GPE-VPN

• LISP-based architecture for SD-WAN
  – programmable **LISP control plane**
  – **VXLAN-GPE data plane** with optional:
    • **ESP** encryption
    • **NSH**-based support for Service Function Chaining

• Mapping System is dynamically programmed
  via **NorthBound API**
  – **Policy rendering** via dynamic **mapping**
    manipulation
Data Plane: GPE Encapsulated Frame

Outer Ethernet Header
Outer IP Header
Outer UDP Header

Reserved

Virtual Network Identifier (VNI) / Instance ID (IID) Reserved

Payload (ethernet, IPv4, IPv6, ESP, NSH, ...)

Next Protocol

GPE Header
Data Plane: GPE with AEAD (ESP-GCM)

Outer Ethernet Header
Outer IP Header
Outer UDP Header

R R V I P R O
Reserved
Virtual Network Identifier (VNI) / Instance ID (IID) Reserved
SPI (32)
Sequence Number (32)

Payload + Padding

ICV

NP = ESP

GPE

AAD

ICV Scope

ESP

NP = IP/Eth
Data Plane: GPE+NSH with ESP+GCM

Outer Ethernet Header

Outer IP Header

Outer UDP Header

Reserved

Virtual Network Identifier (VNI) / Instance ID (IID)

Reserved

SPI (32)

Sequence Number (32)

NSH Base Header

NSH Service Path Header

NSH Context Headers

Payload + Padding

ICV
Mapping Types

• GPE-VPN uses various mapping types to provide finer-grain policy control, and to support different policies
  – Per-destination mapping
    • EID -> RLOC
  – FlowMapping
    • <sEID, dEID, sPort, dPort, Protocol> -> RLOC
    • draft-rodrigueznatal-lisp-multi-tuple-eids
  – Generic Mapping
    • e.g. <NSH SPI, Index> -> RLOC
    • draft-ermagan-lisp-nsh
    • draft-rodrigueznatal-lisp-ms-smr
Dynamic Policy Rendering

- Dynamic mapping manipulation (via NB API) enables GPE-VPN generic policy rendering
  - Forwarding and In-bound load balancing
  - Overlay Re-encapsulation (via RTR)
    - Virtual topologies
    - Hierarchical VPNs
  - Group-based Access Control
  - Support for Service Function Chaining
Key Management Services

- SA provisioning is a **trade-off** between
  - **time** needed to set up the SA on demand
  - overall **security** afforded

- SA provisioning can be done with different mechanisms
  - Use **IKEv2** to negotiate pairwise SAs
  - Use Group Domain of Interpretation (GDOI) **for group key management**
  - Leverage **LISP map-request/reply** to accelerate on demand provisioning of SA
    - e.g. ietf-lisp-crypto
Q&A

Thanks!