Entropy Labels
with deep label stacks
(e.g. SPRING)
draft-kini-mpls-entropy-label-src-stacked-tunnels-01
IETF 88 (Vancouver) - Nov 3-8, 2013
Sriganesh Kini
Kireeti Kompella
Siva Sivabalan
Background

- RFC 6790 specifies Entropy Label for load balancing
- <TL, ELI, EL> is inserted by ingress LSR
- Transit LSR " ...SHOULD use as much of the label stack as feasible as keys for the load balancing..."
Deep label stacks and load balancing in SPRING

• Ingress LSR pushes several tunnel labels making deep label stacks more prevalent
  o Depth dependent on explicit-route
  o Depth changes at LSRs along the path

• Explicit-route requires load balancing
  o ECMP in the shortest-path hops in an explicit-route
  o LAG in explicit-route
Single EL for entire label stack

EL is deep down the stack
EL per tunnel in the stack

<table>
<thead>
<tr>
<th>TL(n)</th>
<th>ELI</th>
<th>EL</th>
<th>...</th>
<th>TL1</th>
</tr>
</thead>
</table>

Label stack depth is **three** times the number of tunnel labels.
Re-usable EL in a stack

EL is re-used after pop by pushing under next tunnel label
Label operation changes, but end-to-end flow is consistently identified with single EL
EL is at a shallow depth along entire path.
EL at readable stack depths

Ingress LSR determines depth to insert ELs for that LSP via label-depth reading capability advertised (e.g. IGP) by each LSR.

LSRs along the explicit-route until \( T_{n-2} \) are able to read label stack depth until EL.
Other work related to EL

• draft-ietf-mpls-forwarding
• draft-ravisingh-mpls-el-for-seamless-mpls
Questions/comments?

Would like to ask for WG adoption.