Application-aware G-SRv6 network enabling 5G services

Cheng Li, Jianwei Mao, Shuping Peng, Yang Xia, Zhibo Hu, Zhenbin Li
Huawei Technologies, Beijing, China
{c.l, maojianwei, pengshuping, yolanda.xia, huzhibo, lizhenbin} @huawei.com

***Abstract*—This demo showcased how application-aware G-SRv6 network provides fine-grained traffic steering with more economical IPv6 source routing encapsulation, effectively supporting 5G eMBB, mMTC and uRLLC services. G-SRv6, a new IPv6 source routing paradigm, introduces much less overhead than SRv6 and is fully compatible with SRv6. Up to 75 percent overhead of an SRv6 SID List can be reduced by using 32-bit compressed SID with G-SRv6, allowing most merchant chipsets to support up to 10 SIDs processing without introducing packet recirculation, significantly mitigating the challenges of SRv6 hardware processing overhead and facilitating large-scale SRv6 deployments. Furthermore, for the first time, by integrating with Application-aware IPv6 networking (APN6), the G-SRv6 network ingress node is able to steer a particular application flow into an appropriate G-SRv6 TE policy to guarantee its SLA requirements and save the transmission overhead in the meanwhile.**

Keywords—SRv6 Compression, G-SRv6, APN6

I. INTRODUCTION

As 5G and industry verticals evolve, ever-emerging new services with diverse but demanding requirements such as low latency and high reliability are accessing to the network. Different applications have differentiated network Service Level Agreement (SLA). For instance, on-line gaming has highly demanding requirements on latency, live video streaming has high requirements on both latency and bandwidth, while backup traffic mainly requires more bandwidth but is less sensitive of latency. However, in current networks, the operators remain unaware of the traffic type traversing their network, making the network infrastructure essentially dumb pipes and losing application performance optimization opportunities. To solve this issue, Application-aware IPv6 networking(APN6) [1] is proposed, which takes advantage of the programmable space in the IPv6/SRv6 packet encapsulations to convey application-aware information into the network layer, and makes network aware of applications and their requirements in order to provide fine-grained application-aware services.

SRv6 [2], as the underlying network protocol supporting APN6, enables the ingress node to explicitly program the forwarding path of packets by encapsulating/inserting ordered Segment ID (SID) list into the Segment Routing Header (SRH) at the ingress node, where each SID is 128-bit long. The SLA can be satisfied by steering the application packets into an explicit SRv6 programmable forwarding path. However, in some scenarios such as strict Traffic Engineering(TE), many SIDs will have to be inserted in the SRH, resulting in a lengthy SRH which imposes big challenges on the hardware processing, and affects the transmission efficiency especially for the small size packets in 5G uRLLC or mMTC scenarios. For instance, the size of an SRv6 encapsulation with 10 SIDs is 208 bytes, which exceeds the parser window of most merchant silicon chipsets (e.g., Jericho2) and causes expensive packet recirculation. This has become a big obstacle for SRv6 deployment in practice.

We proposed Generalized Segment Routing over IPv6 (G-SRv6) [3][4][5] to address the challenges of SRv6 overhead. While compatible with SRv6, G-SRv6 provides a mechanism to encode Generalized SIDs (G-SID) in the Generalized SRH (G-SRH), where a G-SID can be a 128-bit SRv6 SID, a 32-bit compressed SID (C-SID) or some other types. A 32-bit C-SID saves 75% overhead of the SID, so that the size of SRH can be significantly compressed. It also supports incremental upgrade from SRv6 by encoding both SRv6 SIDs and C-SIDs in the SRH. With G-SRv6, most the merchant chipsets can support up to 10 SIDs processing without packet recirculation so that the challenges of SRv6 hardware processing is mitigated, facilitating the large-scale SRv6 deployment. So far, G-SRv6 has been implemented in Linux Kernel, and hardware devices from more than 10 vendors.

This demo showcases that APN6 over G-SRv6 enables fine-grained traffic scheduling and efficient IPv6 source routing encapsulation for services in 5G scenarios, and what benefits G-SRv6 can provide over SRv6. Using APN6, the eMBB, mMTC, and uRLLC traffic is forwarded following the high-bandwidth path, the Service Function Chain (SFC) path, and the lowest latency path, respectively. Using APN6 over G-SRv6, over 50% transmission overhead is reduced, and the Flow-Completion Time (FCT) is shortened from 923s to 102s. Comparing to SRv6 (with 10 SIDs in SRH), the forwarding rate of an SRv6 endpoint node is raised by 55% from 400Mpps to 620Mpps. In summary, the application-aware G-SRv6 helps network operators to reduce the cost and generate more revenue in the 5G area.

II. APPLICATION-AWARE G-SRV6

Normally, SRv6 SIDs are allocated from an address block within an SRv6 domain, so the SIDs share the common prefix (CP) of the address block[5]. An SRv6 SID has the format shown in Fig. 1.

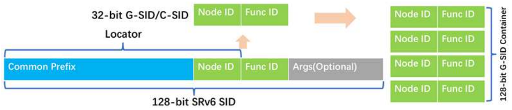
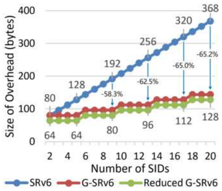


Fig. 1. Format of the128-bit SRv6 SID and 32-bit G-SID

In most cases, only Node ID and Function ID are different among the SIDs in a SID list, while the common prefix and argument parts are redundant. Removing the redundant parts of the SID list can reduce the overhead. Generalized SRv6 (G-SRv6) realizes this idea. It only carries the compressed SID consisting of node ID and Function ID in the SRH, so that the size of the SRH is compressed. Theoretically, up to 75% overhead of the SRv6 SID list can be reduced. The SID size comparison between SRv6 and G-SRv6 with 32-bit C-SIDs is shown in Fig. 2.



Scheme	Application Throughput *	Network Throughput *	FCT *	RTT **	Forwarding Rate **	Bandwidth Utilization *
Best Effort (no APN)	0.94Gbps	0.94Gbps	923s	300.114 ms	/	10.28%
APN SRv6	7.48Gbps	9.01Gbps	114s	0.259 ms	400Mpps	83.07%
APN G-SRv6	**8.36Gbps**	**9.01Gbps**	**102s**	**0.259 ms**	**620Mpps**	**92.78%**

Source of Test Results: * eMBB, analogy with mMTC ** uRLLC

Fig. 2. Comparison between SRv6 and G-SRv6

In order to locate the 32-bit C-SID within the 128-bit space located by Segment Left (SL) in SRH, Segment Index (SI) is defined, and it is the least 2 bits in the argument of the active SID in the IPv6 destination address (DA) field. Furthermore, a Continuation of Compression (COC) flavor is defined [5] to instruct the Segment Endpoint Node to continue to process the 32-bit C-SID in the SRH. When an SRv6 endpoint node receives a SID with COC Flavor, it updates the 32-bit G-SID in the IPv6 DA with the next 32-bit G-SID, and the next G-SID is located at SRH[SL][SI]. Otherwise, the node performs normal SRv6 processing[5]. In application-aware G-SRv6 networks, APN6 ID is added into the IPv6 Hop-by-Hop (HBH) header by application clients and servers to convey the application information to the network layer, so that the network nodes can be aware of the application type of a user group and its requirements. When APN6 packets with APN6 ID are received at the G-SRv6 ingress node, the node steers the packets into corresponding G-SRv6 tunnel based on the APN6 ID and associated policies.

III. DEMONSTRATION

We have implemented APN6 function in Linux kernel to support adding APN6 ID to packets. Next, we enhanced Nginx to set APN6 ID for each socket. In addition, we developed a G-SRv6 kernel module, which supports selecting and encapsulating different G-SRv6 tunnels according to the APN6 IDs. As shown in Fig.3, a 5G transport testbed network is set up with Huawei NE40E routers and Huawei 2288H V5 servers, which consists of an access network, a metro network, and a backbone network. An APN6-capable user equipment (UE) is connected to the access network, an MEC is connected to the metro network and a cloud DC is connected to the backbone network. All links in the testbed are 10Gbps fiber links except the link connects to the MEC that is a 1Gbps cable link.

As mentioned above, three paths are provided for large file downloading, IoT metadata transmission and real-time message exchanging applications in 5G eMBB, mMTC and uRLLC scenarios, and they are identified by APN-ID 0xA, 0xB and 0xC, respectively. For comparison, six TE tunnels are built using SRv6 and G-SRv6, and the APN6 IDs are carried within the inner IPv6 HBH options header. The results of performance improvement are described as below.

1) eMBB, large file downloading from the Nginx server in the remote cloud DC (File size: 106.87 GB, Packet payload size >1024 Bytes) over a 14-hop path: Without APN6 and SRv6/G-SRv6, the traffic is forwarded through a path with small bandwidth, and the transmission rate is 0.94Gbps, FCT is 923s. Using APN6 over SRv6 TE tunnel, the traffic is forwarded in the high-bandwidth path, the transmission rate is 7.48Gbps, FCT is 114s, and bandwidth utilization is 83.07%. Using APN6 over G-SRv6 TE tunnel, the transmission rate is 8.36Gbps, FCT is 102s, 53.33% transmission overhead is reduced, and bandwidth utilization is increased from 83.07% to 92.79%.

2) mMTC, IoT metadata transmission (Payload size: 128 Bytes) over a 10-hop path: Without APN6, the traffic is forwarded following the shortest path. Using APN6 over SRv6/G-SRv6, the traffic is forwarded over the Service Function Chain (SFC) path with a firewall deployed in MEC for security checking. Comparing to SRv6, the SID list (10 SIDs) is compressed from 160 bytes to 64 bytes in G-SRv6. In this situation, the forwarding rate of an SRv6 endpoint node is raised by 55% from 400Mpps to 620Mpps in G-SRv6 due to no packet recirculation.

3) uRLLC, real-time message exchanging traffic (Payload size:128 Bytes) over the 9-hop shortest path: Using APN6, the traffic is forwarded through the lowest latency path, and the latency is shortened from 300.114ms to 0.259ms comparing to another path. Comparing to SRv6, 45.45% transmission overhead is reduced in G-SRv6, and bandwidth utilization is increased from 42.11% to 57.14%.

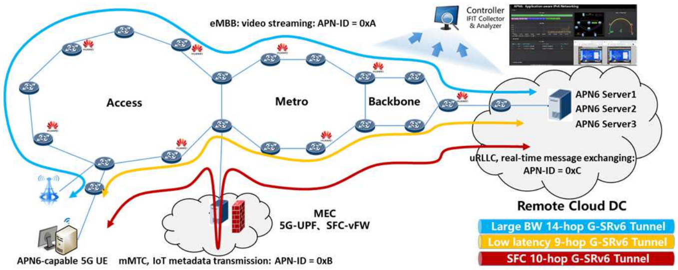


Fig. 3. Application-aware G-SRv6 demo setup

The demo shows the application-aware G-SRv6 provides a fine-grained traffic scheduling and a more economical and efficient encapsulation for the eMBB, mMTC and uRLLC applications' traffic in 5G transport network, respectively, which reduces transmission overhead and increases bandwidth utilization and more revenue.

ACKNOWLEDGMENT

The authors would like to acknowledge the team: Yaqun Xiao and Haoyu Song, as well as the IETF standard partners: Weiqiang Cheng(CMCC), Chongfeng Xie(China Telecom), Hui Tian(CAICT), Francois Clad(Cisco), Aihua Liu(ZTE), Yisong Liu(CMCC), Shay Zadok(Broadcom), Feng Zhao(CAICT) and Cong Li (China Telecom).

REFERENCES

[1] S. Peng, J. Mao, R. Hu and Z. Li, "Demo Abstract: APN6: Application-aware IPv6 Networking," *IEEE INFOCOM 2020*, pp. 1330-1331, doi: 10.1109/INFOCOMWKSHPS50562.2020.9162934.

[2] C. Filsfils, D. Dukes, S. Previdi, J. Leddy, S. Matsushima, D. Voyer, "IPv6 Segment Routing Header (SRH)", IETF RFC8754, Oct. 2019.

[3] Cheng, W., Li, Z., Li, C., Xie, C., Cong, L., Tian, H., and F. Zhao, "Generalized SRv6 Network Programming",IETF, March 2020.

[4] Li, Z., Li, C., Cheng, W., Xie, C., Cong, L., Tian, H.,and F. Zhao, "Generalized Segment Routing Header", IETF, February 2020.

[5] Cheng, W., Li, Z., Li, C., Clad, F., Aihua, L., Xie, C., Liu, Y., and S. Shay, "Generalized SRv6 Network Programming for SRv6 Compression", IETF, May 2020.