SRv6 Operations
Internet-Draft
Intended status: Informational
Expires: 5 December 2026

C. Filsfils
P. Camarillo, Ed.
K. Michielsen
Cisco Systems
A. Gorovoy
Nebius
N. Leymann
Deutsche Telekom

3 June 2026

SRv6 End-to-End DC Frontend and WAN
draft-filsfils-srv6ops-srv6-e2e-dc-frontend-wan-02

Abstract

The SRv6 Network Programming architecture allows an application to control the end-to-end journey of traffic flows through different network domains in a unified and stateless manner, without requiring intermediate network devices to maintain per-flow information.

This document covers its application to the integration of data center (DC) and wide area network (WAN) architectures using SRv6 with uSID (NEXT-CSID). It describes a unified IPv6 data plane from tenant workloads through the DC frontend to Internet peering, replacing designs that stitch separate overlay protocols at the Data Center Interconnect (DCI). The solution enhances scalability and enables flexible stateless service insertion by unifying underlay, overlay, and service steering under SRv6.

Table of Contents

1. Introduction

Traditionally, Data Centers (DCs) and Wide Area Networks (WANs) are designed and operated as independent network domains, each with its own architecture. This separation introduces operational complexity and makes it challenging to seamlessly integrate and deploy network services, such as security appliances or load balancers, across the end-to-end path.

Segment Routing [RFC8402] over IPv6 (SRv6) [RFC8986] enables a unified approach to networking by leveraging source routing.

SRv6 uSID (NEXT-CSID) [RFC9800] provides:

* *Stateless Traffic Engineering*: Any path in the fabric can be enforced by simply encoding a sequence of SRv6 instructions in the packet header. There is *no per-flow state in the fabric* (unlike MPLS RSVP-TE).

* *Stateless Service Insertion*: Any service (e.g., Firewall, IDS) may be inserted in the packet delivery path. This is achieved simply by adding the corresponding segment to the packet. *There is no per-flow state required on the services*.

* *Integrated Overlay*: SRv6 provides per-tenant segmentation, ensuring flows from different tenants are isolated while sharing the same physical infrastructure. This capability may be combined with stateless traffic engineering and stateless service insertion, *without requiring any additional overlay encapsulations*.

* *End-to-end, with minimum overhead*: A simple *IPv6 encapsulation suffices to encode up to six uSIDs in the Destination Address*. These instructions encode a path end-to-end across domains in a provider network. If more instructions are required, a simple Segment Routing extension Header (SRH) [RFC8754] can be added.

This document presents an SRv6 end-to-end solution for integrating the DC frontend with the WAN, eliminating the complexity and inefficiencies of traditional fragmented architectures. Section 1.2 defines the scope of this document and its relationship to the companion AI backend document.

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.

1.2. Scope and Document Relationship

This document addresses the *DC frontend*: the tenant-facing portion of a data center that connects workloads to external destinations through a WAN and the Internet. Typical elements include Top-of-Rack (TOR) switches, leaf and spine nodes, optional service leaf (SL) nodes for security appliances, a DCI node at the DC edge, WAN provider (P) routers, and Border Routers (BR) toward the Internet.

In production-oriented designs, a *cloud gateway (CGW)* on the host or hypervisor often performs SRv6 encapsulation and L3VPN PE functions on behalf of virtual workloads, while the TOR, leaf, spine, and DCI provide IPv6 locator forwarding only. The illustrations in this document use a TOR-based PE for clarity; the same SRv6 mechanisms apply when the PE role is implemented on a CGW. The legacy baseline is a fragmented stack: VXLAN EVPN in the DC (often between the CGW and the DCI) and L3VPN over SR-MPLS in the WAN, with protocol stitching at the DCI.

Operational experience from a large-scale cloud deployment is summarized in Section 6.

This document does *not* cover the *AI backend* fabric: the high-radix, often multi-plane Clos network that interconnects GPUs and AI accelerators for training and inference. That environment is described in [I-D.filsfils-srv6ops-srv6-ai-backend], which documents deterministic path placement, multipath spraying, and transport integration in hyperscale AI clusters. Together, the two documents describe a consistent SRv6 operational model: the AI backend draft for east-west GPU traffic inside the training fabric, and this draft for north-south and DC-to-Internet connectivity through the frontend and WAN.

Both documents are informational use-case descriptions within the SRv6 Operations (srv6ops) work. They assume a single administrative domain or closely coordinated domains where the operator controls locator allocation, IGP/BGP design, and SR Policy provisioning end to end.

2. Terminology

SRv6 Segment Routing over IPv6 [RFC8986].

uSID Micro-segment Identifier, formally defined as NEXT-CSID in [RFC9800].

The term _uSID (micro SID)_ predates the formal naming and has been widely adopted across the industry - including operators with large-scale deployments, vendors, open-source implementations, and used consistently in multi-vendor interoperability reports.

To maintain alignment with the formal specification while also acknowledging the widespread and practical use of the term, this document uses uSID and NEXT-CSID interchangeably.

ECMP Equal-Cost Multi-Path

uN uSID associated with a Node, the short notation for the End behavior with NEXT-CSID, PSP, and USD flavors as defined in [RFC9800].

uA uSID associated with an Adjacency, the short notation for the End.X behavior with NEXT-CSID, PSP, and USD flavors as defined in [RFC9800].

DC frontend Tenant-facing DC edge: TOR, leaf, spine, optional SL, DCI, and their connectivity toward the WAN and Internet.

AI backend GPU or accelerator fabric inside the DC, as described in [I-D.filsfils-srv6ops-srv6-ai-backend].

VXLAN Virtual Extensible LAN [RFC7348]

DCI Data Center Interconnect, interconnecting DC and WAN domains

PE Provider Edge router

TOR Top-of-Rack router

BR Border Router, interconnecting WAN domain and the Internet

CGW Cloud Gateway: virtual or host-based router that connects tenant workloads to the DC fabric and may act as the SRv6 L3VPN PE toward the WAN

3. Initial Fragmented Network Architecture

3.1. Overview

In a common baseline (the *legacy* design), the WAN core runs L3VPN over SR-MPLS while the DC frontend runs an EVPN/VXLAN overlay between workloads, a cloud gateway, and the DCI. The DCI performs *stitching*: it terminates EVPN/VXLAN from the DC and attaches traffic to L3VPN/SR-MPLS toward provider edge routers and the BR. This split increases operational overhead and limits traffic engineering and service chaining across the full path.

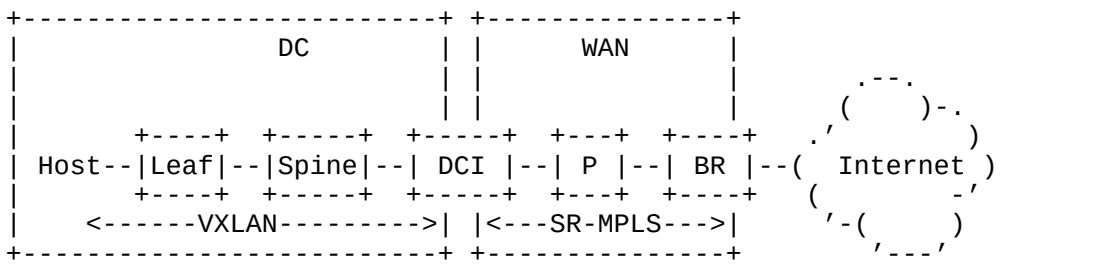

Figure 1: Traditional data center architecture

This is depicted in Figure 1. A workload on the host sends traffic toward the Internet; VXLAN encapsulation starts on the hypervisor (cloud gateway). Traffic is carried in the DC overlay (VXLAN EVPN) to the DCI. At the DCI, the packet is translated from the DC overlay into the WAN VPN (SR-MPLS). The BR ultimately provides Internet reachability after MPLS/SR processing in the WAN.

3.2. Challenges with a Fragmented DC/WAN

This design presents the following challenges:

* *DCI Protocol Translation*:
  - Performance Penalty: Some hardware incurs a performance degradation.
  - Availability: Not all vendors support the functionality homogeneously.
  - Scalability bottleneck: These boxes maintain VRF state, reducing scalability.
  - Complexity: Information like Group Policy Objects (GPO), tags, or context must be conveyed across and between the DCs.
  - Reliability: The increased complexity introduces additional potential points of failure.
  - Cost: The increased functionality typically incurs an additional cost.

* *Limited traffic engineering end to end*: SR-MPLS in the WAN does not extend into the DC frontend, and VXLAN offers no traffic-engineering capabilities there. There is no uniform source-routed path from the workload across DC and WAN.

* *Service Insertion*: Service chaining with VXLAN often requires rigid designs (policy-based routing, default gateways, or VRF/VLAN hand-offs). Firewalls are typically deployed as large clusters near the perimeter; as clusters grow, state synchronization across instances leads to significant state explosion and operational cost.

* *IPv4 Loopback dependency*: Many vendors still require IPv4 loopbacks on both VXLAN and SR-MPLS networks, adding unnecessary operational complexity and limiting pure IPv6 deployments.

4. End-to-End SRv6 across DC Frontend and WAN

The transition from a fragmented network architecture to an end-to-end SRv6 uSID design enables seamless communication between data center (DC) workloads and the wide area network (WAN).

The key benefits of this architecture include:

* *Elimination of protocol translation*: Removing EVPN/VXLAN-to-SR-MPLS stitching at the DCI improves efficiency and reduces processing overhead.

* *Enhanced scalability*: The nature of SRv6 supports flexible traffic engineering with no per-flow state in the fabric.

* *Simplified service chaining*: Native SRv6 capabilities enable the efficient insertion of services without requiring additional protocols or complex policy-based routing policies.

* *Optimized forwarding with minimum MTU*: A simple IPv6 encapsulation allows encoding up to six instructions in the IPv6 Destination Address. If additional instructions are needed, a Segment Routing extension Header can be added to encode additional instructions.

* *IPv6-native*: The architecture enables a fully IPv6-native deployment, eliminating dependencies on IPv4 loopbacks while simplifying management. Transitioning away from fragmented IPv4 dependencies eases uniform addressing across the entire network fabric.

4.1. DC Integration Model

End-to-end DC/WAN integration is clearer when examining the layers are separately; this is *unified* design in contrast to the legacy design in Section 3):

* *Underlay*: A single IPv6 topology across leaf, spine, DCI, and WAN provider routers. Nodes forward on summarized locator prefixes; only PE nodes and service nodes terminate SRv6 behaviors.

* *Overlay*: SRv6 L3VPN services on DC and WAN edge PEs (CGW or TOR, and BR). VRFs, route targets, and End.DT service SIDs provide tenant isolation and Internet/VPN reachability without a separate VXLAN or MPLS VPN encapsulation stitched at the DCI.

* *Services*: Optional service insertion via uA SIDs on an SL and SR Policy / Color steering on the PE headends. Firewalls may be SRv6-unaware NFVs on hosts, connected to an SRv6-capable SL.

Roles: The *PE* at the DC edge (CGW or TOR) originates the SRv6 network program toward the BR PE. The *spine* and *DCI* are transit routers in the locator underlay (no overlay translation). The *BR* is the WAN/Internet PE. This model removes the DCI stitching function required in the legacy VXLAN-EVPN plus SR-MPLS design.

4.2. Service Insertion

Service insertion typically relies on complex policy-based routing mechanisms that direct traffic through predefined service clusters, such as firewalls. These approaches often lead to scalability limitations and suboptimal traffic paths. By using SRv6 services can be seamlessly integrated into the forwarding plane without requiring additional tunneling mechanisms or extensive policy configurations.

Services are inserted into the forwarding path for a specific flow simply by adding the corresponding uSID to the source routed network program. This allows:

* *Stateless steering*: SRv6 uSID (NEXT-CSID) encoding removes the need for per-flow state on intermediate nodes, improving scalability.

* *Flexible service chaining*: Multiple services, such as firewalls, load-balancers, intrusion detection (IDS), or deep packet inspection (DPI), can be chained together dynamically by appending additional uSIDs.

Using the firewall service as an example, when a workload within the DC initiates traffic requiring inspection, the ingress node includes the necessary SIDs in the SRv6 network program. These SIDs guide the packet through the firewall, ensuring compliance with security policies. The firewall processes the traffic and forwards it along the next segment in the chain, with minimal overhead.

4.3. Unified SRv6 Architecture

Figure 2 depicts the unified end-to-end architecture. A single SRv6/ uSID domain spans the DC frontend and the WAN. A host-based *cloud gateway (CGW)* or a TOR and the BR act as SRv6 PE nodes; leaf, spine, DCI, and P routers forward IPv6 on locator prefixes only.

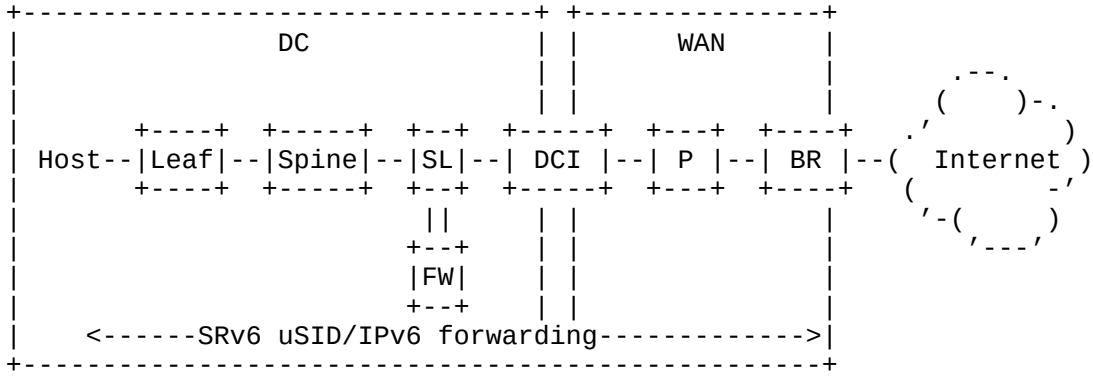

Figure 2: Unified SRv6 DC frontend and WAN architecture

Workloads attach to hypervisors or TOR ports. The CGW (or TOR) encapsulates traffic with an SRv6 network program (uSID container and/or SRH) toward a remote PE SID, an intermediate service SID, or both. Intermediate nodes forward on the longest-match locator prefix. No EVPN/VXLAN-to-SR-MPLS stitching is required at the DCI. Services (e.g., Firewall service) may be inserted into the forwarding path dynamically.

4.4. Role of the DCI in SRv6 Mode

In the fragmented design (Section 3), the DCI often implements a *protocol gateway*: it terminates EVPN/VXLAN from the DC and inserts traffic into L3VPN/SR-MPLS toward the WAN. In the unified SRv6 design, the DCI is an ordinary *IPv6 router* in the locator topology, alongside the spine. It participates in the same SRv6 underlay, with no protocol boundary in between. It does not maintain per-tenant VXLAN VNIs or perform MPLS label imposition on behalf of the DC overlay.

Functions that may remain at the DC edge, on the DCI, CGW, TOR, or BR, are orthogonal to segment routing: Internet peering policy, NAT, DDoS mitigation, and L3VPN route reflection. SRv6 L3VPN termination and service insertion are anchored on the DC-edge PE (CGW or TOR) and on the BR; the spine and DCI only contribute to locator reachability.

4.5. Control Plane Overview

The control plane ties locator advertisement, L3VPN services, and traffic steering together:

* *Locator allocation*: The operator allocates an SRv6 SID space once (for example 5f00:0::/32 or fc00:0::/32) and assigns /48 locators per SRv6-enabled node (CGW or TOR, SL, BR). Hierarchical addressing allows summarization at spine, DCI, and WAN boundaries, keeping the IGP small and avoiding later renumbering when extending SRv6 across domains. Locator reachability is advertised in the DC (typically via BGP) and in the WAN (typically via IS-IS or BGP). Operators SHOULD set the SRv6 Locator attribute consistently at redistribution boundaries, as specified in [RFC9800].

* *SRv6 L3VPN*: PE nodes (CGW or TOR, and BR) use BGP L3VPN with SRv6 service SIDs (End.DT4, End.DT6, End.DT46) per [RFC9252] and [RFC8986].

* *SR Policy and Color steering*: For traffic that must traverse a service (firewall) or a specific path, the operator provisions SR Policies on headends [RFC9256]. BGP Color Extended Communities [RFC9012] signal the mapping of routes to SR Policies, as specified in [RFC9256]. Color-Only BGP Destination Steering can be used to reduce the number of SR Policies.

* *Redundant BRs*: Multiple BRs may advertise the same anycast locator for Internet-facing service SIDs depending on the operator’s redundancy model. DC-edge PEs (CGW or TOR) select the appropriate import path and SR Policy per VRF and color.

Section 5 provides concrete forwarding examples.

4.6. Egress Peer Engineering

Egress Peer Engineering (EPE) is the practice of steering outbound traffic toward specific external peers, links, or link groups at the BR. Therefore, the BR allocates SRv6 Peering SIDs [RFC9087]. The ingress headend then imposes a segment list whose final segment is the chosen peering SID, instructing the BR to forward the packet out the selected external interface to the selected peer. Because the

egress choice is carried in the segment list, the WAN core remains stateless: P routers forward on locator prefixes only.

5. Illustration

In this section we illustrate how an end-to-end SRv6 converged DC/WAN architecture operates.

5.1. Unsecured Traffic Flow

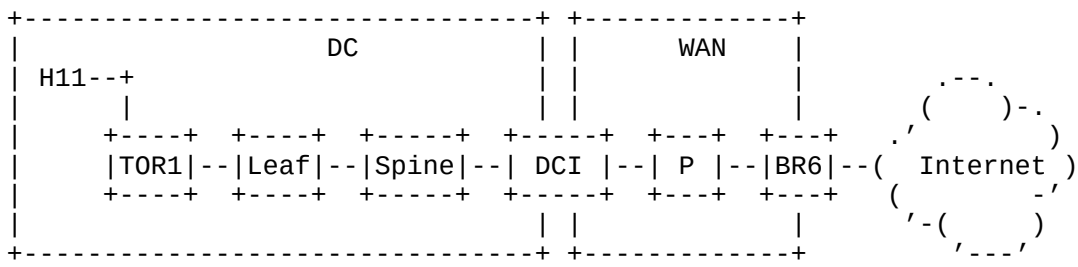

Figure 3: Reference network topology

Figure 3 illustrates one DC and the WAN. The network within the DC has a spine-leaf Clos design. BGP is used as the routing protocol in the DC. The WAN network uses IS-IS as the IGP. Other routing protocols may be used in the DC and the WAN. The Border Router (BR) connects the WAN to the Internet.

SRv6 is enabled on nodes TOR1 and BR6 in this example. SRv6 endpoint behavior is not required on leaf, spine, and DCI, since they only perform IPv6 locator forwarding. In a CGW-based deployment, only the CGW and BR6 require SRv6 endpoint behavior; leaf, spine, and DCI perform IPv6 locator forwarding only.

The operator provisions the following:

* SRv6 SID Space in the fabric 5f00:0::/32

* TOR1 locator prefix: 5f00:0:1::/48. This prefix is advertised in the network.

* BR6 locator prefix: 5f00:0:6::/48. This prefix is advertised in the network.

Host H11 represents a virtual workload. The TOR1 (or an equivalent cloud gateway on the host) and BR6 act as SRv6 Provider Edge (PE) nodes, providing SRv6 L3VPN connectivity between the workload and the Internet. The SRv6 PEs advertise their VPN routes in BGP [RFC9252].

As an L3VPN PE, BR6 advertises its IPv4 and IPv6 Internet reachability in its VRF INTERNET to the TORs using BGP. The BR typically advertises its Internet reachability as a default route. A single BR is illustrated, but multiple BRs may exist for redundancy.

As an SRv6 L3VPN PE, BR6 advertises the VPN routes with an SRv6 L3VPN service SID of behavior End.DT4 for IPv4, End.DT6 for IPv6, or End.DT46 for both IPv4 and IPv6 [RFC8986]. BR6 advertises the SRv6 L3VPN service SID 5f00:0:6:e000:: for the Internet routes, with 5f00:0:6::/48 an SRv6 locator of BR6.

TOR1 is configured with a VRF named UNSECURED for unsecured connectivity. Host H11 is connected to TOR1 in VRF UNSECURED. TOR1 imports the L3VPN routes advertised by BR6 into this VRF.

TOR1 advertises the IPv4 and IPv6 service routes in VRF UNSECURED with an SRv6 L3VPN service SID 5f00:0:1:e001::. BR6 imports the routes of this VRF into its single local VRF INTERNET.

TOR1 steers a traffic flow from H11 in VRF UNSECURED to the Internet via BR6. As a result:

* TOR1: encapsulates packets with IPv6 Destination Address: 5f00:0:6:e000::

* Leaf, Spine, DCI, P: perform IPv6 routing based on the prefix 5f00:0:6::/48

* BR6: receives the packet with IPv6 DA 5f00:0:6:e000::. This matches a FIB entry locally instantiated as an SRv6 SID associated with the End.DT behavior. As a result, it decapsulates the packet and forwards the exposed inner packet to the internet.

The reverse direction is symmetrical.

SRv6 encapsulation and PE functions may be implemented in either of two ways: on a *cloud gateway* on the host (hypervisor), with the physical fabric forwarding on IPv6 locators only, or on a *TOR* switch as illustrated in Figure 3. Both options use the same SRv6 L3VPN and locator-forwarding mechanisms.

5.2. Secured Traffic Flow

This section describes the scenario of an SRv6-unaware [I-D.ietf-spring-sr-service-programming] standalone firewall service. SR Policies [RFC9256] steer the packet flows requiring inspection through the firewall.

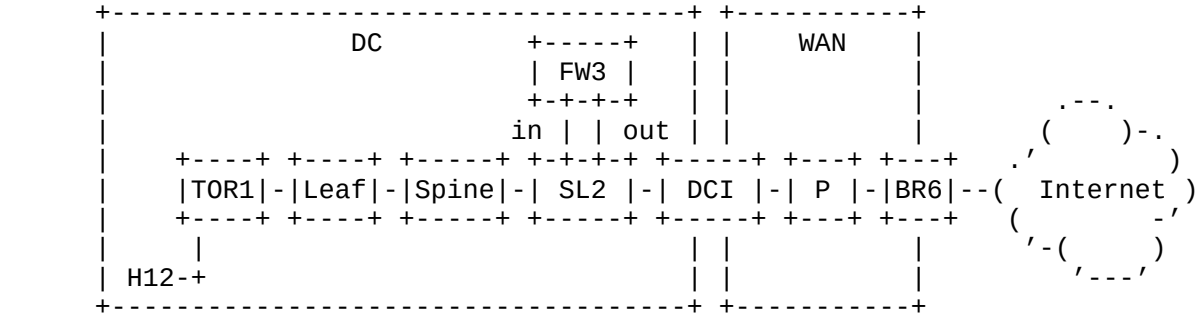

Figure 4: Reference network topology with standalone firewall service

Figure 4 illustrates the network topology with a firewall service FW3. This firewall service is connected to service leaf (SL) node SL2. The firewall FW3 is dual-connected to the SL node via an inside interface (FW3-IN) and outside interface (FW3-OUT).

FW3 inspects the received packets and forwards the allowed ones. FW3 inspects the inner packet and does not modify the outer IPv6 header, apart from the HL field.

The SL node SL2 is SRv6-enabled and advertises a locator prefix 5f00:0:2::/48.

SL2 allocates and installs a uA SID (5f00:0:2:e000::) directing packets to the inside interface of the firewall (FW3-IN) and a uA SID (5f00:0:2:e001::) directing packets to the outside interface of the firewall (FW3-OUT).

Traffic from the Host to the Internet:

TOR1 encapsulates traffic from H12 with IPv6 DA 5f00:0:2:e000:6:e000:: (a uSID container). The network program sends the packet to SL2, which executes its local uA 5f00:0:2:e000::, forwarding the packet towards the inside interface of FW3 (FW3-IN). FW3 inspects the inner packet and returns the whole packet to SL2, which then forwards it to BR6 via End.DT service SID 5f00:0:6:e000::. BR6 decapsulates and sends the inner packet to the Internet. Leaf, spine, DCI, and P forward on locator prefixes 5f00:0:2::/48 and 5f00:0:6::/48 without SRv6 endpoint processing.

The reverse direction is symmetrical: BR6 applies a segment list to first steer the packet towards the outside interface of FW3 (FW3-OUT), using the uA 5f00:0:2:e001:: of SL2. FW3 inspects the inner packet and returns the whole packet to SL2, which then forwards it to the TOR via the service SID.

This configuration ensures that forward and reverse traffic flows pass through the same firewall.

Note that in a typical deployment the service (e.g, Firewall) may be clustered. This is grouping multiple service instances as a single logical device. The connections are synchronized across the instances, which ensures redundancy. SRv6 uSID supports this easily by leveraging an anycast service SID. This is detailed in Section "Cluster Firewall Service".

In Section "SRv6 Service Considerations" we remind the difference between SRv6-aware and SRv6-unaware services, and provide considerations for each of them.

6. Deployment Experience

This architecture has been applied in a large-scale GPU cloud with an IPv6-only DC frontend, a WAN built with two different network vendors, and in-house cloud gateway and firewall NFVs on hosts. SRv6 endpoint behavior is limited to the cloud gateway and BR; the physical leaf, spine, and DCI forward on IPv6 locators only. An open-source cloud gateway implementation based on VPP is used for SRv6 PE functions on the host.

Further operational detail is available in the Nebius presentation at MPLS & SRv6 World Congress, Paris, March 2026: https://www.segment-routing.net/conferences/Paris26-Nebius-Alexey-Gorovoy/.

A related operator architecture discussion was presented by Deutsche Telekom at the same congress. That work is at an early planning stage and explores similar unified SRv6 integration themes across DC and WAN domains: https://www.segment-routing.net/conferences/Paris26-Deutsche-Telekom-Nicolai-Leymann/.

7. Migration and Brownfield Integration

The architecture in this document can be introduced incrementally into existing SR-MPLS deployments without a full network redesign, service interruption, or immediate hardware refresh across the entire infrastructure. Section 7.1 describes the end-to-end migration path from the legacy fragmented design in Section 3. Section 7.2 describes the control-plane steps that enable SRv6 to coexist with existing SR-MPLS and IPv4 services in the WAN and data center.

7.1. End-to-End Migration Path

Operators typically migrate from a fragmented EVPN/VXLAN DC plus SR-MPLS WAN design in phases, keeping production traffic on the legacy path until the SRv6 frontend is validated. Deployments use coexistence of legacy and SRv6 stacks rather than a single cutover.

* *Phase 1 — Locator blueprint*: Allocate the SRv6 SID space once (for example /32) with hierarchical /48 locators per future PE, SL, and BR. Advertise locators and verify IPv6 reachability end to end while the workload overlay (VXLAN EVPN) may still be in use. Complete the data center enablement steps in Section 7.2.2 as part of this phase.

* *Phase 2 — Parallel WAN and pilot PEs*: Run SR-MPLS and SRv6 in parallel in the WAN using the dual-stack integration steps in Section 7.2.1. Enable SRv6 L3VPN on the cloud gateway (or TOR) and BR for pilot VRFs while legacy VPNs continue to serve production tenants. Use the dual PE practices in Section 7.2.3 to manage coexistence.

* *Interim stitching (optional)*: Where a domain cannot move in one step, transitional gateways may stitch VXLAN EVPN to SRv6 at the DC edge, or SRv6 to SR-MPLS toward legacy WAN PEs, until locators and VPN routes are ready on both sides. These gateways are retired as Phase 3 completes.

* *Phase 3 — Per-VRF cutover and DCI simplification*: Migrate tenants VRF by VRF from EVPN/VXLAN and SR-MPLS to SRv6 L3VPN on the CGW/TOR and BR. Once locators are reachable through the DCI as plain IPv6 routes, remove EVPN/VXLAN-to-SR-MPLS translation on the DCI; the DCI and spine remain IPv6 transit only.

* *Phase 4 — Services and policies*: Move firewall and other service insertion to SRv6 uA and SR Policy. Decommission rigid PBR or perimeter-only clusters where SRv6 provides equivalent policy.

7.2. Brownfield Control-Plane Integration

The following subsections apply when SRv6 is introduced alongside an existing SR-MPLS WAN and an IPv4-based data center control plane. Only PE nodes and service nodes require SRv6 endpoint behavior; intermediate nodes continue to provide IPv6 locator forwarding as described in Sections 4.3 and 5.1. Locator allocation, advertisement across routing domains, and redistribution semantics are covered in Section 4.5.

7.2.1. SR Dual-Stack Integration in the WAN

In a brownfield WAN deployment, SRv6 can be introduced alongside an existing SR-MPLS infrastructure by enabling IPv6 routing while maintaining existing IPv4-based forwarding and MPLS services. Assuming the existing deployment uses an IPv4-based IGP with MPLS and Segment Routing enabled, integration of SRv6 typically requires:

* Enable IPv6 on all WAN point-to-point interfaces.

* Enable IPv6 address-family routing within the existing IGP instance.

* Enable SRv6 capabilities within the IGP for the IPv6 address family.

SRv6 endpoint behavior can initially be limited to edge nodes participating in SRv6 services, while provider routers in the WAN core continue to forward on locator prefixes only. This phased approach simplifies integration into environments where some platforms may not yet support SRv6 endpoint processing in hardware.

7.2.2. Dual-Stack Integration in the Data Center

When BGP is used as the data center fabric routing protocol, introduction of SRv6 requires IPv6 reachability within the fabric in parallel with the existing IPv4 control plane. Typical steps include:

* Enable IPv6 on all point-to-point fabric links.

* Establish BGP peering sessions over IPv6 transport addresses in parallel with existing IPv4-based sessions, as required by the design.

This allows the existing IPv4 control plane and services to remain operational while SRv6 locator prefixes and VPN routes are introduced.

7.2.3. Dual PE Operation

A migration period may require PE routers to simultaneously support SR-MPLS and SRv6 VPN services. A dual-capable PE can originate, receive, and process VPN routes carrying SRv6 service SIDs with IPv6 next hops, and VPN routes carrying MPLS service labels with IPv4 next hops. This enables SR-MPLS and SRv6 services to coexist and provides a straightforward migration path between the two technologies.

Operationally, it is RECOMMENDED that SRv6 VPN routes and MPLS VPN routes be originated by separate route-reflector infrastructures or separate BGP sessions. For example, an SRv6 VPN route reflector may advertise SRv6 service routes while an MPLS VPN route reflector continues to advertise MPLS-based VPN routes.

Where both route types are present, BGP policy SHOULD control route preference during different migration phases. The BGP LOCAL_PREF attribute can prefer either SRv6 VPN routes or MPLS VPN routes until cutover is complete for a given VRF or tenant.

8. Security Considerations

This document is informational and does not define a new protocol. Baseline security for SRv6 data plane and segment routing is specified in [RFC8986].

Source routing trust: In the deployment model described here, DC-edge PE nodes (cloud gateway or TOR) and BR PE nodes originate SRv6 encapsulation on behalf of workloads. A compromised headend could encode segment lists that bypass policy (for example, skip a firewall SID). Operators SHOULD restrict which devices may originate SRv6 programs and SHOULD validate that BGP Color and SR Policy bindings match the intended service topology.

Tenant isolation: Per-tenant separation relies on L3VPN VRFs, route targets, and distinct service SIDs on PE nodes, in addition to locator forwarding in the underlay. Misconfiguration of import/export or SID assignment could leak routes between tenants; standard BGP VPN operational controls apply.

Service bypass: If Color communities or SR Policies are inconsistent between the DC-edge PE and BR, traffic may reach the Internet without passing through a required firewall. Operational change control on BGP policy and SR Policy color values is essential.

Internet exposure: BR nodes terminate VPN and connect to external networks. They SHOULD follow the same hardening as any Internet-facing PE: control-plane filtering, rate limiting, and monitoring. Issues specific to firewalls or IDS products are outside the scope of this document.

9. Acknowledgements

The authors would like to recognize the work of Andrew Tikhonov and Samvel Vartapetov from Nebius.

This use case was presented at MPLS & SRv6 World Congress in Paris in March 2025 and updated in March 2026. Recordings are available here: https://www.segment-routing.net/conferences/Paris25-Nebius-Alexey-Gorovoy/ and https://www.segment-routing.net/conferences/Paris26-Nebius-Alexey-Gorovoy/.

Illustration Extension

Cluster Firewall Service

Clustering groups multiple firewalls in a single logical device. The firewall sessions (connections) are synchronized across the devices in a cluster. This ensures seamless failover and uninterrupted traffic flows. It also allows forward and reverse traffic flows of a firewall session to be hashed on different ECMP paths, passing through different firewall cluster members. For example, for a TCP session between H12 and a server on the Internet, the forward flow may be hashed on firewall cluster member FW3, while the reverse flow may be hashed on member FW5 of the same firewall cluster.

All members advertise the same anycast service SIDs to leverage ECMP load-sharing across firewall cluster members. If the firewall cluster is SRv6-unaware, the SRv6 nodes in front of the firewall instances instantiate the same anycast SIDs, directing the traffic to the appropriate firewall instance.

For example, the service leaf nodes connected to firewall cluster members FW3 and FW5 both advertise the anycast locator 5f00:0:24::/48 and instantiate common uA SIDs 5f00:0:24:e000:: and 5f00:0:24:e001:: for the inside and outside firewall interfaces, respectively.

Figure 5 illustrates the network topology with a cluster firewall service with cluster members FW3 and FW5. These firewall services are connected to service leaf (SL) nodes SL2 and SL4, respectively. Each firewall is dual-connected to the SL node via an inside and outside interface.

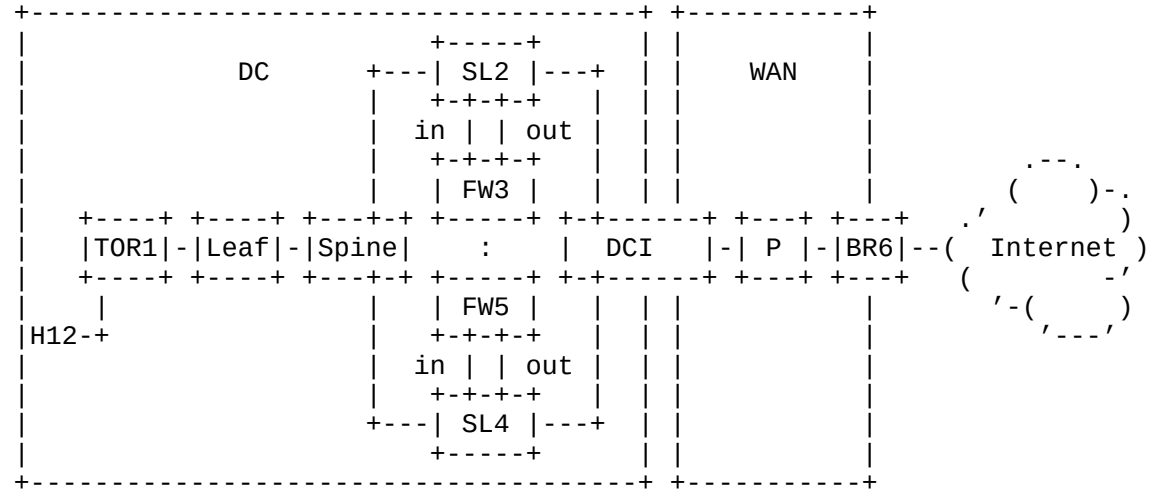

Figure 5: Reference network topology with cluster firewall service

The SL nodes SL2 and SL4 are SRv6-enabled and respectively advertise locator prefixes 5f00:0:2::/48 and 5f00:0:4::/48. In addition, SL2 and SL4 both advertise the same anycast locator prefix 5f00:0:24::/48.

These SL nodes allocate and install a common uA SID (5f00:0:24:e000::) directing packets to the inside interface of the firewall (FW3-IN for FW3, FW5-IN for FW5) and a common uA SID (5f00:0:24:e001::) directing packets to the outside interface of the firewall (FW3-OUT for FW3, FW5-OUT for FW5).

This scenario is similar to the standalone firewall scenario, with the difference being the SID list of the SR Policies to steer the traffic through the firewall service.

SR Policy POLTOR on TOR1 has a SID list <5f00:0:24:e000::>. Due to the anycast locator, this SID list steers the packets to either SL2 or SL4 into the inside interface FW3-IN of FW3 or FW5-IN of FW5, respectively.

SR Policy POLBR on BR6 has a SID list <5f00:0:24:e001::>. This SID list steers the packets to either SL2 or SL4, into the outside interface FW3-OUT of FW3 or FW5-OUT of FW5, respectively.

SRv6 Service Considerations

SRv6-Unaware Service

If the service is SRv6-unaware ([I-D.ietf-spring-sr-service-programming]), it must be connected to an SRv6-capable entity, a physical or virtual element that can handle SRv6 packets. In section 5, this entity is a physical service leaf node. The requirements for the SRv6 entity depend on the service capabilities.

If the service (e.g., firewall, IPS, IDS, etc.) can process the inner packets without modifying the outer IPv6 header, which may include a Segment Routing Header (SRH), it is sufficient for the connected SRv6 entity to implement uN and uA SRv6 SID behaviors ([RFC9800]) to direct the packets into the service. The service returns the processed packets to the SRv6 entity, where they continue their journey to their destinations.

If the service cannot process encapsulated packets, proxy SRv6 SID behaviors ([I-D.ietf-spring-sr-service-programming]) must be implemented on the connected SRv6 entity. The service processes the decapsulated packet, as the proxy entity provides, and returns the processed packets to the proxy. The proxy restores the packet encapsulation and forwards the packet towards its destination.

SRv6-Aware Service

The SRv6-aware Service does not depend on an additional connected SRv6 node to execute the SRv6 SID behaviors related to the service. The SRv6-aware service node is reachable via its SRv6 locator and executes the behavior of its local SID matching the outer IPv6 DA of the received packets.

References

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>.

[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>.

[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>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC9012] Patel, K., Van de Velde, G., Sangli, S., and J. Scudder, "The BGP Tunnel Encapsulation Attribute", RFC 9012, DOI 10.17487/RFC9012, April 2021, <https://www.rfc-editor.org/info/rfc9012>.

[RFC9252] Dawra, G., Ed., Talaulikar, K., Ed., Raszuk, R., Decraene, B., Zhuang, S., and J. Rabadan, "BGP Overlay Services Based on Segment Routing over IPv6 (SRv6)", RFC 9252, DOI 10.17487/RFC9252, July 2022, <https://www.rfc-editor.org/info/rfc9252>.

[RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <https://www.rfc-editor.org/info/rfc9256>.

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

[RFC9087] "Segment Routing Centralized BGP Egress Peer Engineering", RFC 9087, August 2021, <https://www.rfc-editor.org/info/rfc9087>.

[I-D.ietf-spring-sr-service-programming] Abdelsalam, A., Ed., Xu, X., Ed., Filsfils, C., Bernier, D., Li, C., Decraene, B., Ma, S., Yadlapalli, C., Henderickx, W., and S. Salsano, "Service Programming with Segment Routing", Work in Progress, Internet-Draft, draft-ietf-spring-sr-service-programming-12, 3 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-spring-sr-service-programming-12>.

Informative References

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[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>.

[I-D.filsfils-srv6ops-srv6-ai-backend] Filsfils, C., Camarillo, P., Michielsen, K., and A. Gorovoy, "SRv6 for Deterministic Path Placement in AI Backends", Work in Progress, Internet-Draft, draft-filsfils-srv6ops-srv6-ai-backend-04, May 2026, <https://datatracker.ietf.org/doc/html/draft-filsfils-srv6ops-srv6-ai-backend-04>.

Authors' Addresses

Clarence Filsfils
Cisco Systems
Belgium
Email: cf@cisco.com

Pablo Camarillo Garvia (editor)
Cisco Systems
Spain
Email: pcamaril@cisco.com

Kris Michielsen
Cisco Systems
Belgium
Email: kmichiel@cisco.com

Alexey Gorovoy
Nebius
Netherlands
Email: algorovoy@nebius.com

Nicolai Leymann
Deutsche Telekom
Germany
Email: N.Leymann@telekom.de