LISP Working Group
Internet-Draft
Intended status: Informational
Expires: July 30, 2022

S. Barkai
B. Fernandez-Ruiz
R. Tamir
Nexar Inc.
A. Rodriguez-Natal
F. Maino
Cisco Systems
A. Cabellos-Aparicio
J. Paillisse Vilanova
Technical University of Catalonia
D. Farinacci
lispers.net
June 11, 2022

Network-Hexagons: Mobility Geolocation Edge Network Using LISP
draft-ietf-lisp-nexagon-26

Abstract

Geolocation-Services aggregate raw data uploads from vehicles using edge compute locations and process these uploads generating verified, localized, geospatial detection-channels - used by mobility clients to support crowed-sourced dynamic mapping and driving applications. Geolocation Services are broken to shards (areas) and dynamically mapped to compute locations based on road activity. This dynamics combined with clients IP Anchors dynamics creates coherency, context-switching, geo-privacy, and service continuity key issues.
These issues are resolved by dataflow virtualization, communication indirection, between mobility clients and Geolocation Services.
LISP overlay network-virtualization, offers a fully distributed dataflow virtualization solution at the edge networking level.
LISP as a Geolocation mobility-network is described in this document.

Status of This Memo

Table of Contents

1. Introduction

Geolocation-Services aggregate raw data uploads from vehicles using edge compute locations and process these uploads generating verified, localized, geospatial detection-channels - used by mobility clients to support crowed-sourced dynamic mapping and driving applications. Geolocation Services are broken to shards (areas) and dynamically mapped to compute locations based on road activity. This dynamics combined with clients IP Anchors dynamics creates coherency, context-switching, geo-privacy, and service continuity key issues.
These issues are resolved by dataflow virtualization, communication indirection, between mobility clients and Geolocation Services.
LISP overlay network-virtualization, offers a fully distributed dataflow virtualization solution for the edge at the networking level.

Geolocation services shard transformation of inaccurate, duplicated data captured and uploaded by multiple vehicles points of view, into consolidated information channels, can be schematically described by the following key components:

- Addressable queues for uploads from mobility clients in vehicles
- Addressable detection channels subscribed to by mobility clients
- State & functions transforming upload data to detection channels

This service allocation dynamics combined with dynamics of mobility clients in selecting mobile-provider and switching IP Anchors, creates key-issues:

- Coherency of Geolocation shards IP addresses cached in clients
- Context-switching between Geolocation Service shards while driving
- Geo-privacy of clients while interacting with Geolocation shards
- Subscription continuity when switching providers while driving

These issues are resolved by dataflow virtualization, communication indirection, between mobility clients and Geolocation Services. Such communication when based on logical addresses of entities and addressable geospatial topics solves these issues.

Virtual and logical addressing allows for dynamic and portable shard allocation, algorithmic context-switching between shards while driving and channel subscription continuity while preserving the geo-privacy of mobility clients.

LISP network-virtualization can offer such a solution through specific use of Endpoint Identifiers (EID):

- EID addressing of upload queues per geospatial grid identifiers
- EID addressing of detection channels per geospatial topics
- EID addressing of mobility clients renewed periodically

These EIDs are tracked by the LISP mobility-network offering dynamic portability of queues and channels. Functions are assumed to be available in any compute location and geospatial current-state used for consolidation quickly regenerates.

EIDs of mobility clients are mapped across the LISP mobility-network enabling service continuity when mobile carriers are switched by the vehicle systems. These EIDs are ephemeral and make it difficult for just any mobility service provider to track mobility clients movement during the day. The use of LISP as the mobility-network for dataflow virtualization between mobility client to Geolocation Services is described bellow.

Note 1: The breakdown of Geolocations Services to area-shards is done based on geospatial grid lines known to both mobility clients and Geolocation Services. We use H3 [H3] hierarchical hexagonal grid because of its clear shard adjacency properties. Each grid-tile in each resolution has a unique 64bit identifier (HID). In addition to shards and HID to EID mapping, the same grid is used to localize road detections. We refer to h3.rB as the lower resolution shard tile, and h3.rS as the detection higher resolution tile. The mappings: GPS to h3.rS to H3.rB to EID are therefore algorithmic. Sizeof (h3.rB) / Sizeof (h3.rS) x density-factor / MTU (1500 Bytes) = number of message needed for a shard to convey a state snapshot.

Note 2: LISP solution for dataflow virtualization is an application network. In order for clients and services to use it there needs to be a formal provisioning step. For the clients this step will require Authentication Authorization and Accounting (AAA) procedure by which clients are assigned and renew EIDs and XTRs to be used to communicate with services. This process may be done in various vendor specific methods, in this document we use a Diameter[RFC6733] AAA service, this is meant as a life-cycle example only.

Note 3: In order to make the notion of geospatial detection concrete, we add to the 64bit HID of "where" is a detection, 64bit of "what" is the detection. These 64bits are detailed in a bit-mask of possible detections based on a taxonomy defined by Berkeley Deep Drive (BDD) consortium (https://bdd-data.berkeley.edu). It is meant as a baseline that can be extended in additional specifications, yet proven minimal sufficient in current implementations.

```
                                           _____
                                          /     \
Off-Peak Allocation        Addressable >> States >> Addressable
 _  _    _  _             Upload Queues   \_____/     Channels
/ \/ \  / \/ \  ----             /\      f1()..fn()       \/
\_/\_/  \_/\_/  ----
/ \/ \  / \/ \  ----         Peak Geolocation Allocation
\_/\_/  \_/\_/  ----       _  _    _  _    _  _    _  _
/ \/ \  / \/ \  ----      / \/ \  / \/ \  / \/ \  / \/ \  ----
\_/\_/  \_/\_/  ----      \_/\_/  \_/\_/  \_/\_/  \_/\_/  ----
/ \/ \  / \/ \  ----      / \/ \  / \/ \  / \/ \  / \/ \  ----
\_/\_/  \_/\_/  ----      \_/\_/  \_/\_/  \_/\_/  \_/\_/  ----
 ^  ^    ^  ^   ^  ^       ^ ^     ^  ^    ^  ^    ^  ^   ^  ^
 Site   Site  Standby     Site    Site    Site    Site  Standby
```

Figure 1: Geolocation Service Schematics and Dynamic Allocation

2. Definition of Terms

Based on [I-D.ietf-lisp-rfc6830bis][I-D.ietf-lisp-rfc6833bis]

H3ServiceEID: Is an EID addressable Geolocation Service shard. It is a designated destination for geospatial detections, and an (S,G) source of multicast of themed detection channels. It has a light-weight LISP protocol stack to tunnel packets via ServerXTR. The EID is IPv6 and contains HID in the lower bits.

ServerXTR: Is a data-plane only LISP protocol stack implementation, it is co-located with H3ServiceEID process. ServerXTR encapsulates and decapsulates packets to and from EdgeRTRs.

MobilityClient: Is an application that may be a part of a vehicle system, part of a navigation application, gov-muni application etc. It has light-weight LISP data-plane stack to packets via ClientXTR.

MobilityClientEID: Is the IPv6 EID used by the Mobility Clients. The destination of such packets are H3ServiceEIDs. The EID format is assigned as part of the MobilityClient mobility-network AAA.

ClientXTR: Is a data-plane only LISP protocol stack implementation co-located with the Mobility Client application. It encapsulates and decapsulates packets to and from EdgeRTRs.

EdgeRTR: EdgeRTR network connect MobilityClients H3ServiceEIDs. They also manage MobilityClients multicast registrations [RFC8378]. EdgeRTRs aggregate MobilityClients/H3Services using tunnels to facilitate hosting-providers and mobile-providers for accessing the mobility-network. EdgeRTRs decapsulate packets from ClientXTRs, ServerXTRs and re-encapsulates packets to clients and servers via tunnels. EdgeRTRs glean H3ServiceEIDs and MobilityClient EIDs when they decapsulates packets. EdgeRTRs store H3ServiceEIDs and route-locations (RLOC) of where the H3ServiceEID is currently using map-cache. Mappings are registered to the LISP mapping system [I-D.ietf-lisp-rfc6833bis].These mappings may be provisioned when H3Services are assigned EdgeRTRs. EdgeRTRs do not register MobilityClients' EIDs. Enterprises may provide their own EdgeRTRs to protect geo-privacy.

```
                          ___                                  ___
   H3ServiceEIDs   ___  /     \           H3ServiceEIDs ___  /     \
            ___  /     | h3.rB |            ___  /     | h3.rB |
          /     | h3.rB \ ___ /            /     | h3.rB \ ___ /
         | h3.rB \ ___ /  sXTR             | h3.rB \ ___ /  sXTR
          \ ___ /  sXTR    |                   \ ___ /  sXTR     |
            sXTR    |      |                     sXTR     |      |
             |      |      |                      |       |      |
             |      |      |                      |       |      |
             + -  - + - - EdgeRTR          EdgeRTR - + - + - -  +
                             ||  (   (   ((  ||
                          (                        )
                        (      Network Hexagons      )
                      (            H3-LISP              )
                        (      mobility-network       )
                          ((                        )
                            ||  ((   (())   ()  ||
                            ||                  ||
                = = = = = = =                     = = = = = = =
               ||                                             ||
            EdgeRTR                                         EdgeRTR
           ..    ..                                      ..      ..
          ..       ..                                  ..          ..
  ((((|))))    ((((|))))                         ((((|))))    ((((|))))
     /|\    RAN   /|\                               /|\    RAN   /|\
      ..                                                          ..
      ..                                                          ..
   Uploads Upstream                                   Uploads Upstream
 Channels Downstream       h3.rS Tiled Road          Channels Downstream
      ..                   ___    ___    ___                      ..
      ..   << movment << /     \/     \/     \ << cXTR::MobilityClientB
      .. - - - - - - - - -  h3.rS  h3.rS  h3.rS - - - - - - - - - - - -
      MobilityClientA::cXTR >>  \ ___ /\ ___ / >> movement >> .......
```

       Figure 2: Client Mapping h3.rS to h3.rB to LISP routed EID

Figure 2 above describes the following entities interacting:

- MobilityClientA detections used by MobilityClientB, and, vice versa
- Clients: share information via Geolocation Services routed by LISP
- ClientXTR (cXTR): tunnels packets over access networks to EdgeRTR
- ServerXTR (sXTR): tunnels packets over cloud networks to EdgeRTR
- H3-LISP Mobility Network: overlay which spans cXTRs to sXTRs
- Uploads: routed to appropriate Geolocation Service by LISP
- EdgeRTRs: perform multicast replication to EdgeRTRs then to cXTRs

Multicast channels carry detections processed by Geolocation services

3. Deployment Assumptions

The specification described in this document makes the following deployment assumptions:

(1) Unique 64bit HID is associated with each H3 geospatial tile
(2) MobilityClients and H3ServiceEIDs share grid based indexing
(3) 64bit BDD state value is associated with each H3-indexed tile
(4) Tile state is compiled 16 fields of 4-bits, or max 16 enums

```
0       1       2       3       4       5       6       7
+-------+-------+-------+-------+-------+-------+-------+-------+
|-0-|-1-|-2-|-3-|-4-|-5-|-6-|-7-|-8-|-9-|-A-|-B-|-C-|-D-|-E-|-F-|
|0123012301230123012301230123 Index  0123012301230123012301230123
+---------------------------------------------------------------+
```

Figure 3: Nibble based representation of tile state

State is 16 fields x 16 enumerations. Nibbles are named using hexadecimal index according to the position where the most significant nibble has index 0. Values are defined in section 8.

Subscription of MobilityClients to mobility-network is renewed while driving. It is not intended as the basic connectivity. MobilityClients use DNS/AAA described bellow to obtain EIDs/EdgeRTRs and use (LISP) data-plane tunnels to communicate using their EIDs with the dynamically assigned EdgeRTRs. MobilityClients only use the LISP data plane.

In order to get access to the mobility-network, MobilityClients first authenticate with the Mobility AAA. Diameter [RFC6733] based AAA can be used as a solution for the many types of mobility clients: vehicle systems, driving applications, city and consumer applications.

ClientXTRs may perform provision steps to use the mobility-network:
1) obtain the address of the mobility-network AAA server using DNS
2) obtain MobilityClientEIDs and EdgeRTRs from AAA Diameter server
3) renew authorization from AAA while using the mobility-network
They may also be provisioned using vendor tools.

```
     MobilityClient DomainNameServer  Diameter-AAA  MobilityEdgeRTR
     |                    |                   |                   |
     | nslookup <www.xxx> |                   |                   |
     |------------------->|                   |                   |
     |<-------------------|                   |                   |
     |  Mobility AAA IP   |                   |                   |
     |  [RFC6733]         |                   |                   |
     |  AAR(AVP:IMSI/User/Password/Toyota)    |                   |
     |--------------------------------------->|  [RFC6733]        |
     |                    |                   | ACR(AVP ClientEID)|
     |                    |                   |------------------>|
     |                    |                   |<------------------|
     |                    |                   | ACA(AVP ClientEID)|
     |    AAA (ClientEID,EdgeRTR::RLOC)       |                   |
     |<---------------------------------------|                   |
     |                    |                   |                   |
     .                                                            .
     . Upload to IPv6 H3ServiceEID, Subscribe MLDv2 H3ServiceEID  .
     .                                                            .
     |                                                            |
     |----------------------------------------------------------->|
     .                                                            .
     .                                                            .
     |<-----------------------------------------------------------|
     |                                                            |
     .                                                            .
     .   Signal freeing multicast Updates from H3ServiceEID       .
     .                                                            .
     |                    |                   |                   |
     |               AAR(temporary)           |                   |
     |--------------------------------------->|   ACR (temporary) |
     |                    |                   |------------------>|
     |                    |                   |<------------------|
     |                    |                   |   ACA (temporary) |
     |<---------------------------------------|                   |
     |               AAA (temporary)          |                   |
```

Figure 4: DNS and AAA Exchange for lisp mobility-network usage

Using such an AAA procedure we can ensure that:

- MobilityClientEIDs serve as credentials with the EdgeRTRs
- EdgeRTRs are provisioned to approved-list MobilityClientEIDs
- EdgeRTRs are not tightly coupled to h3.rB areas (privacy/balance)
- MobilityClients do not need to update EdgeRTRs while driving

4. Mobility Clients Network Services

The mobility-network functions as a standard LISP overlay.
The overlay delivers unicast and multicast packets across:

- multiple access-networks providers and technologies
- multiple cloud, edge and metro-area network providers

We use data-plane XTRs in the stack of each mobility client/server. ClientXTRs and ServerXTRs are associated to EdgeRTRs. This structure allows for MobilityClients to "show up" at any location behind any network provider in a given mobility-network admin/NAT domain, and for any H3ServiceEID to be instantiated, moved, or failed-over to any cluster in any cloud-provider. LISP overlay enables these roaming mobility-network elements to communicate uninterrupted. This quality is insured by the LISP RFCs. The determination of identities for MobilityClients to always refer to the correct H3ServiceEID is insured by H3 geospatial HIDs.

In this specification we assume semi-random association between ClientXTRs and EdgeRTRs applied by the AAA procedure. We assume that in a given metro edge a pool of EdgeRTRs can distribute the Mobility Clients load randomly between them and that EdgeRTRs are topologically equivalent. Each RTR uses LISP to tunnel traffic to and from other EdgeRTRs forMobilityClient and H3Service exchanges.

```
     MobilityClient == ClientXTR       ClientXTR == MobilityClient
                              ||       ||
                  LISP mobility-network EdgeRTRs
                              ||       ||
       H3ServiceEID == ServerXTR       ServerXTR == H3ServiceEID
```

Figure 5: LISP network connecting MobilityClients and H3ServiceEIDs

5. Mobility Unicast and Multicast

The day in a life of unicast detection upload:

1. A client detects condition of interest using AI camera
2. The client uses its GPS to establish its h3.rS location
3. It then estimates the h3.rS location of the detection
4. The h3.rS used to calculate the h3.rB and H3ServerEID
5. Sends the location-detection (enum) via its ClientXTR

```
  Outer Header src/dest: ClientXTR RLOC, EdgeRTR RLOC
  Inner Header src/dest: ClientEID, H3ServiceEID
```

6. EdgeRTR gleans and caches ClientEID and ClientXTR RLOC
7. EdgeRTR resolved RLOC of remote EdgeRTR, re-tunnels:

```
  Outer Header src/dest: EdgeRTR RLOC, remote EdgeRTR RLOC
  Inner Header src/dest: ClientEID, H3ServiceEID
```

8. Remote EdgeRTR lookups H3ServerEID ServerXTR RLOC, re-tunnels:

```
  Outer Header src/dest: EdgeRTR RLOC, ServerXTR RLOC
  Inner Header src/dest: ClientEID, H3ServiceEID
```

9. ServerXTR delivers ClientEID message to H3ServiceEID

The detection message headers consist of the following fields:

```
Outer headers size = 40 (IPv6) + 8 (UDP) + 8 (LISP) = 56
Inner headers size = 40 (IPv6) + 8 (UDP) + 4 (Nexagon Header) = 52
1500 (MTU) - 56 - 52 = 1392 bytes of effective payload size
```

Nexagon Header allows for key-value (kv) tuples or value-key,key ..(vkkk) using the same formats of key and value oulined bellow

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+\
|  Type         |gzip |        Reserved         | Pair Count = X|Nexagon
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+/
```

Figure 6: Nexagon Header format

Type 0:reserved
Type 1:key-value, key-value.. 1392 / (8 + 8) = 87 pairs
Type 2:value, key,key,key.. (1392 - 8) / 8 = 173 h3.rS IDs
Type 3-255: unassigned
Nexagon Header GZIP field: 0x000 no compression, or GZIP version.
The compression refers to entire kv or vkkk payload.
Nexagon Header Reserved bits
Nexagon Header key and value count (in any format)

* Reserved fields are specified as being set to 0 on transmission and ignored when received. Packets with unsupported gzip version ignored.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|Version| Traffic Class |           Flow Label                  |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|         Payload Length        |  Next Header  |   Hop Limit   |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|                                                               |  |
+                                                               +  |
|                                                               |  |
+                     Source MobilityClientEID                  +  |
|                                                               | IPv6
+                                                               +  |
|                                                               |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|                                                               |  |
+                                                               +  |
|                                                               |  |
+                        Dest H3ServiceEID                      +  |
|                                                               |  |
+                                                               +  |
|                                                               | /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Source Port = xxxx      |       Dest Port = xxxx        | \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ UDP
|           UDP Length          |        UDP Checksum           | /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|  Type         |gzip |        Reserved         | Pair Count = X|Nexagon
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ /
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit State                            +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit State                            +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 7: Uploaded detections packet format

Each H3Service is also an IP Multicast Source used to update subscribers on the aggregate state of the h3.rS tiles in the h3.rB area. We use [RFC8378] signal-free multicast to implement channels in the overlay. The mobility-network has many channels, with thousands subscribers per channel. MobilityClients driving through/subscribing to an h3.rB area can explicitly issue an [RFC4604] MLDv2 in order to subscribe, or, may be subscribed implicitly by the EdgeRTR. MLDv2 signaling messages are encapsulated between the ClientXTR and EdgeRTR.

The day in a life of multicast update:

1. H3ServiceEID determines change or timing requiring update
2. H3ServiceEID sends (S,G) update message via its ServerXTR

   Outer Header src/dest: ServerXTR RLOC, EdgeRTR RLOC
   Inner Header (S,G): H3ServerEID, EID chosen for theme

3. EdgeRTR resolves subscribed remote EdgeRTRs, replicates

   Outer Header src/dest: EdgeRTR RLOC, remote EdgeRTR RLOC
   Inner Header (S,G): H3ServerEID, EID chosen for theme

4. EdgeRTRs lookp subscribed ClientEIDs ClientXTRs RLOCs, replicate

   Outer Header src/dest: EdgeRTR RLOC, ClientXTR RLOC
   Inner Header (S,G): H3ServerEID, EID chosen for theme

5. ClientXTR delivers update message to clientEID

Note: there may be more than one ClientEID in the same process using the same ClientXTR. For example multiple layers of map or heads-up display, Such vendor specific multiplexing implementation is unspecified here.

Multicast update packets are of the following structure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|Version| Traffic Class |           Flow Label                  |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|         Payload Length        |  Next Header  |   Hop Limit   |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|                                                               |  |
+                                                               +  |
|                                                               |  |
+                      Source H3ServiceEID                   +  |
|                                                               | IPv6
+                                                               +  |
|                                                               |  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+  |
|                                                               |  |
+                                                               +  |
|                                                               |  |
+                          Group Address                        +  |
|                                                               |  |
+                                                               +  |
|                                                               | /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Source Port = xxxx      |       Dest Port = xxxx        | \
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ UDP
|           UDP Length          |        UDP Checksum           | /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|                                                               |Nexagon
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ /
~                            Nexagons Payload                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 8: multicast update packet header

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|  Type =   1   |gzip |        Reserved         | Pair Count = X|Nexagon
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ /
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit State                            +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit State                            +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 9: multicast update payload, key-value, key-value..

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ \
|  Type =   2   |gzip |     Reserved            |H3R15 Count = X|Nexagon
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ /
|                                                               |
+                        64bit State                           +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                        64bit h3.rS ID                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 10: multicast update payload, value, key, key.. for larger areas

6. Security Considerations

The LISP mobility-network is inherently secure and private. All information is conveyed using provisioned Geolocation Services. MobilityClients receive information only by geospatial channels originating from a provisioned Geolocation Services, and not from any other source on this application specific LISP network. MobilityClients have no indication as to the origin of the raw data.

In order to be able to use nexagon LISP mobility-network for a given period, the mobility clients go through a DNS/AAA stage by which they obtain their clientEID identifiers-credentials and the RLOCs of EdgeRTRs they may use as gateways to the network. This MobilityClient <> EdgeRTR interface is the most sensitive in this network to privacy and security considerations.

The traffic on the MobilityClient<>EdgeRTR interface is tunneled, and its UDP content may be encrypted; still, the EdgeRTR will know based on the LISP headers alone the MobilityClient RLOC and h3.rB geospatial area to which a given client uploads form / subscribes to.

For this reason we envision the ability of enterprise to "bring their own" EdgeRTRs. For example a car OEM or mobile carrier offering EdgeRTRs on behalf of vehicles for use with Geolocation Services. BYO-RTR masks individual clients' RLOC to h3.rB association. BYO-RTR is pre-provisioned to be able to use the mapping system and be on white-lists of EdgeRTRs aggregating H3ServiceEIDs. If the EdgeRTR functionality is delivered by a carrier then the only entity which can correlate underlay IP, User, and Geo-location is the regulated carrier, which can do so anyway. LISP interoperability enables this.

Beyond this hop, the mapping system does not hold MobilityClientEIDs, and remote EdgeRTRs are only aware of MobilityClient ephemeral EIDs, not actual RLOC or any other mobile-device identifiers. EdgeRTRs register in the mapping (S,G) h3.rB multicast groups. Which clients use which EdgeRTR is not in the mapping system, only the AAA server is aware of that. The H3ServiceEIDs themselves decrypt and parse actual h3.rS detections; they also consider during this MobilityClientEID credentials to avoid "fake-news" or bad poorley localized detections, but again these are only temporary EIDs allocated to clients in order to be able to use the mobility network and not for their actual IP.

H3Services are associated their EdgeRTRs, provisioned n the EdgeRTRs, and optionally also in the mapping system.

In summary of main risk mitigations for the lisp-nexagon interface:

(1) tapping: all communications are through tunnels therefore may be encrypted using IP-Sec or other supported point to point underlay standards.

(2) spoofing: it is very hard to guess a MobilityClientEID valid for a short period of time. Clients and H3Services EIDs are provisioned in EdgeRTRs, Clients using the AAA procedure, H3Services via dev-ops.

(3) credibility: the interface crowd-sources geo-state and does not assume to trust single detections. Credit history track to MobilityClientEIDs by as part of normal H3Services fact checking, aggregate scores affect AAA credentials.

(4) privacy: Only EdgeRTRs are aware of both clients' RLOC and geo-location, only AAA is aware of client IDs credentials and credit but not geo-location. Aggregate credit score span all H3Services administratively without source.

7. Acknowledgments

We would like to kindly thank Joel Halperin for helping structure the AAA section and Geo-Privacy provisions, Luigi Lannone for promoting such LISP Compute First Networking (CFN) use-cases, helping structure the IANA section, and shepherding this draft to completion. We would like to thank George Ericson for help clarifying Geolocation Services terminology through joint work on the AECC specifications and papers, and Lei Zhong for helping term the dataflow virtualization solution.

8. IANA Considerations

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the LISP specification, in accordance with BCP 26 [RFC8126].

IANA is asked to create a registry named NEXAGON Parameters.

Such registry should be populated with the following sub registries.

State Enumeration Field 0x0: Traffic Direction:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Lane North	[This Document]
0x2	Lane North + 30	[This Document]
0x3	Lane North + 60	[This Document]
0x4	Lane North + 90	[This Document]
0x5	Lane North + 120	[This Document]
0x6	Lane North + 150	[This Document]
0x7	Lane North + 180	[This Document]
0x8	Lane North + 210	[This Document]
0x9	Lane North + 240	[This Document]
0xA	Lane North + 270	[This Document]
0xB	Lane North + 300	[This Document]
0xC	Lane North + 330	[This Document]
0xD	Junction	[This Document]
0xE	Shoulder	[This Document]
0xF	Sidewalk	[This Document]

State Enumeration Field 0x1: Persistent Condition:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Pothole Light	[This Document]
0x2	Pothole Deep	[This Document]
0x3	Speed-bump Low	[This Document]
0x4	Speed-bump High	[This Document]
0x5	Icy	[This Document]
0x6	Flooded	[This Document]
0x7	Snow-cover	[This Document]
0x8	Deep Snow	[This Document]
0x9	Cone	[This Document]
0xA	Gravel	[This Document]
0xB	Choppy	[This Document]
0xC	Blind-Curve	[This Document]
0xD	Steep	[This Document]
0xE	Low-bridge	[This Document]
0xF	Other	[This Document]

State Enumeration Field 0x2: Transient Condition:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Jaywalker	[This Document]
0x2	Bike or Scooter	[This Document]
0x3	Stopped Vehicle	[This Document]
0x4	Moving on Shoulder	[This Document]
0x5	First Responder	[This Document]
0x6	Sudden Slowdown	[This Document]
0x7	Oversize Vehicle	[This Document]
0x8	Light/Sign Breach	[This Document]
0x9	Collision Light	[This Document]
0xA	Collision Severe	[This Document]
0xB	Collision Debris	[This Document]
0xC	Collision Course	[This Document]
0xD	Vehicle Hard Brake	[This Document]
0xE	Vehicle Sharp Turn	[This Document]
0xF	Freed-up Parking	[This Document]

State Enumeration Field 0x3: Traffic-light Counter:

Value	Description	Reference
0x0	Null	[This Document]
0x1	1 Second to Green	[This Document]
0x2	2 Second to Green	[This Document]
0x3	3 Second to Green	[This Document]
0x4	4 Second to Green	[This Document]
0x5	5 Second to Green	[This Document]
0x6	6 Second to Green	[This Document]
0x7	7 Second to Green	[This Document]
0x8	8 Second to Green	[This Document]
0x9	9 Second to Green	[This Document]
0xA	10 Second to Green	[This Document]
0xB	20 Second to Green	[This Document]
0xC	30 Second to Green	[This Document]
0xD	60 Second to Green	[This Document]
0xE	Green Now	[This Document]
0xF	Red Now	[This Document]

State Enumeration Field 0x4: Impacted Tile:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Epicenter	[This Document]
0x2	2 Tiles Away	[This Document]
0x3	3 Tiles Away	[This Document]
0x4	4 Tiles Away	[This Document]
0x5	5 Tiles Away	[This Document]
0x6	6 Tiles Away	[This Document]
0x7	7 Tiles Away	[This Document]
0x8	8 Tiles Away	[This Document]
0x9	9 Tiles Away	[This Document]
0xA	10 Tiles Away	[This Document]
0xB	20 Tiles Away	[This Document]
0xC	30 Tiles Away	[This Document]
0xD	60 Tiles Away	[This Document]
0xE	<100 Tiles Away	[This Document]
0xF	<200 Tiles Away	[This Document]

State Enumeration Field 0x5: Expected Duration:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Next 1 Second	[This Document]
0x2	Next 5 Seconds	[This Document]
0x3	Next 10 Seconds	[This Document]
0x4	Next 20 Seconds	[This Document]
0x5	Next 40 Seconds	[This Document]
0x6	Next 60 Seconds	[This Document]
0x7	Next 2 Minutes	[This Document]
0x8	Next 3 Minutes	[This Document]
0x9	Next 4 Minutes	[This Document]
0xA	Next 5 Minutes	[This Document]
0xB	Next 10 Minutes	[This Document]
0xC	Next 15 Minutes	[This Document]
0xD	Next 30 Minutes	[This Document]
0xE	Next 60 Minutes	[This Document]
0xF	Next 24 Hours	[This Document]

State Enumeration Field 0x6: Lane Right Sign:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Yield	[This Document]
0x2	Speed Limit	[This Document]
0x3	Straight Only	[This Document]
0x4	No Straight	[This Document]
0x5	Right Only	[This Document]
0x6	No Right	[This Document]
0x7	Left Only	[This Document]
0x8	No Left	[This Document]
0x9	Right Straight	[This Document]
0xA	Left Straight	[This Document]
0xB	No U Turn	[This Document]
0xC	No Left or U	[This Document]
0xD	Bike Lane	[This Document]
0xE	HOV Lane	[This Document]
0xF	Stop	[This Document]

State Enumeration Field 0x7: Movement Sign:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Keep Right	[This Document]
0x2	Keep Left	[This Document]
0x3	Stay in Lane	[This Document]
0x4	Do Not Enter	[This Document]
0x5	No Trucks	[This Document]
0x6	No Bikes	[This Document]
0x7	No Peds	[This Document]
0x8	One Way	[This Document]
0x9	Parking	[This Document]
0xA	No Parking	[This Document]
0xB	No Standing	[This Document]
0xC	No Passing	[This Document]
0xD	Loading Zone	[This Document]
0xE	Rail Crossing	[This Document]
0xF	School Zone	[This Document]

State Enumeration Field 0x8: Curves & Intersections:

Value	Description	Reference
0x0	Null	[This Document]
0x1	Turns Left	[This Document]
0x2	Turns Right	[This Document]
0x3	Curves Left	[This Document]
0x4	Curves Right	[This Document]
0x5	Reverses Left	[This Document]
0x6	Reverses Right	[This Document]
0x7	Winding Road	[This Document]
0x8	Hair Pin	[This Document]
0x9	Pretzel Turn	[This Document]
0xA	Cross Roads	[This Document]
0xB	Cross T	[This Document]
0xC	Cross Y	[This Document]
0xD	Circle	[This Document]
0xE	Lane Ends	[This Document]
0xF	Road Narrows	[This Document]

State Enumeration Field 0x9: Tile Traffic Speed:

Value	Description	Reference
0x0	Null	[This Document]
0x1	< 1 m/sec	[This Document]
0x2	< 2 m/sec	[This Document]
0x3	< 3 m/sec	[This Document]
0x4	< 4 m/sec	[This Document]
0x5	< 5 m/sec	[This Document]
0x6	< 6 m/sec	[This Document]
0x7	< 7 m/sec	[This Document]
0x8	< 8 m/sec	[This Document]
0x9	< 9 m/sec	[This Document]
0xA	< 10 m/sec	[This Document]
0xB	< 20 m/sec	[This Document]
0xC	< 30 m/sec	[This Document]
0xD	< 40 m/sec	[This Document]
0xE	< 50 m/sec	[This Document]
0xF	> 50 m/sec	[This Document]

State Enumeration Field 0xA: Pedestrian Curb Density:

Value	Description	Reference
0x0	Null	[This Document]
0x1	100%	[This Document]
0x2	95%	[This Document]
0x3	90%	[This Document]
0x4	85%	[This Document]
0x5	80%	[This Document]
0x6	70%	[This Document]
0x7	60%	[This Document]
0x8	50%	[This Document]
0x9	40%	[This Document]
0xA	30%	[This Document]
0xB	20%	[This Document]
0xC	15%	[This Document]
0xD	10%	[This Document]
0xE	5%	[This Document]
0xF	No Peds	[This Document]

State enumeration fields 0xB, 0xC, 0xD, 0xE, 0xF, are unassigned. IANA can assign them on a "First Come First Served" basis according to [RFC8126].

9. Normative References

[I-D.ietf-lisp-rfc6830bis]
Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos-Aparicio, "The Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-rfc6830bis-38 (work in progress), May 2020.

[I-D.ietf-lisp-rfc6833bis]
Farinacci, D., Maino, F., Fuller, V., and A. Cabellos, "Locator/ID Separation Protocol (LISP) Control-Plane", draft-ietf-lisp-rfc6833bis-31 (work in progress), May 2020.

[RFC4604] Holbrook, H., Cain, B., and B. Haberman, "Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast", RFC 4604, DOI 10.17487/RFC4604, August 2006, <https://www.rfc-editor.org/info/rfc4604>.

[RFC6733] Fajardo, V., Ed., Arkko, J., Loughney, J., and G. Zorn, Ed., "Diameter Base Protocol", RFC 6733, DOI 10.17487/RFC6733, October 2012, <http://www.rfc-editor.org/info/rfc6733>.

[RFC8126] Cotton, M., Leiba, B., Narten, T., "Guidelines for Writing an IANA Considerations Section in RFCs", RFC8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8378] Farinacci, D., Moreno, V., "Signal-Free Locator/ID Separation Protocol (LISP) Multicast", RFC8378, DOI 10.17487/RFC8378, May 2018, <https://www.rfc-editor.org/info/rfc8378>.

[H3] Uber Technologies Inc. [n.d.]. H3: Ubers Hexagonal Hierarchical Spatial Index, May 2021, <https://eng.uber.com/h3>.

Authors' Addresses

Sharon Barkai
Nexar
CA
USA

Email: sbarkai@gmail.com

Bruno Fernandez-Ruiz
Nexar
London
UK

Email: b@getnexar.com

Rotem Tamir
Nexar
Israel

rotemtamir@getnexar.com

Alberto Rodriguez-Natal
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

Email: natal@cisco.com

Fabio Maino
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

Email: fmaino@cisco.com

Albert Cabellos-Aparicio
Technical University of Catalonia
Barcelona
Spain

Email: acabello@ac.upc.edu

Jordi Paillisse-Vilanova
Technical University of Catalonia
Barcelona
Spain

Email: jordip@ac.upc.edu

Dino Farinacci
lispers.net
San Jose, CA
USA

Email: farinacci@gmail.com