SRv6 White Paper 2026: Architecture, Design and Deployment

Daniel Navas and the SRv6.md community

October 2026

SRv6 White Paper 2026: Architecture, Design and Deployment

A vendor-neutral white paper on Segment Routing over IPv6 (SRv6) for network architects and engineering leaders. It explains why operators adopt SRv6, how to design an SRv6 network, and what to plan for when migrating from MPLS. Edition of October 2026. Written by Daniel Navas and the SRv6.md community, MIT licensed.

Executive summary

- **SRv6 is source routing on plain IPv6.** The ingress node writes the path into the packet as a list of IPv6 addresses called segment identifiers (SIDs). Every other router either forwards on the destination address like any IPv6 packet, or executes the instruction that the SID encodes.
- **It collapses the transport stack.** One IPv6 data plane replaces MPLS, LDP, RSVP-TE and most of the tunnelling protocols that sit on top of them. VPNs, traffic engineering, fast reroute and network slicing all use the same encapsulation.
- **It is standardized and shipping.** 23 RFCs are explicitly about SRv6, and 8 of the 11 implementations tracked in the support matrix support the compressed uSID encoding.
- **It is in production at large scale.** 19 organizations have documented SRv6 in their own networks, from SoftBank, Rakuten Mobile and Reliance Jio to Microsoft, Meta, Alibaba Cloud and OpenAI.
- **uSID solved the overhead objection.** Compressed SIDs carry up to six hops in a single 128-bit address, so most paths need no extension header at all.
- **AI changed the conversation in 2025–2026.** Hyperscalers now use static SRv6 source routing inside GPU training fabrics to spread traffic across many network planes and to steer around failed paths in tens of microseconds.

1. Why operators adopt SRv6

THE PROBLEM WITH TODAY'S TRANSPORT

A typical service provider or large enterprise backbone runs IS-IS or OSPF, plus LDP or RSVP-TE for MPLS labels, plus BGP for services, plus VXLAN or GRE wherever MPLS does not reach (data centers, public clouds, hosts). Every boundary between these domains needs stitching, and every protocol needs its own operations, troubleshooting and scaling plan.

WHAT SRV6 CHANGES

Area	MPLS / RSVP-TE / LDP	SRv6
Data plane	MPLS labels, one more header format	Native IPv6
Label distribution	LDP or RSVP-TE	None: SIDs are advertised by the IGP
Traffic engineering	Per-path state in every transit router	State only at the headend
Fast reroute	RSVP-TE FRR or LFA with coverage gaps	TI-LFA with full topology coverage
Reach into DC, cloud and hosts	Needs gateways or VXLAN stitching	Any IPv6 node can be an endpoint
Services	L3VPN, EVPN over MPLS	Same services, signalled by BGP with SRv6 SIDs

The operational gain comes from removing protocols, not from adding features. The SRv6 vs SR-MPLS comparison covers the trade-offs in detail, including when SR-MPLS is still the right call.

BUSINESS DRIVERS SEEN IN PUBLIC DEPLOYMENTS

- **5G transport and mobile user plane.** SoftBank built its 5G transport on SRv6 with Flex-Algo and later deployed the SRv6 Mobile User Plane on a commercial network. Rakuten Mobile migrated its IP transport to SRv6 uSID. See 5G transport.
- **One fabric from data center to WAN.** Nebius runs SRv6 uSID L3VPN from the data center front end to the WAN and peering, replacing VXLAN-EVPN and SR-MPLS stitching, and Alibaba Cloud runs full-stack SRv6 on SONiC routers. See deployments and cloud-native backbones.
- **AI training networks.** Microsoft, OpenAI and Oracle use SRv6 inside GPU fabrics. Meta and Alibaba use it for traffic engineering between AI sites. See SRv6 for AI.
- **Public sector and mission-critical networks.** Rijkswaterstaat migrated its national infrastructure backbone from MPLS to SRv6 uSID. Cisco has published a validated design for quantum-safe SRv6 fabrics. See mission-critical networks.

2. Architecture in one page

SIDS ARE INSTRUCTIONS

An SRv6 SID is a 128-bit IPv6 address with three parts: a **locator** that routes the packet to the node, a **function** that tells the node what to do, and optional **arguments** (SID structure). RFC 8986 defines the standard behaviors:

Behavior	What the endpoint does	Typical use
End (uN)	Move to the next segment	Node waypoint for traffic engineering
End.X (uA)	Move to the next segment, send out a specific link	Adjacency steering, TI-LFA repair paths
End.DT4 / End.DT6 / End.DT46	Decapsulate and look up in a VRF	L3VPN
End.DX4 / End.DX6	Decapsulate and send to a fixed next hop	Per-CE VPN
End.DT2U / End.DT2M	Decapsulate into a bridge domain	EVPN
End.B6.Encaps	Apply a new SR Policy (binding SID)	Inter-domain, scaling

THE PACKET

The headend encapsulates the customer packet in an outer IPv6 header. The **active segment is always the destination address**, so transit routers that know nothing about SRv6 forward the packet on a normal IPv6 lookup. When a path needs more SIDs than the destination address can carry, the headend adds a Segment Routing Header (RFC 8754) with the list. The SRH packet walk shows each hop.

USID: THE COMPRESSED FORM

RFC 9800 standardizes compressed SIDs. With the common **F3216** format, a 32-bit block is followed by 16-bit micro-SIDs, so one 128-bit address carries up to **six** instructions. Each endpoint shifts its own uSID out of the address and forwards on the next one. In practice:

- A VPN path with a few traffic engineering waypoints fits in the destination address alone: **40 bytes of overhead, the same as plain IPv6-in-IPv6**, and no SRH.
- Hardware only looks at the first 128 bits for most forwarding decisions, which keeps line rate on merchant silicon.

The uSID page and the SID decoder let you step through the shift operation on real addresses.

3. Designing an SRv6 network

ADDRESSING PLAN

1. **Pick the SID block.** Either use a prefix from the IANA block reserved for SRv6 SIDs, `5f00::/16` (RFC 9602), or carve a block out of your own global prefix. `5f00::/16` makes edge filtering trivial because nothing outside the SR domain should ever use it.
2. **Choose the uSID format.** F3216 (32-bit block, 16-bit uSID) is the default on most platforms and gives 65,536 uSID values per block, enough for node and local IDs in all but the largest single domains.
3. **Give every node a locator**, typically a `/48` inside the block. Keep loopbacks outside the SID block so management traffic and SRv6 traffic can be filtered separately.
4. **Reserve one locator per Flex-Algo** if you plan to slice the network. Each algorithm gets its own locator per node, which is what makes the path selection automatic.

CONTROL PLANE

Function	Recommended choice	Reference
Underlay and locator advertisement	IS-IS (or OSPFv3) with SRv6 extensions	RFC 9352, RFC 9513
Intent-based paths (low latency, disjoint planes)	Flexible Algorithm	RFC 9350, Flex-Algo
Fast reroute	TI-LFA	RFC 9855, TI-LFA
VPN and EVPN services	BGP with SRv6 service SIDs	RFC 9252, BGP overlay
Explicit traffic engineering	SR Policy from a PCE or BGP	RFC 9256, RFC 9603, RFC 9830
Topology export to controllers	BGP-LS	RFC 9514

A sound default is **IS-IS + Flex-Algo for the transport, BGP for services, SR Policy only where Flex-Algo cannot express the intent**. Most networks need far fewer explicit policies than they had RSVP-TE tunnels, because Flex-Algo and TI-LFA cover the common cases with no per-path state.

Large data center fabrics often run BGP as the only routing protocol. SRv6 works there too: locators are advertised as ordinary IPv6 prefixes in BGP. The trade-off is that Flex-Algo and TI-LFA are IGP features, so a BGP-only fabric relies on ECMP and fast BGP convergence

instead.

DATA PLANE AND MTU

- Use **reduced encapsulation** (H.Encaps.Red), which leaves the first SID out of the SRH because it already sits in the destination address.
- Budget **40 bytes** for the outer IPv6 header, plus **8 + 16 × n bytes** when an SRH is needed. With uSID most paths need no SRH.
- Raise the core MTU so that customer packets at 1500 bytes (or 9000 in data centers) never fragment. Fragmentation of encapsulated traffic is the most common avoidable SRv6 incident.
- Check each platform's limits on **how many SIDs it can push** at line rate and **how deep it can parse** the SRH. The support matrix lists uSID support per implementation.

SERVICES

Each VRF gets an End.DT46 (or uDT46) SID from the PE's locator, which BGP advertises with the VPN routes. EVPN uses End.DT2U and End.DT2M. Because the service SID is routable inside the SR domain, any PE reaches any other PE with no tunnels to configure. See VPN services and EVPN multihoming.

4. Security

SRv6 turns IPv6 addresses into instructions, so the security model is about **keeping those instructions inside a trusted domain** (RFC 8754 section 5).

1. **Filter the SID block at every external interface.** Drop packets from outside the domain whose destination falls in the SID block, and drop packets with an SRH arriving from outside. With `5f00::/16` this is one filter line.
2. **Never advertise locators to the Internet.** Summarize the SID block internally and keep it out of eBGP exports.
3. **Use the SRH HMAC** only where SRH packets must cross a less trusted segment. Most designs rely on edge filtering instead.
4. **Rate-limit SRv6 OAM** and ICMPv6 generated by endpoints, as for any IPv6 control traffic.
5. **Encrypt where the transport is not trusted.** SRv6 does not encrypt anything by itself. MACsec on links, or IPsec and post-quantum key exchange on top, are covered on the security page.

5. Operations

- **Reachability and path tracing.** Ping and traceroute work on SIDs with standard IPv6 tools, plus the SRv6 OAM extensions of RFC 9259. See OAM and troubleshooting.

- **Performance measurement.** SR Performance Measurement (TWAMP-based delay and loss per link and per policy) feeds Flex-Algo latency metrics. See SRPM.
- **Flow visibility.** IPFIX exports SRv6 fields (RFC 9487). Streaming telemetry covers locator and SID counters. See telemetry.
- **Packet captures.** Wireshark decodes the SRH natively. The pcap library has reference captures for every common behavior.

6. Migration from MPLS

There is no flag day. Public migrations follow four patterns, detailed on the migration page:

Strategy	When it fits
Greenfield SRv6	New network or new region (Iliad launched a greenfield SRv6 network)
Ship in the night	Run SRv6 services in parallel with MPLS, move VRFs one by one
SR-MPLS first, then SRv6	Hardware not yet SRv6-ready across the whole network
Direct MPLS to SRv6	Hardware is SRv6-ready and an intermediate SR-MPLS step adds nothing (Rijkswaterstaat went straight from MPLS to SRv6 uSID)

Interworking gateways and binding SIDs connect SR-MPLS and SRv6 domains during the transition, so services keep working end to end while the core changes underneath.

A practical sequence:

1. Enable IPv6 everywhere in the core and fix the MTU.
2. Turn on SRv6 locators in the IGP. Nothing uses them yet, so the risk is low.
3. Add TI-LFA for SRv6 and verify repair coverage.
4. Move one low-risk VRF to SRv6 services, then the rest.
5. Replace RSVP-TE tunnels with Flex-Algo or SR Policy.
6. Retire LDP and RSVP-TE.

7. SRv6 in AI networks

AI training clusters with tens of thousands of GPUs need every link busy and recovery from failures faster than any routing protocol can converge. SRv6 gives the **host** control of the path:

- The NIC encapsulates each packet with an SRv6 path chosen from many parallel planes, so traffic spreads evenly instead of colliding on hashed ECMP paths.

- Switches are plain SRv6 transit nodes running a **static** configuration. No routing protocol is involved in choosing paths.
- When a path fails, the NIC retires it within tens of microseconds and the job keeps running.

OpenAI's MRC design, built with Microsoft, AMD, Broadcom and NVIDIA for clusters of more than 100,000 GPUs, is the best documented example. See MRC and SRv6, AI fabric design and SRv6 vs ECMP, adaptive routing and UEC.

8. Adoption and outlook

- **Standards:** the core is complete (architecture, SRH, network programming, IGP and BGP extensions, compression, OAM). Current IETF work focuses on operations (the srv6ops working group), multicast, and path tracing. See the State of SRv6 2026.
- **Silicon and software:** uSID runs at line rate on current merchant silicon and is supported by Cisco, Juniper, Nokia, Arista, Huawei, the Linux kernel, FRRouting, SONiC and several eBPF data planes.
- **Where it goes next:** host-based SRv6 in Kubernetes and clouds (host SRv6), SRv6 in AI back-end and scale-across networks, and SRv6 as the single transport from the GPU to the WAN.

9. Recommendations

1. **Design with uSID (F3216) from day one**, even if the first phase uses few SIDs.
2. **Use** `5f00::/16` or a dedicated block, and filter it at every edge before the first service goes live.
3. **Prefer Flex-Algo and TI-LFA over explicit policies.** Keep SR Policy for the cases that need it.
4. **Fix the MTU first.** Most SRv6 incidents in early deployments are MTU incidents.
5. **Migrate services one VRF at a time** with SRv6 and MPLS running side by side.
6. **Lab it before you buy it.** The free Containerlab and netlab labs on this site reproduce every design element above on a laptop.

How to cite this white paper

> Navas, D. and the SRv6.md community. *SRv6 White Paper 2026: Architecture, Design and Deployment.* SRv6.md, October 2026. https://srv6.md/white-papers/srv6-white-paper-2026/SMARTPANTS_PRESERVED_620SMARTPANTS_PRESERVED_621

The text is MIT licensed: reuse it, translate it or adapt it for internal training, with attribution. Corrections are welcome on GitHub.