

# Multi-Segment Pseudowire Routing

---

## In This Chapter

This section describes advanced multi-segment pseudowire routing configurations.

Topics in this section include:

- [Applicability on page 436](#)
- [Summary on page 437](#)
- [Overview on page 438](#)
- [Configuration on page 441](#)
- [Conclusion on page 487](#)

## **Applicability**

This section is applicable to all of the 7750, 7450 and 7710 SR series and was tested on release 10.0R4. There are no specific pre-requisites for this configuration.

## Summary

Starting with SROS 9.0R3, the SR/ESS portfolio supports the use of Multi-Segment Pseudowire (MS-PW) routing for Epipe services. MS-PW routing is described in draft-ietf-pwe3-dynamic-ms-pw, also known as Dynamic Placement of MS-PW and it is an extension of the procedures proposed in RFC 6073 (static MS-PW) to enable multi-segment pseudowires to be dynamically placed. Ultimately MS-PW Routing provides the capability of setting up MS-PWs without provisioning the S-PE (Switching PEs).

This configuration note will go through the configuration process required to setup MS-PW routing and will provide two configuration examples typically deployed in Service Providers: MS-PW within the same Autonomous System (AS) and MS-PW across two different AS. Different configuration options are tested and described in each example.

## Overview

From a data plane perspective, MS-PW Routing does not introduce any changes with respect to the existing MS-PW architecture. However from the control plane perspective, MS-PW Routing brings a new information model and set of procedures to set up a MS-PW. These are the building blocks defined by the MS-PW Routing feature:

- A new information model is introduced for dynamic MS-PW based on the FEC129, Attachment Individual Identifier (AII) Type 2. Note that static MS-PW uses FEC128 while VPLS with BGP-AD uses FEC129, but with AII Type 1 instead.
  - FEC129 is suitable for applications where the local PE with a SAII (Source Attachment Individual Identifier) must automatically learn the remote TAI (Target Attachment Individual Identifier), normally through BGP, before launching the LDP mapping message for the pseudowire setup. The following figure shows the FEC129 structure:

G.Pwid (0x81)	C	Pw Type	Pw Info Length
AGI Type	Length	Value	
AGI Value (Cont.)			
All Type	Length	Value	
SAII Value (Cont.)			
All Type	Length	Value	
TAII Value (Cont.)			

ACG0004A

**Figure 68: FEC129 Structure**

- The Attachment Group Identifier (AGI) is not used in dynamic MS-PW signaling. In VPLS, it typically carries the instance identifier. It is zero in dynamic MS-PWs.
- The SAII and TAI (or pseudowire end-point identifiers) are encoded in FEC129 and can have two different formats: AII Type 1 or AII Type 2.
- AII Type 1 is composed of a fixed 32-bit value unique on the local PE. This AII Type is used by VPLS when BGP-AD is needed.
- AII Type 2 is composed of `GID:prefix:AC-ID` (Global-ID:prefix:Attachment-Circuit-ID) and allows summarization for scalability in large networks. The GID is normally derived from the AS number, the prefix from the node system address and the AC-ID is the local pseudowire end-point identifier. The combination of the three identifiers gives us a globally unique 96-bit AII value. In general, the same global ID and prefix are assigned for all ACs belonging to the same T-PE. This is not a strict requirement though.

All Type=2	Length	Global ID
	Global ID (Cont.)	Prefix
	Prefix (Cont.)	AC ID
	AC ID (Cont.)	

ACG0004B

**Figure 69: All Type 2 Format**

- A MS-PW routing table must be built in all the T-PEs and S-PEs through one of the following two mechanisms:
  - Multi-Protocol BGP (MP-BGP), using a new NLRI and a new SAFI (pseudowire routing SAFI=6, with AFI=25 L2VPN). The FEC129 All Type 2 global values are mapped in the pseudowire routing NLRI and advertised by BGP. In the tested release, SROS only supports an NLRI comprising a Length, RD, Global ID and 32-bit Prefix, that is, the AC ID is not included in the advertised NLRI. The AC ID is not included as indicated in the draft-ietf-pwe3-dynamic-ms-pw since “the source T-PE knows by provisioning the AC ID on the terminating T-PE to use in signaling. Hence, there is no need to advertise a “fully qualified” 96 bit address on a per pseudowire Attachment Circuit basis. Only the T-PE Global ID, Prefix, and prefix length needs to be advertised as part of well known BGP procedures”. This also minimizes the amount of routing information that is advertised in BGP to only what is necessary to reach the far-end T-PE.

Length	
Route Distinguisher (8 bytes)	
	Global ID
Global ID	Prefix
Prefix	AC ID
AC ID	

ACG0004C

**Figure 70: Pseudowire Routing NLRI (the AC ID is always zero)**

- Static routes, configurable via CLI

- Once the MS-PW routing table is populated, Targeted LDP (TLDP) will make use of it to signal the MS-PW all the way from the originating T-PE to the terminating T-PE as well as in the reverse direction. The following methods will be used:
  - At the originating T-PE<sup>1</sup>, a longest-match lookup will be performed in the pseudowire routing table for the configured TAI. Based on the lookup outcome, a label mapping message will be sent to the Next Signaling Hop (NSH).
  - At the intermediate S-PEs and terminating T-PE, a longest-match lookup between the TAI Type 2 included in the TLDP signaling message and entries installed in the pseudowire routing table will be performed.
  - Alternatively to the pseudowire routing table lookup, TLDP can also use explicit routing, as per section 7.4.2 of draft-ietf-pwe3-dyn-ms-pw. If that is the case, a “path” must be configured at the T-PEs. The originating T-PE will include an ERO (Explicit Route Object) in the TLDP label mapping, containing all the S-PE hops specified in the configured path. Each S-PE along the path will remove its own entry from the ERO and will forward the label mapping message to the next hop.

The SROS, starting from R9.0R3, supports the information model and all the methods described above:

- Dynamic placement through MP-BGP, with the pseudowire routing NLRI
- Static routes
- Explicit paths

In addition to the above, the following features are supported on dynamic MS-PW:

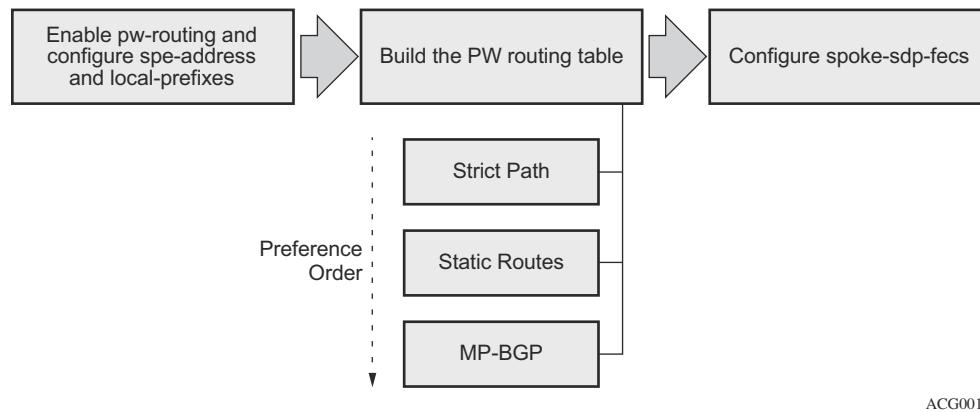
- Auto-configuration of spoke SDPs at T-PE (if enabled on a T-PE, there is no need for configuring the TAI of the remote T-PE. Refer to [Active/Passive Signaling and Auto-Configuration on page 454](#). The auto-configuration is typically used in hub-and-spoke scenarios. The TAI would only be configured on the spoke T-PE while the TAI would be automatically provisioned on the hub T-PE if the auto-config parameter is added.
- OAM using vccv-ping and vccv-trace
- Pseudowire redundancy
- Control word
- Hash label (if the chassis dependency requirements are met)
- Standby-signaling-master and standby-signaling-slave commands
- Filters

---

1. The “originating T-PE” will be the T-PE initiating the MS-PW signaling. Refer to the [Active/Passive Signaling and Auto-Configuration](#) section for further information.

## Configuration

The following flow-chart shows the configuration process to be followed when setting up MS-PW routing. Base IGP and MPLS configuration is assumed to be in place before these configuration tasks can be carried out.



**Figure 71: Configuration Flow Chart**

The following subsections review these three steps, including all the options in detail.

- [Pseudowire Routing Enablement on page 442](#)
- [Building the Pseudowire Routing Table on page 445](#)
- [Spoke-SDP-FEC Timers on page 456](#)

## Pseudowire Routing Enablement

The first step in the configuration is to enable pw-routing and configure the required pw-routing basic parameters: the spe-address (in S-PEs and T-PEs) and the local-prefix/prefixes (only required in T-PEs). A new pw-routing context has been added from 9.0R3 onward. The following CLI example shows the configuration of the spe-address and local-prefixes.

```
A:PE-1>config>service# info
-----
pw-routing
  spe-address 65536:192.0.2.1
  local-prefix 65536:192.0.2.1 create
    advertise-bgp route-distinguisher 65536:11 community 65535:11
    advertise-bgp route-distinguisher 65536:12 community 65535:12
  exit
  local-prefix 65536:192.0.2.11 create
    advertise-bgp route-distinguisher 65536:11 community 65535:11
  exit
exit
```

In order to enable support for MS-PW routing on a 7x50 node, a single, globally unique, S-PE ID (known as the spe-address) is first configured under **config>service>pw-routing** on each 7x50 to be used as a T-PE or S-PE. The S-PE address has the format global-id:prefix. It is not possible to configure any local prefixes used for pseudowire routing or to configure spoke SDPs using dynamic MS-PWs at a T-PE unless an S-PE address has already been configured. The S-PE address is used as the address of a node when populating the switching point TLV in the LDP label mapping message and the pseudowire status notification sent for faults at an S-PE. The following CLI output shows the spe-address configuration format:

```
A:PE-1# configure service pw-routing spe-address
- no spe-address
- spe-address <global-id:prefix>

<global-id:prefix> : <global-id>:{<prefix>|<ipaddress>}
global-id - [1..4294967295]
prefix - [1..4294967295]
ipaddress - a.b.c.d
```

Where:

- <global-id> is normally the 2 or 4-byte ASN identifying the network (although nothing prevents the operator from configuring any value here)
- <prefix> is normally the node's system address (although any value in ip address or decimal format can be used)

If an S-PE is capable of Dynamic MS-PW signaling, but is not assigned with an S-PE address, then on receiving a Dynamic MS-PW label mapping message the S-PE will return a Label Release with the "LDP\_RESOURCES\_UNAVAILABLE" (0x38) status code. Note that the S-PE address



cannot be changed unless the dynamic MS-PW configuration is completely removed; therefore it is recommended to configure the spe-address carefully and keep it for the life of the services.

The second basic pw-routing context parameter is the local-prefix:

```
A:PE-1# configure service pw-routing local-prefix
- local-prefix <local-prefix> [create]
- no local-prefix <local-prefix>

<local-prefix>      : <global-id>:<ip-addr>|<raw-prefix>
ip-addr             - a.b.c.d
raw-prefix          - [1..4294967295]
global-id           - [1..4294967295]

[no] advertise-bgp  - Configure BGP advertisement
```

One or more local (Layer 2) prefixes (up to a maximum of 16), which are formatted in the style of <global-id>:<ipv4-address>, are supported. A local prefix identifies a T-PE in the pseudowire routing domain. When using explicit paths or static-routes, the definition of the local-prefix (or local-prefixes) without any further attribute is enough. However, when BGP is used, the advertise-bgp parameter along with a route-distinguisher (RD) value and an optional BGP community is required.

```
A:PE-1# configure service pw-routing local-prefix 65536:192.0.2.1 advertise-bgp
- advertise-bgp route-distinguisher <rd> [community <community>]
- no advertise-bgp route-distinguisher <rd>

<rd>                : <ip-addr:comm-val>|<2byte-asnumber:ext-comm-val>|
                    <4byte-asnumber:comm-val>
ip-addr              - a.b.c.d
comm-val             - [0..65535]
2byte-asnumber       - [1..65535]
ext-comm-val         - [0..4294967295]
4byte-asnumber       - [1..4294967295]

<community>        : <asnumber:comm-val>
asnumber             - [1..65535]
comm-val             - [0..65535]
```

Up to four unique RDs (and communities) can be configured per each local-prefix. Different RDs for the same prefix allow the operator to advertise the same prefix coming from up to four different next-signaling hops (NSH). Route-Reflectors would reflect the four routes in that case, whereas only one would be reflected should the same RD be used.

```
*A:PE1>config>service>pw-routing>local-prefix# info
-----
advertise-bgp route-distinguisher 400:20
advertise-bgp route-distinguisher 500:3
advertise-bgp route-distinguisher 600:300
advertise-bgp route-distinguisher 700:100
-----

*A:PE1>config>service>pw-routing>local-prefix# advertise-bgp route-distinguisher 800:200
MINOR: SVCNMR #6072 Maximum number of RD's has been reached
```

## Pseudowire Routing Enablement

For each local prefix, BGP then advertises each global ID/prefix tuple and unique RD and community (if configured) using the MS-PW NLRI, based on the aggregated FEC129 AII Type 2 and the Layer 2 VPN/PW routing AFI/SAFI 25/6, to each BGP neighbor, subject to local BGP policies.

## Building the Pseudowire Routing Table

Once the `spe-address` and the `local-prefix(es)` have been configured and before configuring the `Epipe` service itself on the T-PE nodes, we need to populate the pseudowire routing table in all the participating T-PE and S-PE nodes, so that TLDP knows what the Next Signaling Hop (NSH) is and sends LDP Label Mapping messages.

The pseudowire routing table will be populated with local prefixes, static-routes and BGP routes, where the static-routes have preference over the BGP-learned routes. The pseudowire routing table can be overridden by the explicit paths, should the operator want to configure them. Therefore, when TLDP signals an LDP Label Mapping for a given TAIL, it will:

- First check if there is an explicit path configured for that `spoke-sdp-fec`.
- Otherwise it will look up the TAIL prefix into the pseudowire routing table, where static routes take precedence over BGP routes.

An aggregation scheme, similar to that used for classless IPv4 addresses, can be employed in the pseudowire routing table, where a longest match is used to find a route. Except for the default pseudowire route, which is encoded with a 0 mask, masks included in the `pw-routing` table are:

- /64 for regular prefixes, including a `global-id` and `prefix` (as previously mentioned, note that the `AC-ID` is not included in the BGP NLRI).
- /96 for local prefixes, including the `AC-ID`, as well as `global-id` and `prefix`.

Each S-PE and T-PE must have a pseudowire routing table that contains a reference to the TLDP session to use to signal to a set of next hop S-PEs to reach a given T-PE (or the T-PE if that is the next hop). For VLLs, this table contains aggregated AII Type 2 FECs and may be populated with routes that are learned through MP-BGP or that are statically configured.

## Explicit Paths

A set of default explicit routes to a remote T-PE prefix may be configured on a T-PE under **config>services>pw-routing** using the path name command. Explicit paths are used to populate the explicit route TLV used by MS-PW TLDP signaling. Only strict (fully qualified) explicit paths are supported. Note that it is possible to configure explicit paths independently of the configuration of BGP or static routing.

The following CLI excerpt shows an explicit path example for a MS-PW following the PE-1-PE-3-PE-5-PE-2 path (see the diagram in [Figure 72](#)). The IP addresses are the system addresses of all the S-PE and T-PE along the path (except for PE-1).

```
A:PE-1>config>service>pw-routing>path# info
-----
      hop 1 192.0.2.3
      hop 2 192.0.2.5
      hop 3 192.0.2.2
      no shutdown
-----
```

---

## Static Routes

In addition to support for BGP routing, static MS-PW routes may also be configured using the **config>services>pw-routing>static-route** command. Each static route comprises of the target T-PE Global-ID and prefix, and the IP address of the TLDP session to the next hop S-PE or T-PE that should be used:

```
A:PE-1# configure service pw-routing static-route
- no static-route <route-name>
- static-route <route-name>

<route-name>      : <global-id>:<prefix>:<next-hop-ip_addr>
global-id         - 0..4294967295
prefix            - a.b.c.d|0..4294967295
ip_addr           - a.b.c.d
```

If a static route **<global-id>:<prefix>** is set to 0, then this represents the default route.

```
A:PE-1>config>service>pw-routing# info
-----
...
      static-route 0:0.0.0.0:192.0.2.3
      static-route 0:0.0.0.0:192.0.2.4
...
-----
```

Note that, even though you can configure several default-routes, only one default route is added to the pseudowire routing table. The following command shows the pseudowire routing table content where only one default route (out of the two previously configured ones) is added. The default route added to the pseudowire routing table is the first valid route added to the configuration.

```
A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len          Next-Hop      Owner  Age
Route-Distinguisher          Community     Best
-----
0:0.0.0.0:0/0                192.0.2.3    static 19h11m57s
0:0                          0:0          yes
...
```

If a static route exists to a given T-PE, then this is used in preference to any BGP route that may exist.

## BGP Routes

As already mentioned, the dynamic advertisement of the pseudowire routes is enabled for each prefix and RD using the `advertise-bgp` command in the `config>services>pw-routing>local-prefix` context. Note that a BGP export policy is also required in order to export MS-PW routes in MP-BGP. This can be done using a default policy matching all the MS-PW routes, such as the following:

```
A:PE-1>config>router>policy-options# info
-----
      policy-statement "export_ms-pw"
        entry 10
          from
            family ms-pw
          exit
          action accept
          exit
        exit
      exit
A:PE-1>config>router>bgp# info
-----
      group "region"
        family ms-pw
        type internal
        export "export_ms-pw"
        neighbor 192.0.2.3
        exit
        neighbor 192.0.2.4
        exit
      exit
      no shutdown
-----
```

MS-PW routes advertised/received can be debugged and shown on the log sessions (**debug router bgp update**). Note that a new address family and NLRI are used to distribute the MS-PW prefixes:

```
49 2004/11/03 01:21:23.41 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.4
"Peer 1: 192.0.2.4: UPDATE
Peer 1: 192.0.2.4 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 51
  Flag: 0x40 Type: 1 Len: 1 Origin: 2
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    65535:12
  Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
    Address Family MSPW
    NextHop len 4 NextHop 192.0.2.1
    [MSPW] rd: 65536:12, global-id 65536, prefix 192.0.2.1,ac-id 0, preflen 128"
```

MS-PW BGP routes can also be displayed in the pseudowire routing table along with the static-routes and the local-prefixes.

```
*A:PE-1# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community
-----
0:0.0.0.0:0/0            192.0.2.3    static 19h32m08s
0:0                      0:0          yes
65536:192.0.2.1:0/64    192.0.2.1    local  01d17h56m
0:0                      0:0          yes
65536:192.0.2.1:0/64    192.0.2.1    local  01d17h56m
65536:11                 65535:11     yes
65536:192.0.2.1:0/64    192.0.2.1    local  01d17h56m
65536:12                 65535:12     yes
65536:192.0.2.2:0/64    192.0.2.4    bgp    01d15h11m
65536:22                 65535:12     yes
65536:192.0.2.11:0/64   192.0.2.1    local  01d13h02m
0:0                      0:0          yes
65536:192.0.2.11:0/64   192.0.2.1    local  01d13h02m
65536:11                 65535:11     yes
65536:192.0.2.12:0/64   192.0.2.1    local  01d12h57m
0:0                      0:0          yes
65536:192.0.2.12:0/64   192.0.2.1    local  01d12h56m
65536:12                 65535:12     yes
65536:192.0.2.13:0/64   192.0.2.1    local  19h39m31s
0:0                      0:0          yes
65536:192.0.2.14:0/64   192.0.2.1    local  19h39m13s
0:0                      0:0          yes
65536:192.0.2.21:0/64   192.0.2.3    bgp    01d12h59m
65536:21                 65535:11     yes
65536:192.0.2.22:0/64   192.0.2.4    bgp    01d12h56m
65536:22                 65535:12     yes
65536:192.0.2.23:0/64   192.0.2.3    static 19h06m21s
0:0                      0:0          yes
65536:192.0.2.24:0/64   192.0.2.4    static 19h06m13s
0:0                      0:0          yes
-----
Entries found: 15
=====
```

It is important to note that if there are two (or more) equal cost BGP MS-PW routes with identical <global-ID:prefix> and different RDs in the RIB they are both tagged as best/used and both will be added to the pseudowire routing table, however only the one with a higher RD will be shown as “Best” and as a result of that, only that one will be used by TLDP for the NSH. The following CLI output shows an example of two equal cost MS-PW routes. The route 65536:192.0.2.2 with RD 65536:21 and RD 65536:22 are tagged as best/used (u\*>):

```
*A:PE-1# show router bgp routes ms-pw aii-type2 65536:192.0.2.2:0
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
```

## BGP Routes

```
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

### BGP MSPW Routes

```
=====
Flag   Network                RD
      Nexthop             AII-Type2/Preflen
      As-Path
-----
u*>?  65536:192.0.2.2        65536:21
      192.0.2.3          65536:192.0.2.2:0/64
      No As-Path
*?    65536:192.0.2.2        65536:21
      192.0.2.4          65536:192.0.2.2:0/64
      No As-Path
u*>?  65536:192.0.2.2        65536:22
      192.0.2.4          65536:192.0.2.2:0/64
      No As-Path
*?    65536:192.0.2.2        65536:22
      192.0.2.3          65536:192.0.2.2:0/64
      No As-Path
-----
Routes : 4
=====
```

However, only the one with RD 65536:22 (higher RD) is added as “Best” to the pseudowire routing table and TLDP will use 192.0.2.4 as the NSH:

```
*A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
...
65536:192.0.2.2:0/64      192.0.2.3     bgp    01d15h21m
65536:21                  65535:11      no
65536:192.0.2.2:0/64      192.0.2.4     bgp    01d15h21m
65536:22                  65535:12      yes
...
=====
```

How does the 7x50 SR/ESS TLDP process select the NSH (Next-Signaling Hop) for two identical <global-ID:prefix/RD> tuples?

In case the originating T-PE or any intermediate S-PE receives two (or more) equal cost MS-PW routes with the same RD but from different Next-Hops, all the MS-PW routes will be added to the MS-PW routing table. The following output shows two MS-PW routes with the same <global-ID:prefix/RD> but different NH. Both are added to the MS-PW routing table as “Best”.

```
*A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
```



AII-Type2/Prefix-Len Route-Distinguisher	Next-Hop Community	Owner Best	Age
...			
65536:192.0.2.2:0/64 65536:21	192.0.2.3 65535:21	bgp yes	01d15h21m
65536:192.0.2.2:0/64 65536:21	192.0.2.4 65535:21	bgp yes	01d15h21m
...			

If that is the case, TLDP will pick up the NSH out of an ECMP hashing algorithm applied to the <global-ID:prefix:AC-ID> for the SAI and the TAI of the pseudowires pointing at the same prefix. The output of that hashing algorithm will determine what the NSH will be for a given spoke-sdp-fec.

When path diversity for an active and a standby pseudowire (hot standby pseudowire redundancy) is desired and the two pseudowires of the same Epipe end-point are pointing at the same remote <global-ID:prefix> coming from two different NHs, the operator has to make sure TLDP chooses a different NSH for the standby pseudowire. Only in that case hot standby pseudowire redundancy can be achieved. As a rule of thumb, if the SAI/TAI of the active and standby pseudowires are separated by 16 or more AC-ID values, TLDP will select a different NSH for both pseudowires.

For example:

- Given the following SAI/TAI AC-ID values for the active/standby pseudowires on the originating T-PE, TLDP will select the same NSH:
  - Active pseudowire: saii-type2 — 65536:192.0.2.1:1, taii-type2 — 65536:192.0.2.2:1
  - Standby pseudowire: saii-type2 — 65536:192.0.2.1:2, taii-type2 — 65536:192.0.2.2:2
- However, the following SAI/TAI AC-ID values for the active/standby pseudowires on the originating T-PE will allow the ECMP hashing algorithm to make TLDP select different NSHs for the active and the standby pseudowires:
  - Active pseudowire: saii-type2 — 65536:192.0.2.1:1, taii-type2 — 65536:192.0.2.2:1
  - Standby pseudowire: saii-type2 — 65536:192.0.2.1:16, taii-type2 — 65536:192.0.2.2:16

Other AC-ID values greater than 16 (for the standby pseudowire) would also have achieved next hop diversity.

## Configuring Dynamic Pseudowires on the T-PEs

Before any LDP signaling can take place, note that T-LDP sessions must be explicitly configured on T-PEs and S-PEs.

One or more spoke-SDPs may be configured for distributed Epipe VLL services. Dynamic MS-PWs use FEC129 (also known as the Generalized ID FEC) with Attachment Individual Identifier (AII) Type 2 to identify the pseudowire, as opposed to FEC128 (also known as the PW ID FEC) used for traditional single segment pseudowires and for pseudowire switching. FEC129 spoke-SDPs are configured under the `spoke-sdp-fec` command in the CLI. Note that spoke-sdp-fecs (or FEC129 spoke-SDPs) are by default fec-type 129 and aii-type 2 (those are the only values supported for spoke-sdp-fecs in the release 10.0R4). Spoke-sdp-fecs can be part of an endpoint and even an ICB (Inter-Chassis Backup) pseudowire.

```
*A:PE-1>config>service>epipe# spoke-sdp-fec
- no spoke-sdp-fec <spoke-sdp-fec-id>
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create]
- spoke-sdp-fec <spoke-sdp-fec-id> no-endpoint
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create] end-
point <name> [icb]

<spoke-sdp-fec-id> : [1..4294967295]
<fec-type>         : [129..130]
<aii-type>         : [1..2]
<name>             : [32 chars max]
<icb>              : keyword - configure spoke-sdp as inter-chassis backup
```

FEC129 AII Type 2 uses a Source Attachment Individual Identifier (SAII) and a Target Attachment Individual Identifier (TAII) to identify the ends of a pseudowire at the T-PE. The SAII identifies the local end, while the TAIID identifies the remote end. The SAII and TAIID are each structured as follows:

- **Global-ID:** this is a 4 byte identifier that uniquely identifies an operator or the local network. Normally this matches the ASN
- **Prefix:** a 4-byte prefix, which should correspond to one of the local prefixes assigned under pw-routing
- **AC-ID:** a 4-byte identifier for this end of the pseudowire. This should be locally unique within the scope of the global-id:prefix

In terms of the SDP tunnel being used by each spoke-sdp-fec, pw-routing chooses the MS-PW path in terms of the sequence of S-PEs to use to reach a given T-PE. It does not select the SDP to use on each hop, which is instead determined at signaling time. When a label mapping is sent for a given pseudowire segment, an LDP SDP will be used to reach the next-hop S-PE/T-PE if such an SDP exists. If not, and an RFC 3107 labeled BGP SDP is available, then that will be used<sup>2</sup>. Otherwise, the label mapping will fail and a label release will be sent.

The following CLI output shows one example of two spoke-sdp-fecs belonging to an endpoint:

```
*A:PE-1>config>service>epipe# info
-----
      description "ms-pw epipe with bgp - using 2 prefixes"
      endpoint "CORE" create
        description "end-point for epipe A/S PW redundancy"
        revert-time 10
        standby-signaling-master
      exit
      sap ccag-1.b:2 create
      exit
      spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
        precedence primary
        pw-template-bind 1
        saii-type2 65536:192.0.2.11:1
        taii-type2 65536:192.0.2.21:1
        no shutdown
      exit
      spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
        pw-template-bind 1
        saii-type2 65536:192.0.2.12:1
        taii-type2 65536:192.0.2.22:1
        no shutdown
      exit
      no shutdown
-----
```

These are all of the options available under the spoke-sdp-fec context:

```
*A:PE-1# configure service epipe 1 spoke-sdp-fec
- no spoke-sdp-fec <spoke-sdp-fec-id>
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create]
- spoke-sdp-fec <spoke-sdp-fec-id> no-endpoint
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create] end-
point <name> [icb]

<spoke-sdp-fec-id> : [1..4294967295]
<fec-type>        : [129..130]
<aii-type>       : [1..2]
<name>           : [32 chars max]
<icb>           : keyword - configure spoke-sdp as inter-chassis backup
```

- 
- Note that RSVP SDPs might be picked at the T-PE through the use of pw-template <policy-id> [use-provisioned-sdp], however there is no way to select an RSVP SDP on an S-PE.

## Active/Passive Signaling and Auto-Configuration

```
[no] auto-config      - Configure auto-configuration
[no] path             - Configure path-name
[no] precedence       - Configure precedence
[no] pw-template-bi*  - Configure Pseudo-Wire template-binding policy
[no] retry-count      - Configure retry count
[no] retry-timer      - Configure retry timer
[no] sai-type2        - Configure Source Attachment Individual Identifier (SAII)
[no] shutdown         - Administratively enable/disable the spoke SDP FEC binding
                        - Configure Spoke-SDP FEC signaling
[no] standby-signal*  - Enable PW standby-signaling slave
[no] taii-type2       - Configure Target Attachment Individual Identifier (TAII)
```

---

## Active/Passive Signaling and Auto-Configuration

When an MS-PW is signaled, each T-PE might independently initiate signaling of the MS-PW. This could result in a different path being used in each direction of the pseudowire. To avoid this situation one of the T-PE will start the pseudowire signaling (active role), while the other T-PE waits to receive the LDP label mapping message before sending the LDP label mapping message for the reverse direction of the pseudowire (passive role).

By default, the T-PE with SAII>TAII will have the active role and will send the label mapping first:

```
*A:PE2>config>service>epipe# spoke-sdp-fec 21 shutdown
*A:PE2>config>service>epipe# info
-----
      spoke-sdp-fec 21 fec 129 aii-type 2 create
          sai-type2 65536:192.0.2.21:1
          taii-type2 65536:192.0.2.11:1
      exit
      no shutdown
-----
*A:PE2>config>service>epipe# spoke-sdp-fec 21 no shutdown
49 2011/07/28 02:50:59.88 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Label Mapping packet (msgId 26) to 192.0.2.3:0
50 2011/07/28 02:50:59.97 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Recv Label Mapping packet (msgId 30) from 192.0.2.3:0
```

This default behavior can be modified by the signaling command. When set to master, the T-PE will send a label mapping message regardless of the SAII and TAII. By default the parameter is set to auto (which means the T-PE will trigger label mapping if SAII>TAII).

```
*A:PE-1# configure service epipe 1 spoke-sdp-fec 21 signaling
- signaling <signaling>

<signaling>          : auto|master
```

The MS-PW routing implementation on the 7x50 supports single-sided auto-provisioning. This allows it to have “hub” T-PEs where the TAII is not required to be configured and as such simplifies the provisioning. In this case, the spoke T-PE PWs would be configured with specific SAII and TAII as well as signaling master, whereas the hub T-PE PWs would be configured with only the SAII and the auto-config parameter. When the auto-config attribute is set for a spoke-sdp-fec, the T-PE always passively waits for the label mapping to be received before issuing a label mapping message (since it does not know the TAII beforehand). This is a CLI example for a hub T-PE spoke-sdp-fec:

```
*A:PE2>config>service>epipe# info
-----
      spoke-sdp-fec 21 fec 129 aii-type 2 create
        auto-config
          saii-type2 65536:192.0.2.21:1
        exit
      no shutdown
-----
```

## Spoke-SDP-FEC Timers

MS-PW routing provides a few timers that can be configured at the global pw-routing level or at each specific spoke-sdp-fec level:

```
*A:PE1>config>service>pw-routing# info
-----
boot-timer 20
retry-timer 40
retry-count 50

*A:PE1>config>service>epipe# info
-----
spoke-sdp-fec 3 fec 129 aii-type 2 create
retry-timer 10
retry-count 10
```

Where:

- **Boot-timer** (the default is 10 seconds with values 0 — 600 seconds allowed): Configures a hold-off timer for MS-PW routing advertisements and signaling that is used at boot time. This timer helps to make sure all the network infrastructure is up and running before setting up the PWs.
- **Retry-timer** (the default is 30 seconds with values 10 — 480 seconds allowed): The exponential back-off timer that determines the interval between consecutive retries to re-establish a spoke-SDP. The configured value gives the initial retry time. The attempt fails if a label withdrawal is received. If configured at global and spoke-sdp-fec level, the latter overrides the value set by the global settings.
- **Retry-count** (the default 30 with values 10 — 10000): Specifies the number of attempts the system should make to re-establish the spoke-SDP after it has failed. After each successful attempt, the counter is reset to zero. When the specified number is reached, no more attempts are made and the spoke-sdp is put into the shutdown state. Use the **no shutdown** command to bring up the path after the retry limit is exceeded. It is present at the pw-routing level as well as the spoke-SDP level. If configured at global and spoke-sdp-fec level, the latter overrides the value set by the global settings.
- The usual endpoint level timers are also available for MS-PW routing:
  - **Revert-time** <time-value|infinite> (default is 0, values 0-600 sec): configures the time to wait before reverting to the primary spoke-sdp-fec.
  - **Active-hold-delay** (the default is 0, values 0 — 60 deci-seconds): It specifies that the node will delay sending the T-LDP status bits for VLL endpoint when the MC-LAG transitions the LAG subgroup which hosts the SAP from active to standby (MC-Ring or MC-APS are supported too) or when any object in the endpoint, i.e. SAP, ICB, or regular spoke SDP, transitions from up to down operational state. The active-hold-delay range starts from 1 (in units of deci-sec) via CLI, and the only way to get the default value of zero is to use the **no active-hold-delay** command

## Standby Signaling

Just as with a regular endpoint with regular spoke-sdps, there can also be standby-signaling-master and standby-signaling-slave parameters for spoke-sdp-fecs.

The standby-signaling-master command is configured under the end-point context and makes sure that standby signaling (TLDP pseudowire status bits 0x20) is sent for the selected standby pseudowire. It cannot be set if SAPs have been configured at the end-point (for MC-LAG/Ring/APS or ICB).

```
*A:PE1>config>service>epipe>endpoint# info
-----
standby-signaling-master

*A:PE1>config>service>epipe>endpoint# standby-signaling-master
MINOR: SVCMGR #3805 The command is not allowed in an endpoint with sap
```

The standby-signaling-slave can be configured at endpoint or spoke-sdp-fec level (if the spoke-sdp-fec is not part of an endpoint) but never on both at the same time:

```
*A:PE1>config>service>epipe>endpoint# info
-----
standby-signaling-slave

*A:PE1>config>service>epipe>spoke-sdp-fec# standby-signaling-slave
MINOR: SVCMGR #2031 Sdp-bind is in an explicit endpoint

*A:PE1>config>service>epipe# info
-----
sap 1/1/3:3 create
exit
spoke-sdp-fec 11 fec 129 aii-type 2 create
standby-signaling-slave
```

When this parameter is configured, the node will block the transmit forwarding direction of a spoke SDP based on the pseudowire standby bit received from a TLDP peer.

## Spoke-SDP-FEC Templates and Filters

PW-templates are the way to configure the control word for this type of pseudowire as well as ingress/egress filters (ipv4/mac/ipv6). In regards to filters, it is important to note that they are only supported on the T-PEs, since there is no provisioning of a pw-template (or Epipe at all) on the S-PEs.

```
*A:PE1# configure service pw-template 1
*A:PE1>config>service>pw-template# info
-----
controlword
  egress
    filter ip 1
  exit
*A:PE1>config>service>epipe# info
-----
--snip--
      spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
      pw-template-bind 1
```

Note that pw-template changes (just like for VPLS with BGP-AD or BGP-VPLS) are not automatically propagated. A tools perform command is provided to evaluate and distribute the changes at the service level to one or all the services that use that template (if the service ID is omitted, then all the services will be updated).

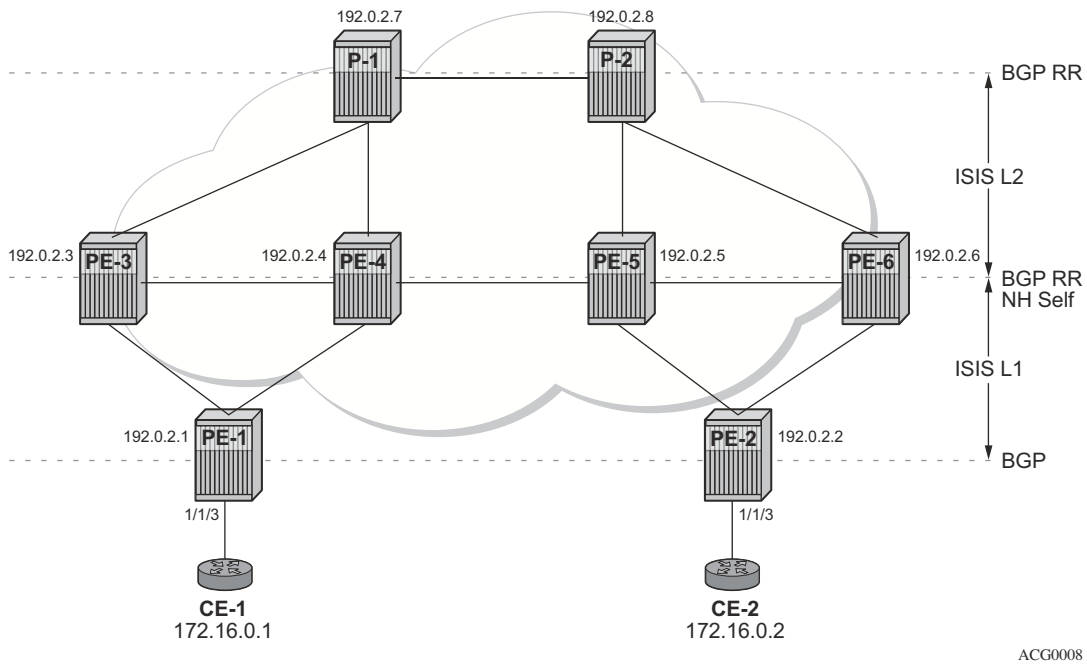
```
*A:PE1# tools perform service id 6 eval-pw-template 1 allow-service-impact
```

---

## Intra-AS MS-PW Routing

This section provides a configuration example for an intra-AS scenario. The following network setup will be used for this section.





**Figure 72: Intra-AS MS-PW Network Topology**

Multiple MS-PW routing Epipes are configured between PE-1 and PE-2, with PE-3, PE-4, PE-5 and PE-6 being S-PE routers. P-1 and P-2 will be pure P routers from a data plane perspective.

All the PEs are pre-configured with ISIS as the IGP, as shown in the figure: PE-1 and PE-2 are level-1 routers, P-1 and P-2 are level-2 only routers and the rest of the routers are level-1/level-2. Link level LDP is also pre-configured on all the network interfaces and targeted LDP is configured between PE-1 and PE-3/PE-4, between PE-2 and PE-5/PE-6 and among PE-3, PE-4, PE-5 and PE-6. There is no targeted LDP sessions configured on P-1 and P-2.

As outlined in [Figure 71](#), the configuration is a three-step process where the pw-routing context is configured first, then the required configuration so that routing tables get populated accordingly and finally the services themselves.

## MS-PW using BGP Routing

In this sub-section, Epipe 2 will be configured between PE-1 and PE-2, where TLDP will use the BGP routes populated in the MS-PW routing table to signal the MS-PW.

The first step is the provisioning of the pw-routing context on all the T-PEs and S-PEs. The spe-address will be configured on all the T-PEs and S-PEs (that is, all the routers except for P-1 and P-2) using the ASN as the global-id and the system address as the prefix. On PE-1 and PE-2 (only) we will configure the prefixes used for setting up Epipe 2. Note that two prefixes are configured per T-PE so that pseudowire redundancy with path diversity for the standby pseudowire can be carried out. The spe-address and local-prefixes for the T-PEs are shown below. Note that the advertise-bgp parameter is required since we are using BGP here.

```
*A:PE-1>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.1
local-prefix 65536:192.0.2.11 create
    advertise-bgp route-distinguisher 65536:11 community 65535:11
exit
local-prefix 65536:192.0.2.12 create
    advertise-bgp route-distinguisher 65536:12 community 65535:12
exit

*A:PE-2>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.2
local-prefix 65536:192.0.2.21 create
    advertise-bgp route-distinguisher 65536:21 community 65535:11
exit
local-prefix 65536:192.0.2.22 create
    advertise-bgp route-distinguisher 65536:22 community 65535:12
exit
```

The second step is the configuration of BGP.

As depicted in [Figure 72](#), BGP is enabled in all the routers. Note that the middle routers (PE-3, PE-4 and PE-5, PE-6) are BGP route-reflectors for PE-1 and PE-2 and they reflect MS-PW routes while changing the next-hop to their own system address. This is required so that TLDP knows where to send the label mapping message for a particular prefix. P-1 and P-2 are regular RRs reflecting routes among all the S-PEs. The BGP configuration of PE-1, PE-3, PE-4 and a P-1 is shown below. Similar commands are configured on the other PEs depending on their T-PE, S-PE or RR function.

The T-PEs have dual-homed BGP sessions to the S-PEs. Example for PE-1:

```
*A:PE-1>config>router>bgp# info
-----
    enable-peer-tracking
    rapid-withdrawal
    group "region"
      family ms-pw
      type internal
      export "export_ms-pw"
      neighbor 192.0.2.3
      exit
      neighbor 192.0.2.4
      exit
    exit
    no shutdown
-----
*A:PE-1>config>router>bgp# show router policy "export_ms-pw"
  entry 10
    from
      family ms-pw
    exit
    action accept
    exit
  exit
```

The S-PEs are reflecting routes and also changing the NH and Local Preference based on the communities accordingly, so that pseudowire diversity can be ensured.

```
A:PE-3>config>router>bgp# info
-----
    rapid-withdrawal
    group "core"
      family ms-pw
      type internal
      export "export_ms-pw_ABR-to-core"
      neighbor 192.0.2.7
      exit
      neighbor 192.0.2.8
      exit
    exit
    group "region"
      family ms-pw
      type internal
  enable-peer-tracking
    cluster 3.3.3.3
    export "export_ms-pw_ABR-to-region"
    neighbor 192.0.2.1
    exit
  exit
  no shutdown
-----
A:PE-3>config>router>bgp# show router policy "export_ms-pw_ABR-to-core"
  entry 10
    from
      protocol bgp
      community "65535:11"
```

## MS-PW using BGP Routing

```
        family ms-pw
    exit
    action accept
        local-preference 150
        next-hop-self
    exit
exit
entry 20
    from
        protocol bgp
        community "65535:12"
        family ms-pw
    exit
    action accept
        local-preference 100
        next-hop-self
    exit
exit
A:PE-3>config>router>bgp# show router policy "export_ms-pw_ABR-to-region"
entry 10
    from
        protocol bgp
        community "65535:11"
        family ms-pw
    exit
    action accept
        local-preference 150
        next-hop-self
    exit
exit
entry 20
    from
        protocol bgp
        community "65535:12"
        family ms-pw
    exit
    action accept
        local-preference 100
        next-hop-self
    exit
exit
```

The second S-PE to which PE-1 is connected has the following BGP configuration:

```
A:PE-4>config>router>bgp# info
-----
    enable-peer-tracking
    rapid-withdrawal
    group "core"
        family ms-pw
        type internal
        export "export_ms-pw_ABR-to-core"
        neighbor 192.0.2.7
        exit
        neighbor 192.0.2.8
        exit
    exit
    group "region"
```

```

        family ms-pw
        type internal
    enable-peer-tracking
        cluster 4.4.4.4
        export "export_ms-pw_ABR-to-region"
        neighbor 192.0.2.1
    exit
    exit
    no shutdown
-----
A:PE-4>config>router>bgp# show router policy "export_ms-pw_ABR-to-core"
  entry 10
    from
      protocol bgp
      community "65535:12"
      family ms-pw
    exit
    action accept
      local-preference 150
      next-hop-self
    exit
  exit
  entry 20
    from
      protocol bgp
      community "65535:11"
      family ms-pw
    exit
    action accept
      local-preference 100
      next-hop-self
    exit
  exit
A:PE-4>config>router>bgp# show router policy "export_ms-pw_ABR-to-region"
  entry 10
    from
      protocol bgp
      community "65535:12"
      family ms-pw
    exit
    action accept
      local-preference 150
      next-hop-self
    exit
  exit
  entry 20
    from
      protocol bgp
      community "65535:11"
      family ms-pw
    exit
    action accept
      local-preference 100
      next-hop-self
    exit
  exit

```

Finally this is the BGP configuration for P-1, a pure RR.

```
A:P-1>config>router>bgp# info
-----
enable-peer-tracking
rapid-withdrawal
group "core"
  family ms-pw
  type internal
  cluster 1.1.1.1
  neighbor 192.0.2.3
  exit
  neighbor 192.0.2.4
  exit
  neighbor 192.0.2.5
  exit
  neighbor 192.0.2.6
  exit
exit
no shutdown
-----
```

After BGP is properly configured and the BGP update exchange takes place, the RIBs are properly populated and the required prefixes uploaded into the MS-PW routing table. An example for PE-1's RIB and pseudowire routing table is provided below.

```
*A:PE-1# show router bgp routes ms-pw
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MSPW Routes
=====
Flag  Network          RD
      Nexthop       AII-Type2/Preflen
      As-Path
-----
u*>?  65536:192.0.2.21  65536:21
      192.0.2.3      65536:192.0.2.21:0/64
      No As-Path
*?    65536:192.0.2.21  65536:21
      192.0.2.4      65536:192.0.2.21:0/64
      No As-Path
u*>?  65536:192.0.2.22  65536:22
      192.0.2.4      65536:192.0.2.22:0/64
      No As-Path
*?    65536:192.0.2.22  65536:22
      192.0.2.3      65536:192.0.2.22:0/64
      No As-Path
□
-----
Routes : 16
=====
*A:PE-1# show service pw-routing route-table
```

```

=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len           Next-Hop      Owner  Age
Route-Distinguisher           Community     Best
-----
65536:192.0.2.11:0/64         192.0.2.1    local  02d13h58m
65536:11                       65535:11     yes
65536:192.0.2.12:0/64         192.0.2.1    local  02d13h53m
65536:12                       65535:12     yes
65536:192.0.2.21:0/64         192.0.2.3    bgp    02d13h56m
65536:21                       65535:11     yes
65536:192.0.2.22:0/64         192.0.2.4    bgp    02d13h53m
65536:22                       65535:12     yes
-----
Entries found: 15
=====

```

It is important to note that the two prefixes advertised by PE-2 are properly learned by PE-1 through two different next hops. Now, use each one with a different pseudowire and make sure that the active and standby pseudowires follow different paths in the network.

Once the routes are installed in the MS-PW routing table, configure the services on PE-1 and PE-2.

```

*A:PE-1# configure service epipe 2
*A:PE-1>config>service>epipe# info
-----
description "ms-pw epipe with bgp - using 2 prefixes"
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
  standby-signaling-master
exit
sap 1/1/3:2 create
exit
spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  sai-type2 65536:192.0.2.11:1
  taii-type2 65536:192.0.2.21:1
  no shutdown
exit
spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
  pw-template-bind 1
  sai-type2 65536:192.0.2.12:1
  taii-type2 65536:192.0.2.22:1
  no shutdown
exit
no shutdown

*A:PE-2# configure service epipe 2
*A:PE-2>config>service>epipe# info
-----
description "ms-pw epipe with bgp - using 2 prefixes"

```

## MS-PW using BGP Routing

```
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
exit
sap 1/1/3:2 create
exit
spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  saii-type2 65536:192.0.2.21:1
  taii-type2 65536:192.0.2.11:1
  no shutdown
exit
spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
  pw-template-bind 1
  saii-type2 65536:192.0.2.22:1
  taii-type2 65536:192.0.2.12:1
  no shutdown
exit
no shutdown
```

-----

The following command can be executed to check that the service and spoke-sdp-fecs are up:

```
*A:PE-1# show service id 2 base
=====
Service Basic Information
=====
Service Id       : 2                Vpn Id           : 0
Service Type     : Epipe
Name             : (Not Specified)
Description      : ms-pw epipe with bgp - using 2 prefixes
Customer Id      : 1
Last Status Change: 11/03/2004 06:13:10
Last Mgmt Change  : 11/03/2004 06:13:52
Admin State      : Up                Oper State        : Up
MTU              : 1514
Vc Switching     : False
SAP Count        : 1                SDP Bind Count    : 2
Per Svc Hashing  : Disabled
Force QTag Fwd   : Disabled
=====
Service Access & Destination Points
=====
Identifier                               Type           AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/3:2                              q-tag         1518    1518    Up   Up
sdp:17406:4294967287 SB(192.0.2.3)       MS-PW         0        1974    Up   Up
sdp:17407:4294967286 SB(192.0.2.4)       MS-PW         0        1974    Up   Up
=====
```

Note that the sdp-binding identifiers and sdp identifiers are automatically generated by the system.



Use vccv-trace to check that the spoke-sdp-fecs for the active and standby pseudowires follow different and disjoint paths:

```
*A:PE-1# oam vccv-trace spoke-sdp-fec 21

VCCV-TRACE with 120 bytes of MPLS payload
1 192.0.2.3 rtt=1.75ms rc=8(DSRtrMatchLabel)
2 192.0.2.5 rtt=5.38ms rc=8(DSRtrMatchLabel)
3 192.0.2.2 rtt=8.34ms rc=3(EgressRtr)

*A:PE-1# oam vccv-trace spoke-sdp-fec 22

VCCV-TRACE with 120 bytes of MPLS payload
1 192.0.2.4 rtt=1.80ms rc=8(DSRtrMatchLabel)
2 192.0.2.6 rtt=5.48ms rc=8(DSRtrMatchLabel)
3 192.0.2.2 rtt=7.83ms rc=3(EgressRtr)
```

## MS-PW using Static Routing

In this sub-section, Epipe 3 will be configured between PE-1 and PE-2, where TLDP will use static-routes in the MS-PW routing table to signal the MS-PW.

On PE-1 and PE-2 (only) we will configure the prefixes used for setting up Epipe 3. Those could be the same as used for Epipe 2, however we will use different ones in this example. Note that the **no advertise-bgp** parameter is required now. The static routes for each remote prefix are also configured. Since we will also have pseudowire redundancy for Epipe 3, two prefixes with static-routes pointing at different next-hops will be used:

```
*A:PE-1>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.1
local-prefix 65536:192.0.2.13 create
exit
local-prefix 65536:192.0.2.14 create
exit
static-route 65536:192.0.2.23:192.0.2.3
static-route 65536:192.0.2.24:192.0.2.4

*A:PE-2>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.2
local-prefix 65536:192.0.2.23 create
exit
local-prefix 65536:192.0.2.24 create
exit
static-route 65536:192.0.2.13:192.0.2.5
static-route 65536:192.0.2.14:192.0.2.6
```

It is important to note that static-routes are also required at all S-PEs along the path (keeping the path diversity for the prefixes as well) and for both directions:

```
A:PE-3>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.3
static-route 65536:192.0.2.13:192.0.2.1
static-route 65536:192.0.2.23:192.0.2.5
-----

A:PE-4>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.4
static-route 65536:192.0.2.14:192.0.2.1
static-route 65536:192.0.2.24:192.0.2.6
-----
```

Finally, once the MS-PW routing tables are properly populated, the services can be configured and brought up:

```
*A:PE-1>config>service>epipe# info
-----
description "ms-pw epipe with static routes"
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
  standby-signaling-master
exit
sap 1/1/3:3 create
exit
spoke-sdp-fec 31 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  saii-type2 65536:192.0.2.13:31
  taii-type2 65536:192.0.2.23:31
  no shutdown
exit
spoke-sdp-fec 32 fec 129 aii-type 2 create endpoint CORE
  pw-template-bind 1
  saii-type2 65536:192.0.2.14:32
  taii-type2 65536:192.0.2.24:32
  no shutdown
exit
no shutdown
-----

*A:PE-2>config>service>epipe# info
-----
description "ms-pw epipe with bgp - using 2 prefixes"
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
  standby-signaling-master
exit
sap 1/1/3:2 create
exit
spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  saii-type2 65536:192.0.2.21:1
  taii-type2 65536:192.0.2.11:1
  no shutdown
exit
spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
  pw-template-bind 1
  saii-type2 65536:192.0.2.22:1
  taii-type2 65536:192.0.2.12:1
  no shutdown
exit
no shutdown
```

Check the status and path of the spoke-sdp-fecs with the proper show commands and oam vccv-trace/ping commands (see previous sub-section).

## MS-PW using Explicit Paths

In this sub-section, Epipe 4 will be configured between PE-1 and PE-2, where TLDP will use explicit paths to signal the MS-PW, overriding the information given by the MS-PW routing table. Although this mode requires the specific configuration of the hops, one by one, the configuration is only done on the T-PEs, as opposed to the static-routes where all the S-PEs must be configured with static routes (a mixed of static-routes and BGP routes can coexist). The local-prefixes shown for Epipe 3 will be re-used here for Epipe 4.

Now path-1 and path-2 will be configured hop by hop, using diverse paths. Note that all the S-PE nodes as well as the terminating T-PE must be included in the path.

```
*A:PE-1>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.1

local-prefix 65536:192.0.2.13 create
exit
local-prefix 65536:192.0.2.14 create
exit

path "path-1" create
  hop 1 192.0.2.3
  hop 2 192.0.2.5
  hop 3 192.0.2.2
  no shutdown
exit
path "path-2" create
  hop 1 192.0.2.4
  hop 2 192.0.2.6
  hop 3 192.0.2.2
  no shutdown
exit
-----
*A:PE-2>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.2

local-prefix 65536:192.0.2.23 create
exit
local-prefix 65536:192.0.2.24 create
exit

path "path-1" create
  hop 1 192.0.2.5
  hop 2 192.0.2.3
  hop 3 192.0.2.1
  no shutdown
exit
path "path-2" create
  hop 1 192.0.2.6
  hop 2 192.0.2.4
  hop 3 192.0.2.1
  no shutdown
exit
```

And now, those paths must be specified when configuring the Epipe:

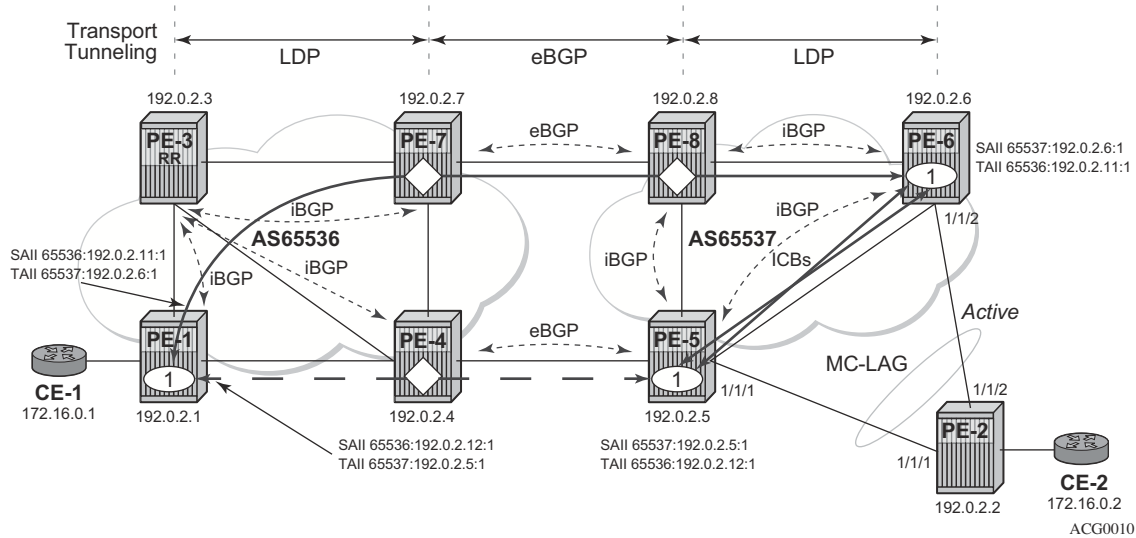
```
*A:PE-1>config>service>epipe# info
-----
description "ms-pw epipe with explicit paths"
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
  standby-signaling-master
exit
sap 1/1/3:4 create
exit
spoke-sdp-fec 41 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  sai-type2 65536:192.0.2.13:41
  tai-type2 65536:192.0.2.23:41
  path "path-1"
  no shutdown
exit
spoke-sdp-fec 42 fec 129 aii-type 2 create endpoint CORE
  sai-type2 65536:192.0.2.14:42
  tai-type2 65536:192.0.2.24:42
  path "path-2"
  no shutdown
exit
no shutdown
-----

*A:PE-2>config>service>epipe# info
-----
description "ms-pw epipe with explicit paths"
endpoint "CORE" create
  description "end-point for epipe A/S PW redundancy"
  revert-time 10
exit
sap 1/1/3:4 create
exit
spoke-sdp-fec 41 fec 129 aii-type 2 create endpoint CORE
  precedence primary
  pw-template-bind 1
  sai-type2 65536:192.0.2.23:41
  tai-type2 65536:192.0.2.13:41
  path "path-1"
  no shutdown
exit
spoke-sdp-fec 42 fec 129 aii-type 2 create endpoint CORE
  sai-type2 65536:192.0.2.24:42
  tai-type2 65536:192.0.2.14:42
  path "path-2"
  no shutdown
exit
no shutdown
-----
```

Now, check the status and path of the spoke-sdp-fecs with the proper **show** commands and **oam vccv-trace/ping** commands (see previous sub-section).

## Inter-AS MS-PW Routing

This section provides a configuration example for an inter-AS scenario, using BGP tunnels between ASBRs and BGP as the MS-PW routing mechanism. The following network setup will be used in this section.



**Figure 73: Inter-AS MS-PW Network Topology**

In this example, only one Epipe is configured (Epipe 1, using MS-PW BGP routing). The T-PEs are PE-1, PE-5 and PE-6. PE-7, PE-8 and PE-4 are S-PEs.

A/S pseudowire redundancy together with MC-LAG at one end will be used, as depicted in the diagram. ICB (Inter-chassis Backup) spoke SDPs between PE-5 and PE-6 are required in order to forward the in-flight packets while MC-LAG and A/S pseudowire are converging, in case of network failures. Those ICBs will also be signalled following the MS-PW routing procedures.

The network setup in the figure is pre-configured with the following settings:

- There are two AS (65536 and 65537) which are connected by two ASBR pairs (PE-7/PE-4 and PE-8/PE-5) running eBGP between them. These eBGP sessions will be used to exchange ipv4-label (to setup the transport BGP-LBL tunnel, according to the RFC 3107) and MS-PW NLRIs.
- Within AS65536, PE-3 is used as an RR to reflect the MS-PW routes. In AS65537 there is a full mesh of iBGP sessions to distribute the MS-PW routes.
- ISIS is used within each AS.

- LDP is used as a transport MPLS signaling protocol within each AS and a BGP tunnel will be used between the ASBRs (note that MS-PW routing supports LDP or BGP tunnels as transport).
- A redundant MC-LAG access to PE-6 and PE-5 is configured.

The next section will go through the configuration required to set up a redundant Epipe between CE-1 and CE-2, by combining A/S pseudowire in the network and MC-LAG at the access.

---

## MS-PW using BGP Routing

Epipe 1 will be configured at the end of this section, including the active and redundant pseudowires from PE-1 to PE-5/PE-6, as well as the required ICBs and SAPs at the access.

As discussed, the first step is the provisioning of the pw-routing context. Again, the spe-address must be provisioned in all T-PEs and S-PEs whereas prefixes are mandatory only on the T-PEs involved in the service. The following CLI output shows the prefixes configured on PE-1, PE-5 and PE-6. Note that two prefixes are needed in PE-1 in order to make sure that active and standby pseudowires follow disjoint paths.

```
*A:PE-1>config>service>pw-routing# info
-----
spe-address 65536:192.0.2.1
local-prefix 65536:192.0.2.11 create
  advertise-bgp route-distinguisher 65536:11 community 65535:11
exit
local-prefix 65536:192.0.2.12 create
  advertise-bgp route-distinguisher 65536:12 community 65535:12
exit
exit

*A:PE-5>config>service>pw-routing# info
-----
spe-address 65537:192.0.2.5
local-prefix 65537:192.0.2.5 create
  advertise-bgp route-distinguisher 65537:5 community 65535:5
exit

*A:PE-6>config>service>pw-routing# info
-----
spe-address 65537:192.0.2.6
local-prefix 65537:192.0.2.6 create
  advertise-bgp route-distinguisher 65537:6 community 65535:6
exit
```

Once the spe-addresses and prefixes have been provisioned, BGP must be configured accordingly. An example of the configuration at PE-1 and PE-6 is shown below. Note that a simple BGP export-policy is used to export all the local MS-PW prefixes.

```

#-----
# PE-1 BGP related configuration
#-----

#-----
echo "Policy Configuration"
#-----
    policy-options
    begin
    policy-statement "export_ms-pw"
    entry 10
    from
    family ms-pw
    exit
    action accept
    exit
    exit
    exit
    commit
exit
#-----
echo "BGP Configuration"
#-----
    bgp
    min-route-advertisement 1
    rapid-withdrawal
    group "intra-AS"
    family ms-pw
    type internal
    export "export_ms-pw"
    neighbor 192.0.2.3
    exit
    exit
    no shutdown
    exit
exit
#-----

# PE-6 BGP related configuration
#-----
#-----
echo "Policy Configuration"
#-----
    policy-options
    begin
    policy-statement "export_ms-pw"
    entry 10
    from
    family ms-pw
    exit
    action accept
    exit
    exit
exit

```



```

        exit
        commit
    exit
#-----
echo "BGP Configuration"
#-----
    bgp
        min-route-advertisement 1
        enable-peer-tracking
        rapid-withdrawal
        group "intra-AS"
            family ms-pw
            type internal
            export "export_ms-pw"
            neighbor 192.0.2.5
            exit
            neighbor 192.0.2.8
            exit
        exit
        no shutdown
    exit
exit

```

At the ASBR, the BGP policies are a little more complex since the following tasks must be accomplished:

- ASBR ipv4 system addresses must be exported to the peer ASBR to establish the RFC 3107 BGP tunnel between ASBRs.
- BGP export policies must be used so that MS-PW NLRI exchange can be controlled and attributes like MED (towards the remote AS) and/or local-preference (towards the local AS) can be modified.
- Finally, BGP import policies must also be used to modify the MS-PW route NH (next-hops) since the TLDP next signaling hop must match a peer TLDP system address

As an example of ASBR BGP configuration, PE-4 and PE-7 are shown below.

Note that the prefixes 65536:192.0.2.11 and 65537:192.0.2.6 must be preferred in the PE-7/PE-8 pair whereas the prefixes 65536:192.0.2.12 and 65537:192.0.2.5 must be preferred in the PE-4/PE-5 pair, so that the pseudowires are established as depicted in [Figure 73](#). The preference can be propagated by using the BGP MED (use the local preference (LP) within the AS (LP is not relevant to eBGP)). The following CLI excerpt shows an example of how to modify MED and LP, as well as changing the NH with an import policy.

```

#-----
# PE-4 BGP related configuration
#-----

#-----
echo "Policy Configuration"
#-----
    policy-options
        begin

```

## MS-PW using BGP Routing

```
prefix-list "system"
  prefix 192.0.2.4/32 exact
exit
community "65535:5" members "65535:5"
community "65535:6" members "65535:6"
community "65535:11" members "65535:11"
community "65535:12" members "65535:12"
policy-statement "ASBR to ASBR"
  entry 10
    from
      protocol bgp
      community "65535:12"
      family ms-pw
    exit
    action accept
      origin igp
      metric set 50
    exit
  exit
entry 20
  from
    protocol bgp
    community "65535:11"
    family ms-pw
  exit
  action accept
    origin igp
    metric set 100
  exit
exit
policy-statement "ASBR to region"
  entry 10
    from
      protocol bgp
      community "65535:5"
      family ms-pw
    exit
    action accept
      origin igp
      local-preference 150
      next-hop-self
    exit
  exit
entry 20
  from
    protocol bgp
    community "65535:6"
    family ms-pw
  exit
  action accept
    origin igp
    next-hop-self
  exit
exit
policy-statement "export_ipv4_system"
  entry 10
    from
```

```

        prefix-list "system"
        exit
        action accept
        origin igp
        exit
    exit
exit
policy-statement "import ms-pw NH change"
    entry 10
        from
            protocol bgp
            family ms-pw
        exit
        action accept
        next-hop 192.0.2.5
        exit
    exit
exit
commit
exit
#-----
echo "BGP Configuration"
#-----
    bgp
        min-route-advertisement 1
        enable-peer-tracking
        rapid-withdrawal
        group "inter-AS"
            family ipv4 ms-pw
            type external
            import "import ms-pw NH change"
            export "export_ipv4_system" "ASBR to ASBR"
            local-as 65536
            peer-as 65537
            neighbor 192.168.45.2
                advertise-label ipv4
            exit
        exit
        group "intra-AS"
            family ms-pw
            type internal
            export "ASBR to region"
            neighbor 192.0.2.3
            exit
        exit
    no shutdown
exit
exit
#-----
# PE-7 BGP related configuration
#-----
#-----
echo "Policy Configuration"
#-----
    policy-options
        begin
            prefix-list "system"

```

## MS-PW using BGP Routing

```
    prefix 192.0.2.7/32 exact
  exit
  community "65535:5" members "65535:5"
  community "65535:6" members "65535:6"
  community "65535:11" members "65535:11"
  community "65535:12" members "65535:12"
  policy-statement "ASBR to ASBR"
    entry 10
      from
        protocol bgp
        community "65535:11"
        family ms-pw
      exit
      action accept
        origin igp
        metric set 50
      exit
    exit
  entry 20
    from
      protocol bgp
      community "65535:12"
      family ms-pw
    exit
    action accept
      origin igp
      metric set 100
    exit
  exit
  policy-statement "ASBR to region"
    entry 10
      from
        protocol bgp
        community "65535:6"
        family ms-pw
      exit
      action accept
        origin igp
        local-preference 150
        next-hop-self
      exit
    exit
  entry 20
    from
      protocol bgp
      community "65535:5"
      family ms-pw
    exit
    action accept
      origin igp
      next-hop-self
    exit
  exit
  policy-statement "export_ipv4_system"
    entry 10
      from
        prefix-list "system"
```

```

        exit
        action accept
            origin igp
        exit
    exit
exit
policy-statement "import ms-pw NH change"
    entry 10
        from
            protocol bgp
            family ms-pw
        exit
        action accept
            next-hop 192.0.2.8
        exit
    exit
exit
commit
exit
#-----
echo "BGP Configuration"
#-----
    bgp
        min-route-advertisement 1
        enable-peer-tracking
        rapid-withdrawal
        transport-tunnel mpls
        group "inter-AS"
            family ipv4 ms-pw
            type external
            import "import ms-pw NH change"
            export "export_ipv4_system" "ASBR to ASBR"
            local-as 65536
            peer-as 65537
            neighbor 192.168.78.2
                advertise-label ipv4
            exit
        exit
        group "intra-AS"
            family ms-pw
            type internal
            export "ASBR to region"
            neighbor 192.0.2.3
            exit
        exit
    no shutdown
exit
exit

```

PE-5 and PE-8 have similar configurations to the one shown above. Note that PE-5 is a T-PE as well as an ASBR, therefore a local MS-PW prefix must be exported as opposed to only remote prefixes (that is, some export entries for the local MS-PW routes will not contain **protocol bgp** in the matching criteria).

## MS-PW using BGP Routing

After BGP is properly configured and the updates get exchanged, the RIBs are populated and the prefixes uploaded onto the MS-PW routing table as shown below for PE-1 and PE-6.

```
*A:PE-1# show router bgp routes ms-pw
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
BGP MSPW Routes
=====
Flag  Network                RD
      Nexthop             AII-Type2/Preflen
      As-Path
-----
u*>i  65537:192.0.2.5        65537:5
      192.0.2.4          65537:192.0.2.5:0/64
      65537
u*>i  65537:192.0.2.6        65537:6
      192.0.2.7          65537:192.0.2.6:0/64
      65537
-----
Routes : 4
```

```
*A:PE-1# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
65536:192.0.2.11:0/64    192.0.2.1    local  30d22h25m
0:0                      0:0          yes
65536:192.0.2.11:0/64    192.0.2.1    local  30d22h24m
65536:11                  65535:11     yes
65536:192.0.2.12:0/64    192.0.2.1    local  30d22h19m
0:0                      0:0          yes
65536:192.0.2.12:0/64    192.0.2.1    local  30d22h19m
65536:12                  65535:12     yes
65537:192.0.2.5:0/64     192.0.2.4    bgp    02h43m25s
65537:5                   65535:5      yes
65537:192.0.2.6:0/64     192.0.2.7    bgp    02h45m49s
65537:6                   65535:6      yes
-----
Entries found: 6
```

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

```
*A:PE-6# show router bgp routes ms-pw
=====
BGP Router ID:192.0.2.6      AS:65537      Local AS:65537
=====
BGP MSPW Routes
=====
Flag  Network                RD
      Nexthop             AII-Type2/Preflen
-----
```

```

As-Path
-----
u*>i 65536:192.0.2.11      65536:11
      192.0.2.8           65536:192.0.2.11:0/64
      65536
u*>i 65536:192.0.2.12      65536:12
      192.0.2.5           65536:192.0.2.12:0/64
      65536
u*>i 65537:192.0.2.5       65537:5
      192.0.2.5           65537:192.0.2.5:0/64
      No As-Path
-----
Routes : 3
=====
*A:PE-6#

*A:PE-6# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
65536:192.0.2.11:0/64    192.0.2.8    bgp    02h52m08s
65536:11                 65535:11     yes
65536:192.0.2.12:0/64    192.0.2.5    bgp    02h38m51s
65536:12                 65535:12     yes
65537:192.0.2.5:0/64     192.0.2.5    bgp    02h38m51s
65537:5                  65535:5      yes
65537:192.0.2.6:0/64     192.0.2.6    local  28d02h16m
0:0                      0:0          yes
65537:192.0.2.6:0/64     192.0.2.6    local  28d02h16m
65537:6                  65535:6      yes
-----
Entries found: 5
=====
*A:PE-6#

```

As can be seen in the show commands, the two PE-1 prefixes are learned on PE-5 and PE-6 through different and disjoint paths, and the PE-5 and PE-6 prefixes are learned by PE-1 through two different and disjoint paths.

The last step is the service configuration on the three T-PEs, as shown below. Note that TLDP sessions must have been previously and explicitly configured between the T-PEs and S-PEs (i.e., between PE-1 and PE-4/7, between PE-4 and PE-5, PE-7 and PE-8 and between PE-6 and PE-5/8).

```

#-----
# PE-1 Service related configuration
#-----
#-----
echo "Service Configuration"
#-----
service
  customer 1 create

```

## MS-PW using BGP Routing

```
        description "Default customer"
    exit

pw-template 1 create
    controlword
exit

epipe 1 customer 1 create
    description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
    endpoint "CORE" create
        description "end-point for epipe A/S PW redundancy"
    exit
    sap 1/1/1:1 create
    exit
    spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
        precedence primary
        pw-template-bind 1
        saii-type2 65536:192.0.2.11:1
        taii-type2 65537:192.0.2.6:1
        no shutdown
    exit
    spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE
        pw-template-bind 1
        saii-type2 65536:192.0.2.12:1
        taii-type2 65537:192.0.2.5:1
        no shutdown
    exit
    no shutdown
    exit
exit

#-----
# PE-5 Service related configuration
#-----
#-----
echo "Service Configuration"
#-----

service
    customer 1 create
        description "Default customer"
    exit

pw-template 1 create
    controlword
exit
epipe 1 customer 1 create
    description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
    endpoint "CORE" create
        description "end-point for epipe A/S PW redundancy"
    exit
    endpoint "ACCESS" create
    exit
    sap lag-1:1 endpoint "ACCESS" create
    exit
    spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
        pw-template-bind 1
        saii-type2 65537:192.0.2.5:1
        taii-type2 65536:192.0.2.12:1
        no shutdown
    exit
    spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE icb
```



```

        pw-template-bind 1
        sai-type2 65537:192.0.2.5:2
        taii-type2 65537:192.0.2.6:2
        no shutdown
    exit
    spoke-sdp-fec 13 fec 129 aii-type 2 create endpoint ACCESS icb
        pw-template-bind 1
        sai-type2 65537:192.0.2.5:3
        taii-type2 65537:192.0.2.6:3
        no shutdown
    exit
    no shutdown
exit
exit
#-----
# PE-6 Service related configuration
#-----
#-----
echo "Service Configuration"
#-----
service
    customer 1 create
        description "Default customer"
    exit

    pw-template 1 create
        controlword
    exit
    epipe 1 customer 1 create
        description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
        endpoint "CORE" create
            description "end-point for epipe A/S PW redundancy"
        exit
        endpoint "ACCESS" create
        exit
        sap lag-1:1 endpoint "ACCESS" create
        exit
    spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
        pw-template-bind 1
        sai-type2 65537:192.0.2.6:1
        taii-type2 65536:192.0.2.11:1
        no shutdown
    exit
    spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE icb
        pw-template-bind 1
        sai-type2 65537:192.0.2.6:3
        taii-type2 65537:192.0.2.5:3
        no shutdown
    exit
    spoke-sdp-fec 13 fec 129 aii-type 2 create endpoint ACCESS icb
        pw-template-bind 1
        sai-type2 65537:192.0.2.6:2
        taii-type2 65537:192.0.2.5:2
        no shutdown
    exit
    no shutdown
exit
exit
exit

```

The following show commands can be executed to check the status of the Epipe 1 and the pseudowire status signaling received:

```
*A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id      : 1                Vpn Id          : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : ms-pw epipe with bgp, inter-AS, MC-LAG redundancy
Customer Id     : 1
Last Status Change: 12/04/2004 03:01:49
Last Mgmt Change  : 12/04/2004 03:42:43
Admin State     : Up                Oper State      : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                SDP Bind Count  : 2
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled
-----
Service Access & Destination Points
-----
Identifier                               Type           AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/1:1                               q-tag          1518    1518    Up   Up
sdp:17405:4294967279 SB(192.0.2.7)  MS-PW         0       1974    Up   Up
sdp:17406:4294967280 SB(192.0.2.4)  MS-PW         0       1974    Up   Up

*A:PE-1# show service id 1 endpoint
=====
Service 1 endpoints
=====
Endpoint name      : CORE
Description        : end-point for epipe A/S PW redundancy
Revert time       : 0
Act Hold Delay    : 0
Standby Signaling Master : false
Standby Signaling Slave  : false
Tx Active (SDP-FEC)  : 11
Tx Active Up Time   : 2d 23:17:51
Revert Time Count Down : N/A
Tx Active Change Count : 4
Last Tx Active Change : 12/04/2004 03:42:43
-----
Members
-----
Sdp-fec: 11 Prec:0                Oper Status: Up
Sdp-fec: 12 Prec:4                Oper Status: Up
=====
```

Note that PE-5 will have the MC-LAG standby interface and as such the SAP will be operationally down and will drive the standby signaling to the remote T-PEs:

```
*A:PE-5# show service id 1 base
=====
Service Basic Information
=====
Service Id       : 1                Vpn Id           : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : ms-pw epipe with bgp, inter-AS, MC-LAG redundancy
Customer Id     : 1
Last Status Change: 11/18/2011 17:12:04
Last Mgmt Change  : 11/18/2011 17:13:11
Admin State     : Up                Oper State       : Up
MTU             : 1514
Vc Switching   : False
SAP Count       : 1                SDP Bind Count   : 3
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled
-----
Service Access & Destination Points
-----
Identifier                               Type           AdmMTU  OprMTU  Adm  Opr
-----
sap:lag-1:1                               q-tag          1518    1518    Up   Down
sdp:17402:4294967245 SB(192.0.2.6) MS-PW         0        1974    Up   Up
sdp:17402:4294967246 SB(192.0.2.6) MS-PW         0        1974    Up   Up
sdp:17403:4294967247 SB(192.0.2.4) MS-PW         0        1974    Up   Up
=====
*A:PE-5# show service id 1 all | match Flag
Flags           : None
Flags           : None
Flags           : None
Flags           : PortOperDown StandByForMcProtocol
```

The following commands are useful on the S-PEs in order to find the PWs automatically created as well as the SDPs automatically used for those PWs.

```
*A:PE-7# show service sdp-using
=====
SDP Using
=====
SvcId      SdpId                Type   Far End      Opr S* I.Label  E.Label
-----
2147483647 17406:4294967294    MS-PW  192.0.2.1    Up    131064 131065
2147483647 17407:4294967295    MS-PW  192.0.2.8    Up    131065 131068
-----
Number of SDPs : 2
=====
```

As it can be seen above, two PWs (type MS-PW) have been automatically created over two also automatically created SDPs: 17406 and 17407. SDP 17406 is built over an LDP tunnel whereas SDP 17407 runs over a BGP tunnel.

## MS-PW using BGP Routing

```
*A:PE-7# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner Encap TunnelId  Pref   Nexthop      Metric
-----
192.0.2.1/32     sdp   MPLS   17406    5      192.0.2.1    0
192.0.2.1/32     ldp   MPLS   -        9      192.168.47.1 20
192.0.2.3/32     ldp   MPLS   -        9      192.168.37.1 10
192.0.2.4/32     ldp   MPLS   -        9      192.168.47.1 10
192.0.2.8/32     sdp   MPLS   17407    5      192.0.2.8    0
192.0.2.8/32     bgp   MPLS   -        10     192.168.78.2 1000

*A:PE-7# show service sdp 17406 detail | match "Active LSP"
Mixed LSP Mode      : Enabled                Active LSP Type      : LDP

*A:PE-7# show service sdp 17407 detail | match "Active LSP"
Mixed LSP Mode      : Enabled                Active LSP Type      : BGP
```

In addition to all of the recommended show commands, **vccv-ping** and **vccv-trace** are two extremely useful commands in this environment. **vccv-trace** can even help to trace the traffic going through the ICBs under failure situations.

## Conclusion

Service Providers are always seeking highly scalable VLL services that can be deployed with the lowest operational cost. The SR OS supports MS-PW routing according to the draft-ietf-pwe3-dynamic-ms-pw. MS-PW routing allows the Service Provider to deploy ELINE services without having to provision services in the core of the network. In other words, MS-PW enables end-point provisioning in highly scalable seamless MPLS networks, through the use of BGP. Alternatively, static MS-PW routes or explicit paths can also be used.

The examples used in this section illustrate the configuration of MS-PW routing in intra-AS and inter-AS scenarios. Show and OAM commands have also been suggested so that the operator can verify and troubleshoot the MS-PW routing paths and procedures.

Conclusion