

Connection-oriented Path in SRv6 Network

draft-du-spring-connection-oriented-srv6-02

Zongpeng Du duzongpeng@chinamobile.com

Peng Liu liupengyiy@chinamobile.com

IETF114

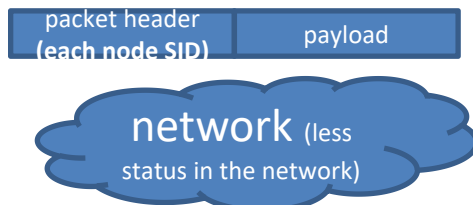
Motivation

- The exact name should be “Connection-oriented, **hop-by-hop switching**” Path in SRv6
 - the “hop-by-hop switching” is just the way that traditional MPLS does on each node, in which the label is locally allocated, and swapped on each node
- As we know, SRv6 packet can contain the whole or partial nodes to do the strict or loose source routing
 - Thus, fewer statuses on each node would be needed
- However, if we need a strict TE path in the network, we need put each node’s SID in every packet header, which is not very convenient

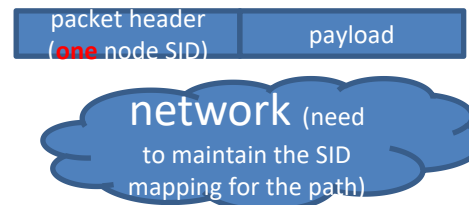
Motivation (cont.)

- For “Connection-oriented” strict TE path in SRv6, we can
 1. put every node’s SID into the packet header
 - if the path is long, the packet header will be large
 - optionally, we can compress the SID list
 2. support both SRv6 and MPLS forwarding in the network, but it is complicated
 3. try to support the “**hop-by-hop switching**” in SRv6 network
- This document talks about the option3
 - based on the uniform SRv6 network programming platform
 - we think the platform can support many network characteristics, and reduce the number of the protocols in the network

Option1:



Option3:



Explanation:

Comparing to option1, option3 is not that scalable, but currently we do not need many “Connection-oriented” paths in the network

New set of SIDs on the data plane

On the data plane, it is easy to replace labels in MPLS with the SIDs in SRv6.
In this document, we call it **End.XCopr** Function, which is supported on each node, and contains an Argument similar to label.

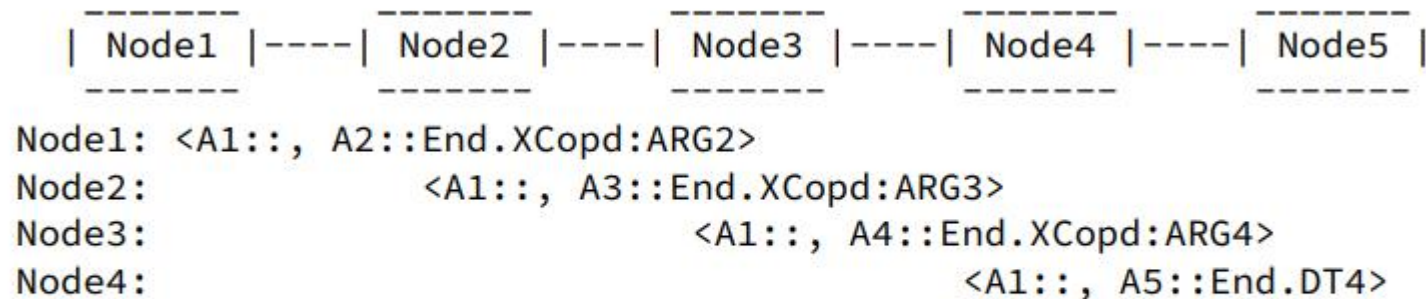


Figure 1: <SA, DA> changes along the Connection-oriented Path on the data plane

- Node2 contains the mapping: in SID A2::End.XCopr:ARG2, out SID A3::End.XCopr:ARG3
- Node3 contains the mapping: in SID A3::End.XCopr:ARG3, out SID A4::End.XCopr:ARG4
- Node4 contains the mapping: in SID A4::End.XCopr:ARG4, out SID A5::End.XCopr:ARG5

New SIDs on the control plane

On the control plane, we have several options

- Option1: a PCE Server can connect to each node, and communicate the “label”
- Option2: to simulate the procedure of RSVP-TE in MPLS by using new SIDs in SRv6

We focus on option 2 here in this document, and we define new **End.Copc** Function, which is supported on each node, and contains an Argument similar to label.

- Label 0000 is special, and is used for confirm the path.

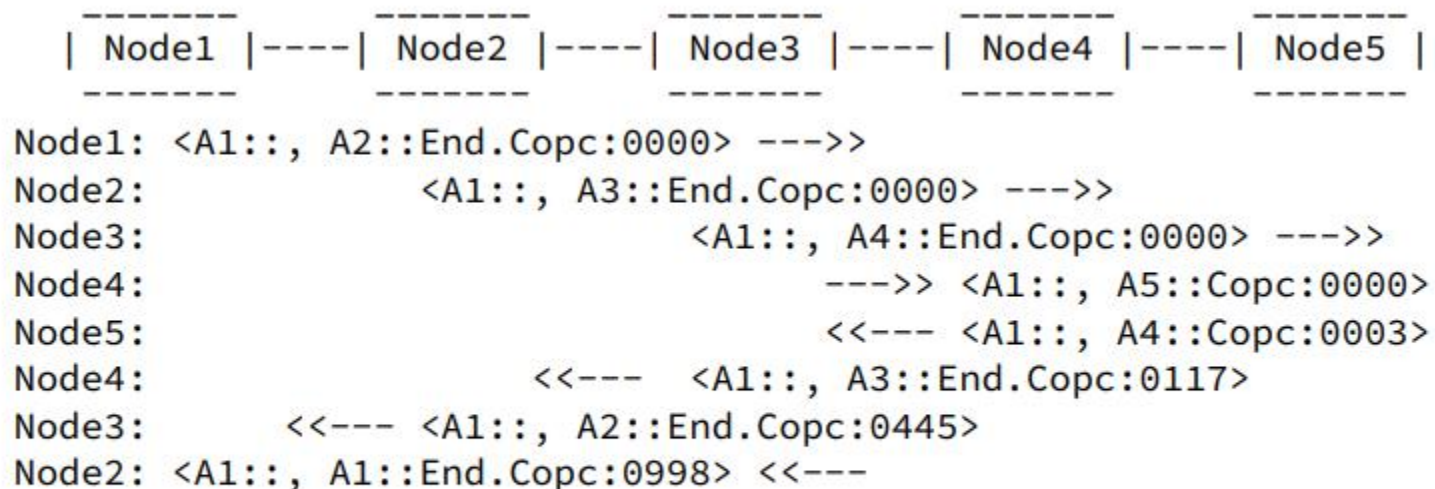


Figure 2: <SA, DA> changes along the Connection-oriented Path on the control plane

Effect of SIDs on the control plane

Steps of **establishing the path**

- Step1: send a packet from the Node1 to Node5, which contains all the node's SIDs
- Step2: each downstream node responses a "label"

The established mapping table

- Node4 : in SID A4::End.XCpd:0117, out SID A5::End.XCpd:0003
- Node3 : in SID A3::End.XCpd:0445, out SID A4::End.XCpd:0117
- Node2 : in SID A2::End.XCpd:0998, out SID A3::End.XCpd:0445

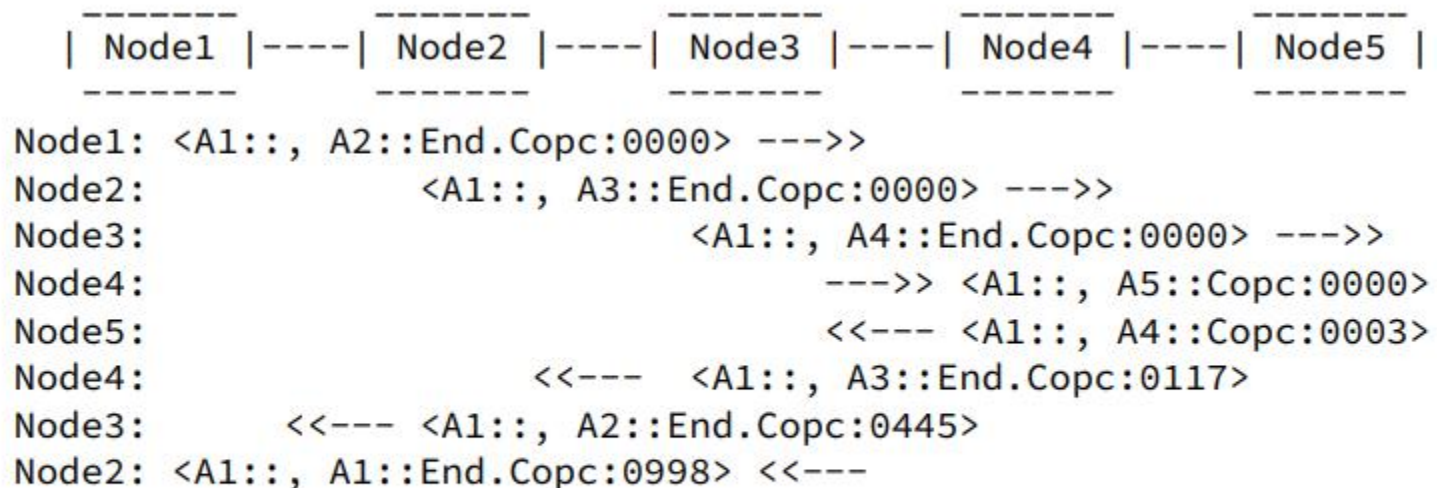


Figure 2: <SA, DA> changes along the Connection-oriented Path on the control plane

Thanks for listening

Welcome for comments &
contributions