SPRING Working Group
Internet Draft
Intended status: Informational
Expires: 05 January 2026

W. Cheng
China Mobile
C. Lin
New H3C Technologies

03 July 2025

A SRv6 Traffic Engineering Application for AI Network
draft-cheng-spring-srv6-for-ai-network-00

Abstract

AI applications require fast processing and responses. Traffic using RoCEv2 has low entropy for ECMP. At the same time, AI elephant flows are predictable. Traffic engineering technology for AI backend networks becomes a possible solution. SRv6 TE can start from the host side, making SRv6 source routing and traffic path control from the host side an optional solution.

This document presents a AI network Traffic Engineering (TE) application scenario for handling link faults and traffic congestion issues in data centers, based on Segment Routing over IPv6 (SRv6) and Compressed Segment Identifier (CSID). The application scenario uses SRv6 CSID Network Programming to directly install all forwarding paths on the head-end device. When a data center experiences a link fault or traffic congestion, the head-end device switches the forwarding path to another optimal path for avoiding the location of link fault or traffic congestion, ensuring optimal AI data flow forwarding.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 05 January 2026.

Table of Contents

1. Introduction

Segment Routing over IPv6 (SRv6) [RFC8402] is the instantiation of Segment Routing (SR) on the IPv6 data plane. Since Traditional SRv6 Traffic Engineering (TE), which require the use of complete 128-bit Segment Identifier (SID) [RFC9602] to define an ordered Segment List for forcing packets to be forwarded along the designated path, has high flexibility and high scalability, but it will lack low packet overhead when a path requires a longer segment list.

As AI resources and services become increasingly rich, AI networks necessitate large-scale, high-bandwidth, and highly reliable features. AI traffic has lower entropy and is primarily composed of elephant flows, which leads to rapid saturation of the link when nodes transmit AI traffic simultaneously. When AI networks employ traditional load balancing techniques (such as ECMP), even with a sufficiently uniform hash algorithm, uneven distribution of low-entropy traffic or link faults can still result in certain links becoming excessively loaded, leading to traffic congestion. And in the event of link failures, whether local or remote, it is necessary to achieve convergence in as short a time as possible to minimize

the impact on network communication. So Data center AI networks require a reliable, flexible, and efficient solution to mitigate the impact of traffic congestion and link faults on communication quality.

From the perspective of source routing, SRv6 TE enables the source node to directly participate in the path selection and planning process. The service traffic within AI networks is highly diverse, with different services having varying requirements for latency, bandwidth, and quality of service (QoS). Leveraging the source routing mechanism, the source node can flexibly determine the forwarding path of packets based on specific service needs, bypassing potentially congested or underperforming links, thereby ensuring efficient transmission for critical services.

AI applications require fast processing and responses. Traffic using RoCEv2 has low entropy for ECMP. At the same time, AI elephant flows are predictable. Traffic engineering technology for AI backend networks becomes a possible solution. SRv6 TE can start from the host side, making SRv6 source routing and traffic path control from the host side an optional solution.

This document presents a SRv6 TE application scenario based on Compressed Segment Identifier (CSID) [RFC9800] to address traffic congestion and link fault issues for AI traffic in data centers. The key idea is to use CSID to design segment list for forwarding paths by SRv6 network path programming [RFC8986], enabling dynamic switchover to the optimal path from the head (source) node to the end node at the head (source) node in the event of traffic congestion or link faults, thus routing traffic around congested or faulty links to alleviate their impact. This application scenario employs CSID, resulting in lower packet overhead, higher flexibility, and scalability.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Application Scenario

This section introduces a SRv6 TE application scenario based on NEXT-CSID flavor [RFC9800] for AI network to mitigate the impact of traffic congestion and link faults on communication quality.

The comprehensive solution of this application scenario builds upon traditional SRv6 TE methods by employing CSID for packet encapsulation and forwarding. CSID compresses the 128-bit SID

[RFC9602] into a shorter SID, such as 16-bit or 32-bit. Multiple CSIDs can be concatenated into a compact list and embedded in the remaining space of a single IPv6 address. When using 16-bit segment CSID for a 32-bit locator block, a single IPv6 address can easily encode a deterministic path with a depth of up to 6 hops.

The topology of the application scenario is shown in Figure 1, which includes a controller and multiple network nodes. The controller collects the status of the entire data center network, such as topology, bandwidth, and latency, and calculates the optimal SRv6 CSID path through algorithms, such as Dijkstra and Path Computation Element (PCE) [RFC4655]. The controller issues SRv6 CSID policies to the head node or modify the CSID sequence list through control management protocols (such as NETCONF [RFC6241], BGP-LS [RFC9514]) to dynamically adjust the packet forwarding path. The SRv6 Policy, including multiple feasible paths, is installed in the head node. Each network node has a CSID and can identify the CSID and perform shift forwarding operations.

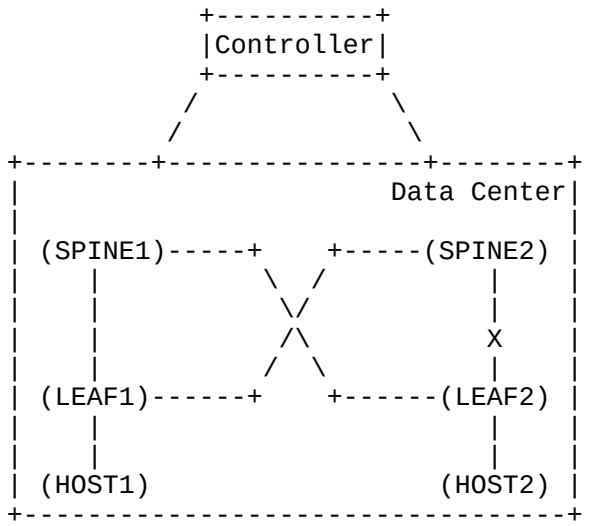


Figure 1: Typical Topology

When congestion or a fault occurs in the application scenario, such as the location between SPINE2 and LEAF2, the procedure is as follows:

* The congested or faulty node (LEAF2) advertises this to the controller to recalculate the optimal CSID path and issue it to the head node (HOST1), or the head node (HOST1) perceives congestion and faults itself through probe packets or responding ACKs, and reselect the optimal CSID path.

* Based on SRv6 Policies that include CSID paths, traffic packets are rerouted to the most optimal forwarding path at the head node (HOST1), avoiding congested or faulty location.

Of course, the prerequisite of this application solution is that there MUST be multiple feasible paths from the head node (HOST1) to the end node (HOST2).

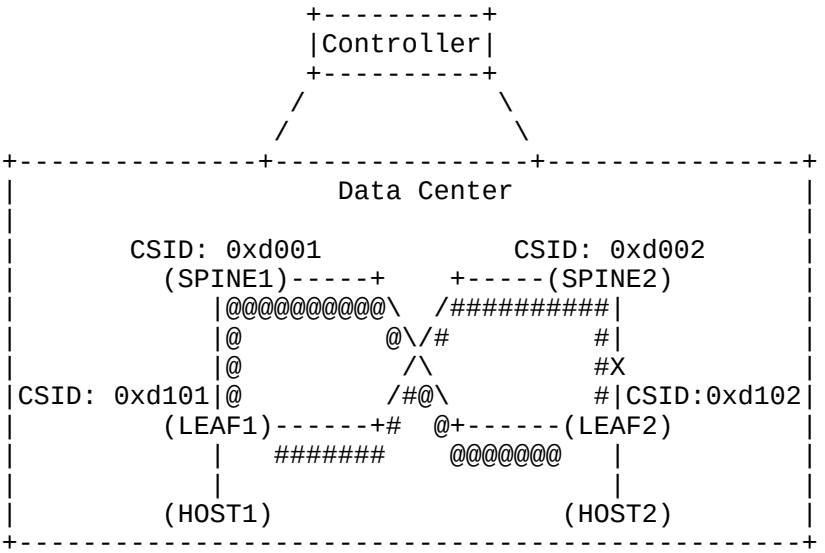
This document assumes that congestion and failures can be probed by the head node or controller, and how to probe congestion and failures is beyond the scope of this document.

3. Illustration

This section provides an illustration of the SRv6 TE application scenario based on NEXT-CSID flavor. The example topology is depicted in Figure 2.

All network nodes in this topology use a global 32-bit Locator-Block, which is 2001:db8::/32. All network nodes use a 16-bit CSID, where the CSID of node LEAF1 is 0xd101, the CSID of node LEAF2 is 0xd102, the CSID of node SPINE1 is 0xd001, and the CSID of node SPINE2 is 0xd002.

The controller, based on global topology information, calculates the optimal path from HOST1 to HOST2 as LEAF1->SPINE2->LEAF2 and issues this to the head node HOST1. Since this path requires passing through three nodes, when HOST1 sends packets, it only needs to set the IPv6 destination address of the packets to 2001:db8:d101:d002:d102::/48.



'X': the location of the traffic congestion or link fault

'#': the optimal path before traffic congestion or link fault occurs
'@': the optimal path after traffic congestion or link fault occurs

Figure 2: Example Topology

When a traffic congestion or link fault occurs between nodes SPINE2 and LEAF2, the controller recalculates the optimal path from HOST1 to HOST2 as LEAF1->SPINE1->LEAF2 when the LEAF2 advertises the traffic congestion or link fault to the controller. The controller issues an update to the head node HOST1, instructing it to set the IPv6 destination address of transmitted packets to 2001:db8:d101:d001:d102::/48, thereby bypassing the location of the traffic congestion or link fault.

4. Operational Considerations

The operation of this application scenario is consistent with [RFC8986] and [RFC9800]. All network nodes related to this application scenario MUST support CSID and execute shift forwarding operation of CSID.

5. IANA Considerations

This document has no IANA actions.

6. Security Considerations

This document does not introduce additional security considerations.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986,DOI 10.17487/RFC8986, February 2021,<https://www.rfc-editor.org/info/rfc8986>.

[RFC9800] Cheng, W., Filsfils, C., Li, Z., Decraene, B., and F. Clad, "Compressed SRv6 Segment List Encoding (CSID)", RFC 9800, DOI 10.17487/RFC9800, July 2025, < https://www.rfc-editor.org/info/rfc9800>.

7.2. Informative References

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <https://www.rfc-editor.org/info/rfc4655>.

[RFC9514] Dawra, G., Filsfils, C., Talaulikar, K., Ed., Chen, M., Bernier, D., and B. Decraene, "Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing over IPv6 (SRv6)", RFC 9514, DOI 10.17487/RFC9514, December 2023, <https://www.rfc-editor.org/info/rfc9514>.

[RFC9602] Krishnan, S., "Segment Routing over IPv6 (SRv6) Segment Identifiers in the IPv6 Addressing Architecture", RFC 9602, DOI 10.17487/RFC9602, October 2024, <https://www.rfc-editor.org/info/rfc9602>.

Authors' Addresses

Weiqiang Cheng
China Mobile
Beijing
China
Email: chengweiqiang@chinamobile.com

Changwang Lin
New H3C Technologies
Beijing
China
Email: linchangwang.04414@h3c.com