MPLS ISD & PSD Processing Analysis

Jie Dong
Design Principles from Hardware’s Perspective

- Minimize the number of headers in packet
  - Better to have one integrated header instead of multiple headers scattered in the packet

- It is more efficient to locate the BoS than a special label at arbitrary position

- Avoid popping the special label and the following data blocks at transit nodes
  - Avoid the special label being exposed as the top label
  - Suggest to put the special label and the data at the BoS

- Avoid adding information into label stack which may change per-packet
  - May have impact on label stack based load balancing

- Header chains are preferred over bitmap based approach
Options of Carrying ADI/AD and some Comparison

<table>
<thead>
<tr>
<th>Option 1</th>
<th>Option 2</th>
<th>Option 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet header overhead</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Ingress node processing complexity</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>Deep of header parsing</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>Cost of finding the ADI</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Cost of parsing ancillary data</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Cost of popping ADI/AD</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Applicability</td>
<td>Suitable for label-swap based MPLS</td>
<td>Reduce the header parsing depth for ISD (if PSD is only for E2E)</td>
</tr>
<tr>
<td>Backward Compatibility</td>
<td>Node popping the last tunnel label need to pop the ADI &amp; ISD, PHP on legacy node is not supported</td>
<td>Intermediate nodes which pop the tunnel label need to pop the ADI &amp; ISD</td>
</tr>
</tbody>
</table>
Considerations

• Can all the types of ancillary data be carried in ISD?
  • If not, we need PSD anyway

• Do we need PSD for HBH data?
  • One example is HBH iOAM
  • Then transit nodes need to be able to parse the PSD

• How much do we plan to change the MPLS label stack?
  • Encoding, parsing, processing...
  • Backward compatibility may become a big concern

• Limitations in scalability and extensibility of MPLS label stack entry
  • If it is known only a subset of the functions can be accommodated in ISD, we may need to revisit whether the changes to MPLS label stack worthwhile?

• Perhaps a better approach is to provide all the new functions with a unified mechanism
Thank You