Label Stacks in Tunnel Encapsulation Attribute

draft-zzhang-idr-tunnel-encapsulation-label-stack

Jeffrey Zhang, Juniper Networks

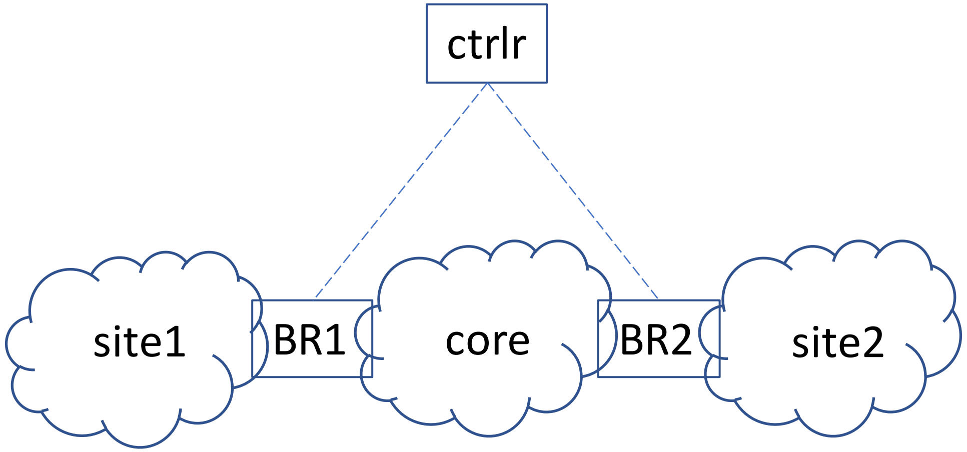
IDR WG, IETF111

MPLS Label Stack sub-TLV in RFC 9012

- "If a packet is to be sent through the tunnel identified in a particular TLV, and if that TLV contains an MPLS Label Stack sub-TLV,   then the label stack appearing in the sub-TLV MUST be pushed onto the packet ***before*** any other labels are pushed onto the packet"
- Why ***before***?
  - Shouldn't we push the tunnel label stack ***after*** pushing service label stack?
- The answer – it is NOT for steering traffic to the tunnel end point
  - Yet this is not clarified in the spec

Intended Use Case

- Traffic steering desired in all sites
  - Routes advertised by the controller
  - Tunnel Encapsulation Attribute received by BRs encodes tunnels in the core
- When BR1 sends traffic to a destination in site2, the label stack for steering in site2 is pre-pushed
  - BR2 may not be able to push a larger stack
  - That label stack is encoded in *MPLS Label Stack* sub-TLV
- How to encode a label stack to steer *to* the tunnel end point?



Tunnel Label Stack

- A new sub-TLV, *Tunnel Label Stack,* is defined to encode the label stack used to steer traffic to the tunnel end point
  - Existing *MPLS Label Stack*, is for steering after the label stack
  - Encoding is the same for the two sub-TLVs (besides the type)

Summary

- A simple small draft that
  - Clarifies the use case for the existing *MPLS Label* ***Stack*** sub-TLV
    - For steering ***after*** tunnel end point
  - Defines a new ***Tunnel Label Stack*** sub-TLV
    - For steering ***to*** the tunnel end point
- Seeking comments and WG adoption