IETF Hackathon
Application-aware G-SRv6 networking

IETF 110
March 1-5, 2021
Online
Cheng Li
Huawei IP Standard Representative
• 30+ IETF drafts, 10 + WG drafts, 1 RFC
• Currently focus on G-SRv6/SRv6, SFC, OAM
• Author of books
  – “SRv6 Network Programming - Ushering in a New Era of IP Networks”
  – “Refactoring Network: Architecture and Implementation of SDN”
• Paper: “Application-aware G-SRv6 network enabling 5G services”, INFOCOM 2021

Jianwei Mao
Huawei IP Senior Engineer for Research
• Currently focus on CFN, G-SRv6/SRv6, APN6
• Author of books
  – “SRv6 Network Programming - Ushering in a New Era of IP Networks”
• Paper
  – “Application-aware G-SRv6 network enabling 5G services”, INFOCOM 2021
  – “CFN-dyncast: Load Balancing the Edges via the Network”, IEEE WCNC 2021
Hackathon Plan

• Develop functions of Generalized SRv6 (G-SRv6), based on Linux Kernel.
• Combine G-SRv6 with APN6, to achieve Application-aware G-SRv6 networking.
  – G-SRv6 IETF drafts:
    draft-lc-6man-generalized-srh  Data plane extension for Generalized Segment Routing Header
    draft-cl-spring-generalized-srv6-np Generalized SRv6 Network Programming
    draft-cl-spring-generalized-srv6-for-cmpr Generalized SRv6 Network Programming for SRv6 Compression
  – APN6 IETF drafts:
    draft-li-6man-app-aware-ipv6-network Data plane extension for Application-aware IPv6 Networking (APN6)
    draft-li-apn-framework Application-aware Networking (APN) Framework
    draft-peng-apn-scope-gap-analysis APN Scope and Gap Analysis
  – Open Communities:
    https://github.com/APN-Community APN6 Community
    https://www.ipv6plus.net IPv6+ Community
G-SRv6 Introduction

Problem Statement

- Transmission overhead of SRv6 is too high.

G-SRv6

- Reduce 75% size of SID List (transmission overhead).
- No new IPv6 address consumption, no new route creation.
- Fully compatible with SRv6, incremental deployment, deploy on demand.

12+ Vendors have PASSED Interop-test

<table>
<thead>
<tr>
<th>Locator</th>
<th>C-SID</th>
<th>C-SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common Prefix</td>
<td>Node-ID1</td>
<td>Func ID1, ArgPadding(opt)</td>
</tr>
<tr>
<td>Common Prefix</td>
<td>Node-ID2</td>
<td>Func ID2, ArgPadding(opt)</td>
</tr>
<tr>
<td>Common Prefix</td>
<td>Node-ID3</td>
<td>Func ID3, ArgPadding(opt)</td>
</tr>
<tr>
<td>Common Prefix</td>
<td>Node-ID4</td>
<td>Func ID4, ArgPadding(opt)</td>
</tr>
<tr>
<td>Common Prefix</td>
<td>Node-ID10</td>
<td>Func ID10, ArgPadding(opt)</td>
</tr>
</tbody>
</table>

SRv6 SID List
16 * 10 = 160 Bytes

G-SRv6 SID List
4 * 10 = 40 Bytes

IETF Hackathon: G-SRv6 & APN6
APN6 Introduction

APN6 makes use of IPv6 Extension Headers

- Convey the application related information, including its SLA requirements, along with the packet to the network.
- Allows the network to quickly adapt and perform the necessary actions for SLA guarantees.
Application-aware G-SRv6 networking

- Enable application-aware fine-grained strict TE, with lower transmission overhead.

  - Application-aware control
  - Based on service requirements

  - More efficient encapsulation
  - SLA-guaranteed transmission resources
  - Network programmability, fully compatible with SRv6
Implemented Functions

• We’ve implemented the demo based on *Linux Kernel* & *Huawei Router*.

• Functions in our demo:
  – G-SRv6:
    1. Identify APN6 info, and select the most suitable G-SRv6 TE tunnel for the specific App / flow.
    2. Encapsulate G-SRv6 Routing Header (*Generalized SRH*).
    3. Implement COC Flavor for End, End.X behavior.
    4. Implement End, End.X, End.DT6 as defined by SRv6 (*G-SRv6 is compatible with SRv6*).
    5. Implement G-SRv6 Local SID Table.
  – APN6:
    1. Encapsulate APN6 Options in IPv6 Hop-by-Hop Options Header, with application-specific info.
**Demo & Result**

- Topology with three layers
  - Three TE paths with 10+ hops (10+ SIDs in the SID List), for Apps with different SLA requirements.
- Apps:
  1. **File Downloading** (Security checking in a SFC)
  2. **Interactive Control** (Live & Short message)
  3. **HD Video on demand**
- G-SRv6’s Forwarding Rate is **55%+** higher than SRv6’s.
- For 128 bytes payload, Overhead is reduced by **50%+**

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Application Throughput *</th>
<th>Network Throughput *</th>
<th>FCT  *</th>
<th>RTT **</th>
<th>Forwarding Rate **</th>
<th>Bandwidth Utilization *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best Effort (no APN)</td>
<td>0.94Gbps</td>
<td>0.94Gbps</td>
<td>923s</td>
<td>300.114 ms</td>
<td>/</td>
<td>10.28%</td>
</tr>
<tr>
<td>APN SRv6</td>
<td>7.48Gbps</td>
<td>9.01Gbps</td>
<td>114s</td>
<td>0.259 ms</td>
<td>400Mpps</td>
<td>83.07%</td>
</tr>
<tr>
<td>APN G-SRv6</td>
<td>8.36Gbps</td>
<td>9.01Gbps</td>
<td>102s</td>
<td>0.259 ms</td>
<td>620Mpps</td>
<td>92.78%</td>
</tr>
</tbody>
</table>
What we learned

Feedback to WG:

• G-SRv6 can improve utilization and value of bandwidth significantly.
• G-SRv6 is fully compatible with SRv6, and can apply to more scenarios.
  • e.g. Real-time control, Video on demand, HD Live streams, SFC, etc.
• Combining with APN6, flows of many kinds of Apps can be distinguished fine-grained, the SLA requirements of specific Apps can be guaranteed better.

In the future:

• We can make more proof of concept tests in wider area networks, such as CENI.
• We may share our codes of this demo openly in our Github community.
• Processing delay in our Linux prototype is higher than SRv6 now (~1ms), welcome to join us to improve it together!
Thank you :)  

Team members:

- Jianwei Mao, Huawei ([maojianwei@huawei.com](mailto:maojianwei@huawei.com))
- Jiang Liu, BUPT ([liujiang@bupt.edu.cn](mailto:liujiang@bupt.edu.cn))
- Weihong Wu, BUPT ([wwh_bupt@foxmail.com](mailto:wwh_bupt@foxmail.com))
- Lin He, Tsinghua ([he-l14@mails.tsinghua.edu.cn](mailto:he-l14@mails.tsinghua.edu.cn))
- Cheng Li, Huawei ([c.l@huawei.com](mailto:c.l@huawei.com))
- Shuping Peng, Huawei ([pengshuping@huawei.com](mailto:pengshuping@huawei.com))

Open Communities:

- [https://github.com/G-SRv6](https://github.com/G-SRv6)
- [https://github.com/APN-Community](https://github.com/APN-Community)
- [https://www.ipv6plus.net](https://www.ipv6plus.net)

G-SRv6 Documents:

- [draft-lc-6man-generalized-srh](draft-lc-6man-generalized-srh)
- [draft-cl-spring-generalized-srv6-np](draft-cl-spring-generalized-srv6-np)
- [draft-cl-spring-generalized-srv6-for-cmpr](draft-cl-spring-generalized-srv6-for-cmpr)

APN6 Documents:

- [draft-li-6man-app-aware-ipv6-network](draft-li-6man-app-aware-ipv6-network)
- [draft-li-apn-framework](draft-li-apn-framework)
- [draft-peng-apn-scope-gap-analysis](draft-peng-apn-scope-gap-analysis)