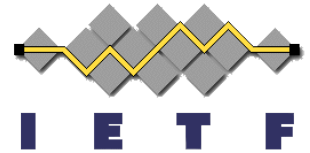


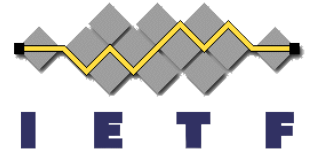
# **BESS BGP MUP SAFI**

## **IETF-118 Prague**

### **November, 2023**

Keyur Patel, Arccus  
Tetsuya Murakami, Arccus  
Satoru Matsushima, SoftBank  
Jeffrey Zhang, Juniper  
Swadesh Agrawal, Cisco





# Contributors

- Katsuhiro Horiba (SoftBank)
- Yuya Kawakami (SoftBank)
- Derek Yeung (Arrcus)
- Kalyani Rajaraman (Cisco)

# Motivation

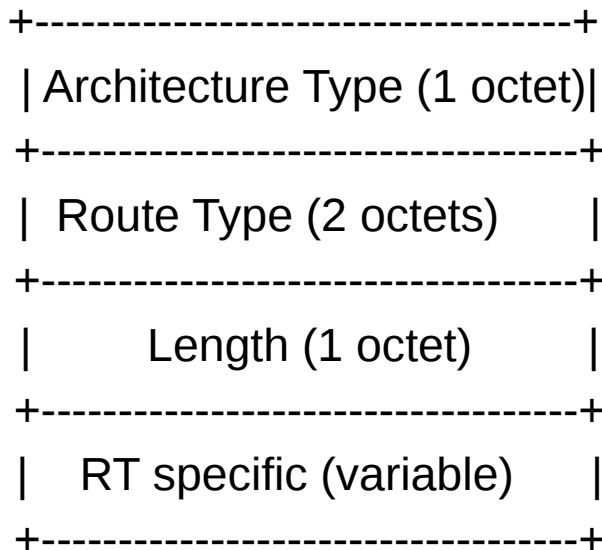
- draft-mhkk-dmm-srv6mup-architecture defines SRv6 Mobile User Plane (MUP) architecture for distributed mobility management
- This architecture integrates mobile user plane into SRv6 data plane
  - It does that by transforming the session information from the mobility management system to the necessary routing information
- As part of the architecture, it defines the following
  - Two new SRv6 Segment types of MUP Segment
    - A PE connects Direct and/or Interwork Segment
  - Two new Session Transformed routes
    - A MUP Controller advertises the session transformed routes

# Motivation (Cont'd)

- draft-mhkk-dmm-srv6mup-architecture depicts the MUP segments in 5G specific example case:
  - Interwork Segment for N3RAN
    - N3 Interface between gNodeBs and UPFs on RAN side
  - Direct Segment for N6DN
    - N6 Interface between UPFs and Data Networks (DN) on DN side
- This draft defines a new SAFI known as BGP Mobile User Plane (MUP) SAFI to support the MUP Extensions of the architecture document
  - Also defines a new Extended Community

# BGP MUP SAFI

- New SAFI to carry MUP routing information
- NLRI Format



- Architecture type defined: 3gpp-5g

# BGP MUP SAFI (Cont'd)

- 4 new Routes Types define for BGP-MUP SAFI 3gpp-5g architecture type:
  - Interwork Segment Discovery Route
  - Direct Segment Discovery Route
  - Type 1 Session Transformed (ST) Route
  - Type 2 Session Transformed (ST) Route
- Route Types can be shared by any new architecture types defined in Future

# BGP MUP SAFI (Cont'd)

- Two wellknown 5G specific segments of routing instances depicted:
  - N3RAN and N6DN routing instances in a PE
  - Interwork Segment Discovery Route associated with N3RAN routing instances MUP segment
    - Carries N3RAN prefix for gNodeB(es), Prefix SID attribute with the PE locator followed by GTP4/6.E function
  - Direct Segment Discovery Route associated with N6DN routing instances MUP segment
    - Carries Address of MUP Segment, Prefix SID attribute with the PE locator followed by End.DT4/6, DX2/4/6 function, etc., for example

# BGP MUP SAFI (Cont'd)

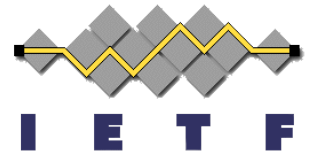
- Type 1 ST Route imported by N6DN routing instances and carries UE reachability information, Tunnel Endpoint address of GTP, TEID, QFI for DL & an optional Source address
  - TEID, QFI and Tunnel Endpoint address are carried in the architecture specific part of the NLRI
  - Tunnel Endpoint address should be resolved using Interwork Segment Discovery route – extract locator and prefix SID
  - Forwarding SID for GTP4/6.E is generated based on the procedures mentioned in draft-ietf-dmm-srv6-mobile-uplane.txt
  - Source address is optional. If it is not present then the implementation manually configures it



# BGP MUP SAFI (Cont'd)

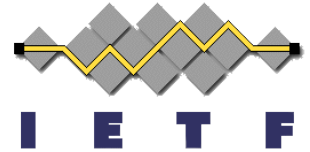
- Type 2 ST Route imported by N3RAN routing instances and carries an Endpoint address and associated GTP Tunnel information for UL
  - TEID is carried in the architecture specific part of the NLRI
  - BGP MUP Extended community carries the MUP segment value present in the Direct Segment Discovery Route – Used to resolve appropriate Direct Segment routing instance and forward it to the address of the MUP Segment
- Route target extended communities are carried to ensure import happens properly

# Draft Version 3

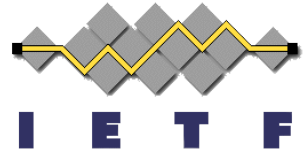


- Draft version 2 added support for End.M.GTP4/6.E for Direct Segment Route
- Draft version 3 added optional support for Source address to be carried on Type 1 ST Route

# Next Steps



- Draft is in a pretty good shape
  - Version 3 of the draft is submitted
  - Multiple Interoperable implementation exists
  - Major Customer trials are underway
  
- Consider WG Adoption



Thank you