Configuring Policy-Based Routing (PBR)

Policy-Based Routing

Information About Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- Route map statement marked as permit is processed as follows:
  - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.

    For example:

    match length A B

    match ip address acl1 acl2

    match ip address acl3

A packet is permitted if it is permitted by match length A B or acl1 or acl2 or acl3

- If the decision reached is permit, then the action specified by the set command is applied on the packet .
- If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence (the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.

- For PBR, route-map statements marked as deny are not supported.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

Policy-Based Routing Using Object Tracking

You can configure policy-based routing (PBR) to use object tracking to verify the most viable next-hop IP address to which to forward packets, using an Internet Control Message Protocol (ICMP) ping as the verification method.

PBR with Object Tracking is most suitable for devices that have multiple Ethernet connections as the next hop. Normally, Ethernet interfaces connect to DSL modems or cable modems, and do not detect a failure upstream, in the ISP broadband network. The Ethernet interface remains up, and any form of static routing points to that interface. PBR with object tracking allows you to back up two Ethernet interfaces, determine the interface that is available by sending ICMP pings to verify reachability, and then route traffic to that interface.

To verify the next-hop IP address for the device, PBR informs the object-tracking process that it is interested in tracking a certain object. The tracking process, in turn, informs PBR when the state of the object changes.

Note VRF is not supported with PBR using Object Tracking.

How to Configure PBR

- To use PBR, you must have the feature set enabled on the switch or active stack.
- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch supports PBR based on match length.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.

- You can define a maximum of 128 IP policy route maps on the switch or switch stack.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.
- When configuring match criteria in a route map, follow these guidelines:
  - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flappping.
- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.
- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- PBR based on TOS, DSCP and IP Precedence are not supported.
- Set interface, set default next-hop and set default interface are not supported.
- **ip next-hop recursive** and **ip next-hop verify availability** features are not available and the next-hop should be directly connected.
- Policy-maps with no set actions are supported. Matching packets are routed normally.
- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**route-map** *map-tag* [**permit**] [*sequence number*] **Example:** `Device(config)# route-map pbr-map permit`	Defines route maps that are used to control where packets are output, and enters route-map configuration mode. • *map-tag* — A meaningful name for the route map. The **ip policy route-map** interface configuration command uses this name to reference the route map. Multiple route-map statements with the same map tag define a single route map. • (Optional) **permit** — If **permit** is specified and the match criteria are met

	Command or Action	Purpose
		for this route map, the route is policy routed as defined by the set actions. • (Optional) *sequence number* — The sequence number shows the position of the route-map statement in the given route map.
Step 3	**match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* |...*access-list-name*] **Example:** `Device(config-route-map)# match ip address 110 140`	Matches the source and destination IP addresses that are permitted by one or more standard or extended access lists. ACLs can match on more than one source and destination IP address. If you do not specify a **match** command, the route map is applicable to all packets.
Step 4	**match length** *min max* **Example:** `Device(config-route-map)# match length 64 1500`	Matches the length of the packet.
Step 5	**set ip next-hop** *ip-address* [*...ip-address*] **Example:** `Device(config-route-map)# set ip next-hop 10.1.6.2`	Specifies the action to be taken on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).
Step 6	**set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*] **Example:** `Device(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100`	Configures the route map to verify the reachability of the tracked object. **Note** This command is not supported on IPv6 and VRF.
Step 7	**exit** **Example:** `Device(config-route-map)# exit`	Returns to global configuration mode.
Step 8	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to be configured.
Step 9	**ip policy route-map** *map-tag* **Example:** `Device(config-if)# ip policy route-map pbr-map`	Enables PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in the order of sequence

	Command or Action	Purpose
		number until the first match. If there is no match, packets are routed as usual.
Step 10	**ip route-cache policy** **Example:** `Device(config-if)# ip route-cache policy`	(Optional) Enables fast-switching PBR. You must enable PBR before enabling fast-switching PBR.
Step 11	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.
Step 12	**ip local policy route-map** *map-tag* **Example:** `Device(config)# ip local policy route-map local-pbr`	(Optional) Enables local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch, and not to incoming packets.
Step 13	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 14	**show route-map** [*map-name*] **Example:** `Device# show route-map`	(Optional) Displays all the route maps configured or only the one specified to verify configuration.
Step 15	**show ip policy** **Example:** `Device# show ip policy`	(Optional) Displays policy route maps attached to the interface.
Step 16	**show ip local policy** **Example:** `Device# show ip local policy`	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Verifying Next-Hop IP Using Object Tracking

To verify the next-hop IP address using PBR with object tracking, perform the following steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**track** *object-number* **ip sla** *entry-number* **Example:** `Device(config)# track 100 ip sla 100`	Tracks the state of an IP SLA object.
Step 3	**ip sla** *operation-number* **Example:** `Device(config)# ip sla 100`	Starts a Cisco IOS IP Service Level Agreement (SLA) operation configuration, and enters IP SLA configuration mode.
Step 4	**icmp-echo** *ip-address* **source-ip** *ip-address* **Example:** `Device(config-ip-sla)# icmp-echo 172.19.255.253 source-ip 172.19.255.47`	Configures an IP SLA Internet Control Message Protocol(ICMP) echo probe operation, and enters Echo configuration mode.
Step 5	**frequency** *seconds* **Example:** `Device(config-ip-sla-echo)# frequency 2`	(Optional) Sets the rate, in seconds, at which a specified IP SLA operation is repeated.
Step 6	**threshold** *milliseconds* **Example:** `Device(config-ip-sla-echo)# threshold 1000`	(Optional) Sets the length of time, in ms, required for a rising threshold event to be declared.
Step 7	**timeout** *milliseconds* **Example:** `Device(config-ip-sla-echo)# timeout 1500`	(Optional) Sets the maximum time, in ms, required for the IP SLA operation to be completed.
Step 8	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh* : *mm[:ss]* *[month day | day month]* | **pending** | *now* | *after hh* : *mm* : *ss}*] [**ageout** *seconds*] **Example:** `Device(config)# ip sla schedule 100 life forever start-time now`	Configures the scheduling parameters for a single Cisco IOS IP SLA operation.
Step 9	**route-map** *map-tag* [**permit** | **deny**] *[sequence-number]* **Example:** `Device(config)# route-map alpha permit 10`	Specifies a route map and enters route-map configuration mode.
Step 10	**match ip address** *[access-list-name]* **Example:** `Device(config-route-map)# match ip address exlist`	Distributes routes that have a destination IPv4 network number address that is permitted by a standard access list.

	Command or Action	Purpose
Step 11	**set ip next-hop verify-availability** *[next-hop-address sequence* **track** *object]* **Example:** `Device(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100`	Configures the route map to verify the reachability of the tracked object.

Feature Information for Configuring PBR

Table 1: Feature information for PBR

Feature Name	Releases	Feature Information
Policy-Based Routing	Cisco IOS Release 15.2(6)E2	Policy-based routing is used to configure a defined policy for traffic flows.