SPRING Working Group Louis Chan
INTERNET-DRAFT
Intended status: Standard Track Juniper Networks
Expires: Jan 4, 2023 Jul 4, 2022

Sub-slicing for SRv6
draft-chan-spring-srv6-sub-slice-00.txt

Abstract

This document describes how to achieve further slicing or traffic engineering interoperability between vendors without the use of SRH.

Slicing or traffic engineering information is encapsulated as part of the SRv6 SID. Use of IP longest prefix match approach to identify the further slicing via sub-slice identifier.

The traffic engineering from one end to another end is seen as segment by segment approach. This approach could solve the scalability of traffic engineering tunnels required in a huge network, which order of N^2 has be considered.

Table of Contents

1. Introduction

The purpose of this document is to describe to a way to signal the desired slicing or sub-slicing information with the SRv6 endpoint behavior SID.

The FUNCT:ARG portion of SRv6 SID is encoded with certain format to achieve such.

In the transit router, when the SRv6 packet is received, it is processed with IPv6 longest prefix match (LPM) approach, which in turn, could point the packet to another tunnel, likely a SRv6-TE tunnel. The effect is similar to a binding SID approach.

The benefits of this approach are

- Provide an easy interoperability method between vendors for slicing without the full SRH header. This format looks legitimate to any vendors.
- Reduce the tunnels to be provisioned in the network based on tactical TE strategy. It would give less work to controller to handle huge number of tunnels in a big scale network.
- Provide an intrinsic backup path. Secondary path provisioning is not a requirement.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in RFC 2119.

3. Encoding sub-slice in FUNCT:ARG

The method is to encode the sub-slice information into the FUNCT:ARG or the endpoint portion

```
+------------+--------------+-------------------------+
|   Locator  |  Sub-slice ID |  Remainder for behavior  |
+------------+--------------+-------------------------+
             |<-        Endpoint Behavior             ->|
```

This format could be used in conjunction WITH or WITHOUT Flex-Algo. If it is used with Flex-Algo, network slicing is expected. Therefore, the use of the term, sub-slice, is to allow further level of slicing within Flex-Algo.

The encoding of sub-slice information is right from the originator of endpoint behavior. For example, the sub-slice information is encoded in END.DT4 and END.DT6 from the originator of the VPN.

No SRH is required in the first ingress PE to send out the packet.

Each node in transit would evaluate the IPv6 header according to longest prefix match rule as normal IP processing. It will forward the packet according to locator routes it learnt from routing protocol.

When a more specific IPv6 route with "locator + sub-slice", which is a longer prefix, is programmed in the routing table, the forwarding decision would be redirected to another TE tunnel. In this case, sub-slice is achieved.

4. Example of operation

```
R1--/--C2--/--C3--/--R4        ;  "/ " means network in between
```

Between R1 and R4, it runs two L3VPN with END.DT4 SID.

Sub-slice TE tunnel is created between C2 and C3 for one of the VPN above.

The locator is encoded as
FC00:xxxx:nnnn::/48, where xxxx is the Flex-Algo portion. Here is 0200 as

example
                              nnnn is the locator portion. R1: 0001, R4: 0004

For the END.DT4 announced from R4

VPN1: FC00:0200:0004::1000
VPN2: FC00:0200:0004:0008::2000   ; 0008 here is a sub-slice ID

From R1, it sends packets for both VPNs without any SRH. Below example only shows the headers but not the payload.

In C2, its routing table has two entries.

a) FC00:0200:0004::0/48 - This is learnt via routing protocol
b) FC00:0200:0004:0008::0/64 - This entry is programmed by other means for TE

For VPN1 traffic, C2 will forward the packet using (a) route.

For VPN2 traffic, C2 will forward the traffic to another path via (b) route, and attach tunnel information, like SRv6-TE. An example of additional hops programmed in SRH sending out from C2 to C3 could be

```
 FC00:0200:pppp::1            ; add SRH with nodes
 FC00:0200:qqqq::1            ; where pppp and qqqq represent intermediate nodes
```

The SRv6-TE tunnel could be in the form of either insert mode or encapsulation mode. C3 could received the packet with or without SRH depending on the configuration.

C3 should remove SRH header if it is the endpoint of the tunnel. It would continue to forward the packet according to FC00:0200:0004:0008::2000.

Between C2 and C3, sub-slice is thus achieved for VPN2 traffic.

It depends on C3 or subsequent routers' forwarding table programming. Another TE action could be imposed based on /64 LPM interpretation. This allows easy interoperability between vendors. C3 might be the border router from another vendor domain.

If the above C2 to C3 tunnel is down, C2 would forward the packet using /48 route, which is the default Flex-Algo route. Hence, backup path is readily available. Secondary tunnel provisioning is therefore optional.

5. Solution to possible looping issue

There is a possibility for routing loop in certain scenario. The options to minimize the risk are

- Running OAM to detect the TE tunnel reachability up to the final end node.
- TE tunnel for sub-slice should be terminated on a trusted node, probably an ABR.
  When the packet arrives this trusted node, either the packet is forwarded properly, or discarded silently.
- Use of controller to detect the possibility of routing loop. Avoid or disable the TE path for sub-slice when necessary.

6. Compatibility with SRv6 compression

TBD for sub-slice notation.

For the TE tunnel portion, it would be compatible to any SRv6 compression scheme.
This sub-slice would provide an interoperability between different domains which run different kind of SRv6 compression mechanism. The lowest denominator is an SRv6 packet without SRH.

7. Interoperability consideration

The benefit of this approach is to allow a clean handover of a SRv6 packet from one domain to another domain.

With the use of Flex-Algo [FLEXALGO] in SRv6, there would be more granularity of slicing information present in the header. A further example from section 4,

```
FC00:0200:0004:0008::2000      - 0200 means Flex-Algo A, sub-slice 0008

FC00:0300:0004:0009::2000      - 0300 means Flex-Algo B, sub-slice 0009
```

The two SRv6 connected domains do not require use of the same tunnel technology, like plain Flex-Algo, SRv6-TE with or without compression. Each domain would have enough information from the packet header to steer the traffic into specific tunnel, if required.

On the other hand, the overhead of traffic engineering header is reduced from end to end. If there is TE header overhead, it is reduced to local domain consumption only.

8. Multi-level sub-slicing

Multi-level of sub-slice is easily achievable via different prefix length. For example, /48 up to the locator, /56 for first level sub-slice and /64 for second level sub-slice.

As an example, /56 case could be used as an aggregate for a group of the endpoint behavior function, and /64 are used on individual.

```
a) FC00:0200:0004::0/48        - No sub-slice
b) FC00:0200:0004:01xx::0/56   - 01 denotes a sub-slice
```

c) FC00:0200:0004:0103::0/64 - 0103 denotes further sub-slice. A more specific TE

With a longer prefix length, the traffic could be directed to more specific traffic engineering path. The TE path could use any kind of tunnel method, like SRv6-TE without header compression, or with compression.

9. Security Considerations

TBD

10. Others

This proposed method also allow SRv6 traffic to tunnel through non-SRv6 domain in the middle. Router C2 in the above example could initiate other kinds of tunnel, which could be RSVP LSP, SR-TE LSP and etc.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

11.2. Informative References

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

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

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

[FLEXAGLO] S. Hegde, P. Psenak and etc, IGP Flexible Algorithm <https://datatracker.ietf.org/doc/draft-ietf-lsr-flex-algo>

12. Acknowledgments

The following people have contributed to this document:
Salih K A, Juniper Networks

Author Address

Louis Chan (editor)
Juniper Networks
2604, Cityplaza One, 1111 King's Road
Taikoo Shing
Hong Kong

Phone: +852-25876659
Email: louisc@juniper.net