IPv6 Support for Segment Routing: SRv6+

draft-bonica-spring-srv6-plus-04

R. Bonica, S. Hegde, Y. Kamite, A. Alston, D. Henriques, J. Halpern, J. Linkova, G. Chen

Juniper Public

Topological Instructions Versus Service Instructions

Juniper Public

Topological Instructions

- Executed on segment ingress node
- Cause a routing action
  - Forward the packet to the segment egress node
- Details
  - Overwrite IPv6 Destination Address with the address of the segment egress node
  - Forward the packet to the segment egress node, either through a specified link or least cost path
- Encoded in IPv6 Routing header (RH)

Juniper Public

Service Instructions

- Executed on segment egress node
- Per-segment service instructions
  - Executed on any segment endpoint
  - Typically do not influence routing
    - Example: expose a packet to a firewall rule
  - Encoded in Destination Option header (DOH) that precedes RH
- Per-path service instructions
  - Executed on final segment endpoint
  - Typically influence demultiplexing and forwarding of packet payload
    - Example: de-encapsulate and forward the payload through VPN interface
  - Encoded in DOH that precedes upper-layer header

Juniper Public

Why Decouple Topological Instructions from Service Instructions

Juniper Public

Using The Most Appropriate IPv6 Extension Header

- RH and topological instructions
  - Both intended to affect routing
  - Both executed on segment ingress node
- DOH preceding RH and per-segment service instruction
  - Both have can have scope beyond routing
    - Example: both can carry an OAM instruction
  - Both executed on any segment egress node
- DOH preceding upper-layer header and per-path service instruction
  - Both have can have scope beyond routing
    - Example: both can influence de-encapsulation and payload forwarding
  - Both executed on the final segment egress node only (i.e., the path egress node)

Simplified Identifier Semantics

- A service instruction identifier (SII) identifies a service instruction
  - Appears in a DOH
  - Not polluted with SID or IPv6 Address semantics
- A SID identifies a segment and the topological instruction that controls it
  - Appears in the RH
  - Not polluted with SII or IPv6 address semantics
- An IPv6 address identifies an interface
  - Appears in IPv6 header
  - Not polluted with SII or SID semantics
- Never copy an identifier of one type into a field that is meant for an identifier of another type

Cost / Benefit Analysis

Juniper Public

Cost

- One more layer of indirection
  - SFIB maps SIDs to IPv6 addresses
  - Required to maintain separation between SIDs and IPv6 addresses
- One more RH type
  - Albeit, simpler
- Two new Destination Options

Benefits

- Simplified RH
  - No need for Tag field
  - No need for TLVs
- SID identifies, but does not contain, an instruction
  - Therefore, the SID can be encoded in relatively few bits
- The RH can be short, even when the SID list that it contains is long
  - Regardless of how strictly and loosely routed segments are interspersed in the SID list
  - Regardless of the network numbering scheme
- No need to augment IPv6 OAM

Juniper Public

Benefits (continued)

- Mix and Match deployment
  - RH with legacy demultiplexing (e.g., RH followed by vxlan header)
  - Least cost routing (no RH) with DOH for demultiplexing
- IPv6 Authentication header can be used to authenticate RH and DH
- Overall simplicity
  - Existing draft cover the subject

Juniper Public

SRH Versus CRH Overhead

SIDs	SRv6 SRH (128-bit SID)	SRv6+ CRH (16-bit SID)	SRv6+ CRH (32-bit SID)
1	24	16	16
2	40	16	16
3	56	16	24
4	72	16	24
5	88	24	32
6	104	24	32
7	120	24	40
8	136	24	40
9	152	32	48
10	168	32	48
11	184	32	56
12	200	32	56
13	216	40	64
14	232	40	64
15	248	40	72
16	264	40	72

Implementation

- LINUX Demo
- JUNOS PoC

Juniper Public

Next Steps

- SPRING WG to adopt draft-bonica-spring-srv6-plus
- 6man WG to adopt
  - Draft-bonica-6man-com-rtg-hdr
  - Draft-bonica-6man-vpn-dest-opt
  - Draft-bonica-6man-seg-end-opt

Juniper Public