Generic Multi-Access (GMA) Convergence Encapsulation Protocols
draft-zhu-intarea-gma-04

Status of this Memo

Copyright Notice

Abstract

   Today, a device can be simultaneously connected to multiple
   networks, e.g. Wi-Fi, LTE, 5G, and DSL. It is desirable to combine
   them seamlessly to improve quality of experience. Such
   optimization requires additional control information, e.g.
   Sequence Number, in each (IP) data packet. This document presents
   a new light-weight and flexible encapsulation protocol for this
   need. The solution has been developed by the authors based on
   their experiences in multiple standards bodies including the IETF
   and 3GPP, but is not an Internet Standard and does not represent
   the consensus opinion of the IETF. This document will enable other
   developers to build interoperable implementations.

Table of Contents

1. Introduction

   Figure 1 shows the user-plane Generic Multi-Access (GMA) protocol
   stack, which has been used in today's multi-access solutions
   [ATSSS] [MAMS] [LWIPEP] [GRE].

```
        +-------------------------------------------------------+
        |                      IP PDU                           |
        |-------------------------------------------------------|
        |               Convergence Sublayer                    |
        |-------------------------------------------------------|
        | Adaptation     | Adaptation     | Adaptation          |
        | Sublayer       | Sublayer       | Sublayer            |
        | (optional)     | (optional)     | (optional)          |
        |-------------------------------------------------------|
        | Access #1 IP   | Access #2 IP   | Access #3 IP        |
        +-------------------------------------------------------+
```

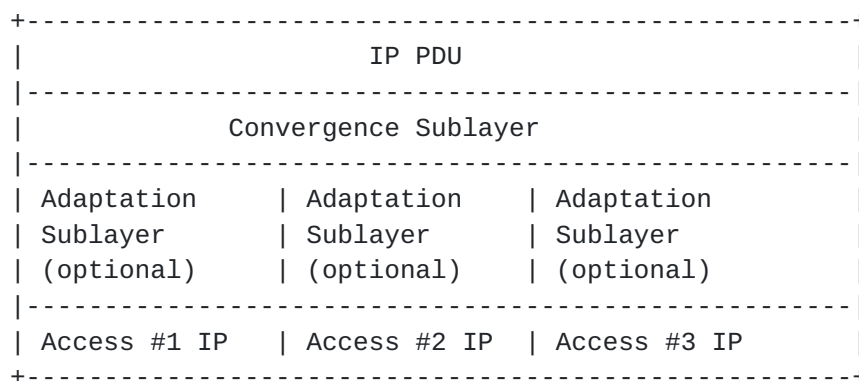   Figure 1: GMA User-Plane Protocol Stack

   It consists of the following two Sublayers:

     o Convergence sublayer: This layer performs multi-access
        specific tasks, e.g., multi-link (path) aggregation,
        splitting/reordering, lossless switching/retransmission,
        fragmentation, concatenation, etc.
     o Adaptation sublayer: This layer performs functions to handle
        tunnelling, network layer security, and NAT (network address
        translation).

   The convergence sublayer operates on top of the adaptation
   sublayer in the protocol stack. From the Transmitter perspective,
   a User Payload (e.g. IP packet) is processed by the convergence
   sublayer first, and then by the adaptation sublayer before being
   transported over a delivery connection; from the Receiver
   perspective, an IP packet received over a delivery connection is
   processed by the adaptation sublayer first, and then by the
   convergence sublayer.

   IP-over-IP tunneling has been used in today's multi-access
   solutions, e.g. [LWIPEP] [GRE], to insert the GRE header, and then
   encode additional control information in the GRE header fields,
   e.g. Key, Sequence Number. However, there are two main drawbacks
   with this approach: 1) IP-over-IP tunneling leads to higher
   overhead especially for small packet. For example, the overhead of
   IP-over-IP/GRE tunneling with both Key and Sequence Number is 32
   Bytes, which is 80% of a 40 Bytes TCP ACK packet; 2) It is
   difficult to introduce new control fields with the GRE header
   format.

   This document presents a light-weight GMA encapsulation protocol
   for the convergence sublayer. It avoids IP-over-IP tunneling to
   minimize overhead, and introduces new control fields to support
   fragmentation and concatenation at the convergence sublayer, which
   are not available in today's GRE-based solutions [LWIPEP] [GRE].

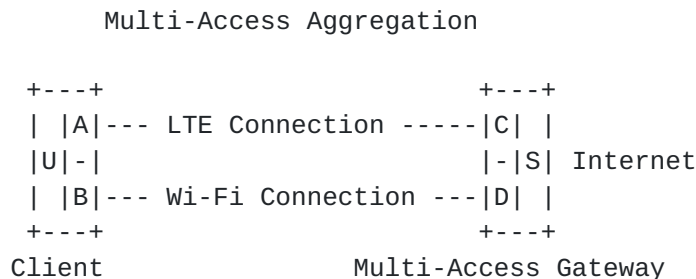   GMA protocol SHALL only operate between endpoints that have been
   configured to operate with GMA through additional control messages
   and procedures, for example [MAMS].

2. Conventions used in this document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL
   NOT",   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT
   RECOMMENDED", "MAY",   and "OPTIONAL" in this document are to be

interpreted as described   in BCP 14 [RFC2119] [RFC8174] when,
and only when, they appear in all capitals, as shown here.

3. Use Case

```
                    Multi-Access Aggregation

            +---+                      +---+
            | |A|--- LTE Connection -----|C| |
            |U|-|                        |-|S| Internet
            | |B|--- Wi-Fi Connection ---|D| |
            +---+                      +---+
              Client              Multi-Access Gateway
```

A: The adaptation sublayer endpoint of the LTE connection
resides in the client

B: The adaptation sublayer endpoint of the Wi-Fi connection
resides in the client

C: The adaptation sublayer endpoint of the LTE connection
resides in the Multi-Access Gateway, aka N-MADP (Network
Multi-Access Data Proxy) in [MAMS]

D: The adaptation sublayer endpoint of the Wi-Fi connection
resides in the Multi-Access Gateway

U: The convergence sublayer endpoint resides in the client

S: The convergence sublayer endpoint resides in the Multi-
Access Gateway

Figure 2: GMA-based Multi-Access Aggregation

As shown in Figure 2, a client device (e.g. Smartphone, Laptop,
etc.) may connect to Internet via both Wi-Fi and LTE connections,
one of which (e.g. LTE) may operate as the anchor connection, and
the other (e.g. Wi-Fi) may operate as the delivery connection. The
anchor connection provides the IP address and connectivity for
end-to-end Internet access, and the delivery connection provides
additional path between client and Multi-Access Gateway for multi-
access optimizations.

For example, per-packet aggregation allows a single IP flow to use
the combined bandwidth of the two connections. In another example,
packets lost over one connection due to temporarily link loss may
be retransmitted over the other connection. Such multi-access

optimization requires additional control information, e.g.
Sequence Number, in each IP data packet, which can be supported by
the GMA encapsulation protocol described in this document.

[4]. GMA Trailer Format

```
+-------------------------------------------------------+
| IP hdr |         IP payload           | GMA Trailer |
+-------------------------------------------------------+
```
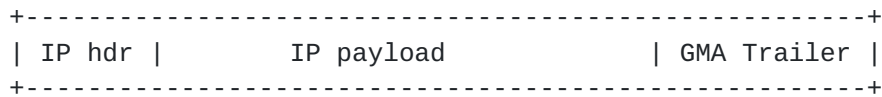   Figure 3: Trailer-based Encapsulation PDU Format

Figure 3 shows the trailer-based encapsulation GMA PDU (protocol
data unit) format for the convergence sublayer. A (GMA) PDU may
carry one or multiple IP packets, aka (GMA) SDUs (service data
unit), in the payload, or a fragment of the SDU.

The Protocol Type field in the IP header of the GMA PDU MUST be
changed to 114 (Any 0-Hop Protocol) [IANA] to indicate the
presence of the GMA trailer. The following three IP header fields
SHOULD also be changed:

  o IP Length: add the length of "GMA Trailer" to the length of
     the original IP packet
  o Time To Live (TTL) or Hop-Limit (HL): set the HL field to "0"
     if the original IP packet is IPv6, and set the TTL field to "1"
     if the original IP packet is IPv4.
  o IP checksum: recalculate after changing "Protocol Type", "TTL
     or HL" and "IP Length"

However, if UDP tunneling is used at the adaptation sublayer to
carry the GMA PDU, the above three IP header fields MAY remain
unchanged, and the receiver will determine the GMA PDU length
based on the UDP packet length.

The GMA (Generic Multi-Access) trailer MUST consist of the
following two mandatory fields. The Next Header field is always
the last octet at the end of the PDU, and the Flags field is the
second last. The Receiver SHOULD first decode the Flags field to
determine the length of the GMA trailer, and then decode each
optional field accordingly.

  o Next Header (1 Byte): the IP protocol type of the (first) SDU
     in a PDU
  o Flags (1 Byte): Bit 0 is the least significant bit (LSB), and
     Bit 7 is the most significant bit (MSB).

+ Checksum Present (bit 0): If the Checksum Present bit is set
to 1, then the Checksum field is present and contains valid
information.
+ Concatenation Present (bit 1): If the Concatenation Present
bit is set to 1, then the PDU carries multiple SDUs, and the
First SDU Length field is present and contains valid
information.
+ Connection ID Present (bit 2): If the Connection ID Present
bit is set to 1, then the Connection ID field is present and
contains valid information.
+ Flow ID Present (bit 3): If the Flow ID Present bit is set
to 1, then the Flow ID field is present and contains valid
information.
+ Fragmentation Present (bit 4): If the Fragmentation Present
bit is set to 1, then the PDU carry a fragment of the SDU and
the Fragmentation Control field is present and contains valid
information.
+ Sequence Number Present (bit 5): If the Sequence Number
Present bit is set to 1, then the Sequence Number field is
present and contains valid information.
+ Timestamp Present (bit 6): If the Timestamp Present bit is
set to 1, then the Timestamp field is present and contains
valid information.
+ TTL Present (bit 7): If the TTL Present bit is set to 1,
then the TTL field is present and contains the valid
information.

The GMA (Generic Multi-Access) trailer MAY consist of the
following optional fields:

o Checksum (1 Byte): to contain the (one's complement) checksum
sum of all the 8 bit in the trailer. For purposes of computing
the checksum, the value of the checksum field is zero. This
field is present only if the Checksum Present bit is set to
one.
o First SDU Length (2 Bytes): the length of the first IP packet
in the PDU, only included if a PDU contains multiple IP
packets.
o Connection ID (1 Byte): an unsigned integer to identify the
anchor and delivery connection of the GMA PDU.
+ Anchor Connection ID (MSB 4 Bits): an unsigned integer to
identify the anchor connection
+ Delivery Connection ID (LSB 4 Bits): an unsigned integer to
identify the delivery connection
o Flow ID (1 Byte): an unsigned integer to identify the IP flow
that a PDU belongs to, for example Data Radio Bearer (DRB) ID
[LWIPEP] for a cellular (e.g. LTE) connection.

o Fragmentation Control (FC) (e.g. 1 Byte): to provide necessary
  information for re-assembly, only needed if a PDU carries
  fragments.
o Sequence Number (4 Bytes): an auto-incremented integer to
  indicate order of transmission of the SDU (e.g. IP packet),
  needed for lossless switching or multi-link (path) aggregation
  or fragmentation. Sequence Number SHALL be generated per flow.
o Timestamp (4 Bytes): to contain the current value of the
  timestamp clock of the transmitter in the unit of 100
  microseconds.
o TTL (1 Byte): to contain the TTL value of the IP header if the
  GMA SDU is an IPv4 packet, and the Hop-Limit value of the IP
  header if the GMA SDU is an IPv6 packet.

Figure 4 shows the GMA trailer format with all the fields present.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Timestamp                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Sequence   Number                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    FC        |    Flow ID     | Connection ID |   First SDU    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  Length       |    Checksum    |    Flags      | Next Header   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
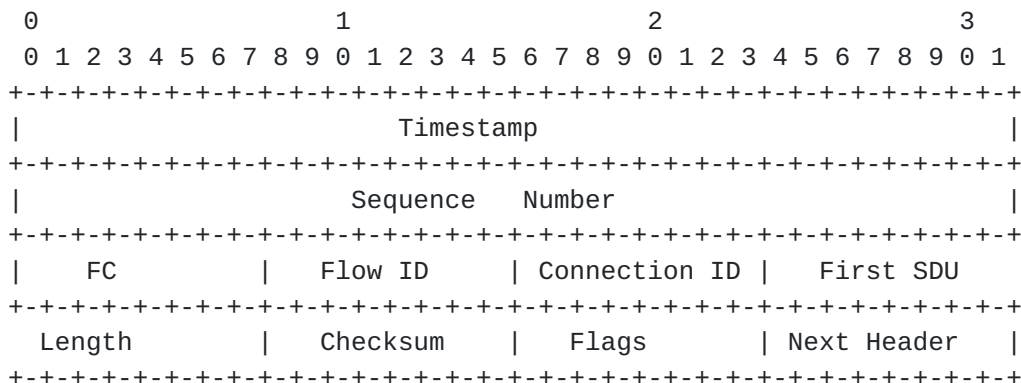
Figure 4: GMA Trailer Format

5. Fragmentation

The convergence sublayer MAY support fragmentation if a delivery
connection has a smaller maximum transmission unit (MTU) than the
original IP packet (SDU). The fragmentation procedure at the
convergence sublayer is similar to IP fragmentation [RFC791] in
principle, but with the following two differences for less
overhead:

o The fragment offset field is expressed in number of fragments
  not 8-bytes blocks
o The maximum number of fragments per SDU is 2^7 (=128)

The Fragmentation Control (FC) field in the trailer contains the
following bits:

        o Bit #7: a More Fragment (MF) flag to indicate if the fragment
          is the last one (0) or not (1)
        o Bit #0~#6: Fragment Offset (in units of fragments) to specify
          the offset of a particular fragment relative to the beginning
          of the SDU

   A PDU carries a whole SDU without fragmentation if the FC field is
   set to all "0"s or the FC field is not present in the trailer.
   Otherwise, the PDU contains a fragment of the SDU.

   The Sequence Number (SN) field in the trailer is used to
   distinguish the fragments of one SDU from those of another. The
   Fragment Offset (FO) field tells the receiver the position of a
   fragment in the original SDU. The More Fragment (MF) flag
   indicates the last fragment.

   To fragment a long SDU, the transmitter creates two PDUs and
   copies the content of the IP header fields from the long PDU into
   the IP header of both PDUs. The length field in the IP header of
   PDU SHOULD be changed to the length of the PDU, and the protocol
   type SHOULD be changed to 114.

   The data of the long SDU is divided into two portions based on the
   MTU size of the delivery connection. The first portion of the data
   is placed in the first PDU. The MF flag is set to "1", and the FO
   field is set to "0". The second portion of the data is placed in
   the second PDU. The MF flag is set to "0", and the FO field is set
   to "1". This procedure can be generalized for an n-way split,
   rather than the two-way split described the above.

   To assemble the fragments of a SDU, the receiver combines PDUs
   that all have the same Sequence Number (in the trailer). The
   combination is done by placing the data portion of each fragment
   in the relative order indicated by the Fragment Offset in that
   fragment's trailer. The first fragment will have the Fragment
   Offset set to "0", and the last fragment will have the More-
   Fragments flag reset to "0".

6. Concatenation

   The convergence sublayer MAY support concatenation if a delivery
   connection has a larger maximum transmission unit (MTU) than the
   original IP packet (SDU). Only the SDUs with the same client IP
   address, and the same Flow ID MAY be concatenated.

   The First SDU Length (FSL) field SHOULD be included in the trailer
   to indicate the length of the first SDU.

```
+-------------------------------------------------------------+
|IP hdr| IP payload    |IP hdr|   IP payload  | GMA Trailer |
+-------------------------------------------------------------+
```
              Figure 5: GMA PDU Format with Concatenation

   To concatenate two or more SDUs, the transmitter creates one PDU
   and copies the content of the IP header field from the first SDU
   into the IP header of the PDU. The data of the first SDU is placed
   in the first portion of the data of the PDU. The whole second SDU
   is then placed in the second portion of the data of the PDU
   (Figure 5). The procedure continues till the PDU size reaches the
   MTU of the delivery connection. If the FSL field is present in the
   trailer, the IP length field of the PDU SHOULD be updated to
   include all concatenated SDUs and the trailer, and the IP checksum
   field SHOULD be recalculated. However, if UDP tunnelling is used
   at the adaptation sublayer to carry the GMA PDU, both the IP
   Length field and the checksum field of the PDU MAY remain
   unchanged, and the receiver will determine the GMA PDU length
   based on the UDP packet length.

   To disaggregate a PDU, the receiver first obtains the length of
   the first SDU from the FSL field in the trailer, and decodes the
   first SDU. If the FSL field or the trailer is not present, the
   receiver obtains the length of the first SDU directly from the IP
   length field of the PDU. The receiver then obtains the length of
   the second SDU based on the length field in the second SDU IP
   header, and decodes the second SDU. The procedure continues till
   no byte is left in the PDU.

   If a PDU contains multiple SDUs, the SN field in the trailer is
   for the last SDU, and the SN of other SDU carried by the same PDU
   can be obtained according to its order in the PDU. For example, if
   the SN field is 6 and a PDU contains 3 SDUs (IP packets), the SN
   is 4, 5, and 6 for the first, second, and last SDU respectively.

7. GMA Header Format

```
+-----------------------------------------------------+
|GMA Header  | IP hdr |        IP payload          |
+-----------------------------------------------------+
```
     Figure 6: Header-based Non-IP Encapsulation GMA PDU Format

   Figure 6 shows the header-based encapsulation GMA PDU (protocol
   data unit) format for the convergence sublayer. A (GMA) PDU may
   carry one or multiple IP packets, aka (GMA) SDUs (service data
   unit), in the payload, or a fragment of the SDU.

If the adaptation sublayer, e.g. UDP tunnelling, supports non-IP
packet format, the GMA PDU format as shown in Figure 6 may be used
without any modification.

```
+------------------------------------------------------+
|IP hdr | GMA Header  | IP hdr |       IP payload     |
+------------------------------------------------------+
Figure 7: Header-based IP Encapsulation GMA PDU Format
```

If the adaptation sublayer supports only IP packet format, the GMA
PDU format SHOULD be modified as shown in Figure 7. The IP header
of the GMA SDU (e.g. IP packet) is copied to the front of the GMA
header so that the GMA PDU becomes an IP packet. Moreover, the
Protocol Type field in the IP header of the GMA PDU MUST be
changed to 114 (Any 0-Hop Protocol) [IANA] to indicate the
presence of the GMA Header. The following three IP header fields
SHALL also be changed:

   o IP Length: add the length of "GMA Header" and "IP Header" to
     the length of the original IP packet
   o Time To Live (TTL) or Hop-Limit (HL): set the HL field to "0"
     if the original IP packet is IPv6, and set the TTL field to "1"
     if the original IP packet is IPv4.
   o IP checksum: recalculate after changing "Protocol Type", "TTL
     or HL" and "IP Length"

```
 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

 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

 |   Flags       |         Timestamp

 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

                 |              Sequence    Number

 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

                 |    FC        |   Flow ID    | Connection ID |

 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

 |  Checksum     |

 +-+-+-+-+-+-+-+-+
```

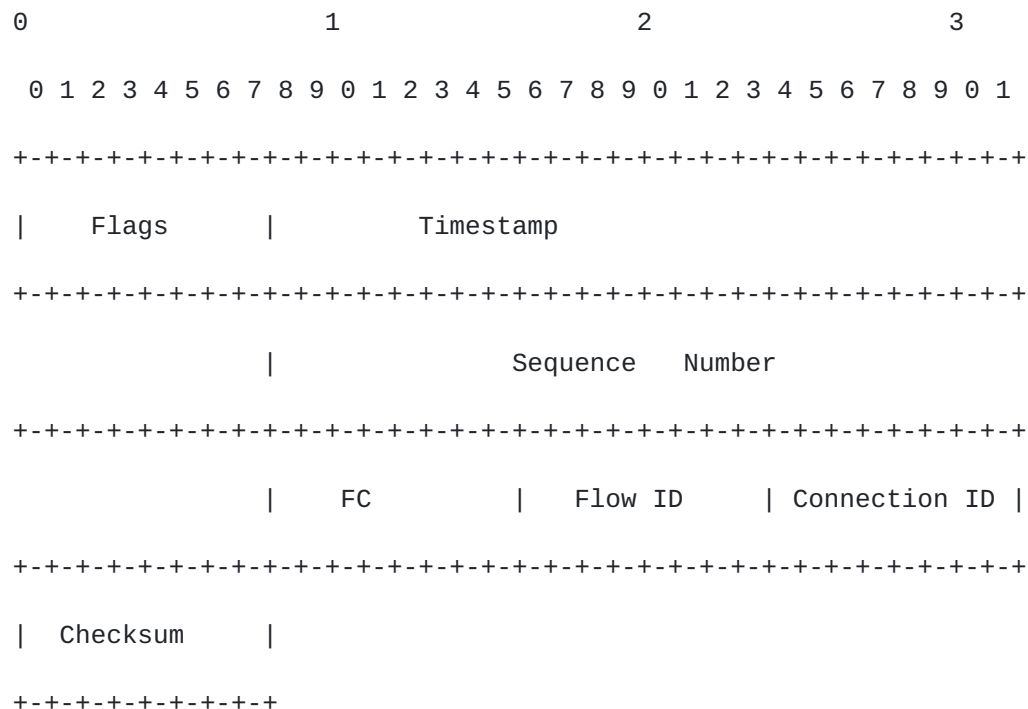   Figure 8: GMA Header Format

   Figure 8 shows the GMA header format with all the fields present.
   In comparison to the GMA trailer format, there are the following
   key differences:

     o The "Flags" field is moved to the beginning of the GMA header
     o The "First SDU Length" field is removed
     o The "Next Header" field is removed

   Fragmentation is supported in the same way as using the trailer-
   based GMA PDU format.

   Concatenation is also supported. The "Concatenation Present" bit
   in the Flags field is used to indicate if a GMA PDU carries one or
   multiple SDUs. However, since the IP header of the first SDU
   encapsulated in a GMA PDU is not changed during the encapsulation
   process, the "First SDU Length" field is not included in the GMA
   header even if the "Concatenation Present" bit is set to 1.

   The GMA endpoints: client and multi-access gateway (see Figure 2)
   MAY determine which GMA PDU format (header or trailer) to use
   through control signalling or pre-configuration.

8. Security Considerations

   Security in a network using GMA should be relatively similar to
   security in a normal IP network. The GMA protocol at the
   convergence sublayer is a 0-hop protocol, and relies on the
   security of the underlying network transport paths. When this
   cannot be assumed, appropriate security protocols (IPsec, DTLS,
   etc.) SHOULD be configured at the adaptation sublayer. On the
   other hand, packet filtering requires either that a firewall looks
   inside the GMA packet or that the filtering is done on the GMA
   endpoints. In those environments in which this is considered to be
   a security issue it may be desirable to terminate the GMA
   operation at the firewall.

   The balance of local-only packet leak prevention (HL=0) and
   security (HL=255) SHOULD be on preventing the leak of the local-
   only GMA PDUs outside the "local domain" to the Internet or to
   another domain which could use the same IP protocol type, i.e.
   114.

9. IANA Considerations

   This draft makes no requests of IANA

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174 May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10.2. Informative References

[MAMS] S. Kanugovi, S. Vasudevan, F. Baboescu, and J. Zhu, "Multiple Access Management Protocol", https://tools.ietf.org/html/draft-kanugovi-intarea-mams-protocol-03

[IANA]   https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml

[LWIPEP] 3GPP TS 36.361, "Evolved Universal Terrestrial Radio Access (E-UTRA); LTE-WLAN Radio Level Integration Using Ipsec Tunnel (LWIP) encapsulation; Protocol specification"

[RFC791] Internet Protocol, September 1981

[ATSSS] 3GPP TR 23.793, Study on access traffic steering, switch and splitting support in the 5G system architecture.

[GRE] RFC 8157, Huawei's GRE Tunnel Bonding Protocol, May 2017

Authors' Addresses

Jing Zhu

Intel

Email: jing.z.zhu@intel.com

Satish Kanugovi

Nokia

Email: satish.k@nokia.com