Ingress Replication P-Tunnels in MVPN

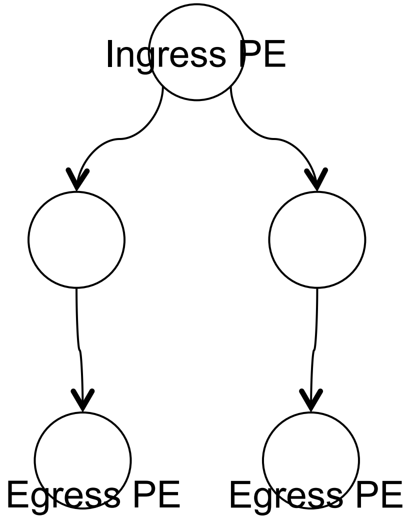
- *Ingress Replication (IR)* is one of the MVPN P-tunnel technologies
- But there's a lot of confusing text in the documents
  - Sometimes an IR tunnel is discussed as if it were just a unicast tunnel, or perhaps a set of unicast tunnels
  - But there are places in the spec where one is told to:
    - advertise (to multiple PEs) the (single) tunnel on which a given flow is sent
    - discard packets from the wrong PE (how do you know the ingress PE of a unicast tunnel, if it's an LDP-created LSP)
    - discard packets that come from an unexpected tunnel (extranet)
    - change the upstream multicast hop for a given tunnel (i.e., prune yourself from a given tunnel and rejoin it at a different place)
- This text is about some kind of P2MP tunnel, not about unicast tunnels
- There seems to be some concept of IR tunnel in which an IR tunnel is a P2MP tunnel, but the transport mechanism is unicast encapsulation.

Purpose of the IR Draft

- When implementing/deploying IR capability, we discovered quite a few questions whose answers were not obvious
- draft-rosen-l3vpn-ir attempts to clear up the issues around IR tunnels:
  - establish clear conceptual model
  - explain how an IR tunnel is identified
  - explain how to join/leave an identified IR tunnel
  - how to apply the discard from the wrong PE/tunnel policy to IR tunnels
  - set out the requirements on MPLS label allocation
  - explain how to switch from one IR tunnel to another in “make before break” fashion
  - explain how to change your UMH within a given IR tunnel, again in “make before break” fashion.

What is an IR Tunnel?

- An IR tunnel is a P2MP tree:
  - Root node, leaf nodes, possibly intermediate nodes
  - Each node maintains multicast state
  - Traffic from a parent node to each of its child nodes is carried through a unicast tunnel
- IR tunnels can be segmented or non-segmented
  - Non-segmented: root node is ingress PE, leaf nodes are egress PEs, no intermediate nodes
  - Segmented: multi-level P2MP tree, with ABRs/ASBRs as intermediate nodes
- Each edge is a unicast tunnel:
  - Sequence of routers that do not maintain multicast state
  - Unicast tunnels may carry packets of multiple IR tunnels, along with “real” unicast packets
  - Tree must be identifiable from packet encapsulation (label)



IR Tunnel Setup Protocol

- Only P2MP tunnel type that doesn’t come with own setup
- All setup is done using MVPN BGP A-D routes
  - Advertise tree (and bind flow(s) to it) via I/S-PMSI A-D route
  - To join a tree:
    - choose a parent node
    - create a Leaf A-D route that identifies the tree
    - “target” Leaf A-D route to parent node by attaching IP-address-specific RT identifying the parent node
- Problem: how to identify an IR tree
  - For most tunnel types, the identifier is in the PMSI Tunnel attribute, but not for IR tunnels!

IR Tunnels and the PMSI Tunnel Attribute

- PMSI Tunnel Attribute (PTA) has:
  - Tunnel type field
  - Tunnel identifier field
  - MPLS label (usually upstream-assigned, for VPN multiplexing)
- In I/S-PMSI A-D route, if type is IR, identifier and label fields unused!
- In Leaf A-D route, if type is IR, PTA identifier field contains unicast IP address of the originator of the route
  - Where’s the tree identifier?
  - The NLRI of the A-D route (identifying the flow) also has to be thought of as the tunnel identifier
- In Leaf A-D route, PTA label field contains downstream-assigned label
  - Are there any requirements on the label allocation policy?
  - Can the PTAs of different Leaf A-D routes use the same label?

What goes in the Leaf A-D Route PTA?

- When Leaf A-D route is sent from child to parent, RT identifies parent, child identified in both NLRI and PTA “tunnel id” field
- Not much information provided about the unicast tunnel between parent and child
  - only child IP address provided
  - unicast tunnel type must be known *a priori*
- Child provides MPLS label (downstream-assigned) that parent uses when transmitting through the IR tunnel to the child
  - MPLS label field of Leaf A-D route PTA
  - On data packets, label is carried inside a unicast encapsulation (which is likely to itself be MPLS, possibly with implicit null)

MPLS Label Allocation Policy (1)

- Every IR tunnel has a “root node” and a “root RD”
- Egress PE policies:
  - Never assign same label to IR tunnels that have different root nodes
    - Otherwise “discard from wrong PE” policy cannot be applied
  - If changing parent nodes on a given tree, change the label also
    - During the transient, one may receive duplicate packets, as old and new parents may both be transmitting
    - Need to use different labels to ensure that one of the duplicates is discarded

MPLS Label Allocation Policy (2)

- Acceptable Egress PE policy for non-extranet:
  - Label unique per <root, parent, egress VRF>
  - Allows “discard from wrong PE” policy to be applied
  - Prevents duplicates during transient changes
  - Allows dispatch to proper VRF context
- Acceptable Egress PE policy for extranet:
  - Label unique for each <root, RD of root, parent>
  - Need uniqueness per ingress VRF, to apply “discard from wrong P-tunnel” policy that is needed for extranet
  - Allows dispatch to multiple VRFs
  - Prevents duplicates during transient changes
- Policy for intermediate node label allocation slightly different, see draft for details

Make before Break (1)

- *Make before break* is desirable when:
  - Changing the IR tree on which a given C-flow is to be received
  - Changing one’s parent node on a given IR tree
- To change parent node, change the RT on Leaf A-D
- Effect*: simultaneously* and immediately prunes from the old parent and joins via the new parent
- But to do make before break, we want to:
  - keep receiving traffic from the old for awhile
  - join the new, but discard traffic from the new for a while
  - start accepting traffic from the new, but discard from the old
  - prune from the old
- Can’t do this with control plane:
  - RT change has simultaneous join-new/prune-old effect
  - Can’t use two RTs because there’s only one PTA

Make before Break (2)

- Make before break must be done with data plane timers
- Parent node actions:
  - When a child node prunes itself from an IR tree, old parent node keeps transmitting to it on that tree, for a period of time
  - When a child node joins a tree via a particular parent, new parent begins transmitting immediately
- Child node actions:
  - When joining a tree via a particular parent, and already joined via a different parent, for a period of time discard from new parent but accept from old parent
  - After a period of time, discard from old parent but accept from new parent
  - Note that this requires different labels to be advertised to the two parents
  - Note also that there is no way to send a Leaf A-D route to both parents at the same time, as each Leaf A-D route has only one PTA and thus assigns only one label

Next Steps

- Please review and comment
- Material in document is:
  - Essential to MVPN architecture
  - Mature, multiple implementations
- We hope to be able to move relatively quickly to WG adoption and then to WG LC