IP Network eBook Series

SRv6

Author: Lanjun Luo

Information Digitalization and Experience Assurance (IDEA) Department

HUAWEI

Copyright

Author:	Lanjun Luo
Key Contributors:	Shuping Peng, Ruiqiang Lu, Wei Li, Chen Jiang

Release Date:	2021-06-08
Issue:	01

Preface

Author Introduction

Lanjun Luo: Joined Huawei in 2010 and has since been engaged in developing documentation for data communication products. He made significant contributions to developing the book *SRv6 Network Programming: Ushering in a New Era of IP Networks*.

About This Book

This book explores the background of Segment Routing over IPv6 (SRv6), reveals why SRv6 has become so popular, explains its technical advantages, and depicts its vast development space. In addition, this book briefly describes SRv6 fundamentals and working modes in Chapter 4 and Chapter 5, respectively, to help you better understand the unique value of SRv6.

Intended Audience

This book is intended for network planning engineers, network design engineers, mid- and senior-level managers at service providers and enterprises, and readers who want to understand cutting-edge IP network technologies. Because SRv6 involves many network concepts, readers of this book should be familiar with IP network basics, such as the IP network architecture, IP routing, and VPN technologies.

Table of Contents

Chapter 1
SRv6 Overview

Despite being developed for more than 20 years, IPv6 has still not been widely deployed or applied. Segment Routing over IPv6 (SRv6), on the other hand, has injected exceptional vitality into IPv6 shortly after it was proposed. As 5G and cloud services develop, the innovation space of IPv6 extension headers is quickly opening up. Applications based on these headers are coming to fruition, and society is entering the IPv6 era at a faster pace.

SRv6 is a next-generation IP bearer protocol that simplifies the complex, traditional network protocols. In the 5G and cloud era, it serves as the foundation for building intelligent IP networks. SRv6 combines the advantages of the source routing mechanism used in Segment Routing (SR) with the simplicity and extensibility that IPv6 offers. In addition, SRv6 provides multi-dimensional programming space and complies with the Software-defined Networking (SDN) paradigm, making it a powerful tool for implementing intent-driven networks.

SRv6 offers a variety of network programming capabilities to better meet the requirements of new network services, and thanks to its compatibility with IPv6, it simplifies the deployment of network services. SRv6 not only breaks the boundary between the cloud and network — allowing carriers to transition away from being simply providers of pipes, and extending networks to user terminals to better share dividends in the information era — but also helps carriers quickly develop intelligent cloud-networks and implement application-level Service Level Agreement (SLA) assurance, bringing significant benefits to various industries.

Chapter 2
SRv6 Background

Only a few years after SRv6 was first proposed, the Internet Engineering Task Force (IETF) adopted multiple SRv6 drafts as RFC standards, and by the end of 2020, SRv6 had been commercially deployed at more than 100 sites worldwide. Such rapid development is uncommon among IP technologies. So, what is SRv6? And what is its historical mission? This chapter begins with the development history of IP networks, analyzes the challenges encountered during the development of IP/Multiprotocol Label Switching (MPLS) networks, and reveals the historical background of SRv6.

2.1 Challenges Facing IP/MPLS Networks

During the early stage of networks, multiple types of networks were developed to meet different service requirements. These networks coexisted and competed with each other, with telecom and computer networks taking center stage. The main technologies used by telecom and computer networks were Asynchronous Transfer Mode (ATM) and IP, respectively. As networks grew larger

and carried more and more services, the complexity involved in ATM allowed the simpler IP to come out on top.

Although the IP network is simpler than an ATM one, it still requires a certain level of QoS guarantee. In addition, the IP network involves table lookups based on the longest match rule, resulting in poor forwarding performance. Consequently, the industry explored numerous ways to overcome these issues, eventually leading to the development of MPLS technology in 1996. MPLS is considered a Layer 2.5 technology because it runs between Layer 2 and Layer 3. As the name suggests, it supports multiple network-layer protocols, such as IPv4 and IPv6, and is compatible with multiple link-layer technologies, such as ATM and Ethernet.

The combination of IP and MPLS can provide QoS guarantee on connectionless IP networks. In addition, MPLS label-based forwarding overcomes the poor forwarding performance issues on IP networks. This is why IP/MPLS became so successful.

However, as networks grow and continue to carry more emerging services, the combination of IP and MPLS faces the following problems and challenges:

1. **Decreasing forwarding advantages**: As algorithms designed to search routing entries improve, and especially as hardware — represented by Network Processors (NPs) — is upgraded, MPLS no longer delivers notable advantages in forwarding performance compared with IP.
2. **Difficult cloud-network convergence**: More and more cloud Data Centers (DCs) are being built to accommodate the continuous development of the Internet and cloud computing. In order to meet the requirements of multi-tenant networking, proposals emerged for multiple overlay technologies, such as Virtual Extensible Local Area Network (VXLAN). Additionally, numerous attempts were made to use MPLS in DCs to provide VPN services. However, these attempts all failed due to factors such as network management boundaries, management complexity, and scalability.
3. **Difficult cross-domain deployment**: MPLS is deployed in different network domains, such as IP backbone, metro, and mobile bearer networks, forming independent MPLS domains and creating new network boundaries. However, many services require End-to-End (E2E) deployment, meaning that services need to be deployed across multiple MPLS domains. This in turn results in complex inter-domain MPLS configuration. Against that backdrop, multiple inter-AS solutions, such as Option A, Option B, and Option C, have been

proposed for MPLS VPN. Even so, each of these solutions still involves relatively complex service deployment.

4. **Complex service management**: When multiple services (such as L2VPN and L3VPN services) coexist, protocols such as Label Distribution Protocol (LDP), Resource Reservation Protocol (RSVP), IGP, and BGP may also coexist on devices. This leads to complex service deployment and management, making it difficult to achieve large-scale service deployment in the 5G and cloud era.
5. **Complex protocol states**: After IGPs are optimized (in terms of their running speeds), they can allocate labels without the need for LDP. In addition, Resource Reservation Protocol-Traffic Engineering (RSVP-TE) is a complex protocol to implement and requires large numbers of protocol packets to be exchanged in order to maintain the connection state. As the number of nodes and tunnels increases, so too does the number of states. The exponential growth of states imposes significant pressure on the performance of transit nodes, hampering the construction of large-scale networks. Because RSVP-TE simulates Synchronous Digital Hierarchy (SDH) pipes, it cannot implement load balancing effectively. And although multiple pipes can be set up manually to implement load balancing, doing so involves much greater complexity.

2.2 Impact of SDN on Networks

Traditional MPLS adopts the distributed architecture, which is the root cause of the issues facing MPLS today. In order to learn its neighbor status, each device — aware of only its own status — needs to exchange a large number of signaling messages.

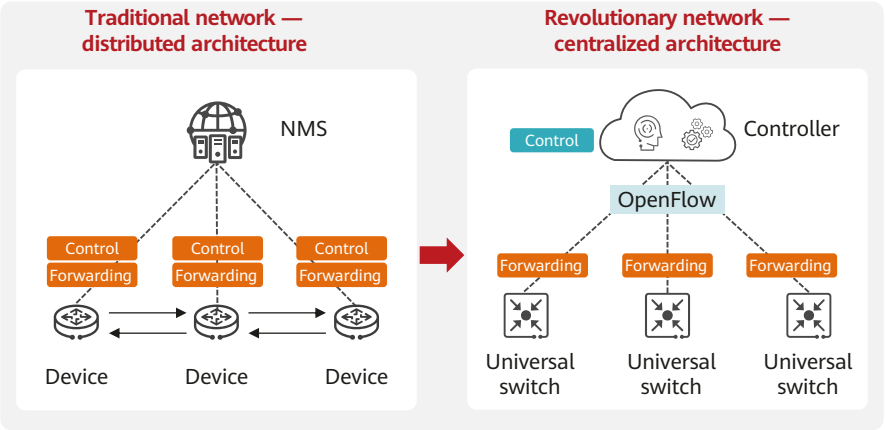
If the centralized architecture is adopted along with a central control node that computes paths and distributes labels in a unified manner, the preceding issues can be eliminated. This is one of the key aspects that SDN is designed to address.

Originally, SDN is represented by OpenFlow, which is a communication protocol for interaction between the SDN control and data planes. As shown in Figure 2-1, an OpenFlow-based network is called a revolutionary network in the industry, as it requires all hardware on the network to be upgraded or replaced.

Although this network features a simple structure and supports centralized programming, it is difficult to implement due to the following factors:

- Performance bottlenecks. The rate at which OpenFlow flow table entries are delivered is subject to bottlenecks. Furthermore, the transmission rate from the controller to the forwarder depends on the network, placing ultra-high requirements on the network.
- Inability to adapt. OpenFlow cannot adapt to complex service deployment on the network, especially when various services, such as L2/L3 VPN, QoS, multicast, and slicing, coexist on the backbone network.
- New hardware required. In order to support OpenFlow, new hardware must be deployed on the network, meaning that existing investments are negated.

Figure 2-1 Traditional network and revolutionary network



From the preceding information, it can be concluded that OpenFlow is applicable to switch-based networks with simple flow tables and fixed forwarding behaviors. However, the bearer network requires a technology that can meet the management and control requirements of SDN as well as the requirements of multi-service bearing, high performance, and high reliability on the bearer network.

2.3 SR Background

OpenFlow aims to function as a brain on networks and implement global optimization through centralized control, thereby avoiding local perspective, decentralized management, and disorder prevalent in traditional networks. However, OpenFlow is not the only solution to centralized network control — the source routing technology is another viable option.

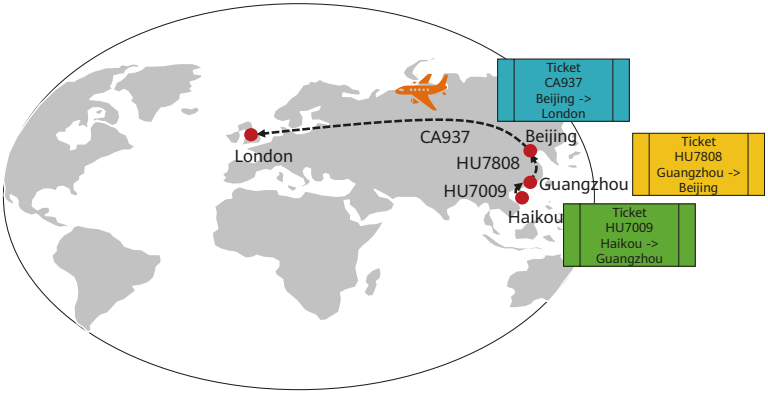
The source routing technology was proposed by Carl A. Sunshine in his paper "Source routing in computer networks", published in 1977. This technology allows the source of a data packet to determine the network path along which the packet will be transmitted, differing from the approach used on traditional networks where each network node selects a shortest path for packet forwarding. But because this technology processes data packets, involving a complex data packet format and increased overheads, it is not widely used in the initial stages when bandwidth resources are insufficient.

In 2013, SR was proposed. Borrowing some ideas from source routing, SR aims to combine different segments into a path and insert segment information into packets at the ingress of the path to guide packet forwarding. A transit node only needs to forward the packet according to the segment information carried in the packet. Each path segment — referred to simply as a segment — is identified by a Segment Identifier (SID).

The design of SR can be compared to many common scenarios. The following uses a journey by plane from Haikou to London to further explain SR, as shown in Figure 2-2. If this journey involves a stop in Guangzhou and in Beijing, we need to buy three tickets: Haikou to Guangzhou, Guangzhou to Beijing, and Beijing to London.

If we buy a ticket for the next leg of the journey at each transfer airport, we may find that no seats are available for our chosen flight. In contrast, if we buy an interline ticket in Haikou, covering the legs from Haikou to Guangzhou, Guangzhou to Beijing, and Beijing to London, we will not worry about a ticket selling out. The ticket states that we will fly from Haikou to Guangzhou on flight HU7009, to Beijing on flight HU7808, and then to London on flight CA937. Using this interline ticket, we are therefore able to fly segment by segment to London. If everyone were to adopt the same approach when planning a trip, local competition for individual segment tickets would be reduced. Although such a trip for an individual might be sub-optimal, it is the optimal choice for groups.

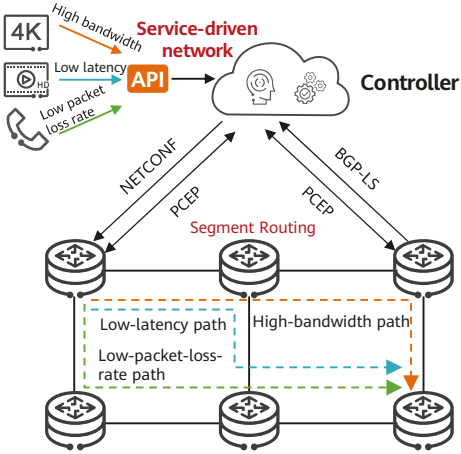
Figure 2-2 Flying from Haikou to London



There are two key points in the preceding process: One is path segmentation (segment), and the other is combining segments on the ingress to determine the entire path (routing) in advance. SR divides a network into segments and combines paths on the ingress, without requiring transit nodes to maintain or even be aware of the path states. This is the source routing concept.

When the network boundary is clear and the service ingress and egress are fixed, the packet forwarding path can be controlled by controlling the ingress. Path adjustment is performed only at the ingress, which can meet the customization requirements of different services. This approach — in line with the SDN paradigm — enables the network to be service-driven and integrate service intents more effectively. Figure 2-3 shows the detailed implementation.

Figure 2-3 Service-driven network



Network optimization based on SR eliminates the need to replace numerous hardware facilities on the live network. As such, SR provides better compatibility with the live network, allowing carriers to upgrade their networks gradually rather than all at once. This innovative incremental evolution is easier to implement, and such networks are called incremental networks in the industry. An incremental network based on SR has the following characteristics:

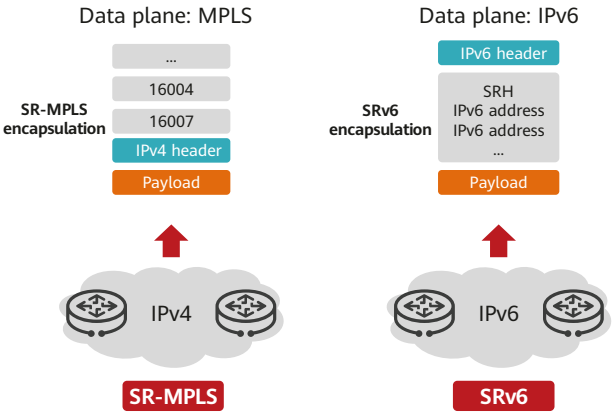
1. Smoother evolution can be achieved by extending existing protocols.
2. A balance between centralized control and distributed forwarding is provided.
3. The source routing technology is used to achieve fast interaction between the network and upper-layer applications in order to quickly meet service requirements.

2.4 What Is SRv6?

As shown in Figure 2-4, SR currently involves two data planes: MPLS and IPv6. When SR is applied to the MPLS data plane, it is called SR-MPLS and uses MPLS

labels as SIDs. When SR is applied to the IPv6 data plane, it is called SRv6 and uses IPv6 addresses as SIDs.

Figure 2-4 SR classification



It is worth noting that SRv6 was mentioned in the SR architecture document as early as 2013, when SR was first proposed.

"The Segment Routing architecture can be directly applied to the MPLS dataplane with no change on the forwarding plane. It requires minor extension to the existing link-state routing protocols.
Segment Routing can also be applied to IPv6 with a new type of routing extension header." — RFC 8402

When SRv6 was first proposed, it lacked the programmability of SRv6 SIDs, as its intention was only to insert IPv6 addresses of nodes and links into Segment Routing headers (SRHs) in order to steer traffic. Unlike SR-MPLS, SRv6 was viewed as a more distant goal and received less attention.

In March 2017, the SRv6 Network Programming draft was submitted to the IETF, upgrading SRv6 to SRv6 Network Programming. Since then, SRv6 has entered a new development phase. SRv6 Network Programming divides a 128-bit SRv6 SID into fields including Locator and Function. Locator provides the routing capability, and Function can represent processing behaviors and identify services. The

ingenious design means that SRv6 SIDs integrate routing and MPLS (in which labels represent services) capabilities, significantly enhancing the programmability of networks and better supporting new services.

At present, various industry activities around SRv6 are in full swing.

- The world's first SRv6 Industry Roundtable was successfully held in Paris, France during MPLS+SDN+NFV World Congress 2019. The experts who attended all agreed that SRv6, a successor of MPLS, will be the next-generation core protocol of IP bearer networks. Bearer networks need to have full SRv6 capabilities if they are to meet intelligent connection and bearer requirements in the 5G and cloud era.
- In December 2019, China's Expert Committee for Promoting Large-Scale IPv6 Deployment hosted the SRv6 industry salon. Experts discussed SRv6 and IPv6 innovation as well as SRv6 technology and industry promotion, and jointly released *SRv6 Technology and Industry White Paper* and *SRv6 Interoperability Test Report*.
- By the end of 2020, the European Advanced Networking Test Center (EANTC) had conducted three successful SRv6 interoperability tests, which covered basic SRv6 VPN service scenarios, SRv6 reliability, and SRv6 ping/tracert. The results obtained from the tests were as expected, fully proving the commercial deployment capability of SRv6.
- By March 2021, the IETF standards work had achieved significant milestones. For example, the SR architecture was standardized in RFC 8402 (*Segment Routing Architecture*), and basic SRv6 features were standardized in RFC 8754 (*IPv6 Segment Routing Header (SRH)*) and RFC 8986 (*SRv6 Network Programming*). Both RFC 8754 and RFC 8986 lay the foundation for future SRv6 development. In addition, IGP, BGP, and VPN extensions for SRv6 are being gradually promoted, and the IS-IS and VPN drafts have passed the Working Group Last Call (WGLC). The maturity of protocols will greatly promote the development of the SRv6 industry.

With the exhaustion of global IPv4 public addresses in 2019, networks are migrating to IPv6, IPv6 development is accelerating, and SRv6-based applications are becoming more and more widely used. The development of SRv6 is overwhelming.

2.5 SRv6 Characteristics

Although SR-MPLS can provide good path programmability, it is unsuitable for services that need to carry metadata, such as Service Function Chaining (SFC) and In-situ Operations, Administration, and Maintenance (IOAM). This is largely due to the relatively poor extensibility provided by MPLS encapsulation. In addition, the mode in which MPLS adds labels to the IP packet header eliminates the universality of IP technology in packets. In this case, network devices need to support MPLS label forwarding hop by hop, raising the requirements on network devices to some extent. As such, MPLS is regarded as a dedicated technology for carriers' backbone networks. It is generally not deployed in data centers; instead, it is limited to only carriers' backbone networks or new metro networks. Because of this, SR-MPLS can only be defined as the next-generation evolution of MPLS given the limiting factor of MPLS.

In contrast, given SRv6 is based on the IPv6 data plane, it not only inherits all the advantages of SR-MPLS, but also provides better compatibility and extensibility than SR-MPLS due to its native IPv6 attribute. SRv6 SID extension provides SRv6 with network programming capabilities that SR-MPLS lacks. Figure 2-5 describes the technical characteristics of SRv6.

Figure 2-5 Technical characteristics of SRv6

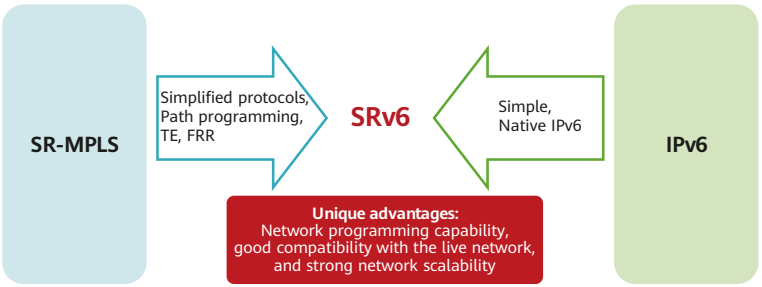


Table 2-1 compares SRv6 and SR-MPLS.

Table 2-1 Comparison between SRv6 and SR-MPLS

Dimension	SRv6	SR-MPLS
Network protocol simplification	Control plane: IPv6 IGP/BGP Data plane: IPv6	Control plane: IPv4/IPv6 IGP/BGP Data plane: MPLS
Programmability	Flexible. The service orchestrator or various apps can specify networks and applications (service chains) based on SLAs and service requirements to provide flexible programmability.	Poor.
Cloud-network synergy	Easy. Data Center Networks (DCNs) can easily support IPv6. With SRv6 technology, carrier networks can be deployed in DCs and even extended to user terminals.	Difficult: It is difficult for DCNs, including virtual machines, to support MPLS.
Terminal collaboration	Easy. Terminals support SRv6. Linux 4.10 and later versions support SRv6, and Linux 4.14 supports most SRv6 functions expressed using the Function field.	Difficult. It is difficult for terminals to support MPLS.
Cross-AS deployment	Easy. With IPv6 reachability, SRv6 can be easily deployed across ASs. Host routes do not need to be flooded across ASs, and only aggregated routes need to be imported. This greatly reduces the number of routes and simplifies routing policies.	Complex. Only SR-MPLS TE can be used across ASs, and inter-AS controllers are required. The local PE requires the loopback host routes of remote PEs. All the loopback host routes of remote PEs need to be leaked.
Large-scale deployment	Easy. SIDs use IPv6 address space and are suitable for large-scale network planning.	Complex. The SID (MPLS label) space is limited, and unified planning and maintenance of device SIDs are complex.

Dimension	SRv6	SR-MPLS
Service provisioning difficulty	SRv6 can be deployed on the same network as common IPv6 devices as long as the ingress and egress support SRv6, making service provisioning more agile.	All devices in an AS need to be upgraded to support SR-MPLS, and service provisioning is complex.
Reliability	Topology-Independent Loop-Free Alternate (TI-LFA)	TI-LFA
Forwarding efficiency	Take L3VPN encapsulation as an example. An IPv6 header of at least 40 bytes is required. Each time a SID is added to an SRv6 SRH, 16 bytes are added.	The SR-MPLS encapsulation header is small. For example, if L3VPN encapsulation is used, at least two MPLS labels (4 bytes each) are required. Each time a SID is added to the SR-MPLS label stack, 4 bytes are added. The forwarding efficiency is high.

Chapter 3
Technical Benefits of SRv6

SRv6 can reduce the number of existing network protocols and simplify network management, thereby better coping with the challenges of network development in the 5G and cloud era. In addition, the core advantages of SRv6 are its native IPv6 attribute and network programming capabilities. Leveraging the native IPv6 attribute, SRv6 can better promote cloud-network convergence, achieve compatibility with existing networks, and improve inter-AS experience. And thanks to its network programming capabilities, SRv6 can better program paths to meet service SLA requirements and connect networks together with applications to build intelligent cloud-networks.

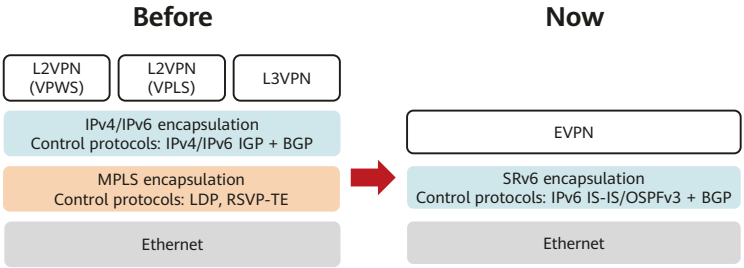
3.1 Simplified Network Protocols

To address the challenges of network development in the 5G and cloud era, IP bearer networks need to be simplified to reduce management complexity and improve O&M. Protocols on IP bearer networks are simplified using SRv6 and Ethernet Virtual Private Network (EVPN).

At the tunnel/underlay layer, IPv6 packet extension replaces the tunnel function, and in turn allows the original MPLS tunneling technologies such as LDP and RSVP-TE to be replaced. SRv6 can implement underlay and tunnel functions through IGP and BGP extension, reducing the number of signaling protocols.

At the service/overlay layer, EVPN integrates L2VPN Virtual Private Wire Service (VPWS, which is based on LDP or MP-BGP), L2VPN Virtual Private LAN Service (VPLS, which is based on LDP or MP-BGP), and L3VPN (based on MP-BGP) technologies. At the service layer, SRv6 SIDs can be used to identify various services, reducing technical complexity.

Figure 3-1 Simplifying existing networks through SRv6



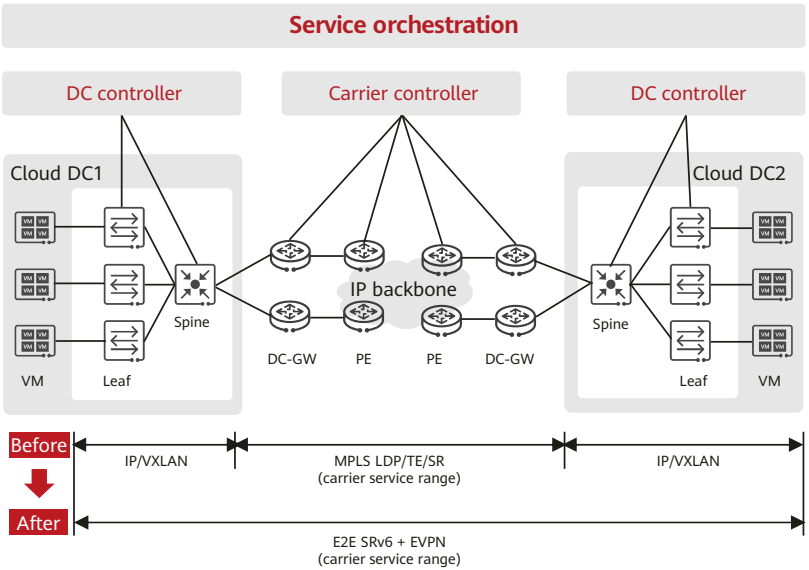
3.2 Promotion of Cloud-Network Convergence

In the Data Center Interconnect (DCI) scenario shown in Figure 3-2, the IP backbone network uses MPLS or SR-MPLS, whereas DCNs use VXLAN. In this case, gateways need to be deployed to implement mapping between VXLAN and MPLS. In turn, this complicates service deployment, without yielding any noticeable benefits.

Because SRv6 has the native IPv6 attribute, and both SRv6 and common IPv6 packets have the same packet header, SRv6 can implement communication between network nodes by leveraging only IPv6 reachability. This also allows SRv6

to break the boundary between carrier networks and DCNs, whereby it can be deployed in DCNs or even on terminals (such as servers).

Figure 3-2 SRv6 application in the cloud DCI scenario



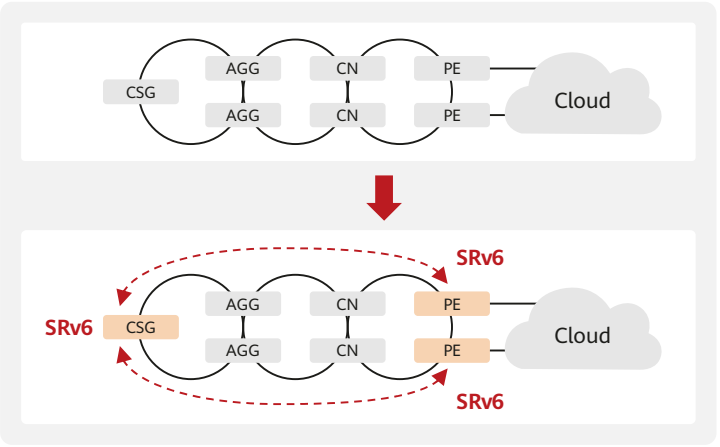
The basic IPv6 header ensures communication between any IPv6 nodes, and multiple IPv6 extension headers provide various functions. SRv6 unlocks the value of IPv6 extensibility and helps build simplified E2E programmable networks, thereby implementing unified service forwarding and connectivity of everything through one network.

3.3 Compatibility with Existing Networks

SRv6 is compatible with existing IPv6 networks, allowing services to be quickly provisioned on demand. Because network-wide upgrade is not required during service deployment, existing investments on the live network are fully protected. In addition, SRv6 only needs to be configured on the ingress and egress, shortening deployment time and improving deployment efficiency.

As shown in Figure 3-3, certain key devices (such as the ingress and egress) are upgraded to support SRv6 during the initial phase. New services are then deployed based on SRv6, with transit devices needing only to support IPv6 and forward packets through IPv6 routes. In the future, it will be possible to upgrade transit nodes on demand to provide value-added services based on SRv6 traffic engineering.

Figure 3-3 On-demand SRv6 upgrade

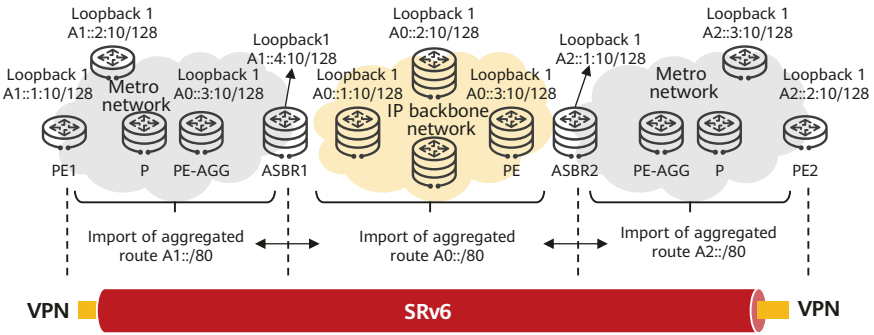


3.4 Improved Inter-AS Experience

Compared with traditional MPLS inter-AS technologies, SRv6 inter-AS deployment is simpler. Because SRv6 has the native IPv6 attribute, all that is needed to deploy inter-AS services is to import IPv6 routes in one AS into another AS through BGP4+. This makes service deployment much easier to perform.

In addition, SRv6 provides higher extensibility in inter-AS scenarios. The native IPv6 attribute enables SRv6 to work based on aggregated routes. As such, only a limited number of aggregated routes need to be imported to edge nodes on a large-scale inter-AS network, as shown in Figure 3-4. This not only reduces the requirements on network device capabilities, but also improves network scalability.

Figure 3-4 SRv6 large-scale networking



The service is reachable as long as the corresponding route is reachable.

3.5 Agile Service Provisioning

As multi-cloud and hybrid-cloud deployments gain popularity, networks need to provide on-demand connections for enterprise customers so that they can flexibly access applications on different clouds. In addition, bearer networks and clouds need to be connected in an agile manner to support flexible scheduling of

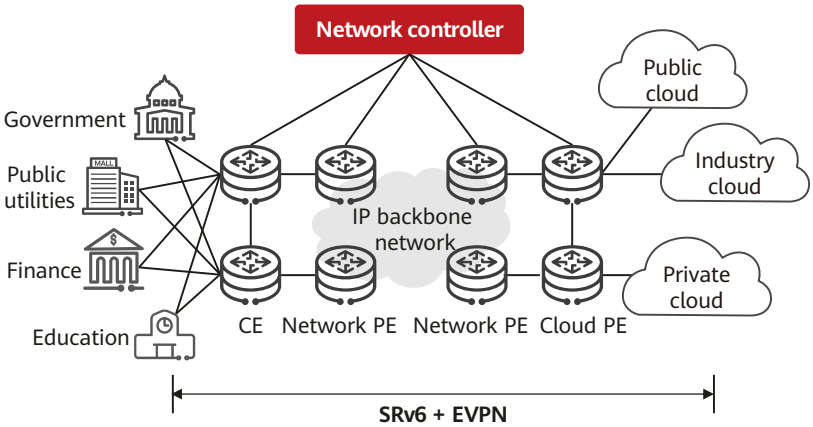
resources between clouds and provide both dynamic and on-demand connectivity for resources on different clouds.

In conventional Layer 2 point-to-point private line scenarios, enterprises need to lease multiple site-to-cloud private lines according to the deployment locations of different clouds. They also need to perform manual adjustments or implement automatic scheduling based on internal networking to access the applications on different clouds, compromising service flexibility and multi-cloud access experience as well as complicating cloud-network synergy.

Due to the lack of a unified interconnected cloud backbone network, if a new cloud DC is deployed in scenarios where multiple networks access their respective clouds, the new DC needs to be connected to all the networks. This results in complex connections, extremely difficult segment-by-segment deployment, and long business monetization time.

The intelligent cloud-network solution uses a cloud backbone network to connect multiple clouds and networks and supports cloud-network connection pre-deployment. The benefit of this is that clouds can be accessed as soon as network access is available. Leveraging SRv6 + EVPN technologies, this solution achieves flexible connection to multiple clouds and agile service provisioning, as shown in Figure 3-5.

Figure 3-5 Agile service provisioning based on SRv6



Chapter 4
SRv6 Fundamentals

This chapter describes SRv6 fundamentals, mainly covering the SRv6 extension header, SRv6 SID, SRv6 packet forwarding process, SRv6-based protocol extensions, and SRv6 reliability. SRv6 inherits the advantages of the source routing mechanism and offers a three-dimensional programming space. Thanks to the native IPv6 attribute of SRv6, SRv6 devices can be deployed together with IPv6 devices on existing networks, facilitating network evolution. Furthermore, SRv6 supports new reliability technologies (for example, TI-LFA and midpoint protection) that use SRv6 explicit paths as post-failure repair paths, enhancing fault protection capabilities on IP networks.

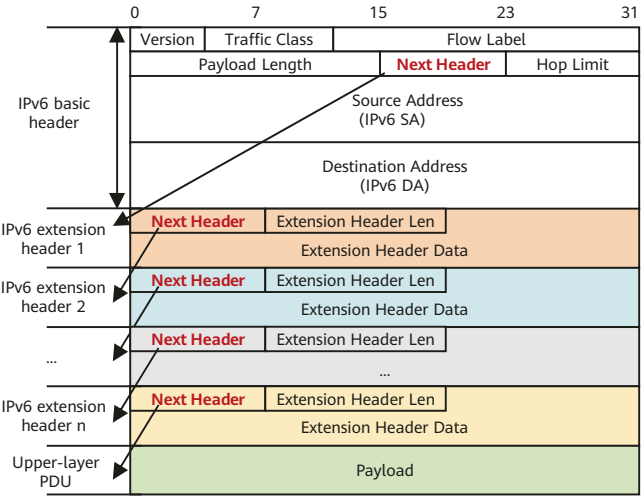
4.1 Why Is SRv6 a Native IPv6 Technology?

SRv6 is inseparable from IPv6. As defined in RFC 8200, an IPv6 packet consists of three parts: IPv6 basic header, IPv6 extension header, and upper-layer Protocol Data Unit (PDU). Figure 4-1 shows the IPv6 packet format.

An IPv6 basic header has eight fields and a fixed length of 40 octets. This header is required in every IPv6 packet to provide basic packet forwarding information, which is parsed by all devices on the corresponding forwarding path.

An upper-layer PDU is usually composed of an upper-layer protocol header and its payload. The PDU can be an ICMPv6, TCP, or UDP packet.

Figure 4-1 IPv6 packet format



The IPv6 packet format is designed to simplify the IPv6 basic header. Typically, a device only needs to process the basic header in order to forward IP traffic. The IPv6 header does not carry fields such as Fragment Offset, Header Checksum, and Options — unlike the IPv4 header — and instead carries the Flow Label field. This makes IPv6 header processing simpler and more efficient. In addition, IPv6 utilizes extension headers to support various options without requiring modification of the existing packet format, offering exceptional flexibility while keeping packet headers simple.

IPv6 extension headers are placed between the IPv6 basic header and upper-layer PDU. An IPv6 packet can carry one or more extension headers, or none at all. The source node of a packet adds one or more extension headers to the packet only when other nodes are required to perform special handling.

If multiple extension headers are used, the Next Header field is used to indicate the type of the next header that follows. As shown in Figure 4-1, the Next Header field in the IPv6 basic header indicates the type of the first extension header, that in the first extension header indicates the type of the second extension header, and so on. In the last extension header, the Next Header field indicates the upper-layer protocol type.

Table 4-1 lists IPv6 extension headers and their corresponding protocol numbers. A routing device determines whether to process an extension header based on the protocol number specified by the Next Header field in the basic header. Not all extension headers need to be checked and processed.

Table 4-1 IPv6 extension headers

IPv6 Extension Header	Protocol Number
Hop-by-Hop (HBH) Options Header	0
Destination Options Header (DOH)	60
Routing Header (RH)	43
Fragment Header (FH)	44
Authentication Header (AH)	51
Encapsulating Security Payload (ESP) Header	50
Upper-Layer Header (ULH)	ICMPv6: 58 UDP: 17 TCP: 6

SRv6 is implemented through the RH extension, without the need to change the encapsulation structure of original IPv6 packets. This means that SRv6 packets are valid IPv6 ones that can be identified by common IPv6 devices. Therefore, SRv6 is considered a native IPv6 technology. Its native IPv6 attribute enables SRv6 devices to interwork with common IPv6 devices, offering excellent compatibility on an existing network.

The transition from IP/MPLS back to native IPv6 is significant because MPLS is eliminated from IP networks, achieving protocol simplification and uniform use

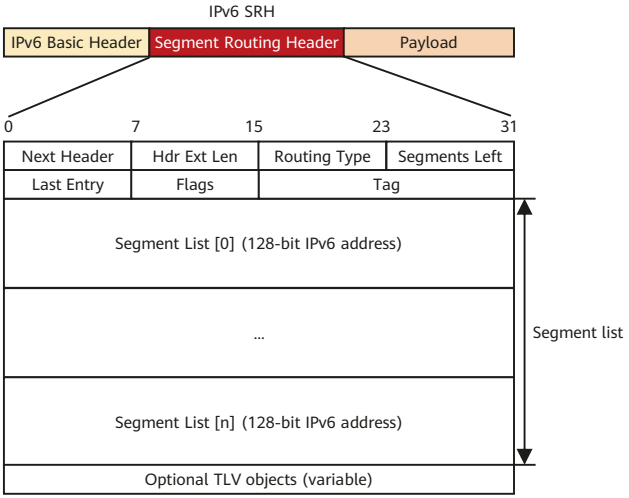
of IPv6. With SRv6, services can reach the destination as long as the corresponding routes are reachable, and they can easily span ASs thanks to the AS spanning capability of routes. This simplifies network deployment and facilitates network expansion.

4.2 How Is IPv6 Extended to Support SRv6?

IPv6 SRH

To implement SR based on the IPv6 forwarding plane, a new type of IPv6 RH called Segment Routing Header (SRH) is defined. The SRH, which the ingress adds to each IPv6 packet, stores IPv6 path constraint information (segment lists) to specify an IPv6 explicit path. Transit nodes forward the packets according to the path information contained in the SRH. Figure 4-2 shows the SRH format.

Figure 4-2 SRH format

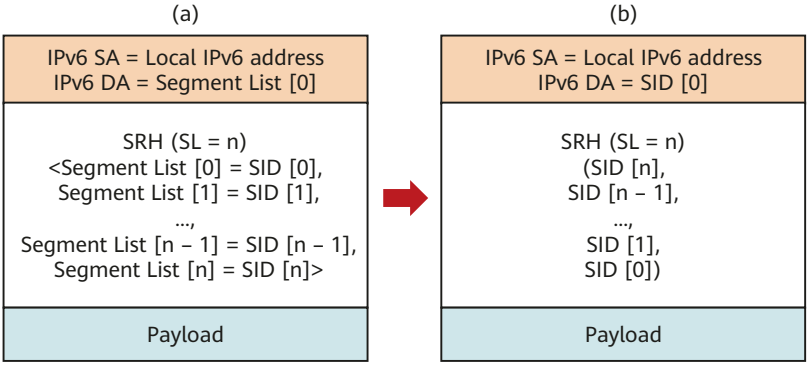


The key information in the IPv6 SRH is as follows:

1. Routing Type: If the value of this field is 4, the packet header is an SRH.
2. Segment List (Segment List [0], Segment List [1], Segment List [2], ..., Segment List [n]): This field indicates network path information.
3. Segments Left (SL): This field is a pointer that indicates the currently active segment.

To make it easier to explain data forwarding, the SRH can be expressed using an abstract format shown in Figure 4-3. The SIDs in (a) are listed in forward order and identified using < >, whereas those in (b) are listed in reverse order and identified using ( ). The reverse order more closely represents actual SRv6 packet encapsulation.

Figure 4-3 Abstract SRH



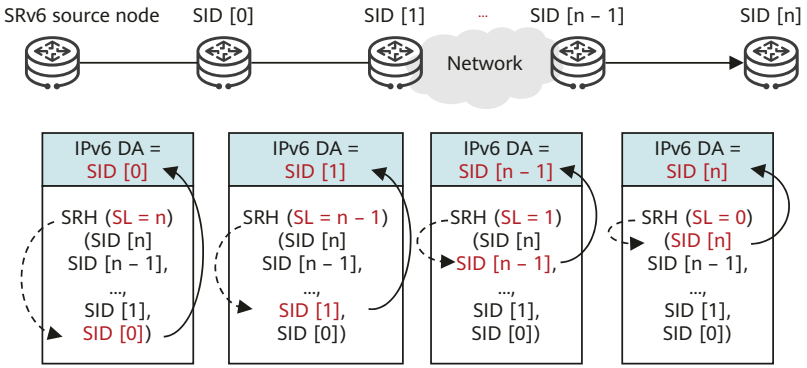
SRv6 SRH Processing

In an SRv6 SRH, the SL pointer and segment list information are used together to determine the IPv6 Destination Address (DA) in the packet header. The value of the SL pointer ranges from 0 (minimum) to the number of SIDs in the SRH minus 1 (maximum). As shown in Figure 4-4, each time a packet passes through an SRv6 node, the value of the SL field is decremented by 1 and the IPv6 DA is updated to the SID that the pointer currently points to.

- If the SL value is n, the IPv6 DA value is equal to the Segment List [n] value.
- If the SL value is n – 1, the IPv6 DA value is equal to the Segment List [n – 1] value.
- ...
- If the SL value is 1, the IPv6 DA value is equal to the Segment List [1] value.
- If the SL value is 0, the IPv6 DA value is equal to the Segment List [0] value.

If a node does not support SRv6, it searches the IPv6 routing table based on the longest match rule for packet forwarding instead of performing the preceding actions.

Figure 4-4 SRH processing



According to the above description, a node operates the SRv6 SRH from the bottom up, which is different from what the node does in SR-MPLS scenarios.

Another difference between SRv6 and SR-MPLS is that a node does not pop segments in the SRv6 SRH after processing them. This is mainly due to the following three reasons:

1. The initial design of the IPv6 RH was not closely related to MPLS, causing the unavailability of the pop option at the time.
2. In contrast to MPLS labels that are independently placed on the top of packets and can therefore be directly removed, SRv6 segments are placed in

the SRH following the IPv6 header and associated with other extension header information, such as security encryption and verification information. Consequently, SRv6 segments cannot be popped.

3. Because the pop operation is not performed, the SRv6 header retains path information that can be used for path backtracking. In addition, some innovative designs attempt to achieve new function extensions by reusing segments retained in the SRH.

4.3 What Makes SRv6 SIDs Distinctive?

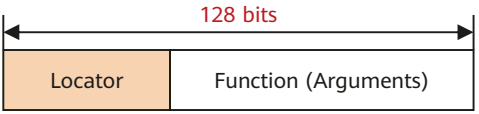
SIDs are expressed using the label format in SR-MPLS and using the IPv6 address format in SRv6. Because SRv6 completes forwarding through SID stack operations, it is regarded as a source routing technology. In order to understand what makes SRv6 SIDs distinctive, you first need to know the SRv6 SID structure.

SRv6 SID Structure

Although SRv6 SIDs use the IPv6 address format, they are not typical IPv6 addresses. Each SRv6 SID has 128 bits, meaning that it can represent almost anything. In order to avoid wasting such a large address space only for route forwarding, SRv6 designers took a clever approach when designing SIDs.

As shown in Figure 4-5, an SRv6 SID usually consists of the Locator and Function parts, which are expressed in the *Locator:Function* format. The Locator part occupies the most significant bits in the IPv6 address, and the Function part occupies the remaining bits.

Figure 4-5 SRv6 SID structure



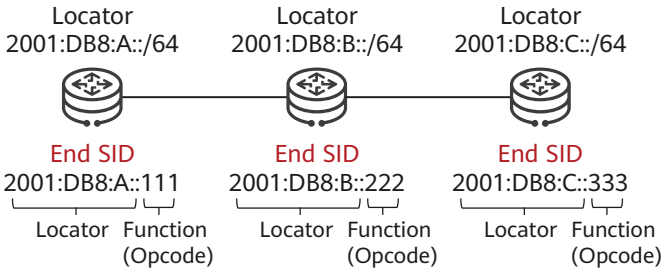
- The Locator part provides the location function, and each locator value is generally unique in an SRv6 domain (the same locator value may be configured for multiple devices in certain scenarios, such as when anycast protection is configured). After a locator value is configured for a node, the system generates a locator route and propagates it throughout the SRv6 domain using an IGP, allowing other nodes on the network to locate that node based on the received route. In addition, all SRv6 SIDs advertised by that node are reachable through the route.
- The Function part identifies an instruction pre-defined on the node that generates the SRv6 SID. It is used to instruct the node to perform the corresponding operation and is explicitly represented by an operation code (opcode).
- The optional Arguments field can be divided from the Function part, in which case the SRv6 SID is expressed in the *Locator:Function:Arguments* format. This field occupies the least significant bits of the IPv6 address and is used to define relevant information, such as packet flow and service information. Currently, one of the important applications is to use the Arguments field to implement split horizon during Broadcast, Unknown-unicast, and Multicast (BUM) traffic forwarding in EVPN VPLS CE multi-homing scenarios.

Because both the Function and Arguments parts can be defined, the SRv6 SID structure enhances network programmability.

The following uses End and End.X SIDs as examples to describe the SRv6 SID structure.

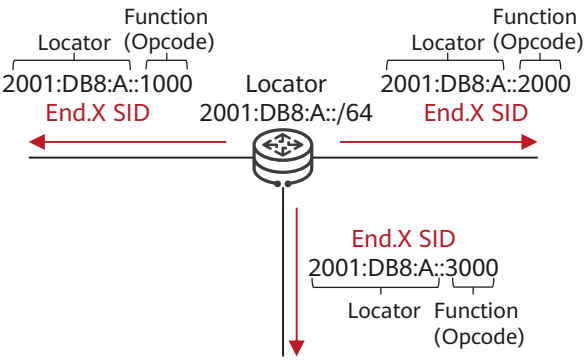
An End SID is an endpoint SID that identifies a destination node. In Figure 4-6, a locator and an opcode (the opcode that represents the specific function) are configured for each node. The locator and opcode on a node are combined to form an End SID, which can be used to represent the node. After an End SID is generated on a node, the node propagates the SID to the other nodes in the SRv6 domain through an IGP. In this way, all nodes in the domain obtain the SID.

Figure 4-6 End SID



An End.X SID is a Layer 3 cross-connect endpoint SID that identifies a link. In Figure 4-7, a locator is configured for each node, and an opcode that represents the specific function is configured for the adjacency in each direction. The locator and opcode are combined to form an End.X SID, which can be used to represent the corresponding adjacency. After an End.X SID is generated on a node, the node propagates the SID to the other nodes in the SRv6 domain through an IGP. In this way, all nodes in the domain obtain the SID.

Figure 4-7 End.X SID



An End SID and an End.X SID represent a node and an adjacency, respectively. They are both path SIDs and can be encapsulated into a SID stack to represent

any network path. The SID stack represents path constraints and is carried in the IPv6 SRH to help implement SRv6 TE.

In addition, SIDs can be allocated to VPN, EVPN, and Ethernet Virtual Private Line (EVPL) instances. In this case, these SIDs represent services. Because the IPv6 address space is sufficiently large, SRv6 SIDs can support numerous services.

Common SRv6 SIDs

Currently, SRv6 SIDs are mainly classified as path or service SIDs. For example, End and End.X SIDs represent nodes and links, respectively, whereas End.DT4 and End.DT6 SIDs represent IPv4 and IPv6 VPN instances, respectively.

With the development of services, more service SIDs are introduced. Table 4-2 describes the commonly used ones.

Table 4-2 Common SRv6 SIDs

SID	Description	Protocol	Type
End SID	Indicates an endpoint SID that identifies a destination node. The corresponding function is to update the IPv6 DA and then search the IPv6 Forwarding Information Base (FIB) for packet forwarding.	IGP	Path SID
End.X SID	Indicates a Layer 3 cross-connect endpoint SID that identifies a link. The corresponding function is to update the IPv6 DA and then forward packets through the outbound interface bound to the SID.	IGP	Path SID
End.DT4 SID	Indicates a PE-specific endpoint SID that identifies an IPv4 VPN instance. The corresponding function is to decapsulate packets and then search the routing table of the involved IPv4 VPN instance for packet forwarding. This SID is equivalent to an IPv4 VPN label and used in L3VPNv4 scenarios.	BGP	Service SID
End.DT6 SID	Indicates a PE-specific endpoint SID that identifies an IPv6 VPN instance. The corresponding function is to decapsulate packets and then search the routing table of the involved IPv6 VPN instance for packet forwarding.	BGP	Service SID

SID	Description	Protocol	Type
	This SID is equivalent to an IPv6 VPN label and used in L3VPNv6 scenarios.		
End.DX4 SID	Indicates a PE-specific Layer 3 cross-connect endpoint SID that identifies an IPv4 CE. The corresponding function is to decapsulate packets and then forward the resulting IPv4 packets through the Layer 3 interface bound to the SID. This SID is equivalent to a label identifying an adjacency to a CE and used in L3VPNv4 scenarios.	BGP	Service SID
End.DX6 SID	Indicates a PE-specific Layer 3 cross-connect endpoint SID that identifies an IPv6 CE. The corresponding function is to decapsulate packets and then forward the resulting IPv6 packets through the Layer 3 interface bound to the SID. This SID is equivalent to a label identifying an adjacency to a CE and used in L3VPNv6 scenarios.	BGP	Service SID
End.DX2 SID	Indicates a Layer 2 cross-connect endpoint SID that identifies an endpoint. The corresponding function is to decapsulate packets, remove the IPv6 header (along with all its extension headers), and then forward the remaining packet data to the outbound interface associated with the SID. This SID can be used in EVPN VPWS scenarios. If a bypass tunnel exists on the network, an End.DX2L SID is generated automatically.	BGP	Service SID
End.DT2U	Indicates a Layer 2 cross-connect endpoint SID that requires unicast MAC table lookup and identifies an endpoint. The corresponding function is to remove the IPv6 header (along with all its extension headers), search the MAC address table for a MAC entry based on the exposed destination MAC address, and then forward the remaining packet data to the corresponding outbound interface based on the entry. This SID can be used in EVPN VPLS unicast scenarios. If a bypass tunnel exists on the network, an End.DT2UL SID is generated automatically. This SID can be used to guide unicast traffic forwarding over the bypass tunnel	BGP	Service SID

SID	Description	Protocol	Type
	when a CE is dual-homed to PEs.		
End.DT 2M SID	Indicates a Layer 2 cross-connect endpoint SID that requires broadcast-based flooding and identifies an endpoint. The corresponding function is to remove the IPv6 header (along with all its extension headers) and then broadcast the remaining packet data in the Bridge Domain (BD). This SID can be used in EVPN VPLS BUM scenarios.	BGP	Service SID
End.OP SID	Indicates an OAM SID. The corresponding function is to send OAM packets to the OAM process. This SID is mainly used in ping/tracert scenarios.	IGP	Service SID

Local SID Table

Each SRv6 node maintains a local SID table that contains all SRv6 SIDs generated on the node, and an SRv6 FIB can be generated based on the table. A local SID table provides the following functions:

- Defines locally generated SIDs, such as End.X SIDs.
- Specifies instructions bound to the SIDs.
- Stores forwarding information related to the instructions, such as outbound interface and next hop information.

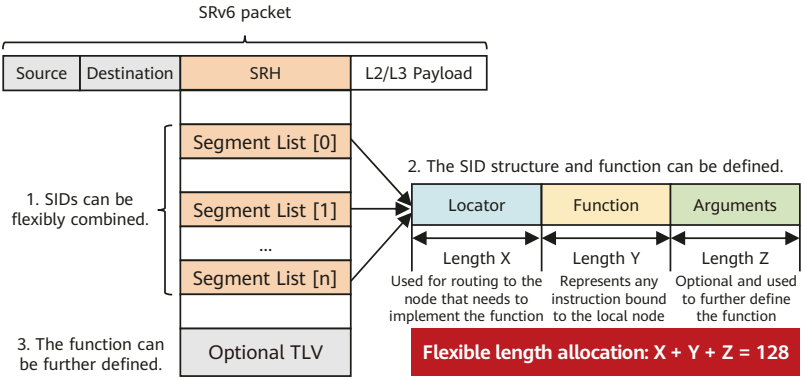
4.4 SRv6-enabled Three-dimensional Programming Space

SRv6 gives more meaning to SIDs, enabling them to represent any user-defined functions in addition to paths and different types of services. As such, SRv6 SIDs enhance network programmability.

SRv6 supports a three-dimensional programming space, as shown in Figure 4-8.

1. SRv6 SIDs can be flexibly combined for path programming. After a service requirement is raised, the controller responds to the requirement and defines a forwarding path accordingly. This approach complies with the SDN paradigm. For example, company A needs to provision services for company B within a month, requiring a large amount of data to be exchanged. Bandwidth must therefore be guaranteed immediately. To achieve this, company A needs to purchase corresponding services from a carrier for one month. Traditionally, service provisioning usually takes months to complete because multiple departments need to work together, meaning that company A will fail to meet the deadline. In contrast, SRv6 path programming enables the carrier's controller to quickly respond to the company's requirement, compute SLA-compliant service paths, and quickly complete service provisioning. Furthermore, the carrier can quickly tear down relevant connections to release network resources after the one-month period expires.

Figure 4-8 SRv6-enabled three-dimensional programming space



2. The Function and Arguments fields can be defined. Device vendors can define the Function field at present, although users will also be able to define it in the future. For example, a device vendor can define this field to instruct an SRv6 egress to forward a received data packet to a VPN instance,

while a user can define it to instruct an SRv6 node to forward a received data packet to an application. Because Linux supports SRv6, different functions can be defined in the future to support a variety of new services based on Linux.

3. The SRH contains optional TLVs to further define functions, for example, carrying the In-situ Flow Information Telemetry (iFIT) instruction header.

4.5 How Is SRv6 Implemented Through Protocol Extensions?

To support SRv6, network nodes need to advertise the following two types of SRv6 information:

1. Locator information: enables other nodes on the network to locate the node that advertises a SID. In this way, the other nodes can execute the instruction bound to the SID. Intra-area locator information is usually propagated through IGP extensions.
2. SID information: completely describes the functions of SIDs, for example, behaviors bound to SIDs. SIDs are classified into path SIDs and service SIDs, both of which are globally visible but locally effective. Path SIDs are mainly used to describe nodes or links and need to be propagated through IGP extensions, whereas service SIDs are closely related to routing information and generally advertised through BGP extensions in BGP Update messages.

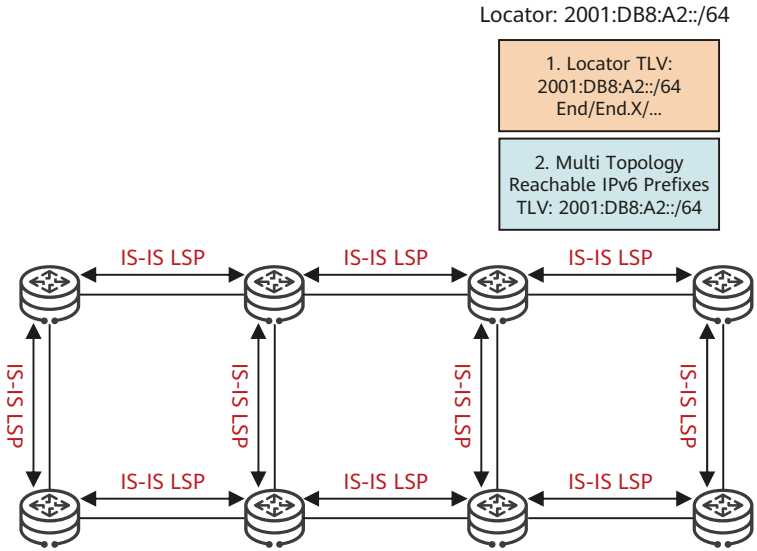
In conclusion, implementing basic SRv6 functions requires at least IGP and BGP extensions.

IGP Extensions

A link-state routing protocol computes the shortest path to a specified address using Dijkstra's Shortest Path First (SPF) algorithm through the following process: Adjacent nodes establish neighbor relationships by exchanging Hello packets and flood their local Link-State PDUs (LSPs) on the entire network to form an identical Link-State Database (LSDB). Each node runs the SPF algorithm based on the LSDB to compute routes.

Figure 4-9 shows how IS-IS SRv6 TLVs are advertised.

Figure 4-9 IS-IS SRv6 TLV advertisement



IS-IS uses the following two TLVs with different functions to advertise a locator's routing information:

1. SRv6 Locator TLV: contains the locator's prefix and mask. This TLV is used to advertise the locator information and enables other SRv6 nodes on the network to learn the locator route. In addition to carrying information used to guide routing, the TLV also carries the SRv6 SIDs that do not need to be associated with IS-IS neighbors, for example, End SIDs.
2. Multi Topology Reachable IPv6 Prefixes TLV: carries the same IPv6 prefix and mask as the locator information carried in the SRv6 Locator TLV. This TLV is an existing IS-IS TLV, which can also be processed by common IPv6 nodes (SRv6-incapable nodes). As such, a common IPv6 node can also use this TLV to generate a locator route, which guides packet forwarding to the node that advertises the corresponding locator. This means that common IPv6 nodes can be deployed together with SRv6 nodes on the same network.

If a device receives both a Multi Topology Reachable IPv6 Prefixes TLV and an SRv6 Locator TLV, the Multi Topology Reachable IPv6 Prefixes TLV is preferentially used.

Table 4-3 describes IS-IS TLV extensions for SRv6.

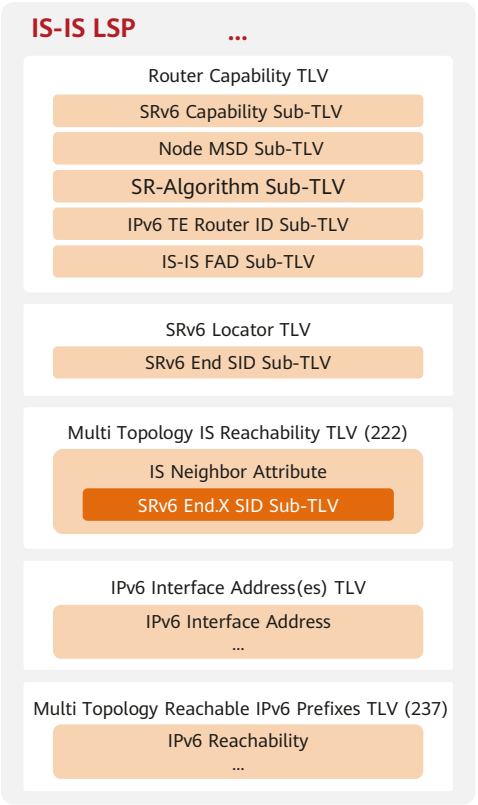
Table 4-3 IS-IS TLV extensions for SRv6

Name	Function	Carried In
SRv6 Locator TLV	Advertises an SRv6 locator and its associated End SIDs.	IS-IS LSP
SRv6 Capabilities sub-TLV	Advertises SRv6 capabilities.	IS-IS Router Capability TLV-242
SRv6 End SID sub-TLV	Advertises SRv6 SIDs.	SRv6 Locator TLV
SRv6 End.X SID sub-TLV	Advertises SRv6 SIDs associated with Point-to-Point (P2P) adjacencies.	IS-IS Extended IS reachability TLV-22 IS-IS IS Neighbor Attribute TLV-23 IS-IS inter-AS reachability information TLV-141 IS-IS Multitopology IS TLV-222 IS-IS Multitopology IS Neighbor Attribute TLV-223
SRv6 LAN End.X SID sub-TLV	Advertises SRv6 SIDs associated with Local Area Network (LAN) adjacencies.	IS-IS Extended IS reachability TLV-22 IS-IS IS Neighbor Attribute TLV-23 IS-IS inter-AS reachability information TLV-141 IS-IS Multitopology IS TLV-222 IS-IS Multitopology IS Neighbor Attribute TLV-223
Node MSD sub-TLV	Advertises the Maximum SID Depth (MSD) supported by a device.	IS-IS Router Capability TLV-242

Name	Function	Carried In
IS-IS FAD sub-TLV	Advertises a Flex-Algo Definition (FAD).	IS-IS Router Capability TLV-242
SR-Algorithm sub-TLV	Advertises the in-use algorithm.	IS-IS Router Capability TLV-242

Figure 4-10 shows the structure of a common IS-IS LSP carrying SRv6 information.

Figure 4-10 IS-IS LSP carrying SRv6 information

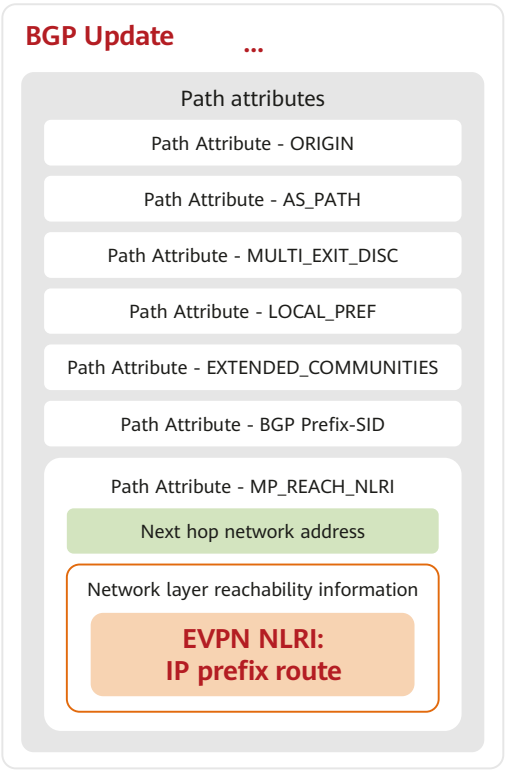


BGP Extensions

BGP extensions include MP-BGP and BGP EVPN. Both L2VPN and L3VPN service SIDs need to be advertised using BGP Update messages.

Figure 4-11 shows the structure of a common BGP EVPN Update message carrying SRv6 information.

Figure 4-11 BGP EVPN Update message carrying SRv6 information



4.6 How Does SRv6 Ensure High Reliability?

High-value services require IP bearer networks to provide high availability. For example, high-quality enterprise private lines, such as those in government, finance, and healthcare sectors, usually require a high availability of 99.99%. In contrast, 5G services, especially Ultra-Reliable Low-Latency Communication (URLLC) services, require an availability of 99.999%. In other cases closely related to social and human safety, such as remote control and high-voltage power supply, these mission-critical services require an extremely high availability of 99.9999%.

Fault recovery within 50 ms has become a basic requirement for IP bearer networks. For example, traditional voice and Internet Protocol Television (IPTV) services require fault recovery to be completed within milliseconds. If fault recovery takes much longer than this (even just a few seconds), services will be severely affected. 5G Enhanced Mobile Broadband (eMBB) and URLLC services have stricter requirements on the E2E latency. For example, eMBB services such as smart home, Virtual Reality (VR), and Augmented Reality (AR) require the latency to be within 10 ms, whereas URLLC services such as autonomous driving, telemedicine, smart energy, and smart manufacturing require the latency to be within 1 ms.

SRv6 provides local protection technologies against E2E failure points on IP networks, thereby achieving local protection switching within 50 ms for any topology. If a failure occurs on an SRv6 network, the node adjacent to the failure point switches traffic to a sub-optimal path, and then route convergence is performed level by level to converge the traffic to the optimal path. SRv6's unique local protection technologies such as TI-LFA and midpoint protection significantly enhance the reliability and fault protection capabilities of IP bearer networks.

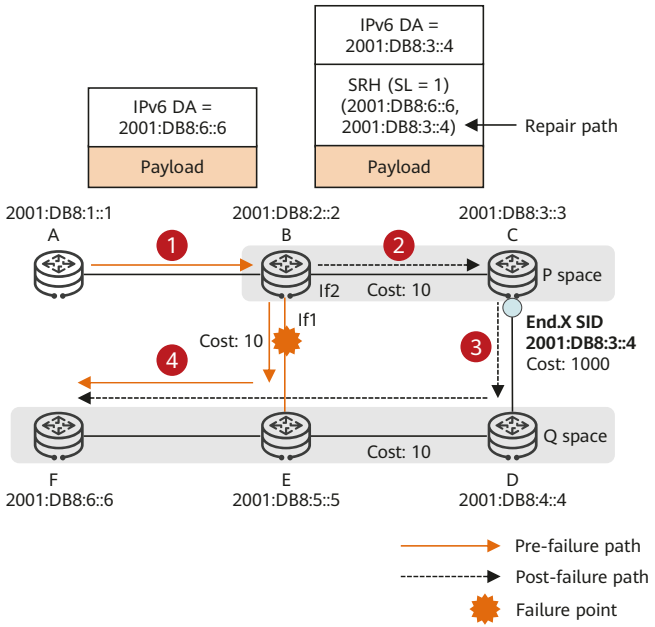
TI-LFA

Figure 4-12 illustrates how SRv6 TI-LFA works. In this figure, the shortest path from node A to node F is A->B->E->F. Node B needs to compute a backup path to node F as follows:

1. Excludes the primary next hop (B-E link) and computes the post-convergence shortest path A->B->C->D->E->F.

2. Computes the extended P space, which is a set of nodes reachable from all the neighbors of a protected link's source node using Shortest Path Trees (SPTs) rooted at the neighbors without traversing the protected link. All nodes within the extended P space are P nodes, which are {node B, node C} in this example.

Figure 4-12 SRv6 TI-LFA protection



3. Computes the Q space, which is a set of nodes reachable from the destination node of a protected link using the reverse SPT rooted at the destination node without traversing the protected link. All nodes in the Q space are Q nodes, which are {node D, node E, node F} in this example.
4. Computes a backup path expressed using a repair segment list. Any path can be represented using the source node->P node->Q node->destination node format. Both the path from the source node to the P node and that from the Q node to the destination node are loop-free. If a PQ node (a node in both the extended P space and Q space) exists, this node can be reached from the source node and can reach the destination node without traversing the

failed path. As such, traffic can be directly forwarded to the PQ node, and the entire path is loop-free. In this case, the repair segment list can be composed of the PQ node's End SID. If no PQ node exists, however, a loop-free forwarding path from the P node to the Q node needs to be specified, and the repair segment list from the P node to the Q node may be a combination of End and End.X SIDs. In this example, the repair segment list from node B's furthest P node (C) to its nearest Q node (D) can be End.X SID 2001:DB8:3::4.

To activate the alternate next hop if the primary next hop fails, the Point of Local Repair (PLR), such as node B in this example, pre-installs the backup forwarding entries (listed in Table 4-4) in its forwarding table according to the TI-LFA computation result, thereby ensuring the reachability to destination node F.

Table 4-4 TI-LFA backup forwarding entries on node B

Route Prefix	Outbound Interface	Segment List	Role
2001:DB8:6::6	If1	-	Primary
	If2	2001:DB8:3::4	Backup

If the B-E link fails, the data forwarding process is as follows:

1. After receiving a packet destined for 2001:DB8:6::6, node B searches the forwarding table based on 2001:DB8:6::6 and finds that the primary outbound interface is If1.
2. Node B detects that the If1 interface is down and therefore uses the backup entry to forward the packet through the backup outbound interface If2. The node also executes the H.Insert behavior: adding an SRH containing the segment list 2001:DB8:3::4 and destination address 2001:DB8:6::6, and initializing the SL value to 1.
3. After receiving the packet, node C finds that 2001:DB8:3::4 is an End.X SID and therefore executes the instruction corresponding to the End.X SID. Specifically, the node decrements the SL value by 1, updates the outer IPv6 address to 2001:DB8:6::6, and forwards the packet to node D along the C-D link based on the outbound interface and next hop that are bound to 2001:DB8:3::4. Because the SL value is now 0, node C can remove the SRH as instructed by the Penultimate Segment Pop of the SRH (PSP) flavor.

4. After receiving the packet, node D searches its IPv6 routing table based on the destination address 2001:DB8:6::6 and then forwards the packet to destination node F over the shortest path.

In conclusion, TI-LFA offers the following benefits in addition to 100% topology protection:

1. In most cases, the TI-LFA backup path is consistent with the post-convergence shortest path due to TI-LFA performing computation based on the post-convergence shortest path. This reduces the number of forwarding path switchovers.
2. Computation of the TI-LFA backup path depends on IGP SRv6, reducing the number of protocols required for deploying reliability technologies.
3. TI-LFA uses existing End, End.X, or a combination of both SIDs to establish a backup path, without needing to maintain additional forwarding states.

SRv6 Midpoint Protection

SRv6 midpoints need to decrement the SL value by 1 and copy the next SID to the DA field in the IPv6 header during SRv6 packet processing. If a midpoint fails, it cannot execute the instruction bound to the corresponding SID, resulting in a forwarding failure.

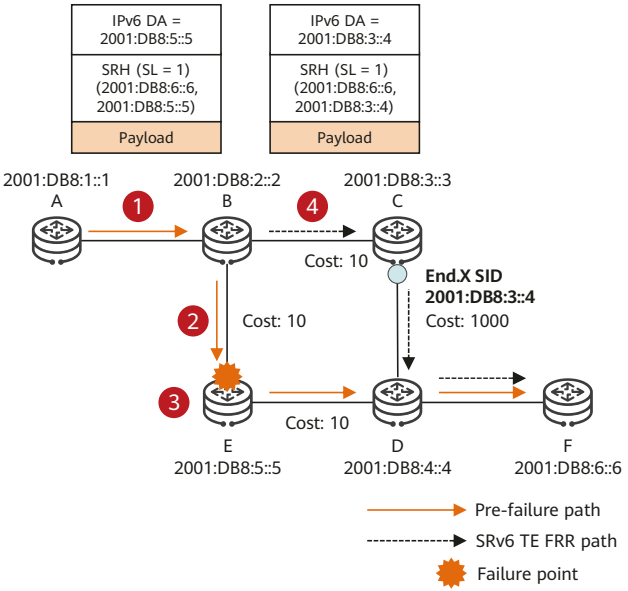
To complete the forwarding and thereby eliminate this issue, a proxy forwarding node (a node upstream to the failed midpoint) needs to take over from the failed midpoint. Specifically, the proxy forwarding node performs the End behavior on behalf of the failed midpoint after detecting that the next-hop interface of the packet fails, the next-hop address is the destination address of the packet, and the SL value is greater than 0. The behavior involves decrementing the SL value by 1, copying the next SID to the DA field in the outer IPv6 header, and then forwarding the packet according to the instruction bound to the SID. This approach bypasses the failed midpoint, achieving SRv6 midpoint protection.

Figure 4-13 shows SRv6 midpoint protection. The detailed process is described as follows:

1. Node A forwards a packet to destination node F, with an SRv6 SRH instructing the packet to pass through node E.

2. If node E fails, node B detects that the next-hop interface of the received packet fails. It also learns that the next-hop address is the current destination address 2001:DB8:5::5 of the packet, and the SL value is greater than 0. In this case, node B functions as a proxy forwarding node. Specifically, it decrements the SL value by 1 and copies the next SID 2001:DB8:6::6 to the DA field in the outer IPv6 header. Because the SL value is now 0, node B can remove the SRH and then search the forwarding table to forward the packet based on the destination address 2001:DB8:6::6.

Figure 4-13 SRv6 midpoint protection



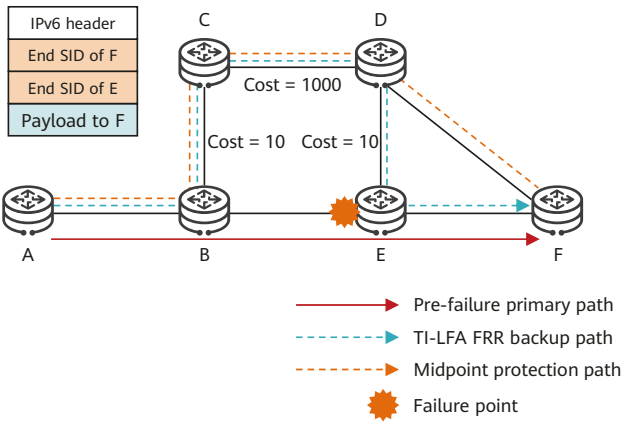
3. Because the primary next hop to the destination address 2001:DB8:6::6 is still node E, node B is not the penultimate hop of the destination address, and the SL value is 0, node B can no longer perform proxy forwarding. Instead, it switches the packet to a backup path according to the normal TI-LFA process, with the repair segment list on the backup path as <2001:DB8:3::4>. As such, node B executes the H.Insert behavior, which involves encapsulating the segment list 2001:DB8:3::4 into the packet, adding an SRH, and then forwarding the packet to node F over the backup path.

4. After detecting that node E fails and IGP convergence is completed, node A deletes the forwarding entry towards node E. Consequently, it cannot find any matching route when searching its forwarding table based on 2001:DB8:5::5. In this case, node A needs to function as a proxy forwarding node to perform proxy forwarding. It therefore decrements the SL value by 1, copies the next SID 2001:DB8:6::6 to the outer IPv6 header, searches the forwarding table based on the destination address 2001:DB8:6::6, and then forwards the packet to node B accordingly. If node B has already converged, it forwards the packet to node F over the post-convergence shortest path; otherwise, it forwards the packet to node F over the backup path according to the TI-LFA process. Through this process, the failed node E is bypassed.

It may be difficult to understand the difference between TI-LFA and SRv6 midpoint protection. In essence, their difference mainly lies in whether the next-hop node is a transit node or an SRv6 midpoint in the destination address.

On the network shown in Figure 4-14, node A sends a packet carrying the segment list <E, F>. Because TI-LFA computes a backup path based on the destination address of the packet, the computed path passes through node E. If node E fails, TI-LFA cannot provide protection. In an SRv6 midpoint protection scenario, however, a backup path is computed based on the next SID to be processed. This ensures that the failed midpoint is bypassed, thereby implementing SRv6 TE Policy midpoint protection.

Figure 4-14 Difference between TI-LFA and midpoint protection



Chapter 5
SRv6 Working Modes

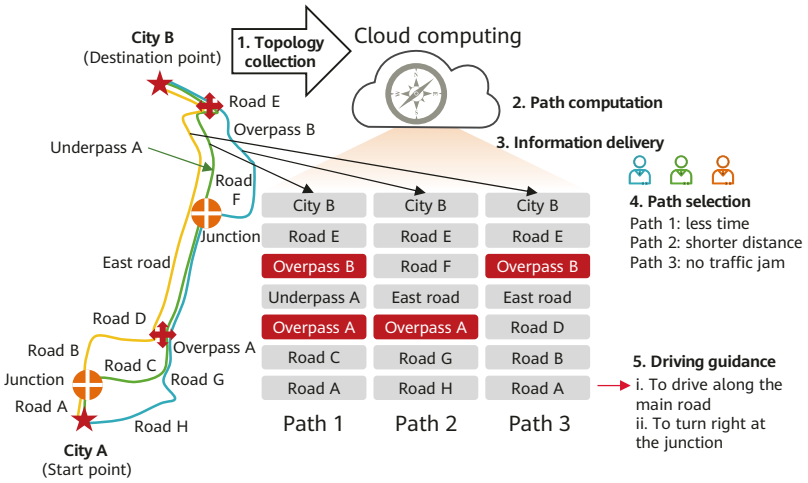
This chapter describes two SRv6 working modes: SRv6 TE Policy and SRv6 Best Effort (BE). Both modes can be used to carry common traditional services, such as BGP L3VPN, EVPN L3VPN, EVPN VPLS/VPWS, and IPv4/IPv6 public network services. In addition to implementing traffic engineering, SRv6 TE Policy can work with a controller to meet differentiated service requirements more effectively, achieving a service-driven network. SRv6 BE is a simplified implementation of SRv6. Typically, it can provide only best-effort forwarding and does not involve SRHs. During the early development of SRv6, SRv6 BE was used to quickly provision services based on IPv6 route reachability, offering unparalleled advantages. In future evolution, transit nodes on the network can be upgraded on demand and SRv6 TE Policy can be deployed to meet the requirements of high-value services.

5.1 SRv6 TE Policy

What Is SRv6 TE Policy?

SRv6 TE Policy leverages the source routing mechanism of SR to guide packet forwarding based on an ordered list of segments encapsulated by the headend. The design of SRv6 TE Policy can be compared to many common scenarios. The following uses a navigation map, as shown in Figure 5-1, to further explain SRv6 TE Policy.

Figure 5-1 Working process of a navigation map



The entire working process can be summarized into five steps:

1. Topology collection: collects information about junctions, lanes, traffic rates, and traffic lights.
2. Path computation: computes paths based on multiple constraints and multi-dimensional SLAs, such as lowest fees, shortest time, shortest distance, and freeway-preferred.

3. Information delivery: delivers the computed path information to the corresponding user terminal.
4. Path selection: requests the user to select a path according to the user's destination address and preference. Each path is a combination of multiple roads and key junctions.
5. Driving guidance: guides driving based on information about each segment of the path.

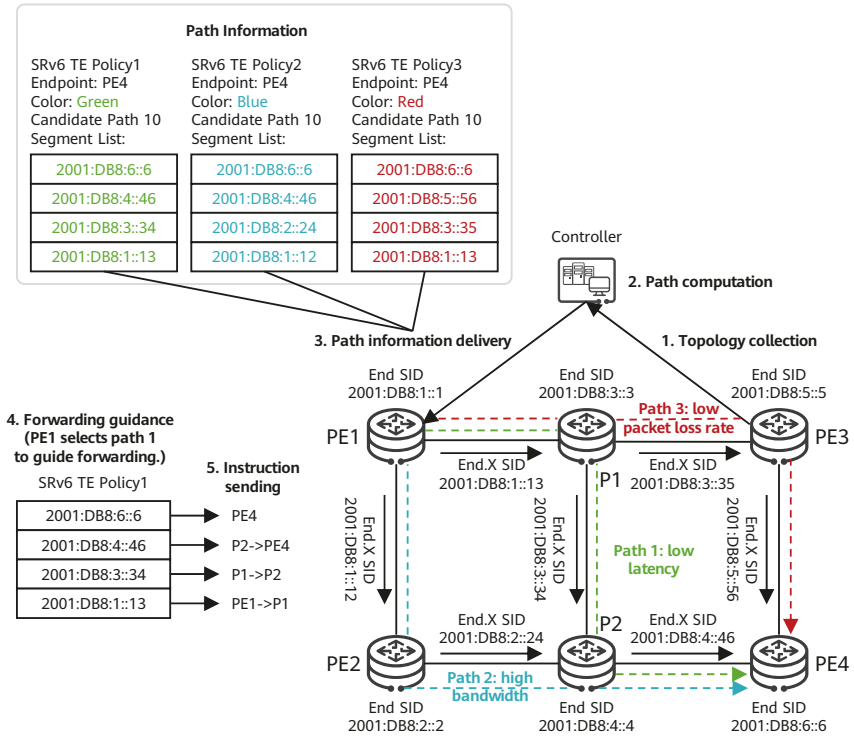
   Before the user approaches a junction, an instruction is sent to ask the user to take an action, such as going straight, turning left, turning right, or making a U-turn.

The working process of SRv6 TE Policy on a network is similar to that of the navigation map. Figure 5-2 shows how SRv6 TE Policy works.

The working process of SRv6 TE Policy also consists of five steps:

1. A forwarder reports network topology information to a network controller through BGP-LS. The information includes TE attributes such as node information (similar to junction information), link information (similar to road information), link cost (similar to the traffic rate), bandwidth (similar to lane information), and latency (similar to traffic light information).
2. The controller analyzes the topology information and computes SLA-compliant paths according to service requirements.
3. The controller uses a BGP SR-Policy extension to deliver path information to the headend, which then generates SRv6 TE Policies containing key information such as the headend address, destination address, and color.
4. The headend selects an appropriate SRv6 TE Policy to guide forwarding.
5. The forwarder executes the instruction bound to the SID advertised by itself to forward data.

Figure 5-2 Working process of SRv6 TE Policy



As shown in Figure 5-2, a series of SRv6 SIDs can be encapsulated into an SRH to explicitly guide packet forwarding along a planned path. This implements fine-grained E2E control over the forwarding path and meets SLA requirements such as low latency, high bandwidth, and high reliability. If the destination address of a service matches the endpoint of an SRv6 TE Policy and the service preference (identified by the color extended community attribute of the corresponding route) is the same as that of the SRv6 TE Policy, the service can be steered to the SRv6 TE Policy for forwarding.

SRv6 uses programmable 128-bit IPv6 addresses to provide diverse network functions, which are expressed using SRv6 instructions that can identify both forwarding paths and Value-Added Service (VAS) devices such as firewalls,

application acceleration gateways, and user gateways. Another prominent feature of SRv6 is its excellent extensibility. Specifically, by simply defining a new instruction, SRv6 can support a new network function without requiring any protocol mechanism or deployment changes. This capability significantly accelerates the rollout of innovative network services. As such, SRv6 TE Policy can meet E2E service requirements and is crucial to SRv6 network programming.

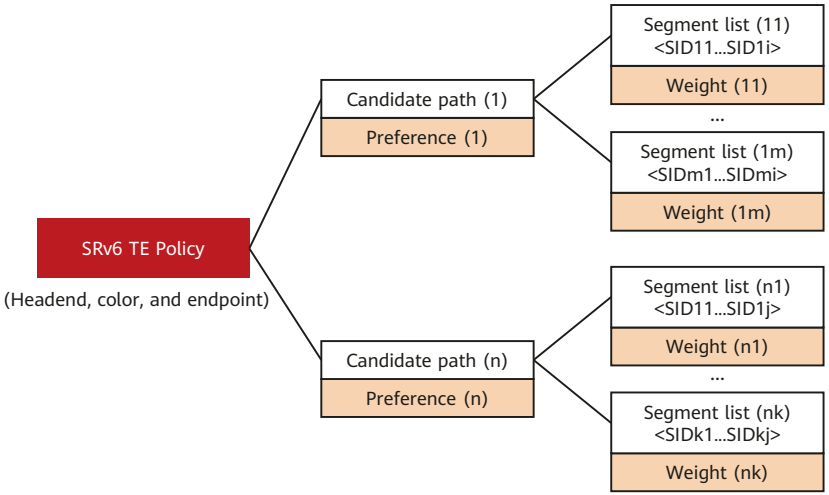
SRv6 TE Policy Structure and Advantages

The SRv6 TE Policy structure is designed to achieve improved reliability and bandwidth utilization, as shown in Figure 5-3.

The SRv6 TE Policy structure consists of the following three elements:

1. Headend: node where an SRv6 TE Policy originates.
2. Color: extended community attribute of an SRv6 TE Policy. A BGP route can recurse to an SRv6 TE Policy if they have the same color attribute.
3. Endpoint: destination address of an SRv6 TE Policy.

Figure 5-3 SRv6 TE Policy structure



The SRv6 TE Policy structure has the following advantages:

1. Flexible traffic steering: Color and endpoint information is added to an SRv6 TE Policy through configuration. The headend steers traffic into an SRv6 TE Policy whose color and endpoint match the color value and next-hop address in the associated route, respectively. The color attribute defines an application-level network SLA policy. This allows network paths to be planned for services according to specific SLA requirements, realizing fine granularity of service value and helping build new business models.
2. High reliability: One SRv6 TE Policy can contain multiple candidate paths with the preference attribute. The valid candidate path with the highest preference functions as the primary path of the SRv6 TE Policy, and that with the second highest preference functions as a backup path.
3. Load balancing: One candidate path can contain multiple segment lists that support both Equal-Cost Multi-Path (ECMP) and Unequal Cost Multi-Path (UCMP) modes. Each segment list carries a weight attribute and functions as an explicit SID stack that instructs a network device to forward packets.

SRv6 TE Policy Implementation

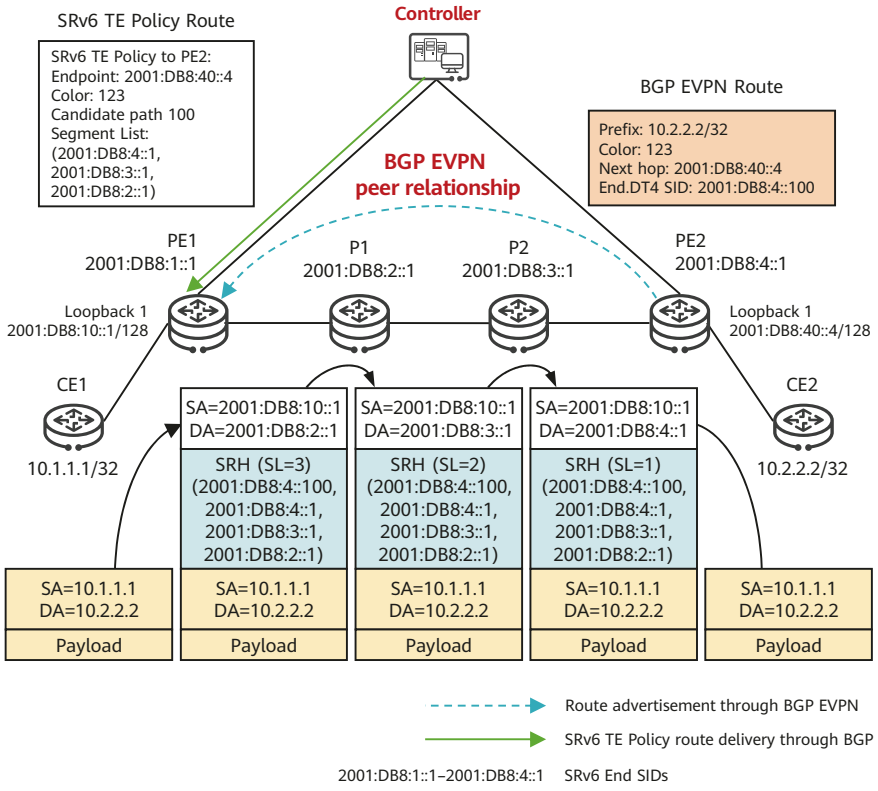
SRv6 TE Policies can carry common traditional services, which are all forwarded in a similar manner. The following example uses an EVPN L3VPNv4 over SRv6 TE Policy scenario to describe how SRv6 TE Policy is implemented.

Figure 5-4 shows the data forwarding process in this scenario. The forwarding process is as follows:

1. The controller delivers an SRv6 TE Policy carrying the color value 123 and endpoint address 2001:DB8:40::4 (PE2's address) to the headend PE1. The SRv6 TE Policy has only one candidate path, which contains only one segment list <2001:DB8:2::1, 2001:DB8:3::1, 2001:DB8:4::1>.
2. The endpoint PE2 advertises the BGP EVPN route 10.2.2.2/32 carrying the color value 123 and next-hop address 2001:DB8:40::4/128 (PE2's address) to PE1.
3. After receiving the route, PE1 recurses it to the corresponding SRv6 TE Policy based on its color and next-hop address.
4. After receiving a common unicast packet from CE1, PE1 searches the routing table of the corresponding VPN instance and finds that the matching VPN

route has been recursed to an SRv6 TE Policy. PE1 then inserts an SRH (which carries the SRv6 TE Policy's segment list whose last SID is the End.DT4 SID corresponding to the VPN route) into the packet, encapsulates an IPv6 header, searches the routing table, and forwards the packet accordingly.

Figure 5-4 SRv6 TE Policy-based data forwarding



5. Transit nodes P1 and P2 forward the packet hop by hop based on the SRH information.
6. After receiving the packet, PE2 searches the local SID table based on the IPv6 destination address 2001:DB8:4::1 in the packet and finds a matching

End SID. According to the instruction bound to the SID, PE2 decrements the SL value by 1 and updates the IPv6 DA field to the VPN SID 2001:DB8:4::100.

7. Based on this VPN SID, PE2 searches the local SID table and finds a matching End.DT4 SID. According to the instruction bound to the SID, PE2 removes the SRH and IPv6 header from the packet, searches the routing table of the VPN instance corresponding to the End.DT4 SID 2001:DB8:4::100 based on the destination address in the inner packet, and then forwards the packet to CE2.

SRv6 TE Policy Reliability Design

Figure 5-5 shows the SRv6 TE Policy reliability design in which TI-LFA can be used to protect the segment lists of a controller-computed SRv6 TE Policy. While this implementation is sufficient in most cases, it is recommended that an SRv6 BE path be used as the best-effort path of an SRv6 TE Policy in cases that require extreme reliability. This ensures that services are switched to the SRv6 BE path for best-effort forwarding if the SRv6 TE Policy fails. For details about SRv6 BE, see 5.2 SRv6 BE.

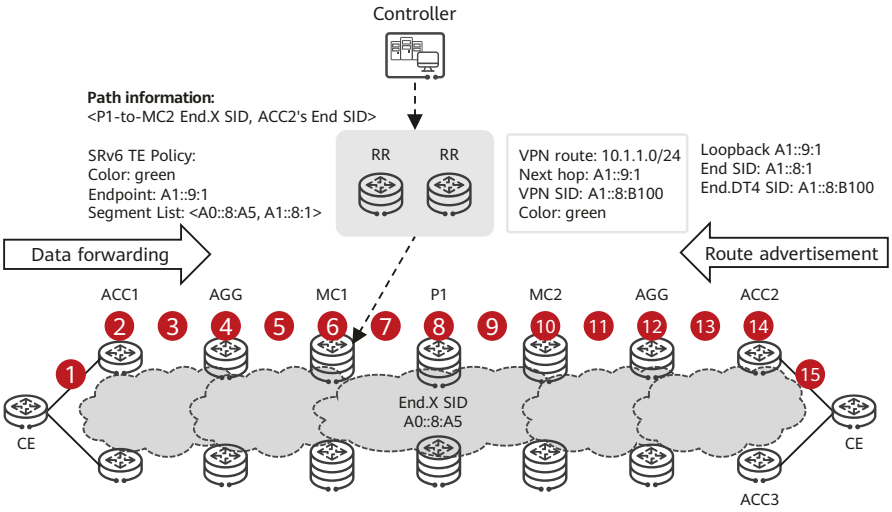
On the network shown in Figure 5-5, different technologies are used to protect services against failures at different locations. Specifically:

- For failure points 1 and 2, link detection can be used to detect failures, and ECMP or IP Fast Reroute (FRR) can be used to protect services.
- For failure points 3 to 7 and 10 to 12, because their SIDs are not in the segment list of the SRv6 TE Policy, TI-LFA FRR can be used to protect services, and link detection or Bidirectional Forwarding Detection (BFD) for IGP can be used to detect failures in order to trigger FRR switching.
- For failure points 8 and 9, because the SID of failure point 9 is in the segment list of the SRv6 TE Policy, midpoint protection can be used to protect services, and link detection or BFD for IGP can be used to detect failures in order to trigger FRR switching.
- For failure points 13 and 14, IP FRR or VPN FRR can be used to protect services, and link detection or BFD for IGP can be used to detect failures in order to trigger FRR switching.

For failure point 15, link detection can be used to detect failures. In L3VPN scenarios, ECMP or VPN-specific IP FRR is used to protect services. (VPN-specific

IP FRR is also called hybrid FRR, which allows traffic to be forwarded to ACC3 over a tunnel and then to the destination CE through VPN forwarding table lookups when the next hop of the route that originates from ACC2 to the CE is unreachable.) In EVPN scenarios, however, ECMP or local-remote-FRR is used to protect services. The latter allows traffic to be detoured to ACC3 from ACC2 and then to the destination CE to reduce packet loss if the link between ACC2 and the CE fails.

Figure 5-5 SRv6 TE Policy reliability design



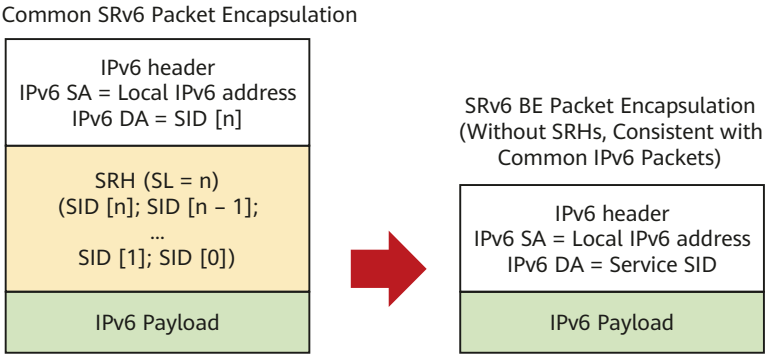
5.2 SRv6 BE

What Is SRv6 BE?

Traditional MPLS involves two control protocols: LDP and RSVP-TE. The former uses IGP path computation results to establish LDP LSPs and guide traffic forwarding, but it does not support traffic engineering. Similar to LDP, SRv6 BE uses only one service SID to guide packet forwarding on an IP network in best-effort mode.

As shown in Figure 5-6, SRv6 BE packets are not encapsulated with SRHs, which are used to represent path constraints. Because the format and forwarding behavior of SRv6 BE packets are the same as those of common IPv6 packets — meaning that common IPv6 devices can also process SRv6 BE packets — SRv6 is compatible with such devices.

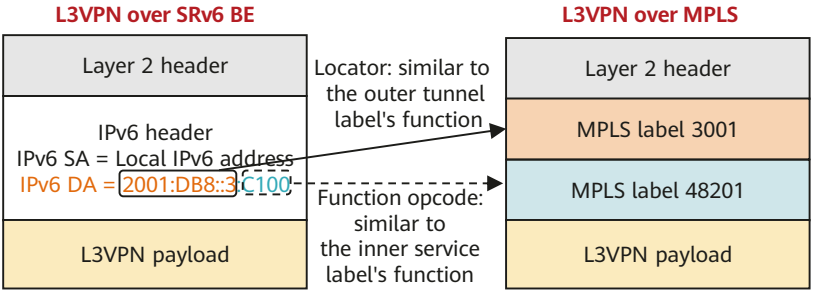
Figure 5-6 SRv6 BE packet encapsulation format



SRv6 BE packet encapsulation differs from common IPv6 packet encapsulation in that the destination address of a common IPv6 packet is a host or subnet, whereas that of an SRv6 BE packet is a service SID. The service SID can direct packet forwarding along the shortest path to the parent node where the SID is generated. After receiving a packet, the parent node executes the instruction corresponding to the service SID.

In L3VPN over MPLS scenarios, two MPLS labels are generally used: the outer label directs packets to a specified PE, while the inner label functions as a service label to identify a VPN instance on the PE. In L3VPN over SRv6 BE scenarios, however, only one SRv6 service SID is needed to implement the functions of both MPLS labels. Take Figure 5-7 as an example, in which the service SID is 2001:DB8:3::C100 (locator: 2001:DB8:3::/64; function opcode: ::C100). The locator 2001:DB8:3::/64 is routable and can direct packets to the corresponding PE. The function opcode ::C100 is a local function configured on the PE to identify services, such as a VPN instance. The combination of the locator and function opcode parts reflects that routing and MPLS (in which labels represent services) capabilities are both integrated into SRv6 SIDs.

Figure 5-7 Two functions of a service SID



SRv6 BE Implementation

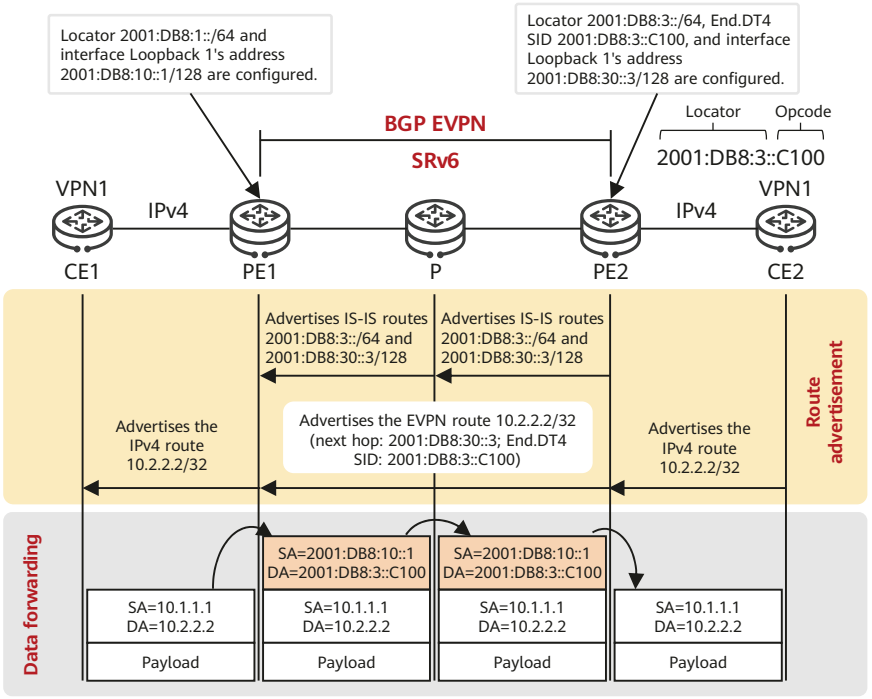
SRv6 BE can carry common traditional services, which are all forwarded in a similar manner. The following example uses an EVPN L3VPNv4 over SRv6 BE scenario to describe how SRv6 BE is implemented.

Figure 5-8 shows the route advertisement and data forwarding processes in this scenario.

In the route advertisement phase:

1. PE2 on which a locator (2001:DB8:3::/64) is configured uses an IGP to advertise the locator route 2001:DB8:3::/64 corresponding to the specified SRv6 SID to PE1, which then installs the route in its IPv6 routing table.
2. After a VPN End.DT4 SID (2001:DB8:3::C100) is configured within the locator range on PE2, PE2 generates a local SID entry.
3. After receiving the VPN IPv4 route advertised by CE2, PE2 converts it into an EVPN IP prefix route and advertises it to PE1 through the BGP EVPN peer relationship. The route carries an SRv6 VPN SID, that is, the VPN End.DT4 SID 2001:DB8:3::C100.
4. After receiving the EVPN route, PE1 leaks it to the IPv4 routing table of the corresponding VPN instance, converts it into a common IPv4 route, and advertises it to CE1.

Figure 5-8 Route advertisement and data forwarding in an EVPN L3VPNv4 over SRv6 BE scenario



In the data forwarding phase:

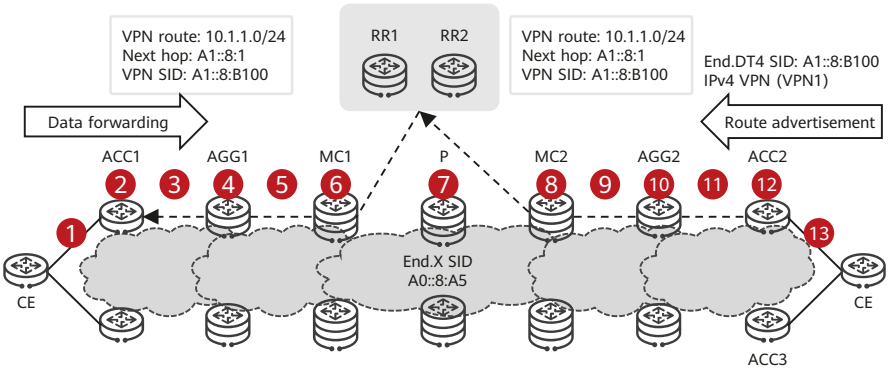
1. CE1 sends a common IPv4 packet to PE1.
2. After receiving the packet through the interface bound to a VPN instance, PE1 searches the IPv4 routing table of the corresponding VPN instance for a matching IPv4 prefix and finds the associated SRv6 VPN SID and next hop. It then encapsulates the packet with an IPv6 header in which the SRv6 VPN SID 2001:DB8:3::C100 is directly used as the destination address.
3. PE1 finds the route 2001:DB8:3::/64 based on the longest match rule and forwards the packet to the P device over the shortest path.

4. Similarly, the P device finds the route 2001:DB8:3::/64 based on the longest match rule and forwards the packet to PE2 over the shortest path.
5. PE2 searches the local SID table based on 2001:DB8:3::C100 and finds a matching End.DT4 SID. According to the instruction bound to the SID, PE2 removes the IPv6 header and searches the IPv4 routing table of the VPN instance corresponding to the End.DT4 SID for packet forwarding.

SRv6 BE Reliability Design

TI-LFA FRR can be used to implement topology-independent protection for the transit nodes on SRv6 BE paths in an IGP domain. During network design, TI-LFA does not require any special considerations other than to be enabled in IGP. Figure 5-9 shows the E2E SRv6 BE reliability design.

Figure 5-9 SRv6 BE reliability design



On the network shown in Figure 5-9, different technologies are used to protect services against failures at different locations. Specifically:

- For failure points 1 and 2, link detection can be used to detect failures, and ECMP or IP FRR can be used to protect services.
- For failure points 3 to 10, TI-LFA FRR can be used to protect services, and link detection or BFD for IGP can be used to detect failures in order to trigger FRR switching.

- For failure points 11 and 12, IP FRR or VPN FRR can be used to protect services, and link detection or BFD for IGP can be used to detect failures in order to trigger FRR switching.
- For failure point 13, link detection can be used to detect failures. In L3VPN scenarios, ECMP or VPN-specific IP FRR is used to protect services. In EVPN scenarios, however, ECMP or local-remote-FRR is used to protect services. The latter allows traffic to be detoured to ACC3 from ACC2 and then to the destination CE to reduce packet loss if the link between ACC2 and the CE fails.

Comparison Between SRv6 BE and SRv6 TE Policy

The main difference between SRv6 BE and SRv6 TE Policy is that SRv6 BE packets do not contain SRH information. For this reason, SRv6 BE does not support traffic engineering. It uses only one service SID to guide packet forwarding to the parent node where the SID is generated and requests the node to execute the instruction bound to the service SID.

Because SRv6 BE needs to be deployed only on the ingress and egress of a network and only requires transit nodes to support IPv6 forwarding, SRv6 BE has unique advantages in common VPN deployment. Take video services as an example, where these services are transmitted between the provincial and municipal centers across the DCN, metro network, and national IP backbone network. During traditional deployment of MPLS VPN, coordination with all kinds of network administrative departments is inevitable, and multi-party collaboration is necessary to successfully perform certain operations. This not only impedes service provisioning, but may also result in the loss of business opportunities. If SRv6 BE is used to carry VPN services, however, the services can be provisioned quickly, as only two PEs that support SRv6 VPN need to be deployed: one in the provincial center and the other in the municipal center. SRv6 BE is therefore more advantageous for carriers to seize business opportunities.

Table 5-1 compares SRv6 BE and SRv6 TE Policy.

Table 5-1 Comparison between SRv6 BE and SRv6 TE Policy

Dimension	SRv6 BE	SRv6 TE Policy
Configuration	Simple.	Complex.
Path	Based on the IGP cost.	Based on TE constraints.

Dimension	SRv6 BE	SRv6 TE Policy
computation		
SRH	Typically, SRHs are not carried during forwarding. They are carried only when traffic is forwarded over a repair path in TI-LFA FRR protection scenarios.	SRHs are carried.
Path programming	Not supported.	Supported.
Whether a controller is required	Not required. IGP path computation eliminates the need for a controller.	Required if SRv6 TE Policies are dynamically delivered. Although SRv6 TE Policies can be manually configured, eliminating the need for a controller, the configuration is complex. Using a controller to dynamically deliver SRv6 TE Policies is therefore recommended to meet service requirements more quickly and achieve service-driven networks.
Protection technology	TI-LFA FRR (50 ms).	TI-LFA FRR (50 ms).
Application scenario	Applies to scenarios where services do not have strict SLA requirements and no path needs to be specified for traffic.	Applies to scenarios where services have strict SLA requirements. For example, traffic needs to be switched to another path if network congestion occurs or needs to be redirected to a specified destination for anti-DoS cleaning.

Chapter 6
SRv6 for 5G and Cloud Services

This chapter describes how SRv6 supports 5G and cloud services. 5G changes the attributes of network connections (which are the core of IP bearer networks), and cloud changes their scope. These changes, in turn, present big opportunities for SRv6 development. As 5G services continue to develop, network connections must meet an increasing number of requirements, such as stronger SLA guarantee and deterministic latency. SRv6 extensions provide the capabilities to meet such requirements. In addition, the development of cloud services has made service processing locations more flexible. And some cloud services (such as telco cloud services) further break the boundary between physical and virtual network devices, integrating services and bearer networks. Such changes have reshaped the scope of network connections. Thanks to its unified programmability for services and bearer networks as well as its native IPv6 attribute, SRv6 enables rapid setup of connections and meets requirements for flexible adjustment of the connection scope.

6.1 SRv6 for Network Slicing

Introduction to Network Slicing

Networks such as IP backbone, metro, and mobile bearer networks have historically been designed to have borders. This not only makes E2E service deployment more difficult, but also complicates network management and maintenance. As society develops, network complexity hinders the development of services. Looking to address this issue, the industry has reached a consensus to build a unified IP bearer network — ultra-broadband, simple, intelligent, reliable, and secure. Against this backdrop, "Everything over IP" is gradually becoming a reality.

Meeting diversified, differentiated, and complex requirements of various services on a unified IP bearer network is a new challenge. Another challenge is how carriers can transition away from being simply providers of pipes to obtain new business value.

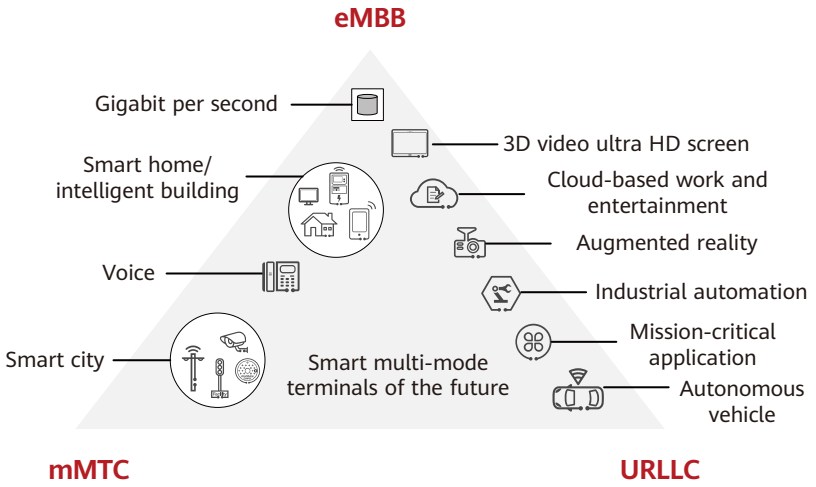
Take 5G services as an example. Because these services have a diverse range of characteristics, they pose distinct requirements on 5G networks. For example, environment monitoring, smart home, smart agriculture, and smart meter reading require huge numbers of device connections and frequent transmission of many small packets. Another two examples are video uploading and mobile healthcare services, which require high transmission rates; and Internet of Vehicles (IoV), smart grid, and industrial control services, which require millisecond-level latency and near-100% reliability. In light of this, 5G must be highly flexible and scalable to penetrate more vertical services, adapt to huge numbers of device connections, and meet diversified user requirements. Furthermore, 5G must meet requirements of different industries by building flexible and dynamic networks, with a focus on vertical industries' requirements — this is in addition to meeting mobile broadband requirements.

As shown in Figure 6-1, the main service requirements in the 5G era are classified into three types:

- eMBB: focuses on bandwidth-intensive services, such as High Definition (HD) video, VR, and AR.

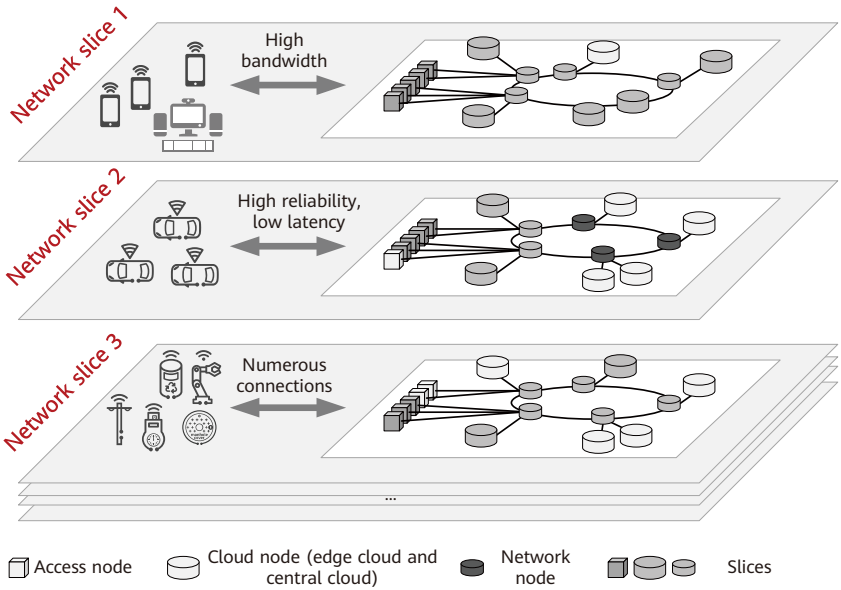
- URLLC: focuses on services that are extremely sensitive to latency and reliability, such as autonomous driving, industrial control, telemedicine, and drone control.
- mMTC: covers scenarios with high connection density, such as smart city and smart agriculture. These scenarios have different kinds of network feature and performance requirements, which cannot be met using a single network.

Figure 6-1 Main service requirements in the 5G era



To meet the differentiated requirements of various services on a physical network, network slicing is introduced as a method for creating multiple virtual networks over a shared physical network. Each virtual network possesses a customized network topology and provides specific network functions and resources to meet the functional requirements and SLAs of different slice tenants. Figure 6-2 shows an example of 5G network slicing.

Figure 6-2 Example of 5G network slicing



Network slicing, as the name suggests, slices a 5G network into logical partitions that can be allocated to different tenants over a shared network infrastructure. In this case, vertical industry customers use the 5G network as slice tenants. Network slices that provide services for such tenants need to be securely isolated, which is crucial for ensuring security and reliability.

- Security: Data/Information is effectively isolated between tenants.
- Reliability: Any network exception or failure that occurs for one tenant does not affect other tenants on the same network.

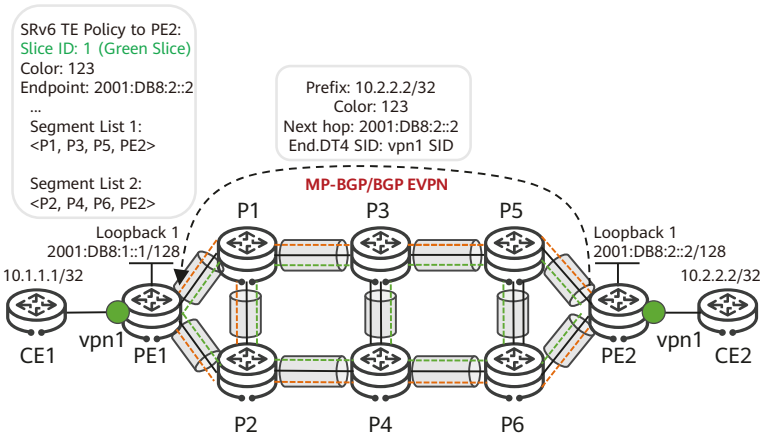
By providing network slicing services, carriers transform their model for generating revenue from a traffic-based one to a service-based one. In the future, carriers will predominantly provide on-demand, customized, and differentiated services, which will enable new value growth.

SRv6-based Network Slicing

In SRv6 scenarios, End SIDs are allocated only to nodes on a physical network, and End.X SIDs are allocated to physical links. No SID is allocated to logical nodes or links in network slices; instead, the SIDs of the physical network are directly used.

This section uses L3VPNv4 over SRv6 TE Policy, shown in Figure 6-3, as an example to describe the route advertisement process in the control plane of network slicing.

Figure 6-3 Route advertisement in the control plane of network slicing

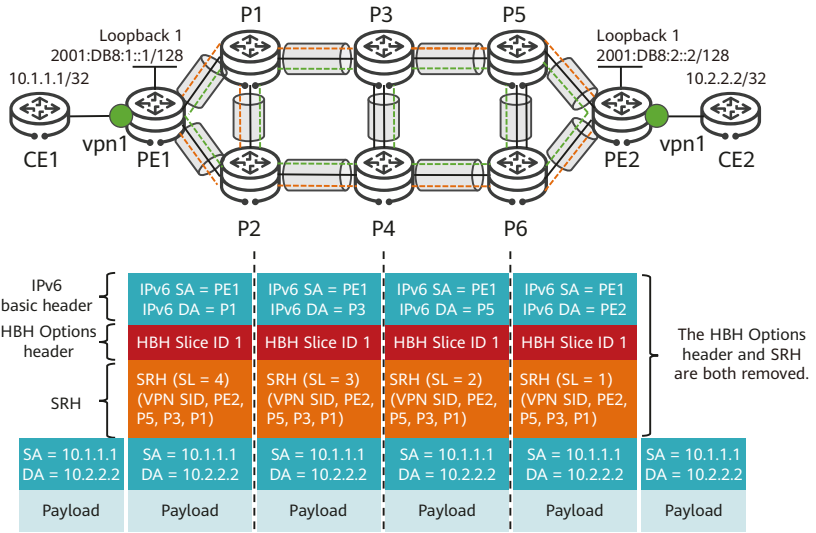


During route advertisement, PE2 can advertise VPN route information (e.g., color, next hop, and VPN SID) to PE1 through the MP-BGP/BGP EVPN peer relationship.

This example assumes that an SRv6 TE Policy is created on PE1 and that a slice ID is configured for the SRv6 TE Policy to associate the two. PE1 recurses the VPN route 10.2.2.2/32 to the SRv6 TE Policy based on the color and next hop information, and then transmits data to the associated network slice for forwarding based on the slice ID of the SRv6 TE Policy.

Figure 6-4 shows the data forwarding process of network slicing in an L3VPNv4 over SRv6 TE Policy scenario.

Figure 6-4 Data forwarding process of network slicing



In the data forwarding phase:

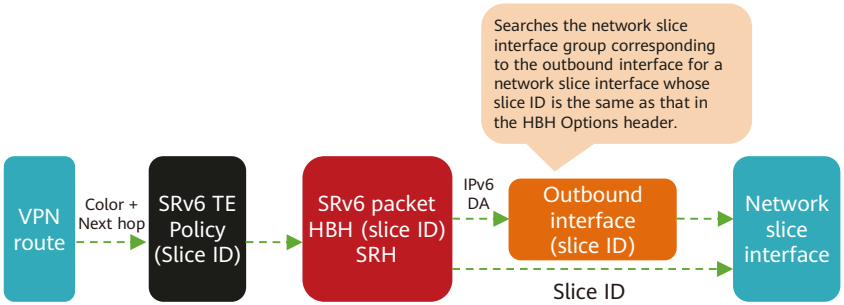
1. CE1 sends a common IPv4 unicast packet to PE1.
2. After receiving the packet from CE1, PE1 searches the routing table of the corresponding VPN instance and finds that the outbound interface of the route is an SRv6 TE Policy. As such, PE1 inserts SRH information into the packet and encapsulates the SID list of the SRv6 TE Policy, the HBH Options header carrying the slice ID of the SRv6 TE Policy, and the IPv6 basic header in sequence. PE1 then forwards the packet, which is associated with the specified network slice interface based on the slice ID, to transit node P1.
3. P1 forwards the packet based on the SRH information, using the slice ID in the HBH Options header to associate the packet with the specified network slice interface. The forwarding process on P3 and P5 is similar to that on P1.

4. After receiving the packet, the egress PE2 searches the local SID table based on the IPv6 destination address in the packet and finds a matching End SID. According to the instruction bound to the SID, PE2 decrements the SL value of the packet by 1 and updates the IPv6 DA field to the corresponding VPN SID.

   Based on the VPN SID, PE2 searches the local SID table and finds a matching End.DT4 SID. According to the instruction bound to the SID, PE2 removes the SRH, HBH Options header, and IPv6 basic header from the packet, searches the routing table of the VPN instance corresponding to the VPN SID based on the destination address 10.2.2.2 in the inner IPv4 packet, and forwards the packet to CE2.

From the preceding process, we can conclude that slice IDs are essential for connecting the control and forwarding planes. The network slicing solution is therefore also referred to as slice ID-based network slicing. For details, see Figure 6-5.

Figure 6-5 Connecting the control and forwarding planes through slice IDs



6.2 SRv6 for iFIT

In the 5G era, the eMBB, URLLC, and mMTC scenarios pose higher requirements on bearer networks. To meet these requirements in terms of network O&M and performance measurement, 5G networks need the following:

- An effective troubleshooting method that can improve O&M efficiency.

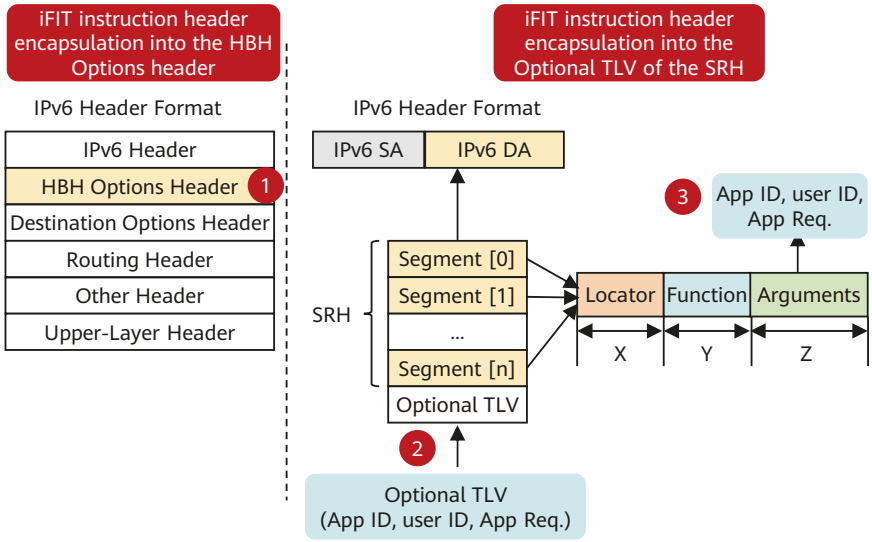
- A service flow-based performance measurement mechanism that can accurately reflect actual user traffic in real time. Existing performance measurement mechanisms support only coarse OAM granularities (such as ports, tunnels, and pseudo wires), which is insufficient in the 5G era.
- Functions that can meet requirements of latency-sensitive 5G services in order to improve user experience. Such functions include network-wide latency visualization, latency abnormality monitoring, and latency-based routing.

To address the fact that current OAM detection technologies cannot adequately meet the performance measurement requirements of 5G bearer networks, Huawei launched the iFIT performance measurement solution with the following highlights:

- Extensibility: iFIT features high measurement precision and easy deployment and can be easily extended in the future.
- Fast fault locating: iFIT provides in-band flow measurement to help measure the latency and packet loss of service flows in real time.
- Visualization: iFIT allows performance data to be displayed on a Graphical User Interface (GUI) so that users can quickly find failure points.

Thanks to SRv6's extensive data-plane programming space for applications, the iFIT instruction header can be encapsulated into the IPv6 HBH Options header or the SRH's Optional TLV. Different encapsulation formats have different processing semantics, offering a wide array of features in SRv6 OAM.

Figure 6-6 SRv6 iFIT encapsulation



All IPv6 forwarding nodes can process the iFIT instruction if it is encapsulated into the HBH Options header, but only SRv6 nodes can process it if it is encapsulated into the SRH. In SRv6 BE or SRv6 TE Policy loose path scenarios where the packet forwarding path is not fixed, encapsulating the iFIT instruction into the HBH Options header helps O&M personnel to know how packets are forwarded hop by hop and facilitates fault demarcation when a fault occurs on the network.

6.3 SRv6 for Telco Cloud

Telecom networks constructed in a traditional manner tightly integrate software and hardware, using dedicated hardware devices purchased from device vendors to provide telecom services. Such devices include Mobility Management Entities (MMEs), Serving Gateways (SGWs), and Provisioning Gateways (PGWs) for mobile data services and Broadband Network Gateways (BNGs) for fixed services. But as telecom services (including emerging services like 5G) rapidly develop, telecom networks need to respond quickly, adapt to diversified service

scenarios, and support frequent service deployments. Given that traditional telecom devices with dedicated hardware rely heavily on device vendors, it often takes several months to expand network capacity or bring a new release online, making service rollout slower and more expensive.

With the development and successful application of cloud-native design in the IT field and Network Functions Virtualization (NFV) technologies, virtualization and cloudification have gradually matured and evolved to provide a new level of productivity, offering a new approach to telecom network construction. Telecom carriers want devices with decoupled software and hardware — similar to those adopted in the IT field — and to use universal or simplified hardware to enhance device capabilities and improve forwarding throughput while lowering costs. Using software decoupled from hardware, carriers can rapidly add new functions and roll out new services in response to customer requirements. Against this backdrop, the cloud-based telecom network (also known as telco cloud) has emerged as a new architecture for telecom networks. Telco cloud construction refers to the NFV/cloudification of telecom service nodes.

Currently, the general solution for edge telco cloud bearing is to integrate the DCN and Wide Area Network (WAN) to form a spine-leaf fabric architecture called Network as a Fabric (NAAF).

As shown in Figure 6-7, NAAF roles are classified as leaf or spine nodes.

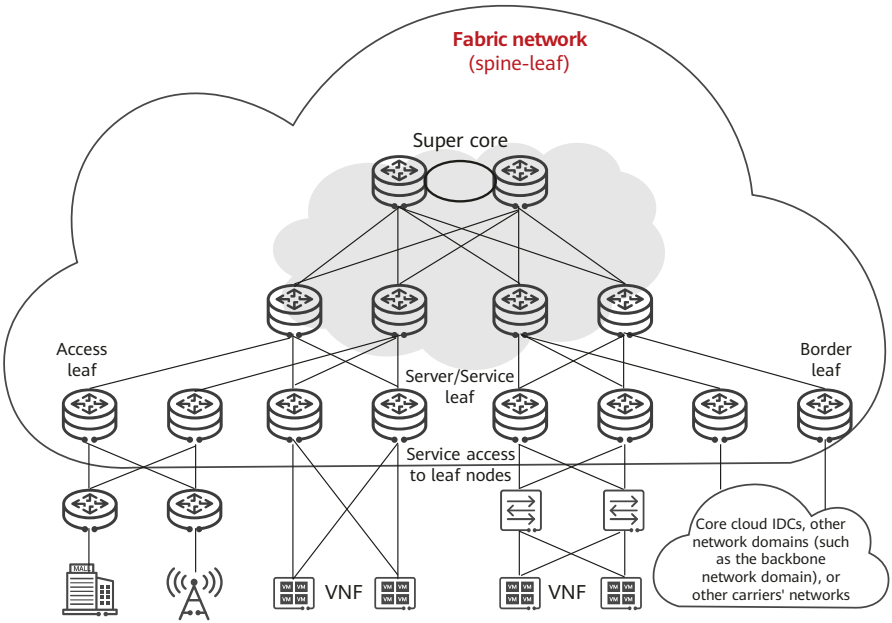
Leaf nodes are access nodes, typically WAN PEs through which various network devices can access a fabric network. Depending on devices that access the network, leaf nodes are classified as access, server/service, or border leaf nodes.

- Access leaf node: used for user access, such as mobile access over a base station or fixed access over an Optical Line Terminal (OLT)
- Server/Service leaf node: used for Virtual Network Function (VNF) service access, including VAS, vCPE, vUPF, and BNG-UP access
- Border leaf node: used for external network connections, such as connections to back-to-back core DCs, other carriers' networks, and other departments' networks

Spine nodes, mainly used for high-speed traffic forwarding, are typically P devices on a WAN and do not function as service access devices. These nodes have the following characteristics:

- They connect to leaf nodes through high-speed interfaces, eliminating the need to establish full-mesh connections between all leaf node pairs. This also enables smooth service/leaf node scale-out, as capacity expansion will not affect existing services.
- They can be interconnected hierarchically on a large network.

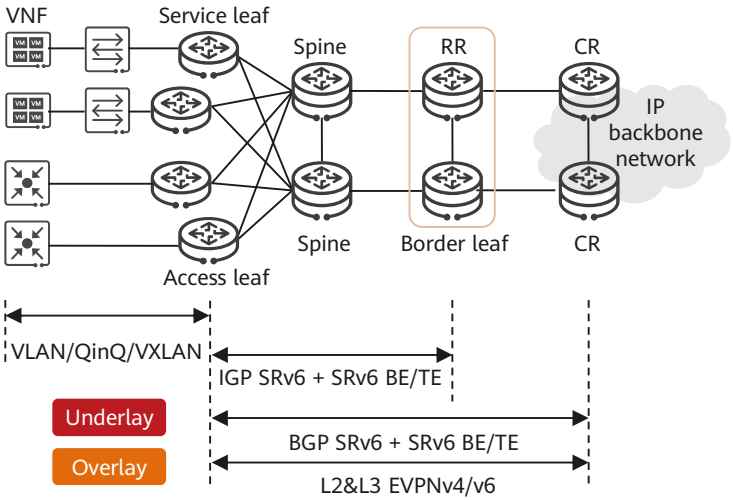
Figure 6-7 Spine-leaf fabric architecture



Because the NAAF edge cloud architecture involves both the DCN and traditional WAN, transmission protocols used on these networks need to be unified. Furthermore, the edge telco cloud has many requirements for telecom connectivity but few for cloudification, meaning that mature telecom transmission solutions (VPN and SRv6) are more suitable for use.

NAAF uses SRv6 + EVPN technologies to support IPv4/IPv6 dual-stack as well as 5G, enterprise, and Mobile Edge Computing (MEC) services. Figure 6-8 shows key NAAF technologies.

Figure 6-8 Key NAAF technologies



The NAAF transport-layer design offers the following benefits:

- Simplified protocols: DCN and WAN bearer solutions are unified, simplifying bearer protocols.
- E2E service capabilities: E2E SRv6 BE/TE provides NAAF with E2E path optimization and network slicing capabilities based on SRv6's excellent programmability.
- Simplified O&M: Carrying VPN services over E2E SRv6 BE/TE eliminates the need to configure inter-AS VPN Option A between DC gateways and PEs and provides E2E OAM capabilities.
- Simplified network layers: DCN and WAN bearer networks are unified, eliminating the need to independently deploy multiple roles such as PEs, DC gateways, and leaf nodes. The roles once played by multiple devices can now be played by a single device, lowering network construction costs.

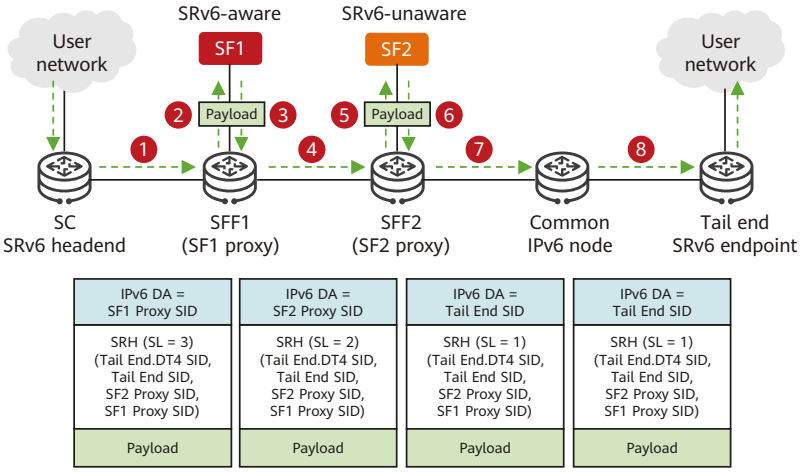
6.4 SRv6 for SFC

The Service Function Chaining (SFC) technology logically connects services on network devices to provide an ordered service set for the application layer. By adding Service Function Path (SFP) information to original packets, SFC enables packets to pass through Service Functions (SFs) along a specified path.

In order to provide users with secure, fast, and stable services as planned, data packets on a network usually need to pass through various SFs, such as firewalls, Intrusion Prevention Systems (IPSs), application accelerators, and Network Address Translation (NAT) devices. The required services can be implemented only when network traffic passes through SFs in a sequence defined by the corresponding service logic.

As shown in Figure 6-9, SF1 and SF2 are SRv6-unaware SFs (meaning that they do not support SRv6). To implement SFC, SF proxy must be configured on SFF1 and SFF2, and SRv6 SIDs must be allocated to SF1 and SF2 proxies. On the Service Classifier (SC), the SF1 Proxy SID, SF2 Proxy SID, and Tail End SID form the segment list of an SRv6 TE Policy that functions as an SFP.

Figure 6-9 SRv6 for SFC



The data forwarding process is as follows:

1. After receiving an original IPv4 packet from the user network, the SC classifies the packet based on certain information (e.g., 5-tuple information) and redirects the classified traffic to an SRv6 TE Policy. Based on the SRv6 TE Policy, the SC encapsulates the packet into an SRv6 one, using the SF1 Proxy SID as the destination address. The SRH contains the SRv6 TE Policy's path information along with the Tail End.DT4 SID that represents VPN or public network services, as shown in Figure 6-9.
2. After receiving the packet, SFF1 executes the instruction bound to the SF1 Proxy SID: decapsulating the packet and sending the original packet to SF1 for processing.
3. After processing the packet, SF1 sends it back to SFF1.
4. Based on information about the interface through which SFF1 receives the IPv4 packet, SFF1 searches for the relevant configuration. After finding the configuration, SFF1 encapsulates the packet into an SRv6 one by re-adding SRH information according to the configuration. In this case, the destination address of the SRv6 packet is the SF2 Proxy SID.
5. After receiving the packet, SFF2 executes the instruction bound to the SF2 Proxy SID: decapsulating the packet and sending the original packet to SF2 for processing.
6. After processing the packet, SF2 sends it back to SFF2.
7. Based on information about the interface through which SFF2 receives the IPv4 packet, SFF2 searches for the relevant configuration. After finding the configuration, SFF2 encapsulates the packet into an SRv6 one by re-adding SRH information according to the configuration. In this case, the destination address of the SRv6 packet is the Tail End SID. SFF2 then forwards the packet to the tail end along the shortest IGP path.
8. After receiving the SRv6 packet, the tail end finds that its own End SID and the destination address of the packet are the same. As such, it executes the instruction bound to the End SID: decapsulating the packet, decrementing the SL value by 1 (the value changes to 0), and updating the IPv6 DA field. Because the IPv6 DA field in the current packet is now the Tail End.DT4 SID, the tail end searches the local SID table and finds a matching Tail End.DT4 SID. It then executes the instruction bound to the SID to forward the original IPv4 packet to the corresponding IPv4 VPN or public network.

6.5 SRv6 for SD-WAN

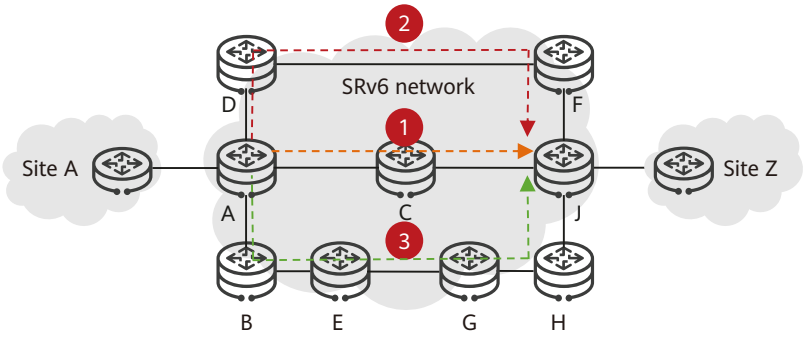
Software-Defined Wide Area Network (SD-WAN) EVPN builds on the existing EVPN technology to separate the overlay service network from the underlay transmission network. This VPN solution is used to interconnect enterprise sites.

SD-WAN EVPN extends Network Layer Reachability Information (NLRI) based on BGP, defining new BGP SD-WAN routes for sites to exchange Transport Network Port (TNP) information, including key information used to establish SD-WAN tunnels between sites. The sites use EVPN IP prefix routes (Type 5 routes) to advertise service routes to each other. After receiving these EVPN routes from the peer site, the local site triggers the establishment of an SD-WAN tunnel to the peer site, based on which the data channel of the underlay network is established. In addition, EVPN routes recurse to the SD-WAN tunnel to establish service paths on the overlay network.

A Service Provider (SP) network supports the creation of paths meeting different SLA requirements upon application requests. Figure 6-10 shows three different paths.

1. Default path (SRv6 BE)
2. Low-latency path created upon an application request
3. 100 Mbit/s-assured path created upon an application request

Figure 6-10 Creating multiple paths that meet different SLA requirements on an SP network



1. Default path (SRv6 BE): <2001:DB8:C9::1>
2. Low-latency path: <2001:DB8:C4::1, 2001:DB8:C6::1, 2001:DB8:C9::1>
3. Assured-100 Mbit/s path: <2001:DB8:C2::1, 2001:DB8:C5::1, 2001:DB8:C8::1, 2001:DB8:C9::1>

To sum up, the SRv6 SD-WAN solution offers the following key benefits:

1. Unified network-wide scheduling: Native IPv6-based SD-WAN that supports both 4G and 5G replaces the original VXLAN and Generic Routing Encapsulation (GRE) solutions.
2. High scalability: The SP network is unaware of any policy change of the SD-WAN instance in terms of which flow to classify, when to steer the flow, and which path to use for flow steering. The SP is mainly responsible for maintaining SRv6 TE Policy states at the network edge and hundreds of SIDs on the network, fully utilizing SRv6's statelessness property.
3. Strong privacy protection: No infrastructure, topology, capacity, or SID information about SP networks is shared, ensuring network privacy security.

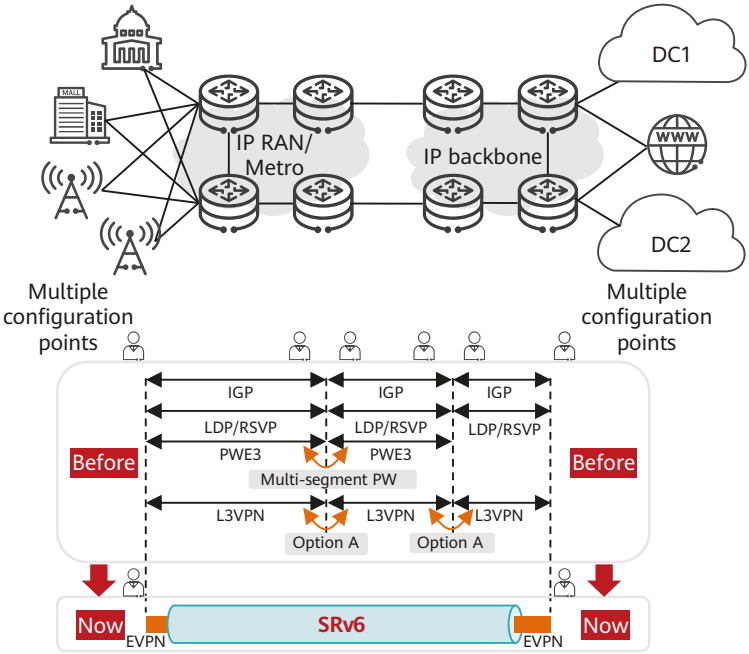
Chapter 7
Successful Applications of SRv6

This chapter describes the successful applications of SRv6. As cloud computing continues to develop, most customers are adopting cloudification to meet their service deployment requirements, favoring either multi-cloud or hybrid cloud. Cloud-network convergence products enable customers to quickly, securely, and reliably access cloud resource pools through cloud private lines. They are interconnected with the cloud pools of multiple cloud service providers through a cloud backbone network in order to achieve multi-cloud access through one connection. In addition, an intelligent cloud-network operations platform is used to build differentiated cloud-network service brands and implement functions such as automatic provisioning of cloud-network services and dynamic bandwidth adjustment.

7.1 Simplified and Unified IP Bearer Network

As shown in Figure 7-1, an IP bearer network carries many cross-domain services, such as mobile 3G/4G, Voice over IP (VoIP), and private line services. Deploying such services segment by segment on a network requires multiple departments to collaborate during the segment-by-segment cross-domain configuration. This not only complicates E2E deployment, but also slows down service provisioning and extends the deployment period to months, negatively affecting service operations. In addition, due to the coexistence of multiple protocols, carriers are in need of a simplified network architecture in order to automate and accelerate service deployment.

Figure 7-1 Simplified and unified IP bearer network



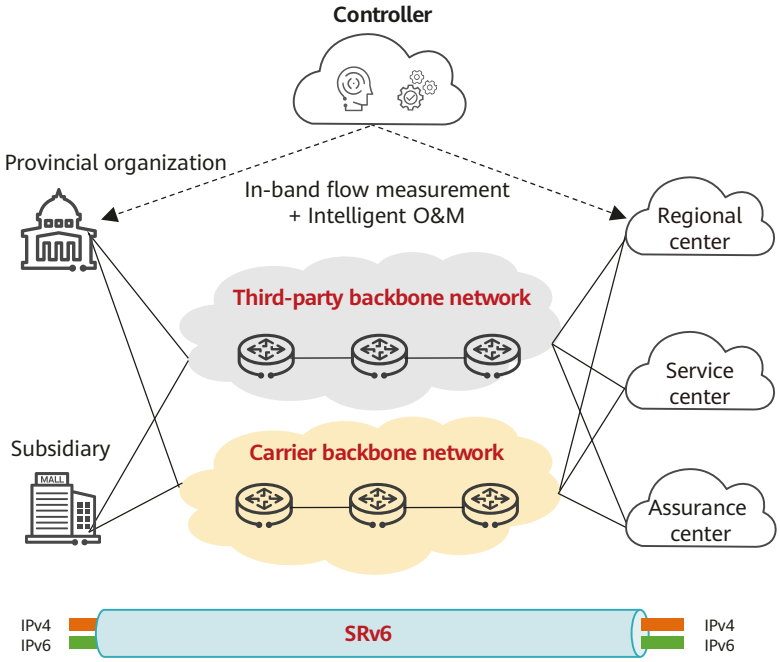
SRv6 is introduced to address the preceding issues, offering the following benefits:

1. Simplified network protocols: SRv6 eliminates the need to use the complex MPLS protocol, thereby radically simplifying the network architecture. The SRv6-based network architecture does not require any MPLS configuration for services and retains only two basic network protocols (IGP and BGP).
2. Fast service provisioning: SRv6 simplifies cross-domain deployment based on IPv6 route reachability. In SRv6, non-key services are carried over SRv6 BE, SRv6 services are deployed only on the ingress and egress, and transit nodes only need to support IPv6, achieving significant improvements in network O&M efficiency. Furthermore, B2B private line services are carried over SRv6 TE Policy, shortening the service provisioning time to one day thanks to automating the provisioning process.
3. Sustainable evolution: SRH programmability enables SRv6 to decouple networks from services. This allows networks to continuously evolve without the need for additional protocols.

7.2 Intelligent and Professional WAN

As shown in Figure 7-2, the service backbone network of a customer needs to be connected to multiple organizations by traversing a third-party backbone network and a carrier's MPLS VPN backbone network. Because the third-party network is not managed by the customer, using traditional SDN technologies cannot achieve intelligent network adjustment or optimization during the network traversal.

Figure 7-2 Intelligent and professional WAN



SRv6 and an SRv6 TE Policy-based L3VPN bearer solution are deployed to achieve SDN optimization across the third-party network.

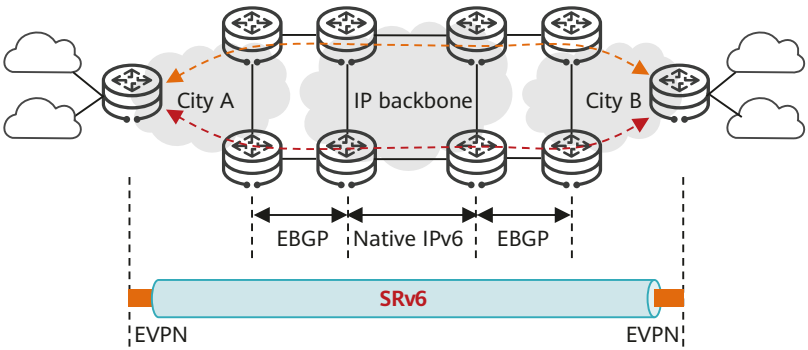
This solution offers the following benefits:

1. Excellent cross-domain service experience: Various services are carried using SRv6, and key services are carried using SRv6 TE Policy, providing differentiated SLA guarantee and ensuring lossless service transmission.
2. E2E high reliability: SRv6 enables E2E fault-triggered switching to be completed within 50 ms. In addition, technologies such as in-band flow measurement and intelligent O&M make networks more reliable and secure.

7.3 Cross-Domain Cloud Backbone Private Line

In the cloud era, enterprise services may concurrently access multiple types of clouds, and the clouds in different regions may need to be interconnected. Traditionally, only solutions such as Option A/B/C and seamless MPLS can be used for interconnecting clouds across different regions. These solutions, however, require multiple geographically dispersed departments to collaborate to provision services. Furthermore, they involve a large number of protocols and complex network states, adversely affecting service O&M and network reliability.

Figure 7-3 Cross-domain cloud backbone private line



As shown in Figure 7-3, a carrier uses the SRv6 Overlay solution to deploy cross-domain cloud backbone private lines on a new type of metro network that features multi-cloud aggregation.

This solution offers the following benefits:

1. Fast service provisioning: Only the metro devices in city A and the DC egress devices in city B need to be upgraded before SRv6 can be deployed. The SRv6 VPN is deployed across the backbone network, quickly constructing cross-province cloud backbone private lines between cities A and B.

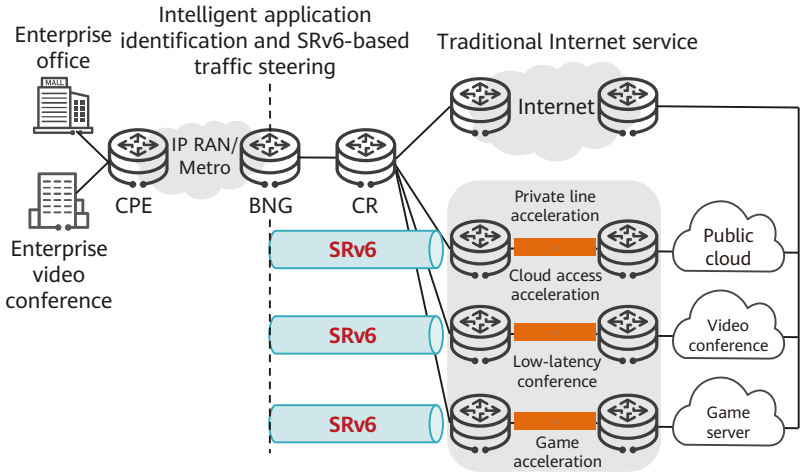
2. On-demand upgrade and evolution: On-demand path selection can be implemented if SRv6 is deployed on key transit nodes. Furthermore, traffic can be directed to traverse the backbone network according to service requirements, facilitating network path optimization.
3. Excellent cross-domain service experience: SRv6 eliminates multi-segment stitching that is used in traditional cross-domain private line scenarios, facilitating cross-domain network access and enabling one-hop cloud access. Furthermore, L3 technologies such as L3VPN and EVPN L3VPN are used to implement cloud private line bearing and inter-cloud interconnection, and the L3 network routing capability enables enterprise sites to achieve one-point and flexible access to any cloud resource pool. By leveraging SRv6 TE Policy, SRv6 provides differentiated SLA guarantee for various cloud applications.

7.4 International Internet Cloud Private Line

Data networks have become as essential as water, electricity, and gas for people. More and more enterprises are quickly adapting to online office scenarios, holding video conferences, collaboratively developing software, and exchanging real-time production data online. And while multinational companies also adapt to online office scenarios, their requirements are typically more extensive and urgent.

SRv6 can easily meet such requirements, as shown in Figure 7-4.

Figure 7-4 International Internet cloud private line



This solution offers the following benefits:

1. Intelligent traffic steering and acceleration: After the BNG intelligently identifies applications, their traffic is flexibly steered to different SRv6 paths and then accelerated according to different policies, improving user experience. The entire process does not require any user operations.
2. Easy business monetization: Because carriers only need to pre-deploy SRv6 and perform few (or no) operations in the future, they can easily achieve business monetization by improving service experience.

7.5 Intelligent Cloud-Network for the Government Sector

As society enters a fully connected intelligence era, traditional networks are also moving towards intelligent cloud-networks. Networks in the future will have at least the following distinct characteristics and requirements:

1. With the rise of multi-cloud applications for enterprise services, cloud computing will enter the multi-cloud era, leading to more urgent requirements for cloud data center interconnection and cloud-network convergence.
2. Due to the different network requirements of various services, networks must change from providing best-effort forwarding to providing deterministic SLA guarantee in order to achieve one-network bearing of multiple services for a wide range of industries.
3. The key requirements will become one-stop acquisition, rapid provisioning, and flexible adjustment of cloud-network services. Building on this, the future intelligent cloud-network will be a converged service network — rather than an independent cloud or bearer network — providing integrated cloud-network products and services for end customers. Customers will be able to select any product portfolio they require as though they are purchasing products on an e-commerce platform, achieving online self-service, rapid provisioning, and process visualization from contract signing to fulfillment.

The intelligent cloud-network solution is based on intelligent IP networks that use SRv6 as the foundation. It is designed to facilitate the digital transformation of various industries, such as government, healthcare, education, finance, and energy, by helping carriers develop solutions for them. This section uses the government sector as an example to describe the intelligent cloud-network solution and introduce how SRv6 is used to implement an intelligent cloud-network.

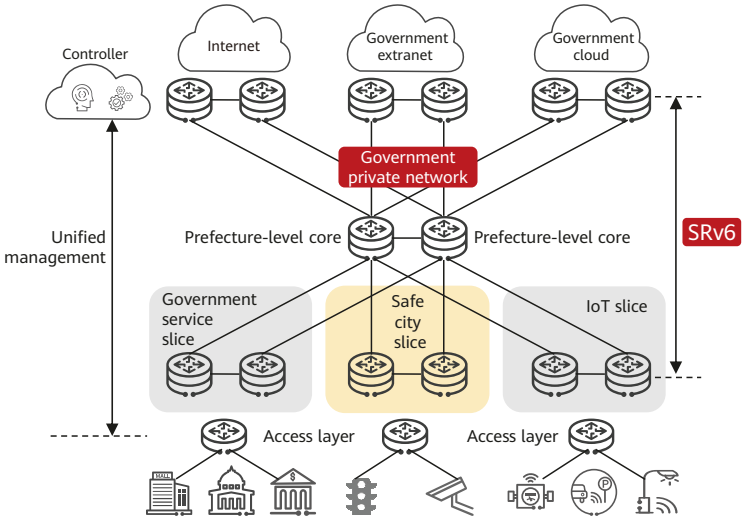
Reforms aimed at achieving smart governments face many issues. For example, different government departments are not interconnected, leading to information silos and difficulty in sharing information. On top of this, government extranets lack sufficient numbers of access points. Smart governments therefore have the following two core requirements:

1. Unified network management: Implement intensive network construction and integrate private networks into government extranets. The key to achieving this is to ensure secure service isolation.
2. One network for all services: Uniformly manage and share data, and upload the huge volumes of Internet of Things (IoT) sensing data and video surveillance data of cities to multi-level government clouds for multiple government departments to invoke. The key to achieving this is to ensure

wide network coverage, fast service access to multiple clouds, and flexible cloud interconnection.

In Figure 7-5, SRv6 is deployed to conveniently and quickly establish basic network connections between the cloud and access devices in different cities, ensuring efficient service provisioning. In the future, network slicing can be used to divide the government private network into different service planes to ensure strict isolation among services (e.g., government office, IoT, and video services) and provide differentiated deterministic SLA guarantee for different services.

Figure 7-5 SRv6-based intelligent cloud-network for the government sector



This solution offers the following benefits:

1. Fast service provisioning: All services are cloudified, and only one network exists below clouds. SRv6 enables different departments to implement one-hop cloud access, achieving fast service provisioning.
2. Excellent service experience: One network is divided into multiple network slices to implement hard resource isolation, meeting the service requirements of different departments with committed SLAs.
3. E2E high reliability: iFIT-based visualized O&M enables fault locating to be completed within minutes.

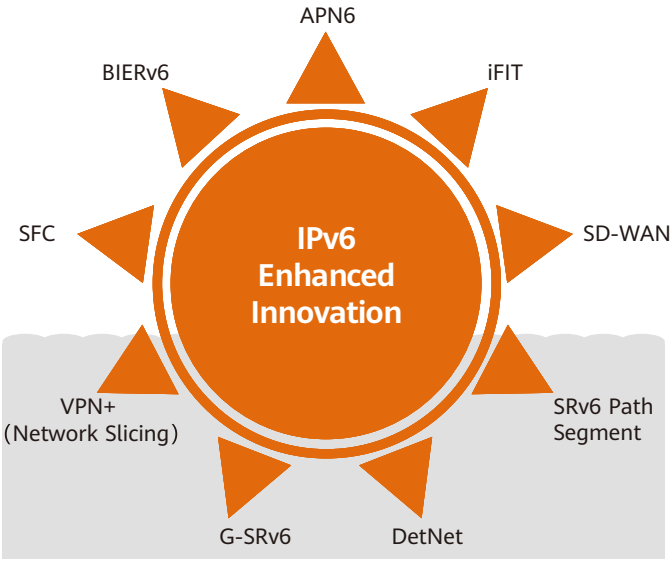
Chapter 8
IPv6 Enhanced Innovations Since SRv6

SRv6 presents new opportunities for large-scale IPv6 deployment, and the use of SRHs inspires people to explore further. As new services continue to develop, SRv6 is no longer the only technology affecting IPv6's future. Specifically, the data plane supports extensions of other types of IPv6 extension headers in addition to SRv6 SRHs. Some examples of these extensions include:

- Implementing Bit Index Explicit Replication IPv6 Encapsulation (BIERv6) based on the DOH
- Implementing network slicing based on the HBH Options header
- Implementing iFIT based on the HBH Options header or the Optional TLV in SRv6 SRHs

Since SRv6 opened the door to innovation based on IPv6 extension headers, new IPv6-based application solutions have continuously emerged. In the IPv6 era, these solutions require new SRv6 SRH extensions or extensions of other IPv6 extension headers. Some of the solutions that may leverage such extensions include VPN+ (network slicing), iFIT, Deterministic Networking (DetNet), SFC, SD-WAN, BIERv6, Generalized SRv6 (G-SRv6), SRv6 Path Segment, and Application-aware IPv6 Networking (APN6), which are shown in Figure 8-1.

Figure 8-1 IPv6 enhanced innovations



As the cloud and network become more converged, more information needs to be exchanged between them, with IPv6 being the most advantageous choice to achieve this. The next-generation Internet requires more than just IPv6, which is only a starting point and platform for innovation of the next-generation Internet. With the large-scale deployment of IPv6, IPv6 enhanced innovation technologies represented by SRv6 will be widely used to build intelligent, simplified, and automated next-generation networks with committed SLAs.

IP Network
eBook Series