**	**half duplex**]	Configures the service to be in half-duplex or full-duplex mode.
Step 5	**probe** {**ctp**} [**frequency** *seconds*] [**timeout** *seconds*] [**retry-down-count** *count*] **retry-up-count** *count*]	Configures the probe for the ePBR service.

	Command or Action	Purpose
	Example: `switch(config)# probe icmp`	The options are as follows: • frequency—Specifies the frequency of the probe in seconds. The range is from 1 to 604800. • retry-down-count —Specifies the number of recounts undertaken by the probe when the node goes down. The range is from 1 to 5. • 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. • timeout —Specifies the length of the timeout period in seconds. The range is from 1 to 604800.
Step 6	**service-endpoint** [**interface** *interface-name interface-number*] **Example:** `switch(config-epbr-svc)# service-end-point interface Ethernet1/3`	Configures service endpoint for the ePBR service. You can repeat steps 2 to 5 to configure another ePBR service.
Step 7	**reverse interface** *interface-name interface-number* **Example:** `switch(config-epbr-fwd-svc)# reverse interface Ethernet1/4`	Defines the reverse interface where the traffic policies are applied.

Step 8	**exit**
	Example:
	`switch(config-epbr-reverse-svc)# exit`
	`switch(config-epbr-fwd-svc)# exit`
	`switch(config-epbr-svc)# exit`
	`switch(config)#`
		Exits ePBR service configuration mode and enters global configuration mode.

Step 9	**epbr policy** *policy-name* **Example:** `switch(config)# epbr policy Tenant_A-Redirect`	Configures the ePBR policy.
Step 10	**match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] | [**l2 address** *l2 acl-name*]} {**drop** | **exclude** | **redirect** | **vlan**{**vlan** | **vlan range** | **all**}} **Example:** `switch (config) # match ip address WEB vlan 10`	Matches an IPv4, or IPv6 address, or a mac address against an IP, or IPv6, or MAC 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. You can repeat this step to match multiple ACLs based on the requirement.
Step 11	[**no**] **load-balance** [**method** { **src-ip** | **dst-ip**}] [**buckets** *count*] [**mask-position** *position-value*] **Example:**	Computes the load balance method and the number of buckets to be used by the ePBR service.

	Command or Action	Purpose
	`switch(config)# load-balance method src-ip mask-position 3`	Beginning with Cisco NX-OS Release 10.4(1)F, the **mask-position** option is provided to choose the bits used for load-balancing in user-defined ACL for IPv4 or IPv6 matches. Default value is 0. 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 bits direction depending on whether the least significant bit or most significant bit is selected. **Note** 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**}] **Example:** `switch(config)# set service firewall fail-action drop`	Configures the fail-action mechanism.
Step 13	**interface** *interface-name interface-number* **Example:** `switch(config)# interface Ethernet1/1`	Enters into interface configuration mode.
Step 14	**epbr** {**l2**} **policy** *policy-name egress-interface interface-name* [**reverse**] **Example:** `epbr l2 policy Tenant_A_Redirect egress-interface Ethernet1/2`	An interface may be associated at any time with one forward policy and one reverse policy of the following: • an IPV4 policy in the forward direction • an IPv4 policy in the reverse direction • an IPv6 policy in the forward direction • an IPv6 policy in the reverse direction • a l2 policy in the forward direction • a l2 policy in the reverse direction
Step 15	**exit** **Example:** `switch(config-if)# end`	Exits policy configuration mode and returns to global 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 type l2*
3. [**no**] **service-endpoint** [**interface** *interface-name*]
4. **service-endpoint** [**interface** *interface-name*]
5. **reverse** [**interface** *interface-name*]
6. **commit**
7. **abort**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**epbr session** **Example:** `switch(config)# epbr session`	Enters ePBR session mode.
Step 2	**epbr service** *service-name type l2* **Example:** `switch(config-epbr-sess)# epbr service TCP_OPTIMIZER`	Specifies the configured ePBR service in the ePBR session mode.
Step 3	[**no**] **service-endpoint** [**interface** *interface-name*] **Example:** `switch(config-epbr-sess-svc)# no service-end-point interface ethernet 1/3`	Disables the configured service endpoint for the ePBR service.
Step 4	**service-endpoint** [**interface** *interface-name*] **Example:** `switch(config-epbr-sess-svc)# service-end-point interface ethernet 1/15`	Add a service endpoint to the service.
Step 5	**reverse** [**interface** *interface-name*] **Example:** `switch(config-epbr-sess-fwd-svc)# reverse interface ethernet 1/4`	Defines the reverse interfaces where the traffic policies are applied.
Step 6	**commit** **Example:** `switch(config-epbr-sess)#commit`	Completes the modification of the ePBR service using the ePBR session. **Note** Restart the ePBR session after you complete this step.
Step 7	**abort** **Example:** `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. **Note** 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** *mac acl-name*]} **vlan** {**all** | **vlan-id** | **vlan-id-range**
4. **match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] | **l2 address** *mac acl-name*]} **vlan** {**all** | **vlan-id** | **vlan-id-range**]
5. *sequence-number* **set service** *service-name* [**fail-action** { **bypass** | **drop** | **forward**}]
6. [**no**] **load-balance** [**method** { **src-ip** | **dst-ip**}] [**buckets** *count*] [**mask-position** *position-value*]
7. **commit**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**epbr session**	
Step 2	**epbr policy** *policy-name* **Example:** `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** *mac acl-name*]} **vlan** {**all** | **vlan-id** | **vlan-id-range** **Example:** `switch(config-epbr-sess-pol)# no match ip address WEB`	Disables the match against IP, IPv6, or L2 ACLs.
Step 4	**match** { [**ip address** *ipv4 acl-name*] | [**ipv6 address** *ipv6 acl-name*] | **l2 address** *mac acl-name*]} **vlan** {**all** | **vlan-id** | **vlan-id-range**] **Example:** `switch(config-epbr-sess-pol)# match ip address HR`	Modifies the match against the IP, IPv6 or L2 ACLs.
Step 5	*sequence-number* **set service** *service-name* [**fail-action** { **bypass** | **drop** | **forward**}] **Example:** `switch(config-epbr-sess-pol-match)# set service firewall fail-action drop`	Configures the fail-action mechanism.
Step 6	[**no**] **load-balance** [**method** { **src-ip** | **dst-ip**}] [**buckets** *count*] [**mask-position** *position-value*]	Configures the load-balance method and buckets for the match.

	Command or Action	Purpose
	Example: `switch(config)# load-balance method src-ip mask-position 3`	**Note** On omitting this configuration in the session context while modifying the service-chain for an existing match, the load-balance configuration for the match will be reset to default. Beginning with Cisco NX-OS Release 10.4(1)F, the **mask-position** option is provided to choose the bits used for load-balancing in user-defined ACL for IPv4 or IPv6 matches. Default value is 0. 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 bits direction depending on whether the least significant bit or most significant bit is selected. **Note** 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** **Example:** `switch(config-epbr-sess)#commit`	Completes the modification of the ePBR policy using the ePBR session.
Step 8	**end** **Example:** `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

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

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

Enforcing Redirection and Drop for Control Traffic

Beginning with Cisco NX-OS Release 10.4(1)F, the following configuration options may be used to control redirection and drop behavior for control traffic through an ePBR L2 policy.

The **all** configuration option is used inside the ACEs in the user-defined match access-list for ePBR, in order to indicate that the highest priority is needed for an ACE . See **Applying an IP ACL Rule Prioritization over SUP Rule** or **Applying a MAC ACL Rule Prioritization over SUP Rule** section of *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* for details on this configuration.

If the **all** option is used, the following behavior is observed:

- For matches with **redirection** or **exclude** action, ePBR generates corresponding redirection ACEs to enforce redirection of all matching traffic, including control traffic toward the specified service devices or the egress interface respectively.
- For matches with **drop** action, ePBR generates deny ACEs to enforce the drop of all matching traffic, including control traffic. If this option is not detected as configured, any control traffic that is typically copied or redirected to the supervisor on Cisco NX-OS 9000 series switches, may continue to do so, even if it matches the ePBR Layer-2 policy definition.

The **all** option has no effect if the match access-lists are used inside ePBR Layer-3 policies.

The **default-traffic-action redirect-all** configuration option is used inside an ePBR Layer-2 policy to specify that any traffic that does not match redirect, exclude, or drop matches, including control traffic must be redirected toward the specified egress interface. If this option is not configured, any control traffic that does not match the access-lists inside the policy, and which is typically copied or redirected to the supervisor on Cisco NX-OS 9000 series switches, may continue to do so, instead of redirecting to the egress interface.

You can configure the default catch-all traffic behavior at a policy level using the following commands.

SUMMARY STEPS

1. **configure terminal**
2. **epbr policy** *policy-name*
3. **default-traffic-action** [**redirect** | **redirect-all**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose
Step 2	**epbr policy** *policy-name* **Example:** `switch(config)# epbr policy p3`	Configures the ePBR policy.
Step 3	**default-traffic-action** [**redirect** | **redirect-all**] **Example:** `switch(config-epbr-policy)# default-traffic-action redirect-all`	Sets the default catch-all behavior for an ePBR policy. • **redirect**: Redirects the data traffic. **redirect** is the default option. • **redirect-all**: Redirects all traffic. **Note** • This option is not supported inside Layer-3 ePBR policies. • This option cannot be modified inside ePBR sessions and requires the policy to be disabled, re-configured, and applied back.

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

	Command or Action	Purpose
Step 1	**show epbr policy** *policy-name* [**reverse**] **Example:** `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**] **Example:** `switch# show ePBR statistics policy pol2`	Displays the ePBR policy statistics.
Step 3	**show tech-support epbr** **Example:** `switch# show tech-support epbr`	Displays the technical support information for ePBR.

	Command or Action	Purpose
Step 4	**show running-config epbr** **Example:** `switch# show running-config epbr`	Displays the running configuration for ePBR.
Step 5	**show startup-config epbr** **Example:** `switch# show startup-config epbr`	Displays the startup configuration for ePBR

Verifying ePBR Configuration

To verify the ePBR configuration, use the following commands:

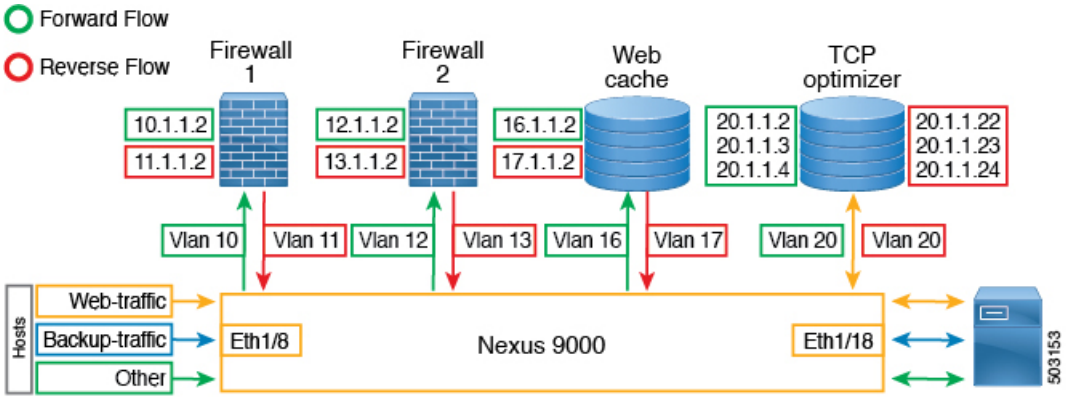
Command	Purpose
show ip access-list <access-list name> dynamic	Displays the traffic match criteria for a bucket access-list.
show ip sla configuration dynamic	Displays the IP SLA configuration generated by ePBR, for the service-end-points in the chain, when probes are enabled.
show track dynamic	Displays the tracks generated by ePBR, for the service-end-points in the chain, when probes are enabled.
show ip access-list summary	Displays the summary of the traffic match criteria for a bucket access-list.
show [ip | ipv6 | mac] access-lists dynamic	Displays the dynamic entries of match criteria.

Configuration Examples for ePBR

Example: ePBR NX-OS Configuration

The following topology illustrates ePBR NX-OS configuration:

Figure 1: ePBR NX-OS Configuration



Example: Service Configuration for Access and Trunk Ports

The following configuration example shows how to perform service configuration for access and trunk ports:

```
epbr infra vlans 100-200

epbr service app_1 type l2
   service-end-point interface Ethernet1/3
    reverse  interface Ethernet1/4

epbr service app_2 type l2
   probe ctp frequency 2 retry-down-count 1 retry-up-count 1 timeout 1
   service-end-point interface port-channel10
    reverse  interface port-channel11

epbr service app_3 type l2
   probe ctp frequency 2 retry-down-count 1 retry-up-count 1 timeout 1
   service-end-point interface Ethernet1/9
    reverse  interface Ethernet1/10

epbr service app_4 type l2
   probe ctp frequency 2 retry-down-count 1 retry-up-count 1 timeout 1
   service-end-point interface port-channel12
    reverse  interface port-channel13
```

Example: Configuring Access Ports

The following example shows how to configure access ports:

```
epbr policy p1
  statistics
  match ipv6 address flow2 vlan 10
    load-balance buckets 2
    10 set service app_1
    20 set service app_3
    25 set service app_4
    30 set service app_2
  match l2 address flow3 vlan 10
    20 set service app_2
    25 set service app_4
    50 set service app_3
  match ip address flow1 vlan 10
    10 set service app_1
    15 set service app_3
    20 set service app_2

interface Ethernet1/1
```

```
  switchport
  switchport access vlan 10
  no shutdown
  epbr l2 policy p1 egress-interface Ethernet1/2

interface Ethernet1/2
  switchport
  switchport access vlan 10
  no shutdown
  epbr l2 policy p1 egress-interface Ethernet1/1 reverse
```

Example: Configuring Trunk Ports

The following configuration example shows how to configure trunk ports:

```
epbr policy p3
  statistics
  match ip address flow1 vlan 10
    load-balance buckets 2
    10 set service app_1
    20 set service app_2
  match ipv6 address flow2 vlan 20
    load-balance buckets 2
    10 set service app_3
    20 set service app_4
  match l2 address flow3 vlan 30
    10 set service app_1
    20 set service app_2

interface Ethernet1/27
  switchport
  switchport mode trunk
  no shutdown
  epbr l2 policy p3 egress-interface Ethernet1/28

interface Ethernet1/28
  switchport
  switchport mode trunk
  no shutdown
  epbr l2 policy p3 egress-interface Ethernet1/27 reverse

Collecting statistics
```

Collecting statistics:

```
itd-san-2# show epbr statistics policy p1

Policy-map p1, match flow2

    Bucket count: 2

      traffic match : bucket 1
        app_1 : 8986 (Redirect)
        app_3 : 8679 (Redirect)
        app_4 : 8710 (Redirect)
        app_2 : 8725 (Redirect)
      traffic match : bucket 2
        app_1 : 8696 (Redirect)
        app_3 : 8680 (Redirect)
        app_4 : 8711 (Redirect)
        app_2 : 8725 (Redirect)

Policy-map p1, match flow3
```

```
    Bucket count: 1

      traffic match : bucket 1
        app_2 : 17401 (Redirect)
        app_4 : 17489 (Redirect)
        app_3 : 17461 (Redirect)

Policy-map p1, match flow1

    Bucket count: 1

      traffic match : bucket 1
        app_1 : 17382 (Redirect)
        app_3 : 17348 (Redirect)
        app_2 : 17411 (Redirect)
```

Example: Viewing ePBR Policy

The following example shows how to view an ePBR policy:

```
show epbr policy p3

Policy-map : p3
Match clause:
ip address (access-lists): flow1
action:Redirect
service app_1, sequence 10, fail-action No fail-action
Ethernet1/3 track 4 [UP]
service app_2, sequence 20, fail-action No fail-action
port-channel10 track 10 [UP]
Match clause:
ipv6 address (access-lists): flow2
action:Redirect
service app_3, sequence 10, fail-action No fail-action
Ethernet1/9 track 13 [UP]
service app_4, sequence 20, fail-action No fail-action
port-channel12 track 3 [UP]
Match clause:
layer-2 address (access-lists): flow3
action:Redirect
service app_1, sequence 10, fail-action No fail-action
Ethernet1/3 track 4 [UP]
service app_2, sequence 20, fail-action No fail-action
port-channel10 track 10 [UP]
Policy Interfaces:
egress-interface Eth1/28
```

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 10.1.1.0/24 any
epbr service s1_l2 type l2
  service-end-point interface Ethernet1/2
    reverse interface Ethernet1/3
epbr policy l2_pol
   statistics
   match ip address acl1 vlan all
   load-balance buckets 4 mask-position 5
   10 set service s1_l2
interface Ethernet1/18
  epbr l2 policy l2_pol egress-interface Ethernet1/19
switch(config-if)# show access-lists epbr_Ethernet1_18_ip dyn
```

```
IP access list epbr_Ethernet1_18_ip
        statistics per-entry
        200001 permit tcp 10.1.1.0 0.0.0.159 any vlan 100 redirect Ethernet1/2 [
match=0]
        200002 permit tcp 10.1.1.32 0.0.0.159 any vlan 100 redirect Ethernet1/2
[match=0]
        200003 permit tcp 10.1.1.64 0.0.0.159 any vlan 100 redirect Ethernet1/2
[match=0]
        200004 permit tcp 10.1.1.96 0.0.0.159 any vlan 100 redirect Ethernet1/2
[match=0]
        4294967295 permit ip any any redirect Ethernet1/19 [match=0]
```