SRv6 for Mobile User-Plane

draft-matsushima-spring-dmm-srv6-mobile-uplane-03

IETF100

S.Matsushima, C.Filsfils, M.Kohno, D.Voyer
Feedbacks after IETF99

• Many people asked: System Impacts?
  • To current control-plane protocol.
  • To current RAN.
  • People really care degree of system impact to change U-plane protocol from current one.

• Benefits?
  • What is able to do with SRv6?
  • Isn’t that possible with current u-plane protocol?
  • Isn’t SRv6 just another tunneling protocol?

• Nobody asked: How SRv6 works for mobile user-plane
  • Sounds good, I did it. 😊
Updates to v03: Answer to the Feedback

• **Introduces “Basic Mode” User-Plane**
  • (It is supposed) **No impact** to control-plane, but no advanced SRv6 features in there.
  • Operator is able to gradually migrate from basic to more advanced mode.

• **Introduces an Use Case “Stateless Interworking with Legacy Access”**
  • (It is supposed) **No impact** to current RAN in control-plane.

• **Introduces “Aggregate Mode” User-Plane**
  • Benefits **seamless deployment** of service-chain, VPNs and TE within the mobile user-plane.
  • Complicated? Mobile control-plane can focus to only manage mobility and keep simple.
    • Implementing other service policies to the user-plane can be done by separate systems.
  • Complicate text? May need to find concise way to describe the procedures.
Leveraging Current Control-Plane

Control-Plane Message

- Tunnel endpoint Address (A::)
- Tunnel Identifier (0x12345678)

MAG/LMA/SGW/PGW/eNB

Control-Plane Entity

- Tunnel endpoint Address (A::)
- Tunnel Identifier (0x12345678)

User-Plane Entity

- SID: A::1234:5678

FIB table

Payload SRH DA=A::1234:5678 SA=S::

Payload DA=D:: SA=S::
Stateless Interworking with Legacy Networks

<table>
<thead>
<tr>
<th>Pay load</th>
<th>DA</th>
<th>SA</th>
<th>Tun-ID</th>
<th>DA</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>v6</td>
<td>v6</td>
<td></td>
<td>v4</td>
<td>v4</td>
</tr>
</tbody>
</table>

IPv6 Network

Internet, Service network

IPv4 header

Tunnel header

IPv4 Network

SRv6 Enabled IPv6 Network

Locator | DAv4 | SAv4 | Tun-ID
--------|------|------|--------
128-a-b-c | a    | b    | c     
Updates to v03: Technical Progress

• Introduces New SRv6 Functions: “End.TM” and “T.Tmap”

• To support Stateless Interworking with legacy user-plane with some parameters.

**End.TM**
(Endpoint function with encaps for mapped tunnel)

SRv6 -> Legacy

1. IF NH=SRH & SL > 0 THEN
2. decrement SL
3. update the IPv6 DA with SRH[SL]
4. push header of TUN-PROTO with tunnel ID from S ;; Ref1
5. push outer IPv4 header with SA, DA from S
6. ELSE
7. Drop the packet

**T.Tmap**
(Transit behavior with decaps tunnel and map SRv6 policy)

Legacy-> SRv6

1. IF P.PLOAD == TUN-PROTO & T.PLOAD == IPv6 THEN ;; Ref1, Ref1bis
2. pop the outer IPv4 header and tunnel headers
3. copy IPv4 DA, SA, TUN-ID to form SID B with IW-IPv6-Prefix
4. insert the SRH (D, B; SL=1) ;; Ref2, Ref2bis
5. set the IPv6 DA = B
6. forward along the shortest path to B
7. ELSE
8. Drop the packet

**Ref1:** TUN-PROTO indicates target tunnel type.

**Ref2:** P.PLOAD and T.PLOAD represent payload protocol of the receiving packet, and payload protocol of the tunnel respectively.
Work in Progress

• **QoS and Accounting**
  • Enables SID to represent QoS and accounting policy.

• **E2E SR and Network Slicing**
  • Enables Apps running on MN be able to designate slices.

• **IPv4 Support**
  • Carries IPv4 user packets.
  • Many IPv6 transition solutions make it can be considered as an user application on IPv6.
    • MAP-E, MAP-T, NAT64, 464XLAT and DS-Lite.

• **Collaborations**
  • 3GPP CT4 to initiate study work of user-plane protocol.
Basic Mode

<table>
<thead>
<tr>
<th></th>
<th>Uplink</th>
<th>Downlink</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Point</td>
<td>T.Insert</td>
<td>End</td>
</tr>
<tr>
<td>L2 Anchor Node</td>
<td>End.B6</td>
<td>End.B6</td>
</tr>
<tr>
<td>L3 Anchor Node</td>
<td>End.T</td>
<td>T.Insert</td>
</tr>
</tbody>
</table>

* SRv6 Network Programming

IPv6 Network
Basic Mode User-Plane Flows (Uplink)

IPv6 Network

IPv6 Header
- SA=S::
- DA=D::
- NH=TCP
- Payload

MN → A1::1 → A2::1 → A3::1

Internet, Service network
Basic Mode User-Plane Flows (Uplink)

IPv6 Network

MN

IPv6 Header

<table>
<thead>
<tr>
<th>SA=S::</th>
<th>DA=D::</th>
<th>NH=TCP</th>
</tr>
</thead>
</table>

Payload

SRH

<table>
<thead>
<tr>
<th>SL=1</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SID[0]=D::</th>
</tr>
</thead>
</table>

| SID[1]=A2::1 |

Payload

A1::1

A2::1

A3::1

Internet, Service network

D::
Basic Mode User-Plane Flows (Uplink)

IPv6 Network

SA=S::
DA=D::
NH=TCP
Payload

SRH
SL=1
SID[0]=D::
SID[1]=A2::1
Payload

SA=S::
DA=A2::1
NH=SRH(43)

SA=S::
DA=A3::1
NH=SRH(43)

SL=0
SID[0]=A3::1
SL=1
SID[0]=D::
SID[1]=A2::1
Payload
Basic Mode User-Plane Flows (Uplink)

IPv6 Network

SA=S::
DA=D::
NH=TCP
Payload

SRH
SL=1
SID[0]=D::
SID[1]=A2::1
Payload

End.B6
SA=S::
DA=A3::1
NH=SRH(43)
Payload

SRH
SL=0
SID[0]=A3::1
SL=1
SID[0]=D::
SID[1]=A2::1
Payload

End.T
SA=S::
DA=D::
NH=TCP
Payload

T.Insert
SA=S::
DA=A2::1
NH=SRH(43)
Payload

IPv6 Header

MN

S::

A1::1

A2::1

A3::1

D::

CN

Internet, Service network

IPv6 Network

IPv6 Header
Basic Mode User-Plane Flows (Uplink)

IPv6 Network

IPv6 Header
- SA=S::
- DA=D::
- NH=TCP

Payload
- SRH
  - SL=1
  - SID[0]=D::
  - SID[1]=A2::1

IPv6 Header
- SA=S::
- DA=A2::1
- NH=SRH(43)

Payload

IPv6 Header
- SA=S::
- DA=A3::1
- NH=SRH(43)

Payload

IPv6 Header
- SA=S::
- DA=D::
- NH=TCP

Payload
Basic Mode User-Plane Flows (Downlink)

IPv6 Network

MN

SA=D::
DA=S::
NH=TCP
Payload
Basic Mode User-Plane Flows (Downlink)

IPv6 Network

SA=D::
DA=A2::2
NH=SRH(43)
SL=1
SID[0]=D::
SID[1]=A2::2
Payload

SA=D::
DA=S::
NH=TCP
Payload
Basic Mode User-Plane Flows (Downlink)

IPv6 Network

MN A1::1

SA=D::
DA=A1::1
NH=SRH(43)
SL=1
SID[0]=D::
SID[1]=A2::2
Payload

A2::2

SA=D::
DA=A2::2
NH=SRH(43)
SL=1
SID[0]=D::
SID[1]=A2::2
Payload

A3::1

End.B6 w/ PSP

T.Insert

Internet, Service network

IPv6 Network

SA=D::
DA=S::
NH=TCP
Payload
IPv6 Network

SA=S::
DA=D::
NH=TCP

Payload

SRH

SL=1
SID[0]=D::
SID[1]=A2::2

Payload

IPv6 Header

SA=D::
DA=A1::1
NH=SRH(43)

Payload

SRH

SL=1
SID[0]=D::
SID[1]=A2::2

Payload

IPv6 Header

SA=D::
DA=A2::2
NH=SRH(43)

Payload

IPv6 Header

SA=D::
DA=S::
NH=TCP

Payload

Basic Mode User-Plane Flows (Downlink)
A Current Mobile Network Example

- Well fragmented to RAN, EPC and SGi.
- Per-session tunnel creation and handling.
- Non-optimum data-path.
What if SRv6 Becomes An Alternative of GTP-U Tunnel?

- Well fragmented to RAN, EPC and SGi.
- Per-session tunnel creation and handling.
- Non-optimal data-path.
- IPv6 integrates networks of the mobile and others.
- A SID represents data-plane role and function.
SRv6 in A Nutshell

**SRH (Segment Routing Header)**

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Header</td>
<td>Hdr Ext Len</td>
<td>Route Type</td>
<td>Segments Left</td>
</tr>
<tr>
<td>Last Entry</td>
<td>Flags</td>
<td>Tag</td>
<td></td>
</tr>
</tbody>
</table>

**Segment List[0]** (128 bits IPv6 address)

... 

**Segment List[n]** (128 bits IPv6 address)

// Optional Type Length Value objects (variable) //
<table>
<thead>
<tr>
<th>SRv6 Function* Name</th>
<th>Forwarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>END</td>
<td>Lookup SRH</td>
</tr>
<tr>
<td>END.X</td>
<td>L3 cross-connect to next-hop</td>
</tr>
<tr>
<td>END.T</td>
<td>L3 lookup IPv6 table</td>
</tr>
<tr>
<td>END.DT6</td>
<td>Decap outer IPv6 hdr and lookup IPv6 table</td>
</tr>
<tr>
<td>END.DT4</td>
<td>Decap outer IPv6 hdr and lookup IPv4 table</td>
</tr>
<tr>
<td>END.DX6</td>
<td>Decap outer IPv6 hdr and IPv6 cross-connect</td>
</tr>
<tr>
<td>END.DX4</td>
<td>Decap outer IPv6 hdr and IPv4 cross-connect</td>
</tr>
<tr>
<td>END.B6</td>
<td>Bound to an SRv6 policy(SID list)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SRv6 Function* Name</th>
<th>Forwarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Pure IPv6 transit</td>
</tr>
<tr>
<td>T.Insert</td>
<td>Insert an SRv6 policy (SID list)</td>
</tr>
<tr>
<td>T.Encaps</td>
<td>Encap SRv6 policy (SID list) by outer IPv6 hdr</td>
</tr>
</tbody>
</table>

* SRv6 Network Programming
E2E Mobile Orchestration with SRv6

- **Data-plane nodes are NOT dedicated to specific roles.**
  -> SID represents each data-plane role.
- **Orchestrator puts SIDs to the nodes with its functions**
  -> It requires some data models to instantiate the data-plane
Data Model for Mobile Orchestration with SRv6

ietf-dmm-fpc.yang

Mobile Control-Plane & Apps

Orchestrator /Controller

UL:T.Insert DL:END.X
UL:END.T DL:T.Insert

SRv6 Network

Internet, Service network
SRv6 for Network Slicing

- A set of SIDs represents Network Slice.
  -> Sharing same prefix among SIDs in a slice would work.
- Then user packets could also indicate Slices by SID.
  -> Applications in a MN could be able to use SID to do that.

<table>
<thead>
<tr>
<th>payload</th>
<th>SRH</th>
<th>DA</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slice SID</td>
<td>IPv6 header</td>
<td>IPv6 header</td>
<td></td>
</tr>
</tbody>
</table>

Contents for NetSlice-A
Contents for NetSlice-B
Contents for NetSlice-C
References

- IPv6 Segment Routing Header (SRH)
  - draft-ietf-6man-segment-routing-header

- SRv6 Network Programming
  - draft-filsfils-spring-srv6-network-programming

- ietf-dmm-fpc.yang
  - A SDO neutral mobile data-plane model as a part of the FPC work in IETF DMM working group.
  - draft-ietf-dmm-fpc-cpdp
Summary

• **SRv6 is expected to make mobile network to be:**
  • Simple to operate in E2E basis.
  • Flexible where to deploy various functions.

• **SID Functions for mobile data-plane represent:**
  • Access point, L2 Anchor, and L3 Anchor node.
  • Interworking node in stateless manner with some new SRv6 function and parameters.

• **Basic Mode vs. Aggregate Mode**
  • Basic mode works with existing c-plane protocol and interwork with current user-plane.
  • Aggregate mode introduces advanced features of SRv6 to seamless deployment which are service chain, VPNs, TE etc., with mobility management.
Next Step

• Be a start point for user-plane optimization work?