SRv6 for Mobile User-Plane

draft-ietf-dmm-srv6-mobile-uplane-03

IETF103

S. Matsushima, C. Filsfils, M. Kohno, D. Voyer, C. Perkins, P. Camarillo
Summary of Updates from v02 to v03

• **Defined “Args.Mob.Session” for SID of SRv6 Mobile User Plane.**
  - QFI, RQI and PDU Session ID are supplied as SID arguments.
  - The SIDs MAY use the arguments if required by UPFs.

• **Modified End.MAP function.**
  - To support more than one SID mapping to received SID in the DA.

• **Added new terminology section for abbreviations and conventions.**
  - See section 2.1 and 2.2.

• **Added new terminology section for pre-defined SRv6 functions.**
  - See section 2.3.

• **Editorial updates to improve readability**
Args.Mob.Session Format

<table>
<thead>
<tr>
<th>PDU Sess(cont')</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDU Session ID</td>
</tr>
<tr>
<td>R</td>
</tr>
<tr>
<td>QFI</td>
</tr>
</tbody>
</table>

---

- **QFI**: QoS Flow Identifier [TS.38415]
- **R**: Reflective QoS Indication [TS.23501]. This parameter indicates the activation of reflective QoS towards the UE for the transferred packet. Reflective QoS enables the UE to map UL User Plane traffic to QoS Flows without SMF provided QoS rules.
- **U**: Unused and for future use. MUST be 0 on transmission and ignored on receipt.
- **PDU Session ID**: Identifier of PDU Session. The GTP-U equivalent is TEID.
Next Steps

• **Extend function coverage**
  • End-Marker support. (either End-Marker flag in SID, or O-bit in SRH.)

• **Examples with different UPFs**
  • e.g, traffic measurement, lawful intercept, multi-homing, charging, etc.,

• **Improve clarity and readability**
  • e.g; anchor and anchoring, etc.,
Appendix
Summary of Updates from v01 to v02

• Clarify supporting PDU types
  • IPv4, IPv6, IPv4v6, Ethernet and Unstructured as the supported PDU types.
  • Supported by corresponding SRv6 functions.

• Introduce some open source implementations in appendix
  • P4 code by ebiken
  • MCORD and OAI. (See I-D.camarillo-dmm-srv6-mobile-pocs)

• Add related references for:
  • Network Slicing
  • Control Plane considerations.

• Miscellaneous corrections
  • Fix some typos.
A View of 3GPP-friendly SRv6 UP Protocol Stack

*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)
A View of 3GPP-friendly SRv6 UP Protocol Stack
(gNB/N3 unchanged scenarios)

*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)
Traditional mode
AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
Mobile user-plane functions are the same ones as with GTP-U. It’s just a data plane replacement.

Equivalent with existing User Plane in terms of functionality.

PDU sessions mapped 1-for-1 with a GTP-U tunnel. In this mode, mapped with SRv6 policy.

gNB is SRv6 capable but from control plane viewpoint there’s no change.

Lower MTU overhead than GTP-U over IPv6/UDP!

draft-dukes-spring-mtu-overhead-analysis-00
Traditional mode - uplink

IPv6 Hdr  SA = A::, DA = Z::, Payload
IPv6 Hdr  SA = G::, DA = U1::, Payload
IPv6 Hdr  SA = A::, DA = Z::, Payload
IPv6 Hdr  SA = G::, DA = U2::, Payload
IPv6 Hdr  SA = A::, DA = Z::, Payload
Traditional mode - downlink

IPv6Hdr, SA = Z::, DA = A::
Payload

IPv6Hdr, SA = U2::, DA = G::1
Payload

IPv6Hdr, SA = U2::, DA = U1::
Payload

IPv6Hdr, SA = Z::, DA = A::
Payload
The “Endpoint function with SID mapping” (End.MAP) is used in several scenarios. Particularly in mobility, is is used in the UPF for

When N receives a packet destined to S and S is a local End.Map SID, N does:

1. look up the IPv6 DA in the mapping table
2. update the IPv6 DA with the new mapped SID ;; Ref1
3. forward according to the new mapped SID
4. ELSE /* if S is NOT a local End.Map SID */
5. Drop the packet

Ref1: SRH is NOT modified if it exists in the header.
Enhanced mode
AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
Enhanced mode

- Several UE share the same SR policy (and its SIDs)
- The SR policy includes Traffic Engineering (C1) and NFV (S1)
- The gNB control-plane (N2 interface) might, or might not be unchanged:
  - If unchanged, we signal a single IP address that the gNB resolves with PCEP, reverse DNS, LISP into a SID list
  - If changed, we signal a full SID list over the N2 interface
Enhanced mode - uplink

End.*: Appropriate SRv6 End function type for the
End.*: Appropriate SRv6 End function type for the purposes.
Enhanced mode with unchanged gNB IPv6 GTP behavior
SRv6 (N3 unchanged – IPv6/GTP)

At the N3 the packet is IPv6/GTP, but IPv6 DA is an SRv6 segment. Routing is based on SRv6.

*: IPv6 header + SRH (variable size: 1 SID = 16Bytes)
No SRH in traditional mode with just an IPv6 header (40Bytes)
C1: Traffic Engineering
S1: Service function instance running on NFV platform
X: GTP-U/SRv6 Interworking

AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
SR GW: Segment Routing Gateway between GTP-U/IPv6 and SRv6
Objective

- SR GW to UPF U2 is SRv6 capable for the underlay, overlay and service chaining
- GTP-U endpoint IPv6 addresses of gNB and SR GW could be treated as SRv6 SID.
- No software changes in the gNB
- To achieve this we deploy an SR GW in between gNB and UPF (N3 interface)
  - Any SRv6 capable router on hardware or software.
- Applies to any kind of PDU session types
Old: IPv6 addr “X::1” + TEID (1234)
New: IPv6 SID “X:A::1” + TEID (1234)

X:A:1 is BSID at X
TEID (1234) could be dummy
Uplink traffic

Old: IPv6 addr “X::1” + TEID (1234)
New: IPv6 SID “X:A::1” + TEID (1234)

X:A:1 is BSID at X
TEID (1234) could be dummy

SRGW advertise SID in LOC(64b of X) + Algorithm(8b of A and B) format:
- X:A::/72 -> Low latency
- X:B::/72 -> High bandwidth

SRGW does End.M.GTP6.D function for X:A:: and X:B::.
X:A::1 is a type of End.M.GTP6.D SID that pops the GTP header without lookup and pushes an outer header with SID list.
X:A:1 could be a flex-alg SID (end-to-end network slicing) for other SRv6 nodes.
X:A::1 can be shared across UEs
Uplink traffic

Old: IPv6 addr “X::1” + TEID (1234)
New: IPv6 SID “X:A::1” + TEID (1234)
X:A::1 is BSID at X
TEID (1234) could be dummy

Without modifying the N2 interface we are steering the UE packets to an SRGW along the designated algorithm path.

• SRGW advertise SID in LOC (64b of X) + Algorithm (8b of A and B) format:
  • X:A::/72 -> Low latency
  • X:B::/72 -> High bandwidth
• SRGW does End.M.GTP6.D function for X:A:: and X:B::.
• X:A::1 is a type of End.M.GTP6.D SID that pops the GTP header without lookup and pushes an outer header with SID list
• X:A::1 could be a flex-alg SID (end-to-end network slicing) for other SRv6 nodes.
• X:A::1 can be shared across UEs

End.*: Appropriate SRv6 End function type for the purposes.

• Old: IPv6 addr “X::1” + TEID (1234)
• New: IPv6 SID “X:A::1” + TEID (1234)
  • X:A::1 is BSID at X
  • TEID (1234) could be dummy

• Without modifying the N2 interface we are steering the UE packets to an SRGW along the designated algorithm path.

End.*: Appropriate SRv6 End function type for the purposes.
End.M.GTP6.D

- The "Endpoint function with IPv6/GTP decapsulation into SR policy" (End.M.GTP6.D) is used to in interworking scenario the direction from legacy user-plane to SRv6 user-plane network.
- When N receives a packet destined to S and S is a local End.M.GTP6.D SID, N does:

  1. IF NH=UDP and UDP.DST_PORT=GTP
  2. pop IP, UDP and GTP headers
  3. push an outer IPv6 header with its own SRH
  4. set the outer IPv6 SA to A
  5. set the outer IPv6 DA to the first segment of the SRv6 Policy
  6. forward according to the first segment of the SRv6 Policy
  7. ELSE
  8. drop the packet
Downlink traffic

Remote Endpoint == G1:: + TEID (3456)
G1:: is a gNB IPv6 addr

• UPF U2 (anchor point) maintains state of A:
  • FIB lookup: A/128 is matched to <C1::, S1::, X::2:3456, G1::>
  • SRGW SID X:: need be pre-configured, or through N4 or other means.
Downlink traffic

Remote Endpoint == G1:: + TEID (3456)
G1:: is a gNB IPv6 addr

• UPF U2 (anchor point) maintains state of A:
  • FIB lookup: A/128 is matched to <C1::, S1::, X::2:3456, G1::>
  • SRGW SID X:: need be pre-configured, or through N4 or other means.

End.*: Appropriate SRv6 End function type for the purposes.

• SRGW does End.M.GTP.E function.
• X::2 is an SRv6 End.M.GTP.E SID.
• Removes SRH, adds UDP and GTP with TEID received in SID ARGs
• IPv6 DA is the last segment of the SRH
• Scales (no state per UE in SR gateway)
• gNB could configure an IPv6 addresses per network slice.
  • G1::1 -> Low latency
  • G1::2 -> High bandwidth
Downlink traffic

- X::2 is an SRv6 End.M.GTP6.E SID.
- Removes SRH, adds UDP and GTP with TEID received in SID ARGs
- IPv6 DA is the last segment of the SRH
- Scales (no state per UE in SR gateway)
- gNB could configure an IPv6 address per network slice.
  - G1::1 -> Low latency
  - G1::2 -> High bandwidth

Remote Endpoint == G1:: + TEID (3456)
G1:: is a gNB IPv6 addr

- UPF U2 (anchor point) maintains state of A:
  - FIB lookup: A/128 is matched to <C1::, S1::, X::2:3456, G1::>
- SRGW SID X:: need be pre-configured, or through N4 or other means.

End.*: Appropriate SRv6 End function type for the
End.M.GTP6.E

- The "Endpoint function with encapsulation for IPv6/GTP tunnel" (End.M.GTP6.E) is used in interworking scenario for the direction from SRv6 user-plane to legacy user-plane network.

- When interworking node N receives a packet P destined to S and S is a local End.RAN SID, N does:

  1. IF NH=SRH & SL = 1 THEN ;; Ref1
  2. decrement SL
  3. store SRH[SL] in variable new_DA
  4. store TEID in variable new_TEID ;; Ref2
  5. pop the (outer) IPv6 header and its extension headers
  6. push an IPv6 header, a UDP header and a GTP-U header
  7. set the IPv6 DA to new_DA
  8. set the GTP_TEID to new_TEID
  9. lookup the new_DA and forward the packet accordingly
  10. ELSE
  11. drop the packet

Ref1: An End.M.GTP6.E SID must always be the penultimate SID.

Ref2: TEID is extracted from the argument space of the current SID.
Enhanced mode with unchanged gNB IPv4 GTP behavior
SRv6 (N3 unchanged – IPv4/GTP)

At the N3 the packet is IPv4/GTP.

*: IPv6 header + SRH (variable size: 1 SID = 16Byte)
No SRH in traditional mode with just an IPv6 header (40Byte)
C1: Traffic Engineering
S1: Service function instance running on NFV platform
X: GTP-U/SRv6 Interworking

AMF: Access & Mobility Function
SMF: Session Management Function
gNB: 5G eNodeB (i.e., base station)
UPF: User Plane Function
N2, N3, N4, N6, N9, N11: 5G reference points (functional block interfaces)
X2: inter-base station reference point
SR GW: Segment Routing Gateway between GTP-U/IPv4 and SRv6
Objective

- SR GW to UPF U2 is SRv6 capable for the underlay, overlay and service chaining
- GTP-U endpoint of gNB and SR GW addresses on N3 is IPv4.
- No software changes in the gNB
- To achieve this we deploy an SR GW in between gNB and UPF (N3 interface)
  - Any SRv6 capable router on hardware or software.
- Applies to any kind of PDU session types
• SRGW does T.M.Tmap function.
• SRGW is a UPF Uplink Classifier. Coexist with current UPFs.
• Steers traffic into SR policy.

IPv4 addr of SRGW + TEID (1234)
• X::G1:SRGW:3456 is an SRv6 End.M.GTP.4.E function
• Removes SRH, adds IPv4/UDP and GTP
• IPv4 SA, DA and GTP TEID are part of the SID’s arguments
• Scales (no state per UE in SR gateway)
The “Endpoint function with encapsulation for IPv4/GTP tunnel” (End.M.GTP4.E) is used to the direction from SRv6 user-plane to legacy user-plane network.

When interworking node N receives a packet destined to S and S is a local End.M.GTP4.E SID, N does:

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
5.    push outer IPv4 header with SA, DA from S
6. ELSE
7.    Drop the packet

Ref1: TUN-PROTO indicates target tunnel type.
The "Transit with mobile tunnel decapsulation and map to an SRv6 policy function (T.M.Tmap) for short) is used to the direction from legacy user-plane to SRv6 user-plane network.

When interworking node N receives a packet destined to a SRGW IPv4 address, N does:

1. IF P.PLOAD == TUN-PROTO THEN    ;; Ref1
2. pop the outer IPv4 header and tunnel headers
3. copy IPv4 DA, SA, TUN-ID to form SID B with SRGW-IPv6-Prefix ;; embedding IPv4 DA/SA/TEID in a SID could be an option.
4. encapsulate the packet into a new IPv6 header ;; Ref2, Ref2bis
5. set the IPv6 DA = B
6. forward along the shortest path to B
7. ELSE
8. Drop the packet

Ref1: P.PLOAD and T.PLOAD represent payload protocol of the receiving packet, and payload protocol of the tunnel respectively.

Ref2: The received IPv6 DA is placed as last SID of the inserted SRH.

Ref2bis: The SRH is inserted before any other IPv6 Routing Extension
A Current Mobile Network Example

- Well fragmented to RAN, EPC and SGi.
- Per-session tunnel creation and handling.
- Non-optimum data-path.

<-- Redundancies lessen TCO
<-- Can be scaled up but costly
<-- Hard to meet Apps reqs
3GPP Rel-15 Architecture (5G Phase.1)

Dramatically Simplified, Why?

- NEF: Network Exposure Function
- NRF: Network Repository Function
- PCF: Policy Control Function
- UDM: Unified Data Management
- AF: Application Function
- AUSF: Authentication Server Function
- AMF: Access & Mobility Management Function
- SMF: Session Management Function
- UE: User Equipment
- (R)AN: (Radio) Access Network
- UPF: User Plane Function
- DN: Data Network
Generic Expectations for 5G Networks

U-Plane must be simplified because to meet Complicated Optimizations

Source: NGMN white-paper
But Today’s U-plane Transports Are Well Complicated Already, Why?

Stacking Multiple Small ID Space Networks to Fulfill Requirements of Reliability, VPNs, etc.,
How to Simplify Such Complicating Stack?

Consolidates All Layers Role Into Single IPv6 Layer

- IPv6 DA (128)
- IPv6 SA (128)
- Segment-ID [0]* (128)
- Segment-ID [1]* (128)
- IPv6 SA (128)
- IPv6 DA (128)

*Exist in Segment Routing Extension Header (SRH)
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.
Multiple UPFs in GTP-U Case (1)
Multiple UPFs in GTP-U Case (2)
Multiple UPFs in A SRv6 Case (1)
Multiple UPFs in A SRv6 Case (2)
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
- SID: A::1234:5678
- SL = 1
- SID[0] = D::
- SID[1] = A::1234:5678

Payload SRH DA=A::1234:5678 SA=S::
SRv6 in A Nutshell

SRH (Segment Routing Header)

Segment List[0] (128 bits IPv6 address)

...
## SRv6 in A Nutshell (Cont’d)

<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>ENDDX6</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>

### SRv6 Function* Name

- **T**
  - Pure IPv6 transit
- **T.Insert**
  - Insert an SRv6 policy (SID list)
- **T.Encaps**
  - Encap SRv6 policy (SID list) by outer IPv6 hdr

* Non exhaustive list of SRv6 Network Programming
References

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

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