

## In This Chapter

This section provides information about Provider Backbone Bridging (PBB) in an MPLS-based network.

Topics in this section include:

- [Applicability on page 516](#)
- [Overview on page 518](#)
- [Configuration on page 519](#)
- [Conclusion on page 554](#)

## Applicability

This chapter is applicable to the 7750 SR, 710 SR and 7450 ESS series and was tested on Release 8.0.R.6. Note that from 8.0R6 onward, the MAC notification and use-sap-bmac features no longer require chassis-mode D, however, a network-mode D (described in the Access Dual-Homing and MAC-notification section) is required .<sup>1</sup>

---

1. Although it can be used in an MPLS-based PBB network as explained in this document, the MAC notification feature for dual-homed access is normally used in native PBB networks.

## Summary

The draft-ietf-l2vpn-pbb-vpls-pe-model-00, *Extensions to LDP Signaling for PBB-VPLS*, describes the PBB-VPLS model supported by the SR OS 7.0. This model expands the VPLS PE model to support PBB as defined by the IEEE 802.1ah.

PBB-VPLS combines the best of the PBB and VPLS technologies to deliver the most scalable multi-point Layer 2 VPN in the market. PBB-VPLS inherits all the benefits derived from MPLS (for example, sub-50ms FRR protection, traffic engineering, no need for MSTP in the backbone) while greatly increasing the scalability of the network by providing MAC hiding, service multiplexing and pseudowire aggregation.

The SR OS 7.0 PBB-VPLS implementation also includes support for:

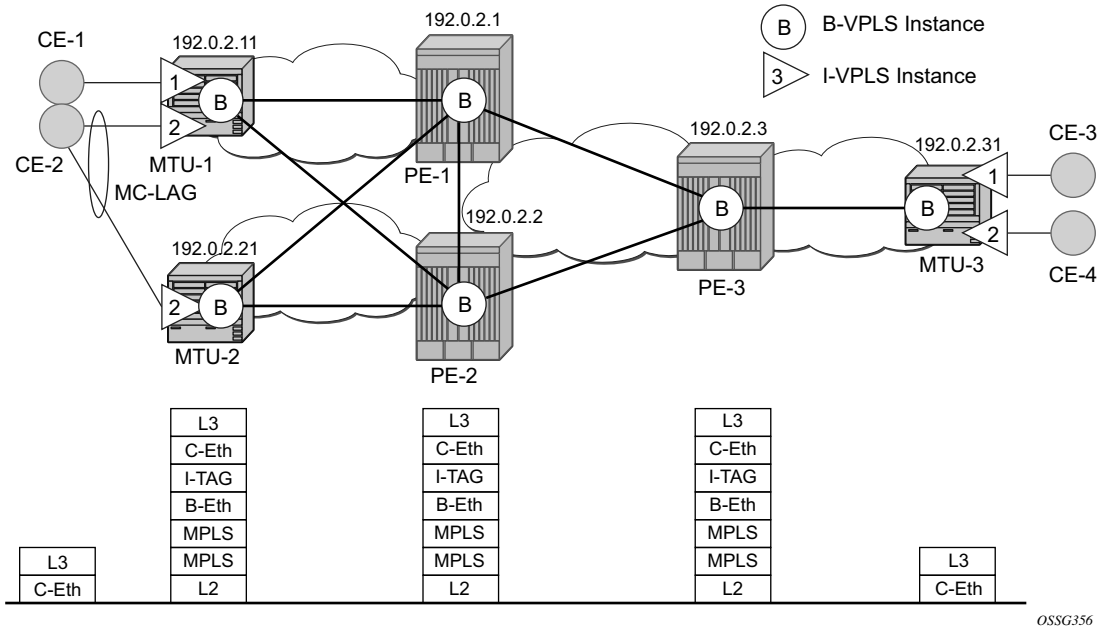
- MMRP (Multiple MAC Registration Protocol, application within IEEE 802.1ak) for flood containment in the backbone instances, as specified in Section 6 of the draft-ietf-l2vpn-pbb-vpls-pe-model.
- Extensions to LDP signaling for PBB-VPLS, according to draft-balus-l2vpn-pbb-ldp-ext-00. These extensions will avoid network blackhole issues, as described in the Section 3 of the mentioned draft.

The objective of this section is to provide the required guidelines to configure and troubleshoot a PBB-VPLS network based on SR OS 7.0.

Knowledge of the VPLS and H-VPLS (RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*) architecture and functionality is assumed throughout this document. The most relevant concepts will be briefly explained throughout the document, taking the network setup shown in the next section as an example. For further information, refer to the relevant Alcatel-Lucent documentation.

# Overview

The following network setup will be used throughout the rest of the chapter.



OSSG356

**Figure 77: Network Topology**

The setup consists of a three 7x50 SR/ESS (PE-1, PE-2 and PE-3) core and three MTU (Multi-Tenant Unit) nodes connected to the core. The MTU nodes can be either 7x50 or 7710 Service Routers running SR OS 7.0. A backbone VPLS instance (B-VPLS 100) will be defined in all the six nodes, whereas a few customer I-VPLS instances will be defined on the three MTU nodes. Those I-VPLS instances will be multiplexed into the common B-VPLS, using the ISID field within the I-TAG as the demultiplexer field at the egress MTU to differentiate each specific customer.

The B-VPLS domain constitutes a H-VPLS network itself, with spoke SDPs from the MTUs to the core PE layer. Active/standby spoke SDPs can be used from the MTUs to the PEs (for example, in the MTU-1 and MTU-2 cases) or single non-redundant spoke SDPs (for example, MTU-3). CE-2 is dual-connected to the service provider network through MC-LAG.

The protocol stack being used along the path between the CEs is represented in [Figure 77](#).

## Configuration

This section describes all the relevant PBB-VPLS configuration tasks for the setup showed in [Figure 77](#). Note that the appropriate associated IP/MPLS configuration is out of the scope of this document. In this particular example, the following protocols will be configured beforehand:

- ISIS-TE as IGP with all the interfaces being Level-2 (OSPF-TE could have been used instead).
- RSVP-TE as the MPLS protocol to signal the transport tunnels (LDP could have been used instead, but without fast re-route).
- LSPs between core PEs will be fast re-route protected (facility bypass tunnels) whereas LSP tunnels between MTUs and PEs will not be protected.
- The protection between MTU-1, MTU-2 and PE-1, PE-2 will be based on the A/S pseudowire protection configured in the B-VPLS.
- BGP is configured for auto-discovery (Layer 2-vpn family), since FEC 129 will be used for the pseudowires between PEs in the core.

Once the IP/MPLS infrastructure is up and running, the service configuration tasks described in the following sections can be implemented.

## PBB-VPLS M:1 Service Configuration

This section explains the process to configure PBB-VPLS services in a M:1 fashion, **M** being the number of customer I-VPLS services multiplexed into the same B-VPLS instance (instance 100). An alternative configuration is 1:1, where each customer I-VPLS has its own B-VPLS. MTU-1 and PE-1 will be picked to show the relevant CLI configuration commands. Note that the bold digits separated by colons **00:xx** are abbreviations for the backbone MAC addresses.

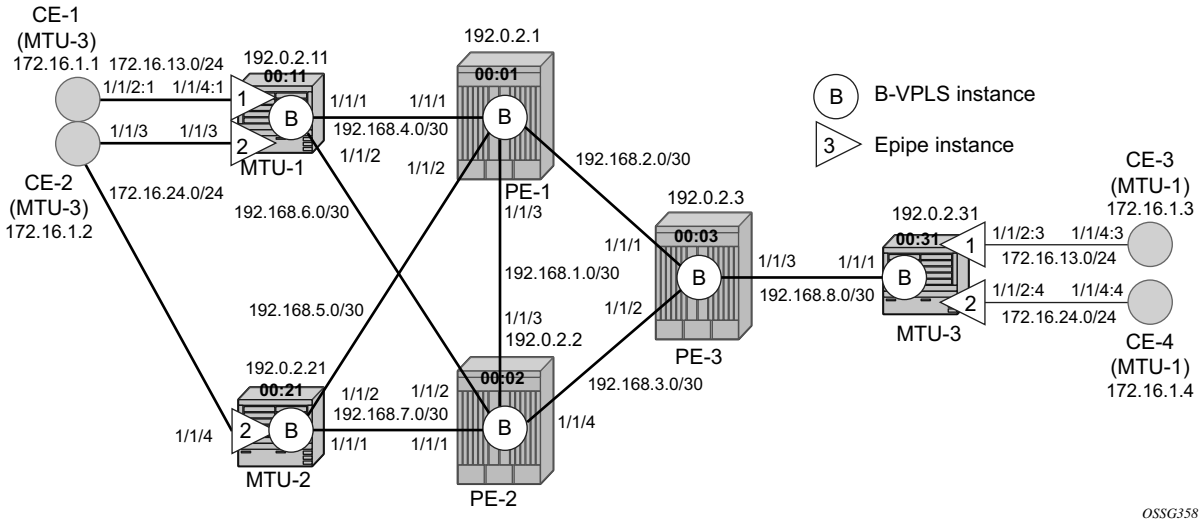


Figure 78: MTU-1 and PE-1 Nodes as Configuration Examples

OSSG358

## B-VPLS Configuration

The first step is to configure the B-VPLS instance that will carry the PBB traffic. The following CLI outputs show the B-VPLS configuration.

Configuration examples:

```
A:MTU-1>config>service# info
-----
onfigure
  service
    vpls 100 customer 1 b-vpls create
      service-mtu 2000
      pbb
        source-bmac 00:11:11:11:11:11
      exit
      stp
        shutdown
      exit
      mrp
        no shutdown
      exit
      endpoint "core" create
        no suppress-standby-signaling
      exit
      spoke-sdp 111:100 endpoint "core" create
        stp
          shutdown
        exit
        precedence primary
      exit
      spoke-sdp 112:100 endpoint "core" create
        stp
          shutdown
        exit
      exit
      no shutdown
    exit
```

```
A:PE-1>config>service# info
-----
onfigure
  service
    pw-template 1 use-provisioned-sdp create
      split-horizon-group "CORE"
    exit
  exit
  vpls 100 customer 1 b-vpls create
    service-mtu 2000
    pbb
      source-bmac 00:01:01:01:01:01
    exit
    bgp-ad
      vpls-id 65000:100
      route-target export target:65000:100 import target:65000:100
      pw-template-bind 1
      no shutdown
```

## PBB-VPLS M:1 Service Configuration

```
exit
stp
    shutdown
exit
mrp
    no shutdown
exit
no shutdown
exit
```

The relevant B-VPLS commands are in **bold**.

Note that the statement **b-vpls** is given at creation time and therefore it cannot be added to a regular existing VPLS instance. Besides the **b-vpls** statement, the B-VPLS is a regular VPLS instance in terms of configuration, with the following exceptions:

- The B-VPLS service MTU must be at least 18 bytes greater than the I-VPLS MTU of the multiplexed instances. In this example, the I-VPLS instances will have the default service MTU (1500 bytes); hence, any MTU equal or greater than 1518 bytes must be configured. In this particular example, a MTU of 2000 bytes is configured in the B-VPLS instance throughout the network.
- The source B-MAC is the MAC that will be sourced when the PBB traffic is originated from that node. Note that you can configure a source B-MAC per B-VPLS instance (if there are more than one B-VPLS) or a common source B-MAC that will be shared by all the B-VPLS instances in the box. If no specific source B-MAC is provisioned, the system MAC address is used as the source B-MAC. Note that when using the Access Multi-Homing feature for Native PBB, the source-bmac must be a configured one and never the chassis mac address. The way to configure a common B-MAC for all the B-VPLS instances is shown below:

```
A:MTU-1>config>service# info
-----
pbb
    source-bmac 00:11:11:11:11:11
```

The following considerations will be taken into account when configuring the B-VPLS:

- B-VPLS SAPs:
  - Ethernet dot1q and null encapsulations are supported
  - Default SAP (:\*) types are blocked in the CLI for the B-VPLS SAP



- B-VPLS SDPs:
  - For MPLS, both mesh and spoke SDPs with split horizon groups are supported.
  - Similar to regular pseudowires, the outgoing PBB frame on an SDP (for example, B-pseudowire) contains a BVID qtag only if the pseudowire type is Ethernet VLAN. If the pseudowire type is **Ethernet**, the BVID qtag is stripped before the frame goes out.
  - BGP-AD is supported in the B-VPLS; therefore, spoke SDPs in the B-VPLS can be signaled using FEC 128 or FEC 129. In this example, BGP-AD and FEC 129 are used. A split-horizon group has been configured to emulate the behavior of mesh-SDPs in the core.
- If a local I-VPLS instance is associated with the B-VPLS, “local frames” originated/terminated on local I-VPLS(s) are PBB encapsulated/de-encapsulated using the PBB etype provisioned under the related port or SDP component.

By default, the PBB etype is 0x88e7 (which is the standard one defined in 802.1ah for the I-TAG) but this PBB etype can be changed if required due to interoperability reasons. This is the way to change it at port and/or SDP level:

```
A:MTU-1# configure port 1/1/1 ethernet pbb-etype
- pbb-etype <0x0600..0xffff>
- no pbb-etype

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex

A:MTU-1# configure service sdp 111 pbb-etype
- no pbb-etype [<0x0600..0xffff>]
- pbb-etype <0x0600..0xffff>

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex
```

The following commands are useful to check the actual PBB etype.

```
A:MTU-1# show service sdp 111 detail | match PBB
Bw BookingFactor      : 100                PBB Etype           : 0x88e7

A:MTU-1# show port 1/1/1 | match PBB
PBB Ethertype       : 0x88e7
```

## I-VPLS Configuration

Once the common B-VPLS is configured, the next step is to provision of the customer I-VPLS instances. The following outputs show the relevant CLI configuration for the two I-VPLS instances represented in [Figure 78](#). The I-VPLS instances are configured on the MTU devices, whereas the core PE's are kept as customer-unaware nodes.

```
vpls 1 customer 1 i-vpls create
    backbone-vpls 100
    exit
    stp
        shutdown
    exit
    sap 1/1/4:1 create
    exit
    no shutdown
exit
vpls 2 customer 1 i-vpls create
    backbone-vpls 100:2
    exit
    stp
        shutdown
    exit
    sap lag-1 create
    exit
    no shutdown
exit
```

The relevant I-VPLS commands are in **bold**.

Note that the statement **i-vpls** is given at creation time and therefore it cannot be added to a regular existing VPLS instance. After creating the I-VPLS instance, it has to be linked to its corresponding transport B-VPLS instance. That link is given by the **backbone-vpls b-vpls:isid** command. If no ISID (20 bit customer identification in the ITAG) is specified, the system will take the ISID from the VPLS instance identifier.

The following considerations will be taken into account when configuring the I-VPLS:

- I-VPLS SAPs:
  - SAPs can be defined on ports with any Ethernet encapsulation type (null, dot1q and qinq)
  - The I-VPLS SAPs can coexist on the same port with SAPs for other business services, for example, VLL and VPLS SAPs.
- I-VPLS SDPs:
  - GRE and MPLS SDPs are supported.
  - No mesh-SDPs are supported, only spoke SDP. Mesh-SDPs can be emulated by using split horizon groups.

Existing SAP processing rules still apply for the I-VPLS case; the SAP encapsulation definition on Ethernet ingress ports defines which VLAN tags are used to determine the service that the packet belongs to:

- Null encapsulation defined on ingress — Any VLAN tags are ignored and the packet goes to a default service for the SAP;
- Dot1q encapsulation defined on ingress — only first VLAN tag is considered;
- QinQ encapsulation defined on ingress — both VLAN tags are considered; wildcard for the inner VLAN tag is supported.
- For dot1q/qinq encapsulations, traffic encapsulated with VLAN tags for which there is no definition is discarded.
- Note that any VLAN tag used for service selection on the I-SAP is stripped before the PBB encapsulation is added. Appropriate VLAN tags are added at the remote PBB PE when sending the packet out on the egress SAP.

Up to 8000 VPLS instances can be defined per system. That number includes I-VPLS, B-VPLS and regular VPLS.

## MMRP for Flooding Optimization

When the M:1 model is used (as in this example), any I-VPLS broadcast/multicast/unknown frame is flooded throughout the B-VPLS domain regardless of the nodes where the originating I-VPLS is defined. In other words, in our example in [Figure 77](#), any broadcast/multicast/unknown frame coming from CE-1 would be flooded in the B domain and would reach PE-2 and MTU-2, even though that traffic only needs to go to PE-3 and MTU-3. In order to build customer-based flooding trees and optimize the flooding, MMRP (Multiple MAC Registration Protocol) must be configured on the B-VPLS.

MMRP can be enabled with its default settings just by executing a **mrp no shutdown** command:

```
A:PE-1>config>service# info
-----
<snip>
vpls 100 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:01:01:01:01:01
  exit
  bgp-ad
    vpls-id 65000:100
    route-target export target:65000:100 import target:65000:100
    pw-template-bind 1
    no shutdown
  exit
  stp
    shutdown
  exit
  mrp
    no shutdown
  exit
  no shutdown
exit
-----
```

There are certain B-VPLS MRP settings that can be modified. These are the default values:

```
*A:MTU-1>config>service>vpls>mrp# info detail
-----
    no attribute-table-size
    attribute-table-low-wmark 90
    attribute-table-high-wmark 95
    no flood-time
    no shutdown
-----
*A:MTU-1>config>service>vpls>mrp#
```

These four attributes can be changed in order to control the number of MMRP attributes per B-VPLS and optimize the convergence time in case of failures in the B-VPLS:

- Controlling the number of attributes per B-VPLS

The MMRP exchanges create one entry per attribute (group B-MAC ) in the B-VPLS where MMRP protocol is running. When the first registration is received for an attribute, an MFIB entry is created for it. The *attribute-table-size* allows the user to control the number of MMRP attributes (group B-MACs) created on a per B-VPLS basis, between 1 and 2048. Based on the configured size, high and low watermarks can be set (in percentage) so that alarms can be triggered upon exceeding the watermarks. This ensures that no B-VPLS will take up all the resources from the total pool. The maximum number of attributes per B-VPLS is 2048 and 4000 can be configured globally on the system.

- Optimizing the convergence time

Assuming that MMRP is used in a certain B-VPLS, under failure conditions the time it takes for the B-VPLS forwarding to resume may depend on the data plane and control plane convergence plus the time it takes for MMRP exchanges to stabilize the flooding trees on a per ISID basis. In order to minimize the convergence time, the PBB SR OS implementation offers the selection of a mode where B-VPLS forwarding reverts for a short time to flooding so that MMRP has enough time to converge. This mode can be selected through configuration using the **flood-time** *value* command where value represents the amount of time in seconds (between 3 and 600) that flooding will be enabled. If this behavior is selected, the forwarding plane starts with B-VPLS flooding for a configurable time period, then it reverts back to the MFIB entries installed by MMRP. The following B-VPLS events initiate the switch from per I-VPLS (MMRP) MFIB entries to BVPLS flooding:

- Reception or local triggering of a TCN (Spanning Tree Topology Change Notification)
- B-SAP failure
- Failure of a B-SDP binding
- Pseudowire activation in a primary/standby H-VPLS resiliency solution
- SF/CPM switchover due to STP reconvergence

The IEEE 802.1ak standard, which defines MRP, requires the implementation of different state machines with associated timers that can be tuned. A full MRP participant maintains the following state machines:

- Registrar state machine
- Applicant state machine
- LeaveAll state machine
- PeriodicTransmission state machine

The two first state machines are maintained for each attribute in which the participant is interested, while the two latter are global to all the attributes.

The job of the registrar function is to record declarations of the attribute made by other participants on the LAN. A registrar does not send any protocol messages, as the applicant looks after the interests of all would-be participants.

The job of the applicant is twofold: on one hand to ensure that this participant’s declaration is correctly registered by other participants’ registrars, and on the other hand to prompt other participants to register again after one withdraws a declaration.

The associated timers can be tuned on a per SAP/SDP basis:

```
*A:MTU-1>config>service>vpls# spoke-sdp 111:100 mrp
- mrp

[no] join-time      - Configure timer value in 10th of seconds for sending
                    join-messages
[no] leave-all-time - Configure timer value in 10th of seconds for
                    refreshing all attributes
[no] leave-time     - Configure timer value in 10th of seconds to hold
                    attribute in leave-state
[no] periodic-time  - Configure timer value in 10th of seconds for
                    re-transmission of attribute declarations
[no] periodic-timer - Control re-transmission of attribute declarations

*A:MTU-1>config>service>vpls# spoke-sdp 111:100 mrp
*A:MTU-1>config>service>vpls>spoke-sdp>mrp# info detail
-----
                    join-time 2
                    leave-time 30
                    leave-all-time 100 # Default Values
                    periodic-time 10
                    no periodic-timer
-----
*A:MTU-1>config>service>vpls>spoke-sdp>mrp#
```

A brief description of the MRP SAP/SDP attributes follow:

- **Join-time** — This command controls the interval between transmit opportunities that are applied to the applicant state machine. An instance of this join period timer is required on a per-port, per-MRP participant basis. For additional information, refer to IEEE 802.1ak-2007 section 10.7.4.1.
- **Leave-time** — This command controls the period of time that the registrar state machine will wait in the leave state before transitioning to the MT state when it is removed. An instance of the timer is required for each state machine that is in the leave state. The leave period timer is set to the value leave-time when it is started. A registration is normally in “in” state where there is an MFIB entry and traffic being forwarded. When a “leave all” is performed (periodically around every 10-15 seconds per SAP/SDP binding – see leave-all-time-below), a node sends a message to its peer indicating a leave all is occurring and puts all of its registrations in leave state. The peer refreshes its registrations based on the

leave all PDU it receives and sends a PDU back to the originating node with the state of all its declarations. Refer to IEEE 802.1ak-2007 section 10.7.4.2.

- **Leave-all-time** — This command controls the frequency with which the leaveall state machine generates leaveall PDUs. The timer is required on a per-port, per-MRP participant basis. The leave all period timer is set to a random value, T, in the range  $\text{leavealltime} < T < 1.5 * \text{leave-all-time}$  when it is started. Refer to IEEE 802.1ak-2007, section 10.7.4.3.
- **Periodic-time** — This command controls the frequency the periodic transmission state machine generates periodic events if the periodic transmission timer is enabled. The timer is required on a per-port basis. The periodic transmission timer is set to one second when it is started.
- **Periodic-timer** — This command enables or disables the periodic transmission timer.

The following command shows the MRP configuration and statistics on a per SAP/SDP basis within the B-VPLS:

```
*A:MTU-1# show service id 100 all | match post-lines 10 MRP
```

```
Sdp Id 111:100 MRP Information
```

```
-----
Join Time           : 0.2 secs           Leave Time          : 3.0 secs
Leave All Time      : 10.0 secs          Periodic Time       : 1.0 secs
Periodic Enabled   : false
Rx Pdus            : 24884              Tx Pdus            : 22834
Dropped Pdus       : 0
Rx New Event       : 0                  Rx Join-In Event   : 47946
Rx In Event        : 1344              Rx Join Empty Evt  : 460
Rx Empty Event     : 1                  Rx Leave Event     : 2
Tx New Event       : 0                  Tx Join-In Event   : 45483
-----
```

```
SDP MMRP Information
```

```
-----
MAC Address         Registered      Declared
-----
```

```
01:1e:83:00:00:01 Yes                Yes
```

```
01:1e:83:00:00:02 Yes                Yes
-----
```

```
Number of MACs=2 Registered=2 Declared=2
-----
```

```
Sdp Id 112:100 MRP Information
```

```
-----
Join Time           : 0.2 secs           Leave Time          : 3.0 secs
Leave All Time      : 10.0 secs          Periodic Time       : 1.0 secs
Periodic Enabled   : false
Rx Pdus            : 0                  Tx Pdus            : 0
Dropped Pdus       : 0
Rx New Event       : 0                  Rx Join-In Event   : 0
Rx In Event        : 0                  Rx Join Empty Evt  : 0
Rx Empty Event     : 0                  Rx Leave Event     : 0
Tx New Event       : 0                  Tx Join-In Event   : 0
-----
```

```
SDP MMRP Information
```

```
-----
MAC Address         Registered      Declared
-----
```

## PBB-VPLS M:1 Service Configuration

```
Number of MACs=0 Registered=0 Declared=0
-----
Number of SDPs : 2
-----
MRP Information
-----
Admin State       : Up                Failed Register Cnt: 0
Max Attributes    : 2048              Attribute Count     : 2
Attr High Watermark: 95%              Attr Low Watermark  : 90%
Flood Time       : Off
-----
Related i-Vpls services for b-Vpls service 100
-----
*A:MTU-1#
```

The following command is also useful to check the MRP configuration and status.

```
*A:MTU-1# show service id 100 mrp
-----
MRP Information
-----
Admin State       : Up                Failed Register Cnt: 0
Max Attributes    : 2048              Attribute Count     : 2
Attr High Watermark: 95%              Attr Low Watermark  : 90%
Flood Time       : Off
-----
*A:MTU-1#
```

In the example we are using throughout the document, as soon as MMRP is enabled, an optimized flooding tree will be built for the ISID 1, since the I-VPLS 1 is only defined in MTU-1 and MTU-3, but not MTU-2. A good way to track the flooding tree for a particular ISID is the following command:

```
*A:MTU-1# show service id 100 mmrp mac
-----
SAP/SDP                MAC Address      Registered  Declared
-----
sdp:111:100            01:1e:83:00:00:01 Yes       Yes
sdp:111:100            01:1e:83:00:00:02 Yes       Yes
-----
A:MTU-2# show service id 100 mmrp mac
-----
SAP/SDP                MAC Address      Registered  Declared
-----
sdp:212:100            01:1e:83:00:00:01 Yes       No
sdp:212:100            01:1e:83:00:00:02 Yes       No
-----
*A:MTU-1#
```

The group B-MAC ending in **01** corresponds to the I-VPLS 1 whereas the one ending in **02** to the I-VPLS 2. Note that MMRP PDUs for the two attributes are sent throughout the loop-tree topology (not over STP blocked ports or standby spoke SDPs and observing the split horizon rules). The two attributes are registered on every B-VPLS virtual port; however, the tree is only built on those ports where the attribute is also declared, and not only registered. For instance, the spoke-sdp 212:100 in MTU-2 will not be part of the ISID 1 or ISID 2 flooding trees. Neither



attribute is declared since: I-VPLS 1 doesn't exist on MTU-2 and I-VPLS 2 is operationally down on MTU-2 (MC-LAG SAP is in standby state hence the I-VPLS down).

Note that as soon as a group B-MAC attribute is registered on a particular port, an MFIB entry is added for that B-MAC on that port, regardless of the declaration state for that attribute on the port. For instance, neither B-MAC is declared on MTU-2 however the two MFIB entries are created as soon as the attributes are registered:

```
A:MTU-2# show service id 100 mfib
=====
Multicast FIB, Service 100
=====
Source Address  Group Address          Sap/Sdp Id              Svc Id  Fwd/Blk
-----
*                01:1E:83:00:00:01      b-sdp:212:100          Local   Fwd
*                01:1E:83:00:00:02      b-sdp:212:100          Local   Fwd
-----
Number of entries: 2
=====
*A:MTU-1#
```

## MAC Flush: Avoiding Blackholes

Both the I-VPLS and B-VPLS components inherit the MAC flush capabilities of a regular VPLS clearing the related C-MAC and respectively B-MAC FIBs. All types of MAC flush – flush-all-but-mine and flush-all-from-me – are supported together with the related CLI. In addition to these features, some extensions have been added so that MAC flush can be triggered on the B-VPLS based on some events happening on the I-VPLS. The following diagram shows a potential scenario where blackholes can occur if the proper configuration is not added.

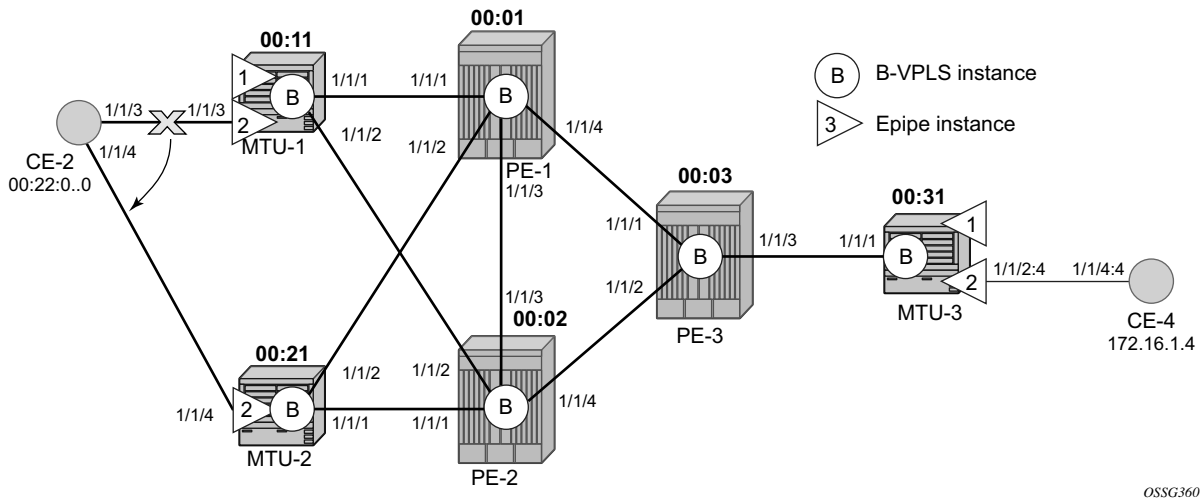


Figure 79: Blackhole

Under normal conditions the I-VPLS 2 FIB on MTU-3 shows that CE-2 MAC address is learnt through B-MAC 00:11 (MTU-1’s B-MAC):

```
A:MTU-3# show service id 2 fdb pbb
=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:22:00:00:00:00  b-sdp:33:100        100      00:11:11:11:11:11 L/0
00:44:00:00:00:00  sap:1/1/2:4         100      N/A                L/0
=====
A:MTU-3#
```

When a failure happens in the CE-2 MC-LAG active link, the link to MTU-2 takes over. However, the FIB on MTU-3 still points at MTU-1’s B-MAC and that will still be the B-MAC used in the PBB encapsulation. Therefore a blackhole occurs until either bidirectional traffic is sent or the FIB aging timer expires.

The following command will solve the blackhole:

```
*A:MTU-1>config>service>vpls# info
-----
      send-bvpls-flush all-from-me
      backbone-vpls 100:2
      exit
      stp
          shutdown
      exit
      sap lag-1 create
      exit
      no shutdown
-----
*A:MTU-1>config>service>vpls# send-bvpls-flush
- send-bvpls-flush {[all-but-mine] [all-from-me]}
- no send-bvpls-flush

<all-but-mine>      : keyword
<all-from-me>      : keyword
```

By enabling **send-bvpls-flush all-from-me** on I-VPLS 2, a failure on the MC-LAG active link on I-VPLS 2 will trigger a LDP MAC flush **flush-all-from-me** into the B-VPLS that will flush the FIB in MTU-3 for I-VPLS 2, avoiding the blackhole. A MC-LAG failure is emulated below:

```
*A:MTU-1# configure lag 1 shutdown

3 2010/04/01 01:47:53.49 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Address Withdraw packet (msgId 20844) to 192.0.2.1:0
MAC Flush (All MACs learned from me)
Service FEC EpipePWE3: ENET(5)/100 Group ID = 0 cBit = 0
Number of PBB-BMACs = 1
BMAC 1 = 00:11:11:11:11:11
Number of PBB-ISIDs = 1
ISID 1 = 2
Number of Path Vectors : 1
Path Vector( 1) = 192.0.2.11
"

A:MTU-3#
3 2010/04/01 01:47:58.63 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Recv Address Withdraw packet (msgId 20844) from 192.0.2.3:0
MAC Flush (All MACs learned from me)
Service FEC EpipePWE3: ENET(5)/100 Group ID = 0 cBit = 0
Number of PBB-BMACs = 1
BMAC 1 = 00:11:11:11:11:11
Number of PBB-ISIDs = 1
ISID 1 = 2
Number of Path Vectors : 3
Path Vector( 1) = 192.0.2.11
Path Vector( 2) = 192.0.2.1
Path Vector( 3) = 192.0.2.3
"

# Immediately after receiving the MAC flush, the CE-2 MAC is flushed and re-learnt but this
time linked to the B-MAC 00:21, which is the MTU-2 B-MAC
```

## PBB-VPLS M:1 Service Configuration

```
show service id 2 fdb pbb

=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:44:00:00:00:00  sap:1/1/2:4         100      N/A                L/0
=====
A:MTU-3# show service id 2 fdb pbb

=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:22:00:00:00:00  b-sdp:33:100        100      00:21:21:21:21:21 L/3
00:44:00:00:00:00  sap:1/1/2:4         100      N/A                L/0
=====
```

The following I-VPLS events are propagated into the B-VPLS depending on the **flush-all-but-mine** or **flush-all-from-me** keywords used in the configuration:

If the **flush-all-but-mine** keyword is configured (positive flush), the following events in the I-VPLS trigger a MAC flush into the B-VPLS:

1. TCN event in one or more of the related I-VPLS/M-VPLS.
2. Pseudowire/SDP binding activation with active/standby pseudowire (standby to active or down to up).
3. Reception of an LDP MAC withdraw flush-all-but-mine in the related I-VPLS.

If the **flush-all-from-me** keyword is configured (negative flush) the following events in the I-VPLS trigger a MAC flush into the B-VPLS:

1. MC-LAG active link failure (in our example).
2. Failure of a local SAP – requires **send-flush-on-failure** to be enabled in I-VPLS.
3. Failure of a local pseudowire/SDP binding – requires **send-flush-on-failure** to be enabled in I-VPLS.
4. Reception of an LDP MAC withdraws flush-all-from-me in the related I-VPLS.

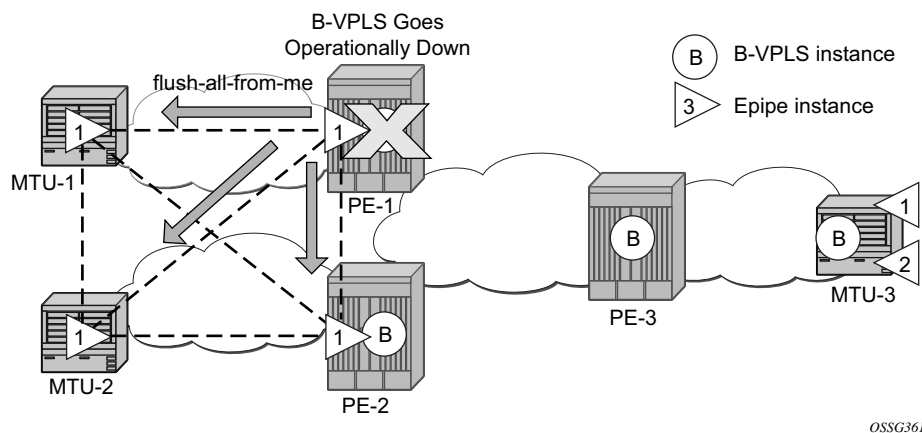
In addition to this and regardless of what type, MAC flush has been optimized to avoid flushing in the core PEs, flushing only the C-MACs mapped to a certain B-MAC (belonging to a specific ISID FIB) and the ability to indicate to core PEs which messages should always be forwarded endpoint-to-endpoint towards all PBB PEs regardless of the propagate-mac-flush setting in B-VPLS. All of this is implemented without the need of any additional CLI commands and it is part of draft-balus-l2vpn-pbb-ldp-ext-00.

Another extension supported to avoid blackholes within this mixed of I- and B-VPLS environments is the block-on-mesh-failure feature in PBB. When the VPLS mesh exists only in I-VPLS or in B-VPLS, and the block-on-mesh-failure feature is enabled, the regular VPLS behavior will apply (when all the mesh SDPs go down an LDP notification with pseudowire status bits = 0x01 - Pseudo Wire Not Forwarding – is sent over the spoke SDPs). When the active/standby pseudowire resiliency is implemented in I-VPLS such that the PBB PE performs the role of a PE-rs, the B-VPLS core replaces the pseudowire (SDP binding) mesh. The block-on-mesh-notification (LDP notification indicating pseudowire not forwarding) will be sent to the MTUs only when the related B-VPLS is operationally down. The B-VPLS core is operationally down only when all of its SAPs and SDPs are down.

The final feature that can be enabled in an I-VPLS with CLI is the send-flush-on-bvpls-failure feature.

```
*A:PE-1>config>service>vpls# send-flush-on-bvpls-failure # only valid for I-VPLS
- no send-flush-on-bvpls-failure
- send-flush-on-bvpls-failure
```

This feature is required to avoid blackholes when there is a full-mesh of pseudowires in the I-VPLS domain and the B-VPLS instance can go operationally down. The following figure shows a typical scenario where this feature is needed (normally when PBB-VPLS and multi-chassis end point are combined together).



**Figure 80: Send Flush on BVPLS Failure Example**

## Access Dual-Homing and MAC Notification

Although this section is focused on PBB in a MPLS based network, Alcatel-Lucent PBB implementation also allows the operator to use a native Ethernet infrastructure in the PBB core. Native Ethernet tunneling can be emulated using Ethernet SAPs to interconnect the related B-VPLS instances. In those cases, there is no LDP signaling available; hence, there is no MAC flush sent when the active link in a multi-homed access device fails.

The SR OS supports a mechanism to avoid potential blackholes in native Ethernet PBB networks. In addition to the source B-MAC associated with each B-VPLS, an additional B-MAC is associated with each MC-LAG supporting Multi-homed I-VPLS SAPs. The nodes that are in a multi-homed MC-LAG configuration share a common B-MAC on the related MC-LAG interfaces. When the **mac-notification no shutdown** command is executed, an Ethernet CFM notification message is sent from the node holding the active link. That message will be flooded in the B-VPLS domain using the MC-LAG SAP B-MAC as the source MAC address. The remote nodes will learn the customer MAC addresses behind the MC-LAG and will link them to this new SAP B-MAC. MC-LAG will keep track of the active link for each particular LAG associated to a SAP B-MAC. Should MC-LAG detect any new active link in a node, a new CFM notification message will be flooded from the new active node.

The following caveats and considerations must be taken into account:

- The MAC notification for access dual-homed devices is only supported in “network mode D”.
  - All network ports and B-SAPs must reside on slots populated with either an IOM3-XP or IMM. All other SAPs may reside on any IOM or IMM populated slot. This is enforced by the CLI.
  - MC-LAG SAPs must be on IOM3-XP/IMM.
  - Other SAPs may be on IOM1/IOM2 (non-PBB services, PBB Epipe SAP with no other SAP on local MC-LAG, PBB I-VPLS SAPs even if some of the other I-SAPs are on the MC-LAG).
  - This is automatically applicable to the 7750 SR-c4/12.
- Only MC-LAG is supported as dual-home mechanism.
- This mechanism is supported for native PBB and/or MPLS-based PBB-VPLS. Although it is mostly beneficial when native PBB is used in the core, it can also help to optimize the re-learning process in a MPLS-based core in case of MC-LAG failures, in addition to the existing LDP MAC flush procedures.

The example of this configuration shows the setup being used in this configuration example. MAC-notification will be configured in MTU-1 and MTU-2 for the dual-homed CE-2.

The first step is to configure the sap-bmac that will be used for the mac-notification messages. The source-bmac-lsb (source backbone MAC least significant bits) command has been added to the mc-lag branch so that the operator can decide the two last octets to be used in the sap-bmac. Those

two last octets can be derived from the lacp-key (if the use-lacp-key statement is used) or can be specifically defined.

```
A:MTU-1>config>redundancy>mc>peer>mc-lag#
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag
  <remote-lag-id>] system-priority <system-priority> source-bmac-lsb
  use-lacp-key
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag
  <remote-lag-id>] system-priority <system-priority> source-bmac-lsb
  <MAC-Lsb>
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag
  <remote-lag-id>] system-priority <system-priority>
- no lag <lag-id>

<lag-id>          : [1..200]
<admin-key>      : [1..65535]
<system-id>      : xx:xx:xx:xx:xx:xx    - xx [00..FF]
<remote-lag-id>  : [1..200]
<system-priority> : [1..65535]
<MAC-Lsb>       : [1..65535] or xx-xx or xx:xxe must be D and above
```

There must be a different sap-bmac per MC-LAG. The use of the lacp-key as a default for two least significant octets makes the operations simpler. In our example, the sap-bmac last two octets will come from the lacp-key:

```
A:MTU-1>config>redundancy>mc>peer>mc-lag# info
-----
lag 1 lacp-key 15 system-id 00:00:00:00:00:01 system-priority 65535
source-bmac-lsb use-lacp-key
no shutdown
-----
```

Therefore, the sap-bmac will be formed in the following way:

[4 first bytes of the source bmac + 2 bytes from source-bmac-lsb]

Finally, enable the mac-notification and instruct the b-vpls to use the sap-bmac at service level:

```
A:MTU-1>config>service# info
-----
pbb
  source-bmac 00:11:11:11:11:11
  mac-name "MTU-1" 00:11:11:11:11:11
  mac-name "MTU-2" 00:22:22:22:22:22
  mac-name "MTU-3" 00:31:31:31:31:31
exit
<snip>

vpls 2 customer 1 i-vpls create
  send-bvpls-flush all-from-me
  backbone-vpls 100:2
  exit
  stp
    shutdown
```

## PBB-VPLS M:1 Service Configuration

```
exit
sap lag-1 create
exit
no shutdown
exit
vpls 100 customer 1 b-vpls create
service-mtu 2000
pbb
    source-bmac 00:aa:aa:aa:aa:aa
    use-sap-bmac
exit
mac-notification
    no shutdown
exit
stp
    shutdown
exit
mrp
    no shutdown
exit
endpoint "core" create
    no suppress-standby-signaling
exit
spoke-sdp 111:100 endpoint "core" create
    stp
        shutdown
    exit
    precedence primary
exit
spoke-sdp 112:100 endpoint "core" create
    stp
        shutdown
    exit
exit
no shutdown
exit
```

The **mac-notification** command activates the described mechanism and has the following parameters:

```
A:MTU-1# configure service vpls 100 mac-notification
- mac-notification
```

```
[no] count          - Configure count for MAC-notification messages
[no] interval       - Configure interval for MAC-notification messages
[no] shutdown       - Configure admin state for MAC-notification messages
```

Where:

- **interval** <value> controls how often the subsequent MAC notification messages are sent. Default = 100 ms. Required values: 100 ms – 10 sec, in increments of 100 ms.
- **count** <value> controls how often the MAC notification messages are sent. Default = 3. Range: 1-10.



Note that the “count” and “interval” parameters can also be configured at the service context. The settings configured at the B-VPLS service context take precedence though.

```
A:MTU-1# configure service mac-notification
- mac-notification

[no] count          - Configure count for MAC-notification messages
[no] interval      - Configure interval for MAC-notification messages
```

The **use-sap-bmac** statement enables (on a per B-VPLS basis) the use of the source B-MAC allocated to the multi-homed SAPs (assigned to the MC-LAG) in the related I-VPLS service (could be Epipe service as well). Note that the command will fail if the value of the source-bmac assigned to the B-VPLS is the hardware (chassis) B-MAC. In other words, the source-bmac must be a configured one. The **use-sap-bmac** statement is by default off.

As soon as the **mac-notification no shutdown** command is executed, an Ethernet CFM notification message is sent from MTU-1, which is the node where the active MC-LAG link resides. The CFM message will have the source mac “00:aa:aa:aa:00:0f” (4 first bytes of the configured source bmac + 2 bytes from the configured source-bmac-lsb, which is 15 in hex) and will be flooded throughout the B-VPLS domain. Should the link between CE-2 and MTU-1 fail, the MC-LAG protocol will activate the redundant link and MTU-2 will immediately issue a CFM message with the shared sourced sap-bmac that will be flooded in the B-VPLS domain.

## PBB and IGMP Snooping

IGMP snooping can be enabled on I-VPLS SAPs and SDPs (it cannot be enabled on B-VPLS). The 7x50/7710 can keep track of IGMP joins received over individual B-SDPs or B-SAPs, and it starts flooding the Multicast Group (and only the multicast group) to ALL B-components (using the Group B-MAC for I-SID) as soon as the first IGMP join for that Multicast Group is received in one of the B-SxP components.

The first IGMP join message received over the local B-VPLS will add all the B-VPLS SAP/SDP components into the related multicast table associated with the I-VPLS context. When the querier is connected to a remote I-VPLS instance, over the B-VPLS infrastructure, its location is identified by the B-VPLS SDP/SAP on which the query was received and also by the source B-MAC address used in the PBB header for the query message, the B-MAC associated with the B-VPLS instance on the remote PBB PE.

The following excerpt shows an I-VPLS with IGMP snooping enabled and some static groups added on a SAP. Note that we are also configuring the location of the querier by adding the B-MAC where the querier is connected to (in this example MTU-3) and adding the two B-VPLS spoke-sdps as mrouter ports (note that the B-VPLS mrouter ports are added under the I-VPLS backbone-vpls context).

## PBB-VPLS M:1 Service Configuration

The **mac-name** command translates MACs into strings so that the names can be used instead of typing the entire MAC address every time needed.

```
*A:MTU-1>config>service>pbb# info
-----
      mac-name "MTU-1" 00:11:11:11:11:11
      mac-name "MTU-2" 00:22:22:22:22:22
      mac-name "MTU-3" 00:31:31:31:31:31
-----

*A:MTU-1>config>service>vpls# info
-----

      send-flush-on-failure
      send-flush-on-bvpls-failure
      send-bvpls-flush all-from-me
      backbone-vpls 100
         igmp-snooping
            mrouter-dest "MTU-3"
         exit
         sdp 111:100
            igmp-snooping
               mrouter-port
            exit
         exit
         sdp 112:100
            igmp-snooping
               mrouter-port
            exit
         exit
      exit
      stp
         shutdown
      exit
      igmp-snooping
         no shutdown
      exit
      sap 1/1/4:1 create
         igmp-snooping
            static
               group 228.0.0.1
                  starg
               exit
               group 228.0.0.2
                  starg
               exit
               group 239.0.0.1
                  source 172.16.99.99
               exit
            exit
         exit
      exit
      no shutdown
-----

*A:MTU-1>config>service>pbb# i
```

As in regular VPLS instances, mrouter ports are added to all the multicast groups:

```
*A:MTU-1> show service id 1 mfib
=====
Multicast FIB, Service 1
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*                *                   b-sdp:111:100      100     Fwd
                  b-sdp:112:100      100     Fwd
*                228.0.0.1          sap:1/1/4:1        Local   Fwd
                  b-sdp:111:100      100     Fwd
                  b-sdp:112:100      100     Fwd
*                228.0.0.2          sap:1/1/4:1        Local   Fwd
                  b-sdp:111:100      100     Fwd
                  b-sdp:112:100      100     Fwd
172.16.99.99    239.0.0.1          sap:1/1/4:1        Local   Fwd
                  b-sdp:111:100      100     Fwd
                  b-sdp:112:100      100     Fwd
-----
Number of entries: 4
=====
```

Note that when the “show service id x mfib” command is issued in a B-VPLS, the group B-MAC entries are shown, whereas when the same command is issued in a I-VPLS, the IGMP (S,G) and (\*,G) entries for the I and B components are shown if IGMP snooping is enabled.

## MMRP Policies and ISID-Based Filtering for PBB Inter-Domain Expansion

As discussed in the [MMRP for Flooding Optimization on page 526](#), MMRP is used in the backbone VPLS instances to build per I-VPLS flooding trees. Each I-VPLS has an associated group B-MAC in the B-VPLS, which is derived from the ISID, and is advertised by MMRP throughout the whole B-VPLS context, regardless of whether a certain I-VPLS is present in one or all the B-VPLS PEs.

In an inter-domain environment, the same B-VPLS can be defined in different domains and as such MMRP will advertise all the group B-MACs in every domain, wasting resources in all the PEs no matter if a particular ISID, and hence its group B-MAC, are not required in one of the domains. When MMRP is enabled in a particular PE, data plane and control plane resources are consumed and they must be taken into consideration when designing PBB-VPLS networks:

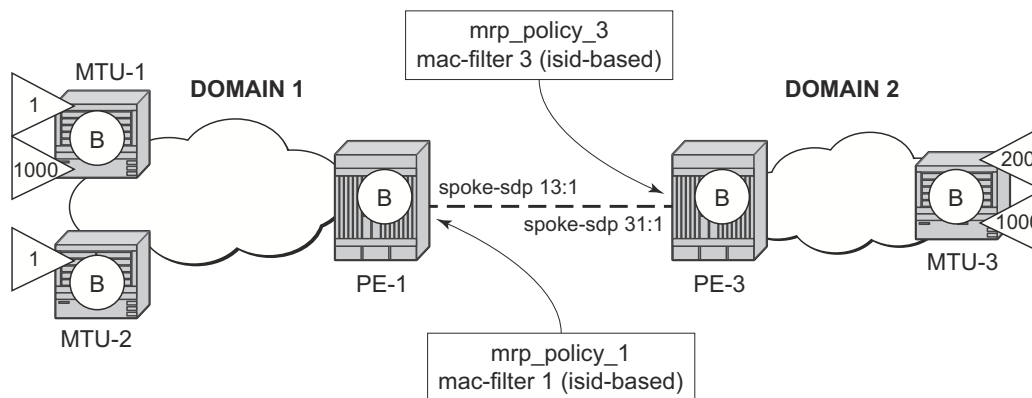
- Control plane – MMRP processing takes CPU cycles and the number of attributes that can be advertised is not unlimited
- Data plane – each group B-MAC registration takes one MFIB entry (the MFIB is shared between MMRP and IGMP/PIM snooping)

From 8.0R1, the 7x50 SR/ESS supports MMRP policies and ISID-based filters so that control plane and data plane resources can be saved when I-VPLS instances are not defined in all the domains.

The following figure illustrates an example of usage for MMRP policies and ISID-based filters that will be configured in this section. “Domain 1” and “domain 2” will have a range of local ISIDs each and a range of “inter-domain” ISIDs:

- Domain 1 local ISIDs: from 1 to 100
- Domain 2 local ISIDs: from 101 to 200
- Inter-domain ISIDs: from 1000 to 2000

By applying the MMRP policies indicated in the figure, domain 1 attributes will be prevented from being declared and registered in domain 2 and vice versa, domain 2 attributes from being declared and registered in domain 1. The egress mac-filters will drop any traffic sourced from a local ISID preventing it to be transmitted to the remote domain.



OSSG667

**Figure 81: Inter-Domain B-VPLS and MMRP Policies/ISID-Based Filters Example**

## MMRP Policies

The following output shows the MMRP policy configuration in node PE-1. This policy will block any registration/declaration except those for ISIDs 1000-2000. Note that packets will be compared against the configured matching ISIDs as long as the pbb-etype matches the one configured on the port or SDP.

```
*A:PE-1>config>service>mrp# info
-----
mrp-policy "mrp_policy_1" create
  default-action block
  description "allow-inter-domain-isids"
  entry 10 create
    action allow
    match
      isid 1000 to 2000
    exit
  exit
exit
-----
```

Once the MMRP policy is configured, it must be applied on the corresponding SAP or sdp-binding. Note that an mrp-policy can be applied to a B-VPLS SAP, B-VPLS spoke-sdp or B-VPLS mesh-sdp:

```
*A:PE-1>config>service>vpls# info
-----
service-mtu 2000
stp
shutdown
-----
```

## PBB-VPLS M:1 Service Configuration

```
        exit
        mrp
            no shutdown
        exit
    ...
    spoke-sdp 13:1 create
        mrp
            mrp-policy "mrp_policy_1"
        exit
    exit
    ...
    no shutdown
-----
```

In the same way, `mrp_policy_3` will be configured in PE-3.

Some additional considerations about the MMRP policies:

- Different entries within the same `mrp-policy` can have overlapping ISID ranges. The entries will be evaluated in the order of their IDs and the first match will cause the implementation to execute the associated action for that entry and then to exit the `mrp-policy`.
- If no ISID is specified in the match condition then:
  - If the action is “end-station”, no entry is added and the action is block.
  - If the action is different from “end-station”, every ISID is considered for that action.
- The `mrp-policy` specifies either a forward or a drop action for the group B-MAC attributes associated with the ISIDs specified in the match criteria.

```
*A:PE-1>config>serv>mrp>mrp-policy>entry# action ?
- action <action>
- no action

<action>                : none|block|allow|end-station
```

- Note that there is an additional action called **end-station**. This action specifies that an end-station emulation is present on the SAP/SDP-binding where the policy has been applied. The matching ISIDs will not get declared/registered in the SAP/SDP-binding (just like the **block** action), however those attributes will get mapped as static MMRP entries on the SAP/SDP-binding, which implicitly get instantiated in the data plane as MFIB entries associated with that SAP/SDP-binding for the related group B-MAC. When the action is “end-station”, the default-action must be block:

```
*A:PE-3>config>serv>mrp>mrp-policy# default-action allow
MINOR: SVCNMR #5904 Mrp-policy default-action must be block when end-station action exists
```

- The **end-station** action can be used in the inter-domain gateways when, for instance, we do not want MMRP control plane exchanges between domains. The following output shows how to define the static MMRP entries 1000-2000 in PE-3 without receiving any declaration for any of those attributes or having any of those locally configured.

```

*A:PE-3# configure service mrp mrp-policy "mrp_policy_3"
*A:PE-3>config>serv>mrp>mrp-policy# info
-----
        default-action block
        entry 10 create
            action end-station
            match
                isid 1000 to 2000
            exit
        exit
-----

*A:PE-3# show service id 8 mfib
=====
Multicast FIB, Service 8
=====
Source Address  Group Address          Sap/Sdp Id                Svc Id  Fwd/Blk
-----
*                01:1E:83:00:03:E8        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:03:E9        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:03:EA        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:03:EB        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:03:EC        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:03:ED        b-sdp:32:8                Local   Fwd
...
*                01:1E:83:00:07:CB        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:07:CC        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:07:CD        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:07:CE        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:07:CF        b-sdp:32:8                Local   Fwd
*                01:1E:83:00:07:D0        b-sdp:32:8                Local   Fwd
-----
Number of entries: 1001
=====
*A:PE-3#

```

- The mrp-policy can be applied to multiple B-VPLS services as long as the scope of the policy is **template** (the scope can also be **exclusive**).
- Any changes made to the existing policy will be applied immediately to all services where this policy is applied. For this reason, when many changes are required on a mrp-policy, it is recommended that the policy be copied to a work-in-progress policy. That work-in-progress policy can be modified until complete and then written over the original mrp-policy. You can use the **config mrp-policy copy** command to work with the policies in this manner. The **renum** command can also help to change the entries sequence order.

```

*A:PE-3# configure service mrp copy
- copy <src-mrp-policy> to <dst-mrp-policy>

<src-mrp-policy>      : [32 chars max]
<dst-mrp-policy>     : [32 chars max]

*A:PE-3# configure service mrp mrp-policy "mrp_policy_3" renum

```

## PBB-VPLS M:1 Service Configuration

```
- renum <src-entry-id> to <dst-entry-id>

<src-entry-id>      : [1..65535]
<dst-entry-id>     : [1..65535]
```

- The **no** form of the **mrp-policy** command deletes the mrp-policy. An mrp policy cannot be deleted until it is removed from all the SAPs/SDP-bindings where it is applied.

### ISID-based filters

The MMRP policies help to control the exchange of group B-MAC attributes across domains. Based on the registration state of a specific group B-MAC on a SAP/SDP-binding, the broadcast/unknown-unicast/multicast traffic for a particular I-VPLS will be allowed or dropped. However, if we want to avoid that ANY local ISID packet is flooded to the remote B-VPLS domain, we also need to filter at the data plane all the packets tagged with the local ISIDs at the gateway PEs. ISID-based filters will prevent the local ISIDs from sending any packet with unicast B-MAC to the remote domain. This is particularly useful for PBB-epipe services across domains, where all the frames use unicast B-MACs and MMRP policies cannot help since they only act on group B-MAC packets.

The following CLI output shows how to configure an ISID-based filter that drops all the traffic sourced from the local ISIDs on PE-1 (note that the default action is drop and it does not show up in the configuration).

```
*A:PE-1>config>filter# info
-----
      mac-filter 1 create
        description "drop_local_isids"
        type isid
        entry 10 create
          match
            isid 1000 to 2000
          exit
          log 101
          action forward
        exit
      exit
-----
```

Once the filter is configured, it must be applied on a B-VPLS SAP or SDP-binding and always at egress.

```
*A:PE-1>config>service>vpls# info
-----
      service-mtu 2000
      stp
        shutdown
      exit
      mrp
        no shutdown
      exit
-----
```



```

...
    spoke-sdp 13:1 create
        egress
            filter mac 1
        exit
    mrp
        mrp-policy "mrp_policy_1"
    exit
exit
...
no shutdown
-----

```

Some additional comments about ISID-based filters:

- The **type isid** statement must be added before introducing any ISID in the match command, otherwise the system will show an error:

```

*A:PE-1>config>filter>mac-filter>entry$ match isid 1000 to 2000
MINOR: FILTER #1533 Can not set ISID values when filter type is not 'isid'

```

```

*A:pe-1>config>filter>mac-filter$ type isid
MINOR: FILTER #1561 Cannot change filter type when filter contains entries

```

- Once the operator sets the “type isid”, the filter cannot be applied at ingress. Only egress ISID-based filters are allowed:

```

*A:PE-1>config>service>vpls>mesh-sdp# ingress filter mac 1
MINOR: SVCMGR #2050 Can not apply filter of type 'isid' on ingress

```

- Like any filter or MMRP policy, the filter can be applied to multiple B-VPLS services as long as the scope of the policy is “template” (the scope can also be “exclusive”).
- The following command shows the filter configuration and packets that have matched the filter (field “Egr. Matches”):

```

*A:PE-1# show filter mac 1
=====
Mac Filter
=====
Filter Id      : 1                               Applied       : Yes
Scope         : Template                       Def. Action   : Drop
Entries       : 1                               Type          : isid
Description   : drop_local_isids
-----
Filter Match Criteria : Mac
-----
Entry         : 10                               FrameType     : Ethernet
Description   : (Not Specified)
Log Id        : 101
ISID          : 1000..2000
Match action: Forward
Next Hop      : Not Specified
Ing. Matches: 0 pkts
Egr. Matches: 4 pkts (728 bytes)

```

## PBB-VPLS M:1 Service Configuration

```
=====
*A:PE-1#
```

- Like any other filter, the matching packets can be “logged”. An example follows (not that the Ethertype is 0x88e7, which is the default standard etype for PBB):

```
*A:PE-1# show filter log 101
```

```
=====
Filter Log
=====
```

```
Admin state : Enabled
Description : Default filter log
Destination : Memory
Wrap        : Enabled
-----
```

```
Maximum entries configured : 1000
Number of entries logged   : 4
2010/12/29 20:53:30 Mac Filter: 1:10 Desc:
Interface: to2 Direction: Egress Action: Forward
Src MAC: 00-11-11-11-11-11 Dst MAC: 01-1e-83-00-03-e8 EtherType: 88e7
Hex: 00 00 03 e8 ff ff ff ff ff ff 26 d9 ff 00 00 00
      08 00 45 00 00 a8 00 00 40 00 01 11 f8 43 01 01
      01 01 7f 00 00 00 c0 09 0d af 00 94 da b8 00 01

2010/12/29 20:53:31 Mac Filter: 1:10 Desc:
Interface: to2 Direction: Egress Action: Forward
Src MAC: 00-11-11-11-11-11 Dst MAC: 01-1e-83-00-03-e8 EtherType: 88e7
Hex: 00 00 03 e8 ff ff ff ff ff ff 26 d9 ff 00 00 00
      08 00 45 00 00 a8 00 00 40 00 02 11 f7 43 01 01
      01 01 7f 00 00 00 c0 09 0d af 00 94 c7 de 00 01
```

## B-VPLS and I-VPLS Show and Debug Commands

The following commands can help to check the B-VPLS and I-VPLS configuration and their related parameters.

```
*A:MTU-1# show service id 100 base # B-VPLS
=====
Service Basic Information
=====
Service Id       : 100                Vpn Id           : 0
Service Type    : b-VPLS
Description        : (Not Specified)
Customer Id       : 1
Last Status Change: 04/01/2010 02:45:27
Last Mgmt Change  : 04/01/2010 02:40:26
Admin State       : Up                  Oper State        : Up
MTU             : 2000                Def. Mesh VC Id  : 100
SAP Count         : 0                  SDP Bind Count   : 2
Snd Flush on Fail: Disabled           Host Conn Verify : Disabled
Propagate MacFlush: Disabled
Oper Backbone Src : 00:11:11:11:11:11
Use SAP B-MAC     : disabled
i-Vpls Count    : 2
Epipe Count       : 0
-----
Service Access & Destination Points
-----
Identifier                                     Type           AdmMTU  OprMTU  Adm  Opr
-----
sdp:111:100 S(192.0.2.1)                       n/a           8000    8000    Up   Up
sdp:112:100 S(192.0.2.2)                       n/a           8000    8000    Up   Up
=====
*A:MTU-1#

*A:MTU-1# show service id 1 base # I-VPLS
=====
Service Basic Information
=====
Service Id       : 1                Vpn Id           : 0
Service Type    : i-VPLS
Description        : (Not Specified)
Customer Id       : 1
Last Status Change: 04/01/2010 00:14:58
Last Mgmt Change  : 04/01/2010 03:07:23
Admin State       : Up                  Oper State        : Up
MTU               : 1514                Def. Mesh VC Id  : 1
SAP Count         : 1                  SDP Bind Count   : 0
Snd Flush on Fail: Enabled           Host Conn Verify : Disabled
Propagate MacFlush: Disabled
b-Vpls Id       : 100                Oper ISID      : 1
b-Vpls Status  : Up
Snd Flush in bVpls: All-from-me
Flsh On bVpls Fail: enabled           Prop Flsh fr bVpls: disabled
-----
Service Access & Destination Points
-----
```

## PBB-VPLS M:1 Service Configuration

```

Identifier                               Type           AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:1                             q-tag         1518    1518    Up   Up
=====
*A:MTU-1#

```

The following command shows all the I-VPLS instances multiplexed into a particular B-VPLS.

```

*A:MTU-1# show service id 100 i-vpls
=====
Related i-Vpls services for b-Vpls service 100
=====
i-Vpls SvcId      Oper ISID      Admin          Oper
-----
1                1             Up             Up
2                2             Up             Up
-----
Number of Entries : 2
=====
*A:MTU-1#

```

Some useful commands to check the I and B VPLS FIBs correlating C-MACs and B-MACs:

```

*A:MTU-1# show service id 1 fdb pbb
=====
Forwarding Database, i-Vpls Service 1
=====
MAC                Source-Identifier  B-Svc  b-Vpls MAC      Type/Age
-----
00:11:00:00:00:00 sap:1/1/4:1       100    N/A              L/0
00:33:00:00:00:00 b-sdp:111:100    100    00:31:31:31:31:31 L/0
=====
*A:MTU-1#

```

```

*A:MTU-1# show service id 100 fdb pbb
=====
Forwarding Database, b-Vpls Service 100
=====
MAC                Source-Identifier  iVplsMACs  Epipes  Type/Age
-----
00:31:31:31:31:31 sdp:111:100       2          0       L/0
24:7f:ff:00:00:00 sdp:111:100       0          0       L/0
=====
*A:MTU-1#

```

If **mac-names** are used in the configuration, the following commands can help to show the translations:

```
A:MTU-1# show service pbb mac-name
=====
MAC Name Table
=====
MAC-Name                               MAC-Address
-----
MTU-1                                   00:11:11:11:11:11
MTU-2                                   00:21:21:21:21:21
MTU-3                                   00:31:31:31:31:31
=====
*A:MTU-1#

A:MTU-1# show service pbb mac-name MTU-3 detail
=====
Services Using MAC name='MTU-3' addr='00:31:31:31:31:31'
=====
Svc-Id                                  ISID
-----
3                                         3
1                                         N/A
-----
Number of services: 2
=====
*A:MTU-1#
```

The following command shows the base MAC notification parameters as well as the source B-MAC configured at the service PBB level. Note that those values are overridden by any potential mac-notification or source B-MAC values configured under the B-VPLS service context.

```
A:MTU-1# show service pbb base
=====
PBB MAC Information
=====
MAC-Notif Count                         : 3
MAC-Notif Interval                       : 1
Source BMAC                              : 00:11:11:11:11:11
=====
*A:MTU-1#
```

If mac-notification is used in a particular B-VPLS, the configured least significant bits for the sap-bmac on a particular MC-LAG can be shown by using the detailed view of the **show lag** command:

```
A:MTU-1# show lag 1 detail
=====
LAG Details
=====
Description                               : N/A
-----
Details
```

## PBB-VPLS M:1 Service Configuration

```
-----  
Lag-id          : 1                Mode           : access  
Adm             : up              Opr            : up  
<snip>  
MC Peer Address : 192.0.2.21      MC Peer Lag-id  : 1  
MC System Id    : 00:00:00:00:01  MC System Priority : 65535  
MC Admin Key   : 15           MC Active/Standby : active  
MC LACP ID in use : true          MC extended timeout : false  
MC Selection Logic : local master decided  
MC Config Mismatch : no mismatch  
Source BMAC LSB : use-lacp-key   Oper Src BMAC LSB : 00:0f  
  
<snip>
```

The following commands allow the operator to check the LDP label mapping, label withdrawal, messages and also the MAC-flush messages for regular VPLS, for I-VPLS and B-VPLS including the PBB extensions and TLVs.

```
*A:MTU-1# show debug  
debug  
  router "Base"  
    ldp  
      peer 192.0.2.1  
        event  
        exit  
        packet  
          init detail  
          label detail  
        exit  
      exit  
      peer 192.0.2.2  
        event  
        exit  
        packet  
          init detail  
          label detail  
        exit  
      exit  
    exit  
  exit  
exit
```

The following debug commands can help the operator to troubleshoot MMRP.

```
*A:MTU-1# debug service id 100 mrp
- mrp
- no mrp

    all-events      - Enable/disable MRP debugging for all events
[no] applicant-sm  - Enable/disable MRP debugging for applicant state
                    machine changes
[no] leave-all-sm  - Enable/disable MRP debugging for leave all state
                    machine changes
[no] mmrp-mac      - Enable/disable MRP debugging for a particular MAC
                    address
[no] mrpdu         - Enable/disable MRP debugging for Rx/Tx MRP PDUs
[no] periodic-sm   - Enable/disable MRP debugging for periodic state
                    machine changes
[no] registrant-sm - Enable/disable MRP debugging for registrant state
                    machine changes
[no] sap           - Enable/disable MRP debugging for a particular SAP
[no] sdp           - Enable/disable MRP debugging for a particular SDP
```

## Conclusion

PBB-VPLS allows the service providers to scale VPLS services by multiplexing customer I-VPLS instances into one or more B-VPLS instances. This multiplexing dramatically reduces the number of services, pseudowires and MAC addresses in the core and therefore allows the service provider to scale Layer 2 multi-point networks and provide services across international backbones.

The example used in this section shows the configuration of the customer and backbone VPLS instances as well as all the related features which are required for this environment. Show and debug commands have also been suggested so that the operator can verify and troubleshoot the service.