Establishing MPLS LSPs Across Multiple Autonomous Systems for Next-Gen Multicast VPNs

Configuration Example

October 2016
# Table of Contents

1. What's Inside This Document? ................................................................. 5
2. Test Setup .................................................................................................. 7
   2.1 Test Topology .................................................................................. 7
3. Inter-Domain RSVP-TE Tunnels: Overview and Configuration ............. 8
   3.1 The Workflow ................................................................................ 8
   3.2 MPLS Inter-Domain ....................................................................... 9
   3.3 MPLS Loose Hop .......................................................................... 9
   3.4 Inter-AS Link ............................................................................... 11
   3.5 IBGP Configuration ....................................................................... 12
   3.6 EBGP on the Inter-AS Links .......................................................... 12
   3.7 BGP for PE-PE Connection ............................................................ 13
   3.8 PE Configuration Example ............................................................. 13
   3.9 ASBR Configuration Example ....................................................... 15
4. Inter-AS LSPs with expand-loose-hop: Verification and Analysis .......... 17
   4.1 Follow the Journey: From Beginning to End .................................... 17
   4.2 A Closer Look at the PATH Messages ........................................... 21
   4.3 OSPF Really Matters .................................................................... 22
5. Inter-AS LSPs with BGP-LS: Overview and Configuration ..................... 24
   5.1 The Ingredients ............................................................................. 24
   5.2 The Main Concept ......................................................................... 24
   5.3 BGP-LS ....................................................................................... 25
   5.4 TE Import/Export Routes ............................................................... 25
   5.5 RSVP and MPLS Interface .............................................................. 26
   5.6 Inter-AS Link ............................................................................... 27
   5.7 EBGP Multihop ............................................................................ 27
   5.8 PE Configuration Example ............................................................. 27
   5.9 ASBR Configuration Example ....................................................... 28
6. Inter-AS LSPs with BGP-LS: Verification and Analysis ......................... 30
   6.1 Scanning the TED ......................................................................... 30
   6.2 Verifying the LSP .......................................................................... 32
7. Node-Link Protection ............................................................................ 33
   7.1 BGP-TE ...................................................................................... 33
8. NG MVPN: Overview and Configuration ................................................. 38
   8.1 The PIM Protocol ......................................................................... 38
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.2</td>
<td>P2MP LSP</td>
<td>39</td>
</tr>
<tr>
<td>8.3</td>
<td>MVPN Signaling</td>
<td>40</td>
</tr>
<tr>
<td>8.4</td>
<td>CE Configuration</td>
<td>42</td>
</tr>
<tr>
<td>8.5</td>
<td>PE Configuration</td>
<td>42</td>
</tr>
<tr>
<td>9</td>
<td>NG MVPN: Monitoring and Verification</td>
<td>45</td>
</tr>
<tr>
<td>9.1</td>
<td>Autodiscovery</td>
<td>45</td>
</tr>
<tr>
<td>9.2</td>
<td>Join Operations at the CE</td>
<td>47</td>
</tr>
<tr>
<td>9.3</td>
<td>Join Operations at the PE</td>
<td>48</td>
</tr>
<tr>
<td>9.4</td>
<td>From to Shared Tree to Source Tree</td>
<td>49</td>
</tr>
<tr>
<td>9.5</td>
<td>Multicast Traffic Flow</td>
<td>52</td>
</tr>
<tr>
<td>10</td>
<td>Conclusion</td>
<td>53</td>
</tr>
</tbody>
</table>
1 What’s Inside This Document?

![Diagram of End-to-End Services with More Than One AS]

Figure 1. End-to-End Services with More Than One AS

When understanding how to deploy services like a Layer 3 VPN, we often consider a single service provider whose devices reside in a single IGP domain or autonomous system (AS). As a large network continues to grow, in order to sustain the increased scale requirements, it might become necessary to partition the network into multiple domains, as shown in Figure 1. This is especially important when low-end devices with limited scalability are deployed. The division can be based on different interior gateway protocol (IGP) areas, different ASs, or a combination of the two. From the MPLS perspective, multi-domain transport architectures bring additional challenges. For instance, a protocol such as RSVP uses either the IGP database or the traffic engineering database (TED), if traffic engineering (RSVP-TE) is enabled, to find out how to reach the tunnel endpoint. Unfortunately, in this case, the tunnel endpoint is in another AS, meaning that RSVP doesn't have all the required information to create the tunnel.

To provide end-to-end services (e.g., L3VPN) across multiple domains, you can consider two high-level approaches: segmented tunnels and non-segmented tunnels.

**Segmented Tunnels**

Transport LSPs span only a local domain, and service-aware “stitching” takes place at the domain boundaries (Area Border Router [ABR], Autonomous System Border Router [ASBR]). This model is conceptually aligned to inter-AS Option B, and in it, next-hop attributes for the service BGP routes typically are changed at the domain boundaries. This approach is not covered in this document.

**Non-Segmented Tunnels**

End-to-end transport label-switched paths (LSPs) are established across domains, and service provisioning is only deployed at the LSP endpoints. There are two flavors:

- **Hierarchical inter-domain tunnels**

  This model is conceptually aligned to inter-AS Option C, and makes use of BGP label unicast. The next-hop attributes for the service BGP routes are not changed at the boundaries.
Flat inter-domain tunnels

This model relies on two main approaches:

*Inter-domain*: Provision this command as part of the MPLS LSP configuration to enable the ingress router to locate the tunnel endpoint, in another domain, by using the IGP database (not the TED) or a BGP route.

*BGP-LS*: A new BGP address family, called BGP-LS (link state) or BGP-TE (traffic engineering) is used to propagate the traffic engineering database, or TED, across different domains.

In this document we will cover the non-segmented flat tunnels approach to establish end-to-end RSVP-TE tunnels across different domains. Because we are using flat tunnels, BGP-LU is not necessary (but can be used). These configurations use the simple BGP unicast family throughout.
2 Test Setup

2.1 Test Topology

Figure 2: The Test Topology

Before going on, it is important to understand some conventions used for addressing, as shown in Figure 2:

- For Loopback, the schema is 192.168.<2-digit AS ending>.dev#, where AS is the AS value and dev# is the device number (for example, for MX240-7, dev# is 7, giving 192.168.36.7 for the IP address).

- Intra-AS links have the schema 10.<2-digit AS ending>.x.y/30, where x and y indicate the devices (for example, 10.37.21.23/30).

- Inter-AS links have the schema 172.16.<2-digit AS ending>.x/30, where the 2-digit AS ending indicates the last two digits of the destination ASN (for example, 172.16.37.1/30).
3 Inter-Domain RSVP-TE Tunnels: Overview and Configuration

As a delicious dish, even here, we need some “ingredients” to establish our RSVP-TE tunnel across multiple autonomous systems.

In this case, we need to configure the following:

- On the ingress provider edge (PE) router, enable the **Inter-domain** command on the MPLS LSP.
- On the ASBRs and ABRs, enable the MPLS loose-hop expansion (**expand-loose-hop**).
- On the inter-AS links between the ASBRs, enable IGP traffic engineering.

3.1 The Workflow

Traffic engineering (TE) information, which is distributed by protocols such as OSPF, IS-IS, and BGP-LS, is collected in a single database, called TED. CSPF actually uses TED as input when performing TE path calculations.

The ingress PE router queries the TED database to establish the RSVP-TE tunnels and compute the Explicit Route Object (ERO). If the endpoint is in another AS, the ingress PE will not find it in the TED, and the tunnel will not be established.

This problem is resolved by instructing the ingress PE to send the RSVP Path message to its closest ABR or ASBR in the same domain; in particular, the ingress PE will set the ERO object so that it can reach the ABR or ASBR.

Once the RSVP Path message reaches the ASBR, this router that has the inter-AS physical link and remote ASBR router-id information in its TED, will compute the ERO to the ASBR in the other AS and send the Path outside the local AS.

Following this method, the RSVP Path message will continue travelling across multiple ASs. Transit routers will always try to determine if the final destination is in the local AS first; if not, they will compute the path until the ASBR will again be able to forward the message to the next AS.

Finally, the Path message will reach the AS where the remote PE resides and, through a canonical CSPF computation, it will reach the Egress router. Resv messages will simply follow the Path message in the backward direction.

The procedure described above is possible by using some tweaks and extensions of well-known protocols such as OSPF, BGP, and RSVP. (We’ll cover these “tricks” in a bit more in detail later.) For example, it’s possible to modify the default behavior of RSVP so that intermediate routers, like an ASBR, are able to modify the ERO and expand an LSP across multiple ASs.
3.2 MPLS Inter-Domain

We have highlighted that one of the biggest challenges is the creation of an RSVP-TE tunnel across multiple ASs. When the egress tunnel endpoint is in another AS, the ingress PE will not find it in the local TED and the tunnel cannot be established.

This challenge is resolved by adding the `inter-domain` command to the MPLS LSP on the ingress PE, as shown in Figure 3. This technique allows the ingress router to resolve the egress point of the MPLS LSP by searching for routes in the IGP database or searching for BGP routes. You need to configure this statement on routers that might be unable to locate a path using intra-domain CSPF (by looking in the traffic-engineering database). In the inter-AS case, where the MPLS tunnel endpoint is present as a BGP route in the ingress PE router routing table, the ingress router will be able to look for the tunnel endpoint as a BGP route. If the egress PE is in another AS, the CSPF algorithm is unable to compute a route toward the egress PE but can only compute a route toward the local ASBR router. The ingress router will only include a loose hop to the ASBR in the initial ERO.

This way, the RSVP ERO is not complete at first, since it will not include the egress tunnel endpoint; however, it includes the path to reach the local ASBR. The ERO will be updated step-by-step until the AS where the egress router is located is reached.

To configure `inter-domain`, simply apply it when configuring the LSP:

```
user@mx80-PE# show protocols
mpls { 
  label-switched-path Inter-AS-LSP { 
    from 192.168.36.23; 
    to 192.168.38.24; 
    inter-domain; 
  } 
}
```

3.3 MPLS Loose Hop
We've just said that, when it is not possible to compute a path to the egress using the TED, a loose hop to the ASBR is added to the ERO—but how can the Path message span multiple ASs? Figure 4 shows how this task is accomplished thanks to the expand-loose-hop option. It gives a transit router the ability to add information and modify the existing ERO of a transit RSVP Path message. When an ASBR configured with expand-loose-hop finds its router-id, as loose next hop, in the transit RSVP Path message, it will search in its TED to compute a path toward the ASBR in the remote AS. The local ASBR will expand the ERO in the Path message by including the IP address of the physical inter-AS link on the other ASBR, and the remote ASBR router-id as the loose next hop. This option is mandatory on all the ABR/ASBR routers; otherwise it will be impossible to cross the AS border and different IGP areas.

Configuring it is easy:

```
user@mx80-ASBR4# show protocols
mpls {
    expand-loose-hop;
}
```
3.4 Inter-AS Link

To establish an RSVP-TE tunnel across multiple ASs, the ASBRs need to retain their TED information about the inter-AS link; in particular, the address on the remote interface must be made available inside the AS. This information is not normally included either in EBGP reachability messages or in OSPF routing advertisements.

To flood this link address information within the AS and make it available for traffic engineering calculations, you must configure OSPF passive mode for traffic engineering on each inter-AS interface. You must also supply the remote address for OSPF to distribute and include in the traffic engineering database.

In practice, as shown in Figure 5, we have to configure OSPF passive mode for traffic engineering on every inter-AS link. This will trigger the flooding, within the area, of the inter-AS link local (and remote) address (subnet). If the area is area 0, an LSA Type 2 (router) will be used. This entry will be installed into the TED, allowing the ASBR to expand the establishment of the RSVP LSP over multiple ASs. Moreover, OSPF passive mode for traffic engineering allows MPLS LSP to dynamically discover OSPF ASBR routers.

Configuration requires the addition of a line under the `[protocols ospf]` hierarchy:

```
user@mx80-ASBR# show protocols
ospf {
  traffic-engineering;
  area 0.0.0.0 {
    interface lo0.0 {
      passive;
    }
    interface xe-2/1/0.0;
    interface xe-2/0/1.0;
    interface xe-2/0/0.0 {
      passive {
        traffic-engineering {
          remote-node-id 172.16.37.2;
        }
      }
    }
  }
```

Figure 5: Tunnel with Inter-AS Link
Please note that it is important to configure the physical interface address of the remote ASBR node. Don’t use the loopback address, otherwise the TED will contain inconsistent information and it will be impossible to correctly forward packets. We will see this better later on.

### 3.5 IBGP Configuration

The ASBRs are IBGP IP unicast (AFI=1, SAFI=1) route reflectors for the PE routers within AS 65536 and AS 65538. The following is the configuration for the ASBR1 in AS 65536:

```bash
user@asbr1# show protocols bgp group IBGP-DOWN
    type internal;
    local-address 192.168.36.5;
    family inet { unicast;
    } export PL-IBGP-DOWN-EXP;
    cluster 192.168.36.5;
    neighbor 192.168.36.21;
    neighbor 192.168.36.22;
```

The policy `PL-IBGP-DOWN-EXP` controls the PE loopbacks from the remote AS 65538 that are exported to the local PE inside AS 65536:

```bash
[edit policy-options]
user@asbr1# show
    policy-statement PL-IBGP-DOWN-EXP {
        term REMOTE-LOOPBACKS {
            from { protocol bgp;
                community CM-REMOTE-LOOPBACKS;
            }
            then { next-hop self;
                accept;
            }
        }
        [other terms]
    }
    community CM-REMOTE-LOOPBACKS members 38:1000;
```

It is fundamental that the PE loopback addresses are exchanged via BGP between the remote PE in different ASs.

### 3.6 EBGP on the Inter-AS Links

EBGP IP unicast (AFI=1, SAFI=1) is used to peer different ASBRs in neighboring ASs. The peering is done using the inter-AS link addresses:

```bash
[edit protocols bgp]
user@asbr1# show
    group EBGP {
        type external;
        family inet { unicast;
        } export PL-EBGP-EXP;
        peer-as 65537;
        neighbor 172.16.37.2;
    }
```

The policy `PL-EBGP-EXP` is used to advertise the local PE router loopback addresses to the remote ASs:
3.7 BGP for PE-PE Connection

Once the remote PE loopback addresses are advertised between different ASs, the PE routers can establish a multihop EBGP IP VPN unicast (AFI=1, SAFI=128) session, such as between PE2 in AS 65536 and PE3 in AS 65538:

```plaintext
user@PE2# show protocols
group EBGP-MHOP-VPN {
    type external;
    multihop {
        ttl 10;
        local-address 192.168.36.23;
        family inet-vpn {
            unicast;
            peer-as 65538;
            neighbor 192.168.38.24;
        }
    }
}
```

As you can see, the multihop parameter is used to indicate that this neighbor relationship will be established between two peers that are not directly connected. Moreover, we can also specify the maximum number of hops we intend to allow. In this example, the two peers can’t be more than 10 hops away.

3.8 PE Configuration Example

Let’s give a look at a possible PE configuration, highlighting the most important aspects that allow the creation of an inter-AS LSP, as shown in Figure 6.
Figure 6: Inter-AS LSP Considerations

RSVP configuration is pretty standard:

```bash
rsvp {
  interface xe-0/0/0.0 {
    interface xe-0/0/2.0 {
      interface lo0.0;
    }
  }
}
```

For MPLS, we define the LSP as inter-domain, and we provide possible paths:

```bash
mpls {
  label-switched-path Inter-AS-65536→65538 {
    from 192.168.36.23;
    to 192.168.38.24;
    node-link-protection;
    inter-domain;
  }
  interface xe-0/0/0.0;
  interface xe-0/0/2.0;
  interface lo0.0;
}
```

Internal BGP:

```bash
group IBGP {
  type internal;
  local-address 192.168.36.23;
  family inet {
    unicast;
  }
  neighbor 192.168.36.21;
  neighbor 192.168.36.7;
  neighbor 192.168.36.5;
```
While EBGP for PE-PE connectivity has to support multihop:

```c
while EBGP-MHOP-VPN {
    type external;
    multihop {
        ttl 10;
    }
    local-address 192.168.36.23;
    family inet-vpn {
        unicast;
    }
    peer-as 65538;
    neighbor 192.168.38.24
}
```

The OSPF configuration is fairly standard too, but remember to include support for traffic engineering:

```c
while OSPF {
    traffic-engineering;
    area 0.0.0.0 {
        interface lo0.0 {
            passive;
        }
        interface xe-0/0/0.0;
        interface xe-0/0/2.0;
    }
}
```

### 3.9 ASBR Configuration Example

Now, let's have a look at an ASBR, as shown in Figure 7.

![ASBR Links](image)

**Figure 7. ASBR Links**

RSVP is again configured without any unusual details. In this section, the RSVP configuration is omitted.

The MPLS configuration has to support `expand-loose-hop`:

```c
while MPLS {
    expand-loose-hop;
    interface xe-2/0/1.0;
}
```
interface xe-2/1/0.0;
interface xe-2/0/0.0;
interface xe-2/2/0.0;
}

The EBGP configuration on the ASBRs is shown below:

```
[edit protocols bgp]
user@asbr1# show

group EBGP {
    type external;
    family inet {
        unicast;
    }
    export PL-EBGP-EXP;
    peer-as 65537;
    neighbor 172.16.37.2;
}
```

IBGP is used for intra-AS communications. The following configuration can be deployed in the PE and ASBR routers:

```
user@asbr1# show protocols bgp group IBGP-DOWN

type internal;
local-address 192.168.36.5;
family inet {
    unicast;
}
export PL-IBGP-DOWN-EXP;
cluster 192.168.36.5;
neighbor 192.168.36.21;
neighbor 192.168.36.22;
```

OSPF requires a few additional lines because we have to take care of the inter-AS link:

```
ospf {
    traffic-engineering;
    area 0.0.0.0 {
        interface lo0.0 {
            passive;
        }
        interface xe-2/1/0.0;
        interface xe-2/0/1.0;
        interface xe-2/0/0.0 {
            passive {
                traffic-engineering {
                    remote-node-id 172.16.37.2;
                }
            }
        }
        interface xe-2/2/0.0 {
            passive {
                traffic-engineering {
                    remote-node-id 172.16.37.6;
                }
            }
        }
    }
}
```
4 Inter-AS LSPs with expand-loose-hop: Verification and Analysis

4.1 Follow the Journey: From Beginning to End

Now that we have enough background, it is time to follow a packet, step-by-step, from the ingress to the egress. At each step we will check that all the pieces of our puzzle are in the right place. In particular, we’re going to monitor:

- Presence of the LSP toward the egress
- RSVP sessions
- BGP route toward the egress
- TED entries to perform CSPF computations

The first step is going from the ingress to the ASBR within the AS, as shown in Figure 8.

Figure 8. From Ingress PE to ASBR in AS 65536

We will use different monitoring commands to understand how things actually work:

```
user@mx80-23# run show mpls lsp detail ingress
Ingress LSP: 1 sessions
192.168.38.24
  From: 192.168.36.23, State: Up, ActiveRoute: 0, LSPname: Inter-AS65538
  ActivePath: (primary)
  PathDomain: Inter-domain
...
  Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 1)
  10.36.2.2 S
  Received RRO: 10.36.2.2 172.16.37.1 10.37.1.1 172.16.38.1 10.38.2.1
Total 1 displayed, Up 1, Down 0
```

There are some important things to notice in this output:

- The `PathDomain` value is `Inter-domain`. This gives the ingress PE the ability to look for the egress router as a BGP route, once it realizes the egress is not in its TED.

- The computed ERO only contains a loose hop representing the ASBR. Up to now, the only thing that mattered for the ingress router was to reach the ASBR (then the ASBR took care of forwarding the packet toward the final destination).
• The path has been already established and we can see the full RRO. Use the
topology in Figure 2 as a reference. Here we can see packets will go through MX240-
7 (ASVR AS 65536) to MX40-1 (AS 65537) to MX240-8 (the ASBR in AS 65537) to
MX80-22 (AS 65538) and finally MX80-24 (egress in AS 65538).

Now we look at the RSVP session:

user@mx80-23> show rsvp session ingress detail
Ingress RSVP: 1 sessions
192.168.38.24
  From: 192.168.36.23, LSPstate: Up, ActiveRoute: 0
  LSPname: Inter-AS-65538, LSPpath: Primary
...
  Expct route: 10.36.2.2
  Record route: <self> 10.36.2.2 172.16.37.1 10.37.1.1 172.16.38.1 10.38.2.1
Total 1 displayed, Up 1, Down 0

Here we identify the ingress session for our LSP. The egress loopback is outside AS 65536,
so it must be resolved by a BGP route. As we’ve already seen, the ERO only contains the
information needed to reach the ASBR.

We can check if there is a BGP route for the egress:

user@mx80-23> show route 192.168.38.24/32 table inet.0
inet.0: 32 destinations, 42 routes (32 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
  192.168.38.24/32  *[BGP/170] 01:58:20, localpref 100, from 192.168.36.5
    AS path: 65537 65538 I, validation-state: unverified
to 10.36.0.2 via xe-0/0/0.0
> to 10.36.2.2 via xe-0/0/2.0

And finally, we check that an entry for that next-hop exists in the TED:

user@mx80-23# run show ted database 192.168.36.7
TED database: 0 ISIS nodes 12 INET nodes
ID Type Age(s) LnkIn LnkOut Protocol
192.168.36.7 Rtr 227 3 4 OSPF(0.0.0.0)
...
To: 10.36.2.1-1, Local: 10.36.2.2, Remote: 0.0.0.0
Local interface index: 0, Remote interface index: 0

You can see that an entry for address 10.36.2.2 exists for the ASBR (192.168.36.7), which
means that CSPF can compute a route to that address and the next-hop can be successfully
resolved.

For the following steps, we use the same commands, so the same considerations we’ve
previously made are still valid.

Now the packet is at the ASBR. This is where the expand-loose-hop is enabled, as shown
in Figure 9.
Figure 9. The ASBR and Expanding Loose Hop

Let's check the RSVP session on MX240-7 in AS 65536:

```
user@mx240-7-re1# run show rsvp session transit detail
Transit RSVP: 1 sessions
192.168.38.24
  Record route: 10.36.2.1 <self> 172.16.37.1 10.37.1.1 172.16.38.1 10.38.2.1
Total 1 displayed, Up 1, Down 0
```

This time the RSVP session is a transit session. We see the router is able to modify the ERO, which we can see when compared to the ERO in the output of the `show rsvp session ingress detail` command from MX80-23 above. The ERO now contains two elements: the directly connected next-hop and the loopback of the egress node (as a loose hop). The first address listed in the ERO (Explet route) is the inter-AS link resolved using the TED and the one obtained thanks to the OSPF-TE passive configuration.

As usual, we check for a BGP route and identify the next-hop:

```
user@mx240-7-re1# run show route 192.168.38.24/32 table inet.0
inet.0: 34 destinations, 49 routes (34 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.38.24/32 *(BGP/170) 02:58:59, localpref 100
  AS path: 65537 65538 I, validation-state: unverified
    > to 172.16.37.1 via xe-2/2/0.0
```

And finally, we look for the BGP next-hop in the TED:

```
user@mx240-7-re1# run show ted database 172.16.37.1
TED database: 0 ISIS nodes 12 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
172.16.37.1                   ---   11031     1      0

user@mx240-7-re1# run show ted database 192.168.36.7
TED database: 0 ISIS nodes 12 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.36.7                  Rtr    2773     3      4 OSPF(0.0.0.0)
  To: 172.16.37.1, Local: 172.16.37.2, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
```

Now we are in AS 65537, and the ASBR has to forward the packet to the other ASBR in the AS, as shown in Figure 10.

Figure 10. From One ASBR to the Other

Let's check RSVP sessions, BGP routes, and TED entries on MX40-1 in AS 65537:
The next step is to check ASBR MX240-8 and, again, make sure that all the data is available and correct, as shown in Figure 11.

**Figure 11. From ASBR to ASBR**

We check that the CSPF computation is successful:

```bash
user@mx240-8-re0> show rsvp session transit detail
Transit RSVP: 1 sessions
192.168.38.24
  ...  Explicit route: 172.16.38.1 192.168.38.24
  Record route: 10.36.2.1 172.16.37.2 10.37.1.2 <self> 172.16.38.1 10.38.2.1
  Total 1 displayed, Up 1, Down 0

user@mx240-8-re0> show route 192.168.38.24/32 table inet.0
inet.0: 33 destinations, 40 routes (33 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.38.24/32  *[BGP/170] 03:17:09, MED 1, localpref 100
AS path: 65538 I, validation-state: unverified
  > to 172.16.38.1 via xe-2/1/0.0

user@mx240-8-re0> show ted database 192.168.37.8
TED database: 1 ISIS nodes 10 INET nodes
ID                Type Age(s) LnkIn LnkOut Protocol
192.168.37.8      Rtr   11839     2      3 OSPF(0.0.0.0)
...  To: 10.37.1.1-1, Local: 10.37.1.1, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0

user@mx240-8-re0> show ted database 172.16.38.1
TED database: 1 ISIS nodes 10 INET nodes
ID                Type Age(s) LnkIn LnkOut Protocol
172.16.38.1       ---   11847     1      0
```

Copyright (©) 2016 Juniper Networks. All Rights Reserved.
When we reach MX80-22, we’re finally in the same AS of the egress router, and this hop is from ASBR to PE, as shown in Figure 12.

![Figure 12. From ASBR to PE Router](image)

The RSVP session on MX80-22 shows us:

```
user@mx80-22> show rsvp session detail transit
Transit RSVP: 1 sessions
192.168.38.24
  ... Exptct route: 10.38.2.1 192.168.38.24
  Record route: 10.36.2.1 172.16.37.2 10.37.1.2 172.16.38.2 <self> 10.38.2.1
Total 1 displayed, Up 1, Down 0
```

This is the final step, and all the required information will be found in the IGP database. We can see this by looking at the route toward the egress:

```
user@mx80-22> show route 192.168.38.24/32 table inet.0
inet.0: 29 destinations, 31 routes (29 active, 0 holddown, 0 hidden)
  + = Active Route, - = Last Active, * = Both
  192.168.38.24/32  * [OSPF/10] 03:20:51, metric 1
  > to 10.38.2.1 via xe-0/0/3.0
```

The TED database, of course, has an entry for the egress loopback:

```
user@mx80-22> show ted database 192.168.38.24
TED database: 0 ISIS nodes 7 INET nodes
ID     Type Age(s) LnkIn LnkOut Protocol
192.168.38.24  Rtr  12062     2      2 OSPF(0.0.0.0)
  To: 10.38.2.1-1, Local: 10.38.2.1, Remote: 0.0.0.0
  Local interface index: 0, Remote interface index: 0
```

### 4.2 A Closer Look at the PATH Messages

Now we take a different approach; we’re going to take a look at the contents of the PATH messages. Only the relevant fields are shown in the outputs below.

**MX80-23 (ingress PE AS 65536)**

```
Mar 16 11:43:39.850978   Hop      Len 12 10.31.2.1/0x80000019 << to the ASBR
Mar 16 11:43:39.851042   SrcRoute Len 12  10.36.2.2 S   << ERO up to ASBR
```

**MX240-7 (ASBR AS 65536)**

```
Mar 16 11:43:39.850978   Hop      Len 12 10.31.2.1/0x80000019 << to the ASBR
Mar 16 11:43:39.851042   SrcRoute Len 12  10.36.2.2 S   << ERO up to ASBR
```

```
Mar 16 11:43:39.850978   Hop      Len 12 10.31.2.1/0x80000019 << to the ASBR
Mar 16 11:43:39.851042   SrcRoute Len 12  10.36.2.2 S   << ERO up to ASBR
Mar 16 11:43:39.851042   RecRoute Len 12  10.36.2.2 S   << received ERO
Mar 16 11:43:39.851042   RecRoute Len 12  10.36.2.1 S   << received RRO
```

```
```
4.3 OSPF Really Matters

We've already highlighted that in order to cross the AS border and move to another AS, we need OSPF-TE passive to inject information about the subnet connecting the two ASs into the IGP database.

Now, let's see a few more details on how this works.

Consider ASBR MX240-7, with an OSPF configuration of:

```bash
user@mx240-7-re1# show protocols ospf area 0.0.0.0 interface xe-2/2/0.0 passive {
    traffic-engineering {
        remote-node-id 172.16.37.1;
    }
}
```

Check that the LSAs actually include this subnet:

```bash
user@mx240-7-re1# run show ospf database opaque-area advertising-router 192.168.36.7 detail OSPF database, Area 0.0.0.0
Type ID Adv Rtr Seq Age Opt Cksum Len bits 0x0, link count 5
... id 172.16.37.0, data 255.255.255.252, Type Stub (3)
    Topology count: 0, Default metric: 1
... It is easy to spot the information about the network and subnet mask configured on the link between the two ASs.

The router will advertise opaque LSAs with local/remote inter-AS link IP addresses and install entries for them in the TED.

We can check the database to verify that information is there:

```bash
user@mx240-7-re1# run show ospf database opaque-area advertising-router 192.168.36.7 detail lsa-id 1.0.0.9 OSPF database, Area 0.0.0.0
Type ID Adv Rtr Seq Age Opt Cksum Len
OpaqueArea*1.0.0.9 192.168.36.7 0x80000005 2119 0x22 0x9432 124
Area-opaque TE LSA
    Link (2), length 100:
        Linktype (1), length 1:
```

4.3 OSPF Really Matters

We’ve already highlighted that in order to cross the AS border and move to another AS, we need OSPF-TE passive to inject information about the subnet connecting the two ASs into the IGP database.

Now, let's see a few more details on how this works.

Consider ASBR MX240-7, with an OSPF configuration of:

```bash
user@mx240-7-re1# show protocols ospf area 0.0.0.0 interface xe-2/2/0.0 passive {
    traffic-engineering {
        remote-node-id 172.16.37.1;
    }
}
```

Check that the LSAs actually include this subnet:

```bash
user@mx240-7-re1# run show ospf database opaque-area advertising-router 192.168.36.7 detail OSPF database, Area 0.0.0.0
Type ID Adv Rtr Seq Age Opt Cksum Len bits 0x0, link count 5
... id 172.16.37.0, data 255.255.255.252, Type Stub (3)
    Topology count: 0, Default metric: 1
... It is easy to spot the information about the network and subnet mask configured on the link between the two ASs.

The router will advertise opaque LSAs with local/remote inter-AS link IP addresses and install entries for them in the TED.

We can check the database to verify that information is there:

```bash
user@mx240-7-re1# run show ospf database opaque-area advertising-router 192.168.36.7 detail lsa-id 1.0.0.9 OSPF database, Area 0.0.0.0
Type ID Adv Rtr Seq Age Opt Cksum Len
OpaqueArea*1.0.0.9 192.168.36.7 0x80000005 2119 0x22 0x9432 124
Area-opaque TE LSA
    Link (2), length 100:
        Linktype (1), length 1:
```
LinkID (2), length 4: 172.16.37.1
LocIfAdr (3), length 4: 172.16.37.2
RemIfAdr (4), length 4: 172.16.37.1

The ASBR is actually advertising the two addresses for the inter-AS link.

user@mx240-7-re1# run show ted database 192.168.36.7
TED database: 0 ISIS nodes 12 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.36.7                  Rtr      35     4     4 OSPF(0.0.0.0)
... To: 172.16.37.1, Local: 172.16.37.2, Remote: 0.0.0.0
Local interface index: 0, Remote interface index: 0

But what if we remove the traffic-engineering option from the configuration?

user@mx240-7-rel1# show protocols ospf area 0.0.0.0 interface xe-2/2/0.0
passive {
    inactive: traffic-engineering {
        remote-node-id 172.16.37.1;
    }
}

Remember that in order to reach the egress router, we need a BGP route toward it. Once we have that, we also require that the BGP route next-hop can be found in the TED.

user@mx240-7-re1# run show route 192.168.38.24/32 table inet.0
inet.0: 34 destinations, 49 routes (34 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.177.33.24/32   *[BGP/170] 03:54:21, localpref 100
    AS path: 65537 65538 I, validation-state: unverified
    > to 172.16.37.1 via xe-2/2/0.0
[BGP/170] 03:54:21, localpref 100, from 192.168.36.5
    AS path: 65537 65538 I, validation-state: unverified
    > to 10.36.4.1 via xe-2/0/1.0

user@mx240-7-re1# run show ted database 192.168.36.7
TED database: 0 ISIS nodes 11 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.36.7                  Rtr      35     3     3 OSPF(0.0.0.0)
... # NOTE: no information about 172.16.37.1 !!!

And, as a result, the LSP is down:

user@mx80-23# run show mpls lsp extensive
Ingress LSP: 1 sessions
192.168.38.24
    From: 192.168.36.23, State: Dn, ActiveRoute: 0, LSPname: Inter-AS-38
    ActivePath: (none)
    PathDomain: Inter-domain
... 1140 Mar 16 15:40:01.514 10.36.2.2: No Route toward dest

We have an analogous scenario if we use the loopback address when configuring the remote side of the inter-AS link. In any case, we won't have the BGP next-hop in the TED, making the establishment of the LSP impossible.
5 Inter-AS LSPs with BGP-LS: Overview and Configuration

5.1 The Ingredients

Before going into details, let’s start by highlighting the main components behind this second solution. Some of them have been already used before, while others are totally new.

We will have to deal with:

- IBGP IP unicast
- BGP-TE
- Inter-AS link
- EBGP multihop

5.2 The Main Concept

This second solution really changes how LSP establishment works. We want the ingress router to be able to compute the full ERO up to the egress router. To accomplish this, the ingress router has to enrich its IGP database content. That is exactly what is done here: routes from other ASs are sent to the ingress router, which installs them in its TED. This is accomplished with BGP using IGP link-state information sent across AS boundaries (BGP-LS).

As a result, the TED is much richer, containing routing information about other areas, too. The ingress router is now able to build a more complete view of the network topology that goes beyond its own AS. This allows the ingress router to compute the full route up to the tunnel destination and have a complete ERO. The role of BGP-LS in this process is shown in Figure 13.

![Figure 13. The Role of BGP-LS and the TED](image)

BGP will take care of distributing routes coming from other ASs. This particular BGP “flavor” is known as BGP-LS. This BGP version carries traffic engineering routes that can be installed in the local TED. More details on this are presented later.

The only special configuration still needed is for the inter-AS link.
5.3 BGP-LS

BGP is the protocol responsible for carrying traffic engineering routes across different ASs. To do this it must be configured to work with a new type of BGP address family called traffic-engineering. This configuration must be applied to all internal and external BGP connections. This is because traffic engineering routes must make their way from the PE in one AS to a PE in another AS.

This is a BGP configuration with traffic engineering enabled:

```
user@mx80-23# show protocols bgp
  group IBGP {
    type internal;
    local-address 192.168.36.23;
    family traffic-engineering {
      unicast;
    }
    neighbor 192.168.36.21;
  }
```

5.4 TE Import/Export Routes

TE routes are not exchanged by default: more configuration is needed to make this happen. To understand this mechanism, some background information must be made clear.

IGP protocols like OSPF and IS-IS are used to populate the TED. TED is used by the CSPF algorithm to compute paths when MPLS tunnels must be created. The role of the TED is shown in Figure 14.

![Figure 14. BGP, IGPs, and the TED](image)

If we want to advertise traffic engineering routes with BGP, then we need an import policy. Traffic engineering routes are converted into Local Node/Link NLRI entries and installed into the Lstdist.0 table, used only for link-state route information distribution. BGP looks into that table and retrieves the routes that should be advertised to its neighbors. This configuration section applies the import policy to MPLS:

```
user@mx240-5-re0# show protocols mpls
```
traffic-engineering {
    database {
        import {
            policy TED-TO-NLRI;
        }
    }
}

This import policy is typically applied to an ASBR router carrying routes over the network. This policy takes link-state routes from OSPF and prepares them for BGP NLRIs:

```
user@mx240-5-re0# show policy-options policy-statement TED-TO-NLRI
term 1 {
    from protocol ospf;
    then accept;
}
term default {
    then reject;
}
```

But more is needed than to get OSPF information into BGP. To install the traffic engineering routes into the local TED, we need an export policy. NLRI entries from the lsdist.0 table are copied into the local TED, a process that enriches its contents. As a result, the router that learned the new traffic engineering routes knows about addresses and subnets outside its AS and can compute paths toward those destinations.

```
user@mx80-23# show protocols mpls traffic-engineering {
    database {
        export {
            policy BGP-TO-TED;
        }
    }

    cross-credibility-cspf;
```

Note the presence of the `cross-credibility-cspf` statement. This is needed because when entries are installed into the TED, a “level of credibility” – a sort of ranking – is given to them. The CSPF starts with the highest credibility level, and works its way down, while trying to find a path within that credibility level.

However, in case of inter-AS LSPs, it is essential to work across different credibility levels to compute inter-AS paths. For example, a node in area 0.0.0.0 trying to compute a path through another area might see area 0.0.0.0 entries installed by OSPF, but with a different credibility level. The cross-credibility option allows path computation over different credibility levels.

An export policy is normally used in a PE router that needs a richer TED to compute paths.

```
user@mx80-23# show policy-options policy-statement BGP-TO-TED
term 1 {
    from family traffic-engineering;
    then accept;
}
```

5.5 RSVP and MPLS Interface

Now the ingress PE can compute the full route up to the egress PE in another AS. This means that we will have a complete ERO. And this method is simpler: when taking part in the LSP establishment process using BGP-LS, transit routers (like the ASBRs) no longer have a special role, but simply follow their default behavior.
5.6 Inter-AS Link

This configuration is needed in order to install information about the inter-AS link into the local TED and is almost identical to the OSPF configuration we saw earlier in Section 4.4.

The different is that we now configure the remote loopback address as well as the remote physical interface:

```
user@mx240-5-re0# show protocols ospf
traffic-engineering;
area 0.0.0.0 {
  interface lo0.0 {
    passive;
  }
  interface xe-2/1/0.0;
  interface xe-2/0/1.0;
  interface xe-2/0/0.0 {
    passive {
      traffic-engineering
      remote-node-id 172.16.37.2;
      remote-node-router-id 192.168.37.6;
    }
  }
}
```

5.7 EBGP Multihop

With the PEs in different ASs, we need a multihop BGP session because the PEs are not directly connected. The configuration is identical to the one we had in the previous scenario. This configuration is listed in Section 4.7 and 4.8, and the multihop details are reproduced here for convenience.

```
user@PE2# show protocols
group EBGP-MHOP-VPN {
  type external;
  multihop {
    ttl 10;
  }
--
```

5.8 PE Configuration Example

Let’s highlight the most important aspects of configuring the PE routers.

**IBGP has to include family** inet unicast and family traffic-engineering:

```
user@mx80-23# show protocols bgp
group IBGP {
  type internal;
  local-address 192.168.36.23;
  family inet {
    unicast;
  }
  family traffic-engineering {
    unicast;
  }
  neighbor 192.168.36.21;
  neighbor 192.168.36.7;
  neighbor 192.168.36.5;
}
```

The EBGP multihop has a standard configuration:
group EBGP-MHOP {
  type external;
  multihop {
    ttl 10;
  }
  local-address 192.168.36.23;
  peer-as 65538;
  neighbor 192.168.38.24;
}

For MPLS we have to configure the interactions with the TED and the policy to control that interaction:

```
user@mx80-23# show protocols mpls traffic-engineering {
  database {
    export {
      policy BGP-TO-TED;
    }
  }
  cross-credibility-cspf;
  label-switched-path Inter-AS-65538 {
    from 192.168.36.23;
    to 192.168.38.24;
  }
  interface xe-0/0/0.0;
  interface xe-0/0/2.0;
  interface lo0.0;
user@mx80-23# show policy-options policy-statement BGP-TO-TED
  term 1 {
    from family traffic-engineering;
    then accept;
  }
```

5.9 ASBR Configuration Example

The ASBR configuration is a bit less than in Section 4 and shares some similarities with the configuration for the PE routers before, MPLS needs to manage the interaction with the TED. Remember to include the inter-AS link interface as well.

```
user@mx240-5-re0# show protocols mpls traffic-engineering {
  database {
    import {
      policy TED-TO-NLRI;
    }
  }
  interface xe-2/0/1.0;
  interface xe-2/1/0.0;
user@mx240-5-re0# show policy-options policy-statement TED-TO-NLRI
  term 1 {
    from protocol ospf;
    then accept;
  }
  term default {
    then reject;
  }
```

RSVP requires the inter-AS link interface to be included too:

```
user@mx240-5-re0# show protocols rsvp
  interface xe-2/0/1.0;
  interface xe-2/1/0.0;
```
The IBGP configuration is identical to the one for the PE given in Section 4.8, so we omit it here.

The EBGP configuration must include the traffic-engineering family:

```bash
user@mx240-5-re0# show protocols bgp
  group EBGP {
    type external;
    family traffic-engineering {
      unicast;
    }
    export Lo0 NLRI-TO-BGP;
    peer-as 65537;
    neighbor 192.36.37.2;
  }
```

The OSPF configuration includes the inter-AS link as passive:

```bash
user@mx240-5-re0# show protocols ospf
  area 0.0.0.0 {
    interface lo0.0 {
      passive;
    }
    interface xe-2/1/0.0;
    interface xe-2/0/1.0;
    interface xe-2/0/0.0 {
      passive {
        traffic-engineering
          remote-node-id 172.16.37.0.2;
          remote-node-router-id 192.168.37.6;
      }
    }
  }
```
6 Inter-AS LSPs with BGP-LS: Verification and Analysis

6.1 Scanning the TED

In this section, we’re going to analyze the content of the TED database to see how traffic engineering routes from different ASs can be part of the same database.

When a traffic engineering route is received via BGP, it is installed in the lsdist.0 table. We can see which routes are received from a BGP neighbor using the `receive-protocol` option in the `show route` command:

```plaintext
user@mx80-23# run show route receive-protocol bgp 192.168.36.5 table lsdist.0
lsdist.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
Prefix               Nexthop            MED     Lclpref    AS path
NODE { AS:65537 Area:0.0.0.0 IPv4:192.168.38.6 OSPF:0 }/1152
  *                         192.168.36.5                 100     65537 I
```

Here we see that a router in AS 65536 has received routes from AS 65537:

- The route is from OSPF area 0.
- 192.168.36.5 is the router that received the route via EBGP.
- The advertised address is 192.168.38.6.

Loopbacks are not the only advertised routes:

```plaintext
NODE { AS:65537 Area:0.0.0.0 IPv4:10.37.0.1-1 OSPF:0 }/1152
  *                         192.168.36.5                 100        32 I
```

Now we see how the lsdist.0 table appears on a router in AS 65536:

```plaintext
user@mx80-23# run show route table lsdist.0
lsdist.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
NODE { AS:65537 Area:0.0.0.0 IPv4:10.37.0.1-1 OSPF:0 }/1152
  *[BGP/170] 2d 20:04:21, localpref 100, from 192.177.31.5
  AS path: 65537 I, validation-state: unverified
  > to 10.36.0.2 via xe-0/0/0.0
  to 10.36.2.2 via xe-0/0/2.0
```

What does this output tell us?

- There exists a BGP route to subnet 10.37.0.0/30.
- The route came by way of IBGP from 192.168.36.5.
- Destination is in AS 65537.
- There are two possible next-hops. As usual, only one is active (>).

The next step is to look at the “updated” TED by looking at the different types of entries we can have. A local entry looks like this:

```plaintext
user@mx80-23# run show ted database 192.168.36.5
TED database: 0 ISIS nodes 18 INET nodes
ID               Type Age[s] LnkIn LnkOut Protocol
192.168.36.5      Rtr  246092     3      3 OSPF(0.0.0.0)
  To: 10.36.1.2-1, Local: 10.36.1.1, Remote: 0.0.0.0
     Local interface index: 0, Remote interface index: 0
  To: 10.36.4.2-1, Local: 10.36.4.1, Remote: 0.0.0.0
     Local interface index: 0, Remote interface index: 0
  To: 192.168.37.6, Local: 31.37.0.1, Remote: 172.16.37.2
     Local interface index: 0, Remote interface index: 0
```

Now we can see that:
• Router 192.168.36.5 is in AS 65536.
• 10.36.1.0/30 and 10.36.4.0/30 are two directly connected subnets in AS 65536.
• 192.168.37.6 is the loopback address of a router in AS 65537. From this entry we can also see the information about the inter-AS link (172.16.37.0.0/30) obtained from OSPF-TE.

Here is a BGP-TE entry:

```
user@mx80-23# run show ted database 192.168.38.24
TED database: 0 ISIS nodes 18 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.38.24                 Rtr    6444     2      2 Exported OSPF(2)
    To: 10.38.1.2-1, Local: 10.38.1.2, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
    To: 10.38.2.1-1, Local: 10.38.2.1, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
```

In this output listing we see that:

• The entry is related to a router in AS 65538.
• The router is connected to two subnets: 10.38.2.0/30 and 10.38.1.0/30.
• The route originated in OSPF.

The TED also contains the inter-AS link:

```
user@mx80-23# run show ted database 192.168.38.6
TED database: 0 ISIS nodes 18 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.38.6                  Rtr  257130     2      2 Exported OSPF(1)
    To: 192.168.36.5, Local: 172.16.38.2, Remote: 172.16.37.1
    Local interface index: 0, Remote interface index: 0
    To: 10.37.0.1-1, Local: 10.37.0.1, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
```

Here, the physical interface addresses are not directly advertised. We configured the `remote-node-router-id` option so the information is related to the loopback. Inter-AS link information can be obtained by looking at the “Local” and “Remote” fields.

We have already seen that TED entries are copied into the lsdist.0 table and are ready to be sent as BGP NLRIs. Let’s see what the translation of a local TED entry to an lsdist.0 table entry looks like:

```
user@mx240-5-re0# run show ted database 192.168.36.23
TED database: 1 ISIS nodes 10 INET nodes
ID                            Type Age(s) LnkIn LnkOut Protocol
192.168.36.23                 Rtr  279113     2      2 OSPF(0.0.0.0)
    To: 10.36.0.1-1, Local: 10.36.0.1, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
    To: 10.36.2.1-1, Local: 10.36.2.1, Remote: 0.0.0.0
    Local interface index: 0, Remote interface index: 0
```

```
user@mx240-5-re0# run show route table lsdist.0 te-link-local-node-ip 192.168.36.23
lsdist.0: 58 destinations, 58 routes (58 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
LINK { Local { AS:65536 Area:0.0.0.0 IPv4:192.168.36.23 }.{ IPv4:10.36.0.1-1 }. OSPF(1) 01:45:16 Fictitious
    Remote { AS:65536 Area:0.0.0.0 IPv4:192.168.36.23 }.{ IPv4:10.36.0.1-1 }. OSPF(0) 01:45:16 Fictitious
```

Copyright (©) 2016 Juniper Networks. All Rights Reserved.
6.2 Verifying the LSP

Now we are ready to take a more detailed look at how the establishment of the LSP worked.

First, let’s check to see if the LSP is up:

```
user@mx80-23# run show mpls lsp ingress name Inter-AS-65538
Ingress LSP: 4 sessions
  To            From            State Rt P     ActivePath       LSPname
  192.168.38.24  192.168.36.23         Up   0 *                         Inter-AS-65538
Total 1 displayed, Up 1, Down 0
```

```
user@mx80-23# run show mpls lsp ingress name Inter-AS-65538 detail
Ingress LSP: 4 sessions
  192.168.38.24
    From: 192.168.36.23, State: Up, ActiveRoute: 0, LSPname: Inter-AS-65538
    ... Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 6)
      10.36.0.2 S 10.36.1.1 S 172.16.37.2 S 10.37.0.2 S 172.16.38.1 S 10.38.2.1 S
```

The most important thing to notice is the ERO. It is no longer a partial ERO up to the ASBR, but it has all the strict hops necessary to reach the egress router. The numbering scheme lets us easily see which links are intra-AS (10.x.y.z) and which are inter-AS (172.x.y.z).

Congratulations! You have now established an MPLS LSP that traverses multiple AS domains and includes traffic engineering.
7 Node-Link Protection

In this section we will examine the use of RSVP link protection to protect inter-AS MPLS LSPs. The goal is to create, in parallel, a bypass LSP to go around a failed link or node. In this case, forwarding continues when a failure takes place.

Let’s take a closer look at how link protection works in the BGP-LS scenario we detailed earlier.

7.1 BGP-TE

To achieve the link protection across different ASs in the inter-AS scenario, we need to configure RSVP cross-credibility-cspf on ASBR routers and on the transit routers. This allows them to compute inter-AS bypass paths toward routers (point of local repair, or PLR) in a different AS. The use of the RSVP cross-credibility-cspf statement enables bypass path computation across different credibility levels (for example, across OSPF and BGP-TE credibility levels).

```
lab@mx240-5-re0# show protocols
rsvp {
  cross-credibility-cspf;
  interface xe-2/0/1.0 {
    link-protection;
  }
} [...]
```

Under normal conditions, the LSP would follow the path highlighted in Figure 15.

![Figure 15: The LSP Path Without Cross-Credibility](image)

Having node-link protection enabled causes routers in AS 65536 to compute and create bypass LSPs in order to maintain connectivity in case of link failures or broken nodes.

Every router in the same AS has the egress router lo0 address in their TED, but we need to remember that only the PE device has a “richer” TED due to the use of BGP-TE.

So what could go wrong in AS 65536 that would require a bypass LSP?

- The network node MX80-21 could fail.
- The link from MX80-21 to MX240-5 could fail.

Let’s focus on MX80-23 and see what bypass LSPs have been established:
user@mx80-23# run show mpls lsp bypass
Ingress LSP: 3 sessions
To                  From             State   Rt Style Labelin Labelout LSPname
192.168.36.5       192.168.36.23   Up       0  1 SE       -   300720 Bypass->10.36.0.2->10.36.1.1
192.168.36.5       192.168.36.23   Up       0  1 SE       -   301280 Bypass->10.36.2.2->10.36.4.1
192.168.36.7       192.168.36.23   Up       0  1 SE       -   301264 Bypass->10.36.2.2
Total 3 displayed, Up 3, Down 0

Note that we have three bypass LSPs:

- The **first one** protects MX80-21 node failure by going through MX240-7.
- The **second one** gives an alternative in case the previous bypass is unavailable; in practice, this path will be the same as the "main" LSP.
- The **third one** is used in case the link between MX80-23 and MX240-7 fails.

Let's take a look at the details of the bypass LSPs on PE (MX80-23):

user@mx80-23# run show mpls lsp bypass detail
Ingress LSP: 3 sessions
192.168.36.5
From: 192.168.36.23, LSPstate: Up, ActiveRoute: 0
LSPname: Bypass->10.36.0.2->10.36.1.1
... Explicit route: 10.36.2.2 10.36.4.1

This first LSP goes to MX240-5 by way of MX240-7. This way we have a path in case MX80-21 fails. There is also a second bypass LSP:

192.168.36.5
From: 192.168.36.23, LSPstate: Up, ActiveRoute: 0
LSPname: Bypass->10.36.0.2->10.36.4.1
... Explicit route: 10.36.0.2 10.36.4.1

This second bypass LSP is a way to maintain connectivity in case the previous bypass has problems. Looking at the ERO, we see that packets will follow the same path identified by the primary LSP, the one that is used under normal conditions. A bypass LSP like this one can still be very useful in case our configuration doesn't provide automatic revert to the primary LSP. This way, if a problem with the bypass LSP occurs, if it is at all possible, traffic will start to flow over the primary path again.

Finally, there is a third LSP:

192.168.36.7
From: 192.168.36.23, LSPstate: Up, ActiveRoute: 0
LSPname: Bypass->10.36.0.2
... Explicit route: 10.36.0.2 10.36.3.1

This third LSP can be seen as a link bypass LSP. It avoids the link between the PE and MX240-7 by going to MX80-21 first and then coming back to MX240-7.

Now let's check every router in AS 65536 where RSVP sessions are established and running, as shown in Figure 16.
On the PE, MX80-23, we have two ingress sessions:

- The “original” LSP
- The bypass to avoid node MX80-21

We can even bypass a failed link partway to MX240-5, as shown in Figure 17.

Here is what the RSVP session looks like on MX80-21:
There are two RSVP sessions:

- An ingress session for an LSP used when the link toward MX240-5 is unavailable
- A transit session for the primary LSP

A look at the RSVP sessions on the other ASBR, MX240-7, shows two transit LSPs, both highlighted in Figure 18.

![Figure 18: Transit LSPs on an ASBR](image)

On this ASBR we only have transit sessions:

- The first LSP originates at MX80-21 and is used when Link A is down.
- The second LSP is used when MX240-21 is down.

On router MX240-5, there are two egress LSPs (this is where the bypass LSPs end) and one transit LSP (the primary LSP goes on from here) in the RSVP sessions, as shown in Figure 19.
Figure 19: Transit and Egress LSP RSVP Sessions

user@mx240-5-re0> show rsvp session
Ingress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
Egress RSVP: 2 sessions
To                From            State   Rt Style Labelin Labelout LSPname
192.168.36.5     192.168.36.21   Up       0  1 SE       3        - Bypass->10.36.1.1
192.168.36.5     192.168.36.23   Up       0  1 SE       3        - Bypass->10.36.0.2->10.36.1.1
Total 2 displayed, Up 2, Down 0
Transit RSVP: 1 sessions
To                From            State   Rt Style Labelin Labelout LSPname
192.168.38.24    192.168.36.23   Up       0  1 SE  311072  307280 Inter-AS-65538
Total 1 displayed, Up 1, Down 0

This ASBR, MX240-5, is used by the primary LSP. Both bypass LSPs see this router as the egress, and this explains the two egress sessions in the output. The primary LSP is seen as a transit session that, after leaving this router, goes on to AS 65537.

The RSVP sessions and LSPs in the other ASs are not discussed further.
8 NG MVPN: Overview and Configuration

Now that we know how to create LSPs that cross AS boundaries, we can configure and deploy a so-called next-generation multicast VPN (NG MVPN) that uses these LSPs.

Let’s start with a look at the main components needed to deploy the service. Configuration is required for two main aspects of the feature:

- The Provider–Customer (PE-CE) connection—This is where the multicast protocol, in this case Protocol Independent Multicast (PIM), runs.
- The core network—There is no multicast routing protocol running here. We simply use BGP to allow the routers to exchange information about VPN membership.

All forwarding uses MPLS LSP tunnels. These tunnels are actually point-to-multipoint LSPs (P2MP LSPs), used to mimic a multicast tree. To create a P2MP LSP that crosses AS boundaries, we use previously described techniques, such as BGP-TE.

8.1 The PIM Protocol

PIM is the most common multicast protocol. PIM is used to create a distribution tree over which multicast packets travel to reach all the destinations interested in the multicast (members of the multicast group).

There are two ways this distribution tree can be built:

- When the source is known, it is possible to build a sourced-tree (S,G). This kind of tree has its root at the multicast source.
- When the source is initially unknown, it is possible to elect a node to act as a rendezvous point (RP). The tree will be initially rooted at the RP. Later, when a multicast source is discovered, the tree will change and will converge into a (S,G) tree.

Both methods are shown in Figure 20.

![Figure 20. Multicast Shared Tree and Sourced Tree Methods](image-url)
Our example uses the second option, and the RP connected to the multicast source is the CE router. In addition, PIM must be configured on all PE and CE devices.

The configuration of the PIM protocol on the CE is simple and easy:

```plaintext
pim {
    rp {
        local {
            family inet {
                address 192.168.100.1;
            }
        }
        interface lo0.1;
        interface xe-0/0/1.1;
        interface xe-0/0/3.0;
    }
}
```

The `local` keyword indicates that the RP is located on this node.

Other CEs or PEs also have to know about the location of the RP router. As a result, the configuration of the PIM protocol slightly changes:

```plaintext
pim {
    rp {
        static {
            address 192.168.100.1;
        }
        interface xe-0/0/1.1;
        interface lo0.1;
    }
}
```

Here we no longer use `local`; instead we use `static` to indicate the RP location.

### 8.2 P2MP LSP

On PE devices facing the core network, such as MX80-23, where PIM is not running, we configure a point-to-multipoint (P2MP) LSP, as shown in Figure 21.

![Figure 21. P2MP LSP in the Core Network](image)

To create this LSP, first we define the LSP as a template:

```plaintext
user@mx80-23# show protocols mpls
label-switched-path inter-as-p2mp-template {
    template;
    p2mp;
    }
```
The template makes it easy to apply P2MP characteristics to an LSP. Next, we move to the routing instance configuration where we use that template:

```
user@mx80-23# show routing-instances NG-MVPN-1
instance-type vrf;
interface xe-0/0/1.1;
interface lo0.1;
route-distinguisher 192.168.36.23:36;
provider-tunnel {
   rsvp-te {
      label-switched-path-template {
         inter-as-p2mp-template;
      }
   }
}
```

The `provider-tunnel` keyword means that traffic will be tunneled across the P2MP tunnel.

When configuring how the provider tunnel should work, remember that two possibilities exist:

- **Inclusive**: This is the simplest option. When a packet enters the tunnel, it is sent to all the endpoints, meaning all the PEs receive the packet. However, this solution could lead to an inefficient use of bandwidth because some PEs might not be connected to any multicast receiver at that time.

- **Selective**: To increase efficiency, we can send packets only where necessary. A selective tunnel works by making it possible to specify a subset of PEs that receive those packets. So only some branches of the tree see traffic, while other branches see none.

The configuration provided earlier is used for an inclusive tunnel, and that is what we are using in this example.

### 8.3 MVPN Signaling

Let's see how multicast VPN (MVPN) information is exchanged among PE routers. Once again, we use BGP as routing protocol. This confirms how BGP can be effectively used to transport “services.”

In Junos OS terms, in order to enable MVPN signaling, we have to configure a new family when defining the EBGP sessions among PE routers. In this case, the new family is “Internet MVPN signaling.”

```
user@mx80-23# show protocols bgp
group EBGP-MHOP {
   type external;
   multihop {
      ttl 10;
   }
   local-address 192.168.36.23;
   family inet-vpn {
      unicast;
   }
   family inet-mvpn {
      signaling;
   }
   peer-as 65538;
   neighbor 192.168.38.24;
   neighbor 192.168.38.1;
}
```

It is important to note that this family is necessary only for EBGP sessions between PEs. When configuring IBGP sessions (PE-P or P-P), we don’t need it. If we decided to use the expand-next-hop solution to build LSPs, we still have to configure the IBGP sessions.
At this point we still need to configure MVPN on the virtual routing and forwarding (VRF) routing instance:

```bash
user@mx80-23# show protocols mvpn
mvpn {
  unicast-uhm-election;
  mvpn-mode {
    rpt-spt;
  }
}
```

Let’s see what the options are for:

- **unicast-uhm-election**: This is necessary with multihomed scenarios. Unicast UHM method elects UHM based on BGP best-path selection rules.

- **rpt-spt**: Defines the `mvpn-mode` used here. Initially traffic flows over the shared tree, `rpt`, until the source is unknown. After that, traffic switches to the source tree, `spt`, once the multicast source is learned. The default mode is `spt-only`, which we aren’t using here.

Here are some additional useful details about the two possible modes:

- **RPT-SPT mode**: There are no restrictions on the C-RP. (The customer RP, or C-RP, is an RP router located somewhere within the VPN and can be either a service provider router or a customer router.) The C-RP can be located on both the CE or the PE under the VRF configuration. Be aware that the multicast source and receiver cannot be directly connected to PEs.

- **SPT-only mode (default)**: Traffic flows directly over the SPT (source tree) in SPT-only mode. The C-RP must be located on one of the PEs under a VRF operating in STP-only mode. If this is not possible, it requires MSDP/Anycast-RP connecting the C-RP (located on a CE router) to one of the PEs. The multicast source and receiver can be directly connected to PEs operating in STP-only mode.

Once we have a complete MVPN configuration it is useful to look at the different types of messages sent among PEs to exchange MVPN information. These are shown in Table 1.

### Table 1: The Seven Types of MVPN Messages

<table>
<thead>
<tr>
<th>Route</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1 Intra-AS I-PMSI AD</td>
<td>Originated by all PE routers and is used for advertising and learning intra-AS MVPN membership information.</td>
</tr>
<tr>
<td>Type 2 Inter-AS I-PMSI AD</td>
<td>Originated by NG MVPN ASBR routers, and used for advertising and learning inter-AS MVPN membership information.</td>
</tr>
<tr>
<td>Type 3 S-PMSI AD</td>
<td>Originated by ingress PE routers, and is used for initiating a selective P-Tunnel for a given C-Source and C-Group multicast stream.</td>
</tr>
<tr>
<td>Type 4 Leaf AD</td>
<td>Originated by Egress PE routers in response to a Type 3 announcement. It used for indicating interest for a given C-Source and C-Group multicast stream.</td>
</tr>
<tr>
<td>Type 5 Source Active AD</td>
<td>Originated by a PE router (ingress PE) when it learns about an active multicast source. The Type 5 route is announced to all egress PEs that belongs to a given NG MVPN.</td>
</tr>
<tr>
<td>Type 6 Shared-Tree Join</td>
<td>Originated by an egress PE when it receives a PIM shared-tree join (C-*_G) from the CE device</td>
</tr>
<tr>
<td>Type 7 Source-Tree Join</td>
<td>Originated by an egress PE when it receives a source-tree join, or when it receives a Type 5 route announcement from an ingress PE.</td>
</tr>
</tbody>
</table>
There are seven different messages:

- Type 1 and Type 2 messages are similar and are used to exchange information about membership to a particular MVPN. This information is useful when building the P2MP because we know which PEs need to be reached.

- Type 3 messages are used when there is the intention to build a selective tree. In response to this message, egress routers can reply with a Type 4 message, indicating interest in a particular source and group.

- Type 5 messages are generated when a source is discovered. It is important to notice that before this message is created, a pure PIM message is generated. PE routers receive that message and translate its content into a Type 5 MVPN signaling message.

- Type 6 and 7 MVPN signaling messages are similar to Type 5 and are sent when an egress router receives a PIM shared-tree join or a PIM source-tree join message. Those messages are translated, and Type 6 or Type 7 MVPN signaling messages are sent. Type 7 messages are also sent in response to a Type 5 route announcement received from an ingress PE.

### 8.4 CE Configuration

The first thing to configure is the PIM protocol running between the CE and the PE.

```
user@mx80-1# show protocols pim
pim {
  rp {
    local {
      family inet {
        address 192.168.100.1;
      }
    }
  }
  interface lo0.1;
  interface xe-0/0/1.1;
  interface xe-0/0/3.0;
}
```

Then we have to define the EBGP session with the PE router:

```
user@mx80-23# show protocols bgp
bgp {
  group EBGP {
    type external;
    export MCAST-SRC;
    peer-as 65536;
    neighbor 10.136.1.1;
    neighbor 10.136.3.1;
  }
}
```

### 8.5 PE Configuration

Now let's look at all the different protocols we have to configure on the PE router. The PE needs to configure a routing instance, and inside the instance we also configure all the necessary protocols, such as PIM, EBGP for a session with the CE, and support for NG MVPN.

```
user@mx80-23# show vrf
instance-type vrf;
interface xe-0/0/1.1;
interface lo0.1;
```
route-distinguisher 192.168.36.23:36;
vrf-target target:360:380;
vrf-table-label;

bgp {
  group MVPN1-EBGP {
    type external;
    peer-as 65500;
    neighbor 10.136.1.2;
  }
}

pim {
  rp {
    static {
      address 192.168.100.1;
    }
  }
  interface xe-0/0/1.1;
  interface lo0.1;
}

mvpn {
  unicast-umh-election;
  mvpn-mode {
    rpt-spt;
  }
}

From the main hierarchy, we need to configure MPLS:

```
user@mx80-23# show protocols mpls
traffic-engineering {
  database {
    export {
      policy BGP-TO-TED;
    }
  }
}
```
cross-credibility-cspf;
lable-switched-path inter-as-p2mp-template {
  template;
p2mp;
}
interface xe-0/0/0.0;
interface xe-0/0/2.0;
interface lo0.0;

In this example, BGP-TE is used, so a TED export policy is defined.

IBGP configuration is a standard one. Just remember to include:

- family inet unicast
- family traffic-engineering when BGP-TE is used

EBGP between PEs is also a multihop session. We also need to configure support for multicast VPN signaling:

```
user@mx80-23# show protocols bgp
```
group EBGP-MHOP {
  type external;
multihop {
    ttl 10;
  }
  local-address 192.168.36.23;
  family inet-vpn {
    unicast;
  }
  family inet-mvpn {
    signaling;
  }
```
peer-as 65538;
neighbor 192.168.38.24;
neighbor 192.168.38.1;
9 NG MVPN: Monitoring and Verification

9.1 Autodiscovery

As soon as the configuration is committed, PE routers discover their neighbors in order to determine the other members of the MVPN. This process is called autodiscovery, as shown in Figure 22, and makes use of the Type 1 MVPN messages described before.

Figure 22. PE Autodiscovery

The MVPN messages are sent and received by BGP:

```
user@mx80-23# run show route receive-protocol bgp 192.168.38.1 table NG-MVPN-1.mvpn.0 detail
NG-MVPN-1.mvpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
  * 1:192.168.138.1:33:192.168.38.1/240 (1 entry, 1 announced)
    Import Accepted
    Nexthop: 192.168.38.1
    AS path: 65538 I
    Communities: target:360:380
    PMSI: Flags 0x0: Label 0: RSVP-TE:
    Session_13[192.168.38.1:0:7209:192.168.38.1]
```

Note the format of the NLRI:

- NLRI-type: in this case 1
- Route distinguisher: 192.168.138.1:33
- Remote PE loopback: 192.168.38.1

We also see information about some common BGP attributes like next-hop or AS-path. The Community attribute is not standard but extended; it contains the route-target. Another important value to keep in mind is the RSVP-TE session number, 7209.

Identical output is obtained by examining routes received from other EBGP neighbors. These routes are installed in the main routing table of the routing instance:

```
user@mx80-23> show route table NG-MVPN-1.mvpn.0
NG-MVPN-1.mvpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
  * = Active Route, - = Last Active, * = Both
    * [MVPN/70] 03:43:12, metric2 1
    Indirect
    * [BGP/170] 01:28:47, localpref 100, from 192.168.38.24
    AS path: 65538 I, validation-state: unverified
to 10.36.0.2 via xe-0/0/0.0, Push 300688, Push 299872(top)
    * [BGP/170] 01:28:47, localpref 100, from 192.168.38.24
    AS path: 65538 I, validation-state: unverified
```

Note the format of the NLRI:
It is easy to pick out two routes, one for each PE router. Also note the presence of two push operations: one for the P2MP LSP, common to all the destinations, and one for the "sub-LSP."

Additional information is available when using the `detail` keyword:

```
user@mx80-23> show route table NG-MVPN-1.mvpn.0 detail
1:192.168.138.1:33:192.168.38.1/240 (1 entry, 1 announced)
  *BGP  Preference: 170/-101
  PMSI: Flags 0x0: Label 0: RSVP-TE: Session_13[192.168.38.1:0:7209:192.168.38.1]
  ... Source: 192.168.38.1 Protocol next hop: 192.168.38.1 ...
  ... Local AS: 65538 Peer AS: 65538 ...
  ... Router ID: 192.168.38.1 Primary Routing Table bgp.mvpn.0
```

Once messages are exchanged, membership information is spread across the network. As a result, the PEs build a list of neighbors:

```
user@mx80-23> show mvpn neighbor instance-name NG-MVPN-1
MVPN instance:
Legend for provider tunnel
S= Selective provider tunnel
Legend for c-multicast routes properties (Pr)
DS -- derived from (*, c-g) RM -- remote VPN route
Family : INET
Instance : NG-MVPN-1
MVPN Mode : RPT-SPT
Neighbor                   Inclusive Provider Tunnel
192.168.38.1               RSVP-TE P2MP:192.168.38.1, 7209,192.168.38.1
```

The list contains two neighbors. They are the two egress routers of our P2MP LSP.

Finally, we can look at the state of the LSP that will carry the traffic over the core:

```
user@mx80-23> show mpls lsp p2mp ingress detail
  Ingress LSP: 1 sessions
  P2MP name: 192.168.136.23:36:mvpn:NG-MVPN-1, P2MP branch count: 2
  192.168.38.24
  From: 192.168.36.23, State: Up, ActiveRoute: 0, LSName:
  ActivePath:  (primary)
  P2MP name: 192.168.136.23:36:mvpn:NG-MVPN-1
  ... Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 5)
  10.36.2.2 S 10.36.37.1 S 10.37.1.1 S 10.37.38.1 S 10.38.2.1 S
  192.168.38.1
  From: 192.168.36.23, State: Up, ActiveRoute: 0, LSName:
  ActivePath:  (primary)
  P2MP name: 192.168.136.23:36:mvpn:NG-MVPN-1
  ... Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 5)
  10.36.2.2 S 10.36.37.1 S 10.37.1.1 S 10.37.38.1.1 S 10.38.0.1
```

Let’s try to understand how to read this output:

- There is one, and only one, ingress LSP session, which is a P2MP one. Then different “sub-LSPs” are listed.
• For each “sub-LSP,” we have all the usual information, like LSP name and ERO. In addition, we have the information about the P2MP LSP they belong to. Note that we have a complete ERO due to the fact that BGP-LS is used in this example.

On the other end, we have three separate ingress RSVP sessions:

user@mx80-23> show rsvp session p2mp ingress
Ingress RSVP: 3 sessions
P2MP name: 192.168.136.23:36:mvpn:NG-MVPN-1, P2MP branch count: 2
To              From            State   Rt Style Labelin Labelout LSPname
192.168.38.24   192.168.36.23   Up       0  1 SE       -   300880
192.168.38.1    192.168.36.23   Up       0  1 SE       -   300880
Total 2 displayed, Up 2, Down 0

Note that regardless of the final destination, the Labelout value is the same: 300800.

At this point, the P2MP is up and PIM is correctly configured. But no receivers or senders have been discovered.

9.2 Join Operations at the CE

When a receiver wants to join the tree to receive multicast traffic, it first sends an IGMP packet to the CE to request membership in a multicast group, as shown in Figure 23.

![Figure 23. Membership Request](image)

Let’s see what happens at a CE:

user@mx80-4> show pim neighbors
B = Bidirectional Capable, G = Generation Identifier
H = Hello Option Holdtime, L = Hello Option LAN Prune Delay,
P = Hello Option DR Priority, T = Tracking Bit
Instance: PIM.master
Interface           IP V Mode        Option       Uptime Neighbor addr
xe-0/0/0.1           4 2             HPLGT      04:36:39 10.138.1.1
xe-0/0/3.0           4 2             HPLG       00:00:19 10.4.4.2

There are two neighbors:

• The PE, connected through xe-0/0/0.1
• The receiver, connected through xe-0/0/3.0

On the CE, we can also look at the PIM groups:

user@mx80-4> show igmp group
Interface: xe-0/0/3.0, Groups: 2
  Group: 233.252.0.1
Another useful PIM monitoring command is:

```
user@mx80-4> show pim join logical-system MCAST-RCVR-1 detail
Instance: PIM.master Family: INET
R = Rendezvous Point Tree, S = Sparse, W = Wildcard
Group: 233.252.0.1
Source: *
  RP: 192.168.100.1
  Flags: sparse,rptree,wildcard
  Upstream interface: xe-0/0/0.1
  Downstream neighbors:
       Interface: xe-0/0/3.0
```

The source is unknown, but the RP address is known. The “Flags” field also tells us we are currently in shared-tree mode (rptree). The upstream interface is the one toward the multicast receiver, while downstream neighbors represent PEs.

### 9.3 Join Operations at the PE

To sum up, the multicast receiver sends an IGMP message to the CE. In response, the CE sends a PIM join shared-tree message to the PE. This is shown in Figure 24.

![Figure 24. PIM Join Shared-Tree Message](image)

We can look at join information for a particular instance of the PE:

```
user@mx80-24> show pim join instance NG-MVPN-1 extensive
Instance: PIM.NG-MVPN-1 Family: INET
R = Rendezvous Point Tree, S = Sparse, W = Wildcard
Group: 233.252.0.1
Source: *
  RP: 192.168.100.1
  Flags: sparse,rptree,wildcard
  Upstream protocol: BGP
  Upstream interface: Through BGP
  Upstream neighbor: Through MVPN
  Upstream state: Join to RP
  Uptime: 00:08:44
  Downstream neighbors:
       Interface: xe-0/0/0.1
       10.138.1.2 State: Join Flags: SRW Timeout: 165
       Uptime: 00:08:44 Time since last Join: 00:00:44
  Number of downstream interfaces: 1
```

Again, no source is specified. The upstream state is currently “Join to RP,” the shared tree. The RP address is available and provided through configuration.
PE routers are responsible for translating PIM messages into MVPN signaling messages. In this case, we should see Type 6 MVPN messages advertised to other PEs:

```
user@mx80-24> show route advertising-protocol bgp 192.168.36.23 table NG-MVPN-1.mvpn.0
NG-MVPN-1.mvpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
       *                         Self                                    I
       Type: RD-MX80-23: AS-MX80-23: C-RP Mask: C-RP Address: C-G Mask: C-Group
```

It is easy to recognize the message, as it starts with a “6”.

On the other side of the core, the ingress PE receives the Type 6 message and performs two actions. First, a route is installed in the routing table:

```
user@mx80-23> show route table NG-MVPN-1.mvpn.0
NG-MVPN-1.mvpn.0: 4 destinations, 6 routes (4 active, 1 holddown, 0 hidden)
   + = Active Route, - = Last Active, * = Both
         [BGP/170] 00:21:10, localpref 100, from 192.168.38.24
         AS path: 65538 I, validation-state: unverified
         > to 10.36.0.2 via xe-0/0/0.0, Push 300672, Push 299872(top)
         to 10.36.2.2 via xe-0/0/2.0, Push 300672, Push 299904(top)
```

Second, the ingress PE translates the message back into a PIM (*,G) shared-tree join message that will be sent to the ingress CE. We can verify this by running the previous PIM command on the CE:

```
user@mx80-1> show pim join extensive
Instance: PIM.master Family: INET
   R = Rendezvous Point Tree, S = Sparse, W = Wildcard
Group: 233.252.0.1
   Source: *
   RP: 192.168.100.1
   Flags: sparse,rptree,wildcard
   Upstream interface: Local
   Upstream neighbor: Local
   Upstream state: Local RP
   Uptime: 00:27:44
   Downstream neighbors:
      Interface: xe-0/0/1.1
      10.136.1.1 State: Join Flags: SRW Timeout: 166
      Uptime: 00:27:44 Time since last Join: 00:00:44
      Number of downstream interfaces: 1
```

### 9.4 From to Shared Tree to Source Tree

At a certain point, the multicast source starts sending traffic. The CE connected to the receiver sees this traffic, learns its location, and performs a join source tree. As a result, it will switch from a shared tree (RPT) to a source tree (SPT). This is shown in Figure 25.
We can check the PIM join messages on the CE:

```
user@mx80-1> show pim join
Instance: PIM.master Family: INET
R = Rendezvous Point Tree, S = Sparse, W = Wildcard
Group: 233.252.0.1
    Source: *
    RP: 192.168.100.1
    Flags: sparse,rptree,wildcard
    Upstream state: Join to RP

Group: 233.252.0.1
    Source: 10.4.2.2
    Flags: sparse,spt
    Upstream state: Join to Source, No Prune to RP
```

There are two results for the group:

- The first refers to the join operation for the shared tree.
- The second refers to the join operation for the source tree.

The difference can be seen by looking at the “Flags” and “Upstream state.” In the second case, we have “Join to source.”

After receiving a PIM join source-tree message from the CE, the ingress PE creates and advertises a Type 5 BGP message:

```
user@mx80-23> show route advertising-protocol bgp 192.168.38.24 table NG-MVPN-1.mvpn.0
NG-MVPN-1.mvpn.0: 6 destinations, 9 routes (6 active, 2 holddown, 0 hidden)
    Prefix     Nexthop     MED   Lclpref    AS path
      *                  Self                             I

user@mx80-23> show route advertising-protocol bgp 192.168.38.1 table NG-MVPN-1.mvpn.0
NG-MVPN-1.mvpn.0: 6 destinations, 9 routes (6 active, 2 holddown, 0 hidden)
    Prefix     Nexthop     MED   Lclpref    AS path
      *                  Self                             I
```
This route is advertised to both EBGP multihop neighbors.

On the other side of the network, receivers will try to join the source tree as well. As a result, new PIM join messages are created:

```
user@mx80-24> show pim join extensive instance NG-MVPN-1
Instance: PIM.NG-MVPN-1 Family: INET
R = Rendezvous Point Tree, S = Sparse, W = Wildcard
Group: 233.252.0.1
Source: *
RP: 192.168.100.1
Flags: sparse,rptree,wildcard
Upstream protocol: BGP
Upstream interface: Through BGP
Upstream neighbor: Through MVPN
Upstream state: Join to RP
Uptime: 00:37:37
Downstream neighbors:
  Interface: xe-0/0/0.1
  10.138.1.2 State: Join Flags: SRW Timeout: 172
  Uptime: 00:37:37 Time since last Join: 00:00:37
Number of downstream interfaces: 1
```

```
user@mx80-24> show route advertising-protocol bgp 192.168.36.23 table NG-MVPN-1.mvnp.0 detail
NG-MVPN-1.mvnp.0: 5 destinations, 6 routes (5 active, 1 holddown, 0 hidden)
  * 7:192.168.136.23:36:36:37:10.4.2.2:37:233.252.0.1/240 (2 entries, 2 announced)
Type: RD-MX80-23: AS-MX80-23: C-S Mask: C-Source Address: C-G Mask: C-Group
```

Figure 26. From PIM Join to BGP NLRI

Figure 26 shows how the egress PE router translates PIM (S,G) join (source tree) into Type 7 NLRI source-tree join (C-S, C-G). Remember that Type 7 messages are also created in response to a Type 5 message.

```
user@mx80-24> show route advertising-protocol bgp 192.168.36.23 table NG-MVPN-1.mvnp.0 detail
NG-MVPN-1.mvnp.0: 5 destinations, 6 routes (5 active, 1 holddown, 0 hidden)
  * 7:192.168.136.23:36:36:37:10.4.2.2:37:233.252.0.1/240 (2 entries, 2 announced)
Type: RD-MX80-23: AS-MX80-23: C-S Mask: C-Source Address: C-G Mask: C-Group
```
Ingress PE router receives and installs the Type 7 source-tree join into its table:

* [PIM/105] 00:15:39
  Multicast (IPv4) Composite
  [BGP/170] 00:15:39, localpref 100, from 192.168.38.24
  AS path: 65538 I, validation-state: unverified
  > to 10.36.0.2 via xe-0/0/0/0.0, Push 300672, Push 299872 (top)
  to 10.36.2.2 via xe-0/0/2.0, Push 300672, Push 299904 (top)

The ingress PE will translate the message back to a PIM one and send it to the CE. Finally, we have our source tree on which traffic can flow from source to destinations.

### 9.5 Multicast Traffic Flow

Now, traffic is ready to flow across the network. This is shown in Figure 27.

![Figure 27. Traffic Flowing on the Network](image)

On the PE, we can look at traffic data:

```
user@mx80-23> show multicast route extensive instance NG-MVPN-1
Instance: NG-MVPN-1 Family: INET
Group: 233.252.0.1/32
  Source: *
  Upstream interface: xe-0/0/1.1
  Downstream interface list:
    xe-0/0/2.0
  Number of outgoing interfaces: 1
  Session description: Organisational Local Scope
  Statistics: 0 kbps, 0 pps, 40 packets
  Next-hop ID: 1048590
  Upstream protocol: MVPN
  Route state: Active
  Forwarding state: Forwarding
  Cache lifetime/timeout: forever
  Wrong incoming interface notifications: 0
  Uptime: 00:55:27
Group: 233.252.0.1
  Source: 10.4.2.2/32
  Upstream interface: xe-0/0/1.1
  Downstream interface list:
    xe-0/0/2.0
  Number of outgoing interfaces: 1
  Session description: Organisational Local Scope
  Statistics: 490 kbps, 1000 pps, 1471501 packets
  Next-hop ID: 1048590
  Upstream protocol: MVPN
  Route state: Active
  Forwarding state: Forwarding
```
The most important thing to notice in this output is the traffic statistics. Data is flowing on the source tree as expected, suggesting that everything is working properly.

Traffic flows on the PMSI to all the destination PEs:

```
user@mx80-23> show mpls lsp statistics p2mp
Ingress LSP: 1 sessions
P2MP name: 192.168.136.23:36:mvpn:NG-MVPN-1, P2MP branch count: 2
To              From            State     Packets            Bytes LSPname
Total 2 displayed, Up 2, Down 0
```

10 Conclusion

This paper has shown how to configure two complex elements of modern networks:

1. Inter-AS LSPs with TE
2. NG MVPN spanning multiple ASs

These are necessary (p.5-6, Section 2) to provide end-to-end services (e.g., L3VPN) across multiple domains. This document covered non-segmented flat tunnels approach to establish end-to-end RSVP-TE tunnels across different domains. Because we are using flat tunnels, BGP-LU is not necessary (but can be used). These configurations used the simple BGP unicast family throughout.

The basic LSP configuration was completed (p. 23, Section 5.3) but did not function correctly. In order to reach the egress router, a BGP route toward it is needed. We also required that the BGP route next-hop be found in the TED. Because these conditions were not satisfied, the LSP remained down.

Even if we used the loopback address when configuring the remote side of the inter-AS link, we would not have the BGP next-hop in the TED, making the establishment of the LSP impossible.

The paper corrected this situation (p.24, Section 6.2) by using a "second solution." We added the ability for the ingress router to compute the full ERO up to the egress router. To accomplish this, the ingress router had to enrich its IGP database content, which we did by configuring BGP to use IGP link-state information sent across AS boundaries (BGP-LS).

As a result, the TED was much richer, and contained routing information about other areas. The ingress router was then able to build a more complete view of the network topology going beyond its own AS. This allowed the ingress router to compute the full route up to the tunnel destination and have a complete ERO. Then (p.32, Section 7.2) we could establish an MPLS LSP that traverses multiple AS domains and includes traffic engineering.

After adding (p. 33, Section 8) link protection to the configuration, we added NG-MVPN capability with the PIM protocol using the sourced tree option in which the RP connected to the multicast source is the CE router (p.39, Section 9.1).

The paper ended with a verification of the solution.