

PBR Programming on M3-Series Linecards

Introduction

This document provides configuration example along with procedure to verify programming of PBR feature on M3-Series Linecards.

Contributed by Nikolay Kartashev, Cisco TAC Engineer.

Prerequisites

Understanding of PBR operation

Components Used

Nexus 7700 Series Switch with M3 linecard.

```
Angeles2-vdcl1# show module
```

Mod	Ports	Module-Type	Model	Status
9	0	Supervisor Module-2	N77-SUP2E	active *
10	0	Supervisor Module-2	N77-SUP2E	ha-standby
16	12	40/100 Gbps Ethernet Module	N77-M312CQ-26L	ok

Mod	Sw	Hw
9	8.0(1)	0.308
10	8.0(1)	0.308
16	8.0(1)	0.8

Configure

Network Diagram

Network setup consists of Nexus 7700 Series switch, with one L3 point-to-point interface, and one L2 trunk interface, as shown in the network diagram below.



Nexus 7700 switch has ingress L3 interface Eth16/2, and two egress interfaces: SVI100 and SVI200, with both vlans 100 and 200 enabled on L2 trunk.

Configuration

Here are basic steps to configure PBR

1. Enable feature PBR

```
Angeles2-vdcl# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Angeles2-vdcl(config)# feature pbr
Angeles2-vdcl(config)# end
Angeles2-vdcl#
```

2. Identify traffic flows that need to be redirected, and specify them in access-list

```
Angeles2-vdcl# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Angeles2-vdcl(config)# ip access-list PBR-ACL
Angeles2-vdcl(config-acl)# permit ip any 101.1.1.1/32
Angeles2-vdcl(config-acl)# end
Angeles2-vdcl#
```

3. Configure PBR route map and apply it on the ingress interface

```
Angeles2-vdcl# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Angeles2-vdcl(config)# route-map TEST-PBR permit 10
Angeles2-vdcl(config-route-map)# match ip address PBR-ACL
Angeles2-vdcl(config-route-map)# set ip next-hop 102.1.1.1
Angeles2-vdcl(config-route-map)# exit

Angeles2-vdcl(config)#
Angeles2-vdcl(config)# interface Ethernet16/2
Angeles2-vdcl(config-if)# ip policy route-map TEST-PBR
Angeles2-vdcl(config-if)# end
Angeles2-vdcl#
```

4. Confirm above ACL and PBR configuration was applied successfully with the following commands

```
Angeles2-vdc1# show run aclmgr

version 8.0(1)
ip access-list PBR-ACL
  10 permit ip any 101.1.1.1/32

Angeles2-vdc1#
Angeles2-vdc1# show running-config rpm

version 8.0(1)
feature pbr

route-map TEST-PBR permit 10
  match ip address PBR-ACL
  set ip next-hop 102.1.1.1

interface Ethernet16/2
  ip policy route-map TEST-PBR

Angeles2-vdc1#
```

Verify

Use the following command to verify PBR configuration in software.

```
Angeles2-vdc1# show ip internal pbr interface ethernet 16/2
Ethernet16/2
IP policy route-map

Routemap Seq permit #
Policy details:   Policy has 1 ACEs.
  (Stats if reported are counters since last upload to policy manager)
  (Group LEVEL Stats are not supported in this ver)

  ip access-list PBR-ACL
  permit   ip   any 101.1.1.1/32 (fl:0x800)

Control packets that were allowed to bypass :0

Set action: Nexthop 102.1.1.1
Angeles2-vdc1#
```

The following command checks ACL programming for PBR feature on M3-series linecard.

```
Angeles2-vdc1# show system internal access-list interface ethernet 16/2 input
statistics module 16

INSTANCE 0x0
-----

Tcam 1 resource usage:
```

```

-----
Label_b = 0x2
Bank 0
-----
IPv4 Class
Policies: PBR(PBR-ACL) [Merged]
Netflow profile: 0
Netflow deny profile: 0
Entries:
[Index] Entry [Stats]
-----
[0018:28242:0004] prec 1 permit-routed ip 0.0.0.0/0 224.0.0.0/4 [134]
[0019:28c42:0005] prec 1 redirect(0x2001d)-routed ip 0.0.0.0/0 101.1.1.1/32 [0]
[001a:29442:0006] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0 [801]

```

Angeles2-vdcl#

The following is the rewrite index table output from M3-series linecard.

Angeles2-vdcl# **slot 16 show system internal aclqos info rit-sram**

```

Instance: 0
=====
RIT_IDX: 0x20000 ==> 0x10000 0x10001 0x10002 0x10003
RIT_IDX: 0x20019 ==> 0x3
RIT_IDX: 0x2001d ==> 0x10005
RIT_IDX: 0x2001e ==> 0x8000

SRAM IDX : RIT IDX MAP
0x3:0x20019 0x8000:0x2001e 0x10000:0x20000 0x10001:0x20000 0x10002:0x20000
0x10003:0x20000 0x10005:0x2001d

```

Using rewrite index, you can find adjacency pointer.

```

module-16# show hardware internal forwarding 13 inst 0 table rw2adj_map_tbl_adjptr
index 0x2001d
adjptr: 0x98
module-16#

```

With the following command, egress LIF can be found.

```

module-16# show hardware internal forwarding 13 inst 0 table adj index 0x98
same_if_mask_sel 0x0
ingress_lif_segid_sel 0x0
format 0x0
fc_iod_drop 0x0
mcast_cpp_lif 0x0
ad_age 0x1
l3_enable 0x1
ad_trig 0x0
valid 0x1
rdt 0x0
peer_id_sel 0x0
no_intra_split_horizon 0x0
egress_lif 0xc8
ri 0x3

```

```

top_sel 0x0
zone_enforce_en_or_use_vft 0x0
filter_en 0x0
frr_te 0x0
usd_da 0x0
gleen_adj 0x0
index_sel_or_bndl_en 0x0
tnl_encap 0x0
rw_hint 0x0
preserve_cos 0x0
ttl_control 0x2

module-16#

```

In the following output you can verify above findings, such as RIT index and egress LIF.

```

module-16# show system internal aclqos info redirect

...

ADJACENCY INFO: INST: 0, DIR: INGRESS Type: PBR_fib_
-----
Adj. Base
-----
Adjacency type: PBR_fib_ sub 0 dir 1 inst 0
SW ID: 0x400020000000004 Adj HW index: 0x4 HW_rit_idx: 0x2001d DIT idx: 0x0
REF SW: 1 HW 1
Sig type: 0x55667788 Flags 0x82 vrf 0
Last tcam res: 0 commit pending 0
L3 Adj Key
-----
AFI: 1 vdc 2 inst 0 drop 0 count 1 nh_sz 8
Next hop:      102.1.1.1      Vlan200(0x90200c8)

...

```

To get more information on rewrite parameters for PBR next hop, use the following command:

```

Angeles2-vdc1# show system internal forwarding route 102.1.1.1/32 detail module 16

RPF Flags legend:
  S - Directly attached route (S_Star)
  V - RPF valid
  M - SMAC IP check enabled
  G - SGT valid
  E - RPF External table valid
102.1.1.1/32      , Vlan200
Dev: 0 , Idx: 0xf21fa , Prio: 0x0      , RPF Flags: VS      , DGT: VPN: 3
RPF_Intf_5:  Vlan200      (0xc8      )
AdjIdx: 0x98      , LIFB: 0      , LIF: Vlan200      (0xc8      ), DI: 0x0
DMAC: 0000.cccc.dddd SMAC: d867.d90b.6b42      RitIdx: 0x2001d
102.1.1.1/32      , Vlan200
Dev: 1 , Idx: 0xf21fa , Prio: 0x0      , RPF Flags: VS      , DGT: VPN: 3
RPF_Intf_5:  Vlan200      (0xc8      )
AdjIdx: 0x98      , LIFB: 0      , LIF: Vlan200      (0xc8      ), DI: 0x0
DMAC: 0000.cccc.dddd SMAC: d867.d90b.6b42      RitIdx: 0x2001d

Angeles2-vdc1#

```