INTAREA
Internet Draft
Intended status: Standards Track
Expires: July 4,2019
J. Zhu
Intel
January 4, 2019

Generic Multi-Access (GMA) Convergence Encapsulation Protocols
draft-zhu-intarea-gma-00

Abstract

Today, a device can be simultaneously connected to multiple communication networks based on different technology implementations and network architectures like WiFi, LTE, 5G, and DSL. In such scenario, it is desirable to combine multiple access networks to improve quality of experience. For example, an IP flow may be delivered over both LTE and Wi-Fi network for higher end-to-end throughput. In another example, lost packets may be retransmitted when the device switches from Wi-Fi to LTE. However, such multi-access optimization requires inserting additional control information, e.g. Sequence Number, into each IP packet.

Unfortunately, there is no existing IP protocol for such purpose. As a result, IP-over-IP tunneling has been used in today's 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 using the existing GRE header format.

This document presents a new light-weight and flexible encapsulation protocol to support multi-access solutions [ATSSS] [MAMS] [LWIPEP] without IP-over-IP tunneling.

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 tunneling, network layer security, and NAT.

The convergence sublayer operates on top of the adaptation sublayer in the protocol stack. From the Transmitter perspective, a User Payload (e.g. IP PDU) 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.

This document presents a light-weight encapsulation protocol for the convergence sublayer. It avoids IP-over-IP tunneling to minimize the 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].

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. GMA Trailer Format

```
+-------------------------------------------------------+
| IP hdr |        IP payload             | GMA Trailer |
+-------------------------------------------------------+
```

Figure 2: Trailer-based Encapsulation PDU Format

Figure 2 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 GMA (Generic Multi-Access) trailer SHOULD consist of the following two mandatory fields:

- Next Header (1 Byte): the IP protocol type of the (first) SDU in a PDU
- Flags (1 Byte): Bit 0 is the least significant bit (LSB), and Bit 7 is the most significant bit (MSB).
  - Concatenation Present (bit 0): 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.
  - Fragmentation Present (bit 1): 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.
  - Flow ID Present (bit 2): If the Flow ID Present bit is set to 1, then the Flow ID field is present and contains valid information.
  - Checksum Present (bit 3): If the Checksum Present bit is set to 1, then the Checksum field is present and contains valid information.
  - Sequence Number Present (bit 4): If the Sequence Number Present bit is set to 1, then the Sequence Number field is present and contains valid information.
  - Timestamp Present (bit 5): If the Timestamp Present bit is set to 1, then the Timestamp field is present and contains valid information.
  - Bit 6-7: reserved

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

- First SDU Length (2 Bytes): the length of the first IP packet in the PDU, only included if a PDU contains multiple IP packets.
- Fragmentation Control (FC) (e.g. 1 Byte): to provide necessary information for re-assembly, only needed if a PDU carries fragments.

o Flow ID (1 Byte): an unsigned integer to identify the IP flow that a PDU belongs to. It is composted of the following two sub-fields:
  + Connection ID (MSB 4 Bits): an unsigned integer to identify the connection of the IP flow
  + Traffic Class ID (LSB 4 Bits): an unsigned integer to identify the traffic class of the IP flow, for example Data Radio Bearer (DRB) ID [LWIPEP] for a cellular (e.g. LTE) connection

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 Sequence Number (3 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 milliseconds.

Figure 3 shows the GMA trailer format with all the fields present. 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 trailer, and then decode each field accordingly.

```
      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         |   Checksum     |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |   Flow ID      |      FC       |    First SDU Length          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |    Flags       | Next Header   |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: GMA Trailer Format

Moreover, the Protocol Type field in the IP header of the PDU SHOULD be changed to "xyz" to indicate the presence of GMA trailer. The following two IP header fields MAY also be changed:

- o IP Length: add the length of "GMA Trailer" to the length of the original IP packet
- o IP checksum: recalculate after changing "Protocol Type" and "IP Length"

4. 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 "xyz".

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".

5. 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.

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 4). 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.

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.**

```
     +-------------------------------------------------------------+
     | IP hdr | IP payload  | IP hdr |  IP payload  | GMA Trailer |
```

```
     +------------------------------------------------------------+
                Figure 4: GMA PDU Format with Concatenation
```

6. Security Considerations

The convergence sublayer 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.

7. IANA Considerations

The trailer based encapsulation protocol in this document requires the allocation of a new value in the "IP Protocol Type" field within the IP header (e.g. xyz GMA Generic-Multi-Access-convergence-encapsulation).

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. 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