             Transmission of IPv6 packets over ITU-T G.9959 Networks
                          draft-ietf-6lo-lowpanz-03

Abstract

   This document describes the frame format for transmission of IPv6
   packets and a method of forming IPv6 link-local addresses and
   statelessly autoconfigured IPv6 addresses on ITU-T G.9959 networks.

Requirements Language

   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].

to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   The ITU-T G.9959 recommendation [G.9959] targets low-power Personal
   Area Networks (PANs). This document defines the frame format for
   transmission of IPv6 [RFC2460] packets as well as the formation of
   IPv6 link-local addresses and statelessly autoconfigured IPv6
   addresses on G.9959 networks.

The general approach is to adapt elements of [RFC4944] to G.9959
networks. G.9959 provides a Segmentation and Reassembly (SAR) layer
for transmission of datagrams larger than the G.9959 MAC PDU.

[RFC6775] updates [RFC4944] by specifying 6LoWPAN optimizations for
IPv6 Neighbor Discovery (ND) (originally defined by [RFC4861]). This
document limits the use of [RFC6775] to prefix and Context ID
assignment. An IID may be constructed from a G.9959 link-layer
address, leading to a "link-layer-derived IPv6 address". If using
that method, Duplicate Address Detection (DAD) is not needed.
Alternatively, IPv6 addresses may be assigned centrally via DHCP,
leading to a "non-link-layer-derived IPv6 address". Address
registration is only needed in certain cases.

In addition to IPv6 application communication, the frame format
defined in this document may be used by IPv6 routing protocols such
as RPL [RFC6550] or P2P-RPL [RFC6997] to implement IPv6 routing over
G.9959 networks.

The encapsulation frame defined by this specification may optionally
be transported via mesh routing below the 6LoWPAN layer. Routing
protocol specifications are out of scope of this document.

1.1. Terms used

6LoWPAN: IPv6-based Low-power Personal Area Network

ABR: Authoritative Border Router ([RFC6775])

AES: Advanced Encryption Scheme

EUI-64: Extended Unique Identifier

HomeID: G.9959 Link-Layer Network Identifier

IID: Interface IDentifier

MAC: Media Access Control

MTU: Maximum Transmission Unit

NodeID: G.9959 Link-Layer Node Identifier (Short Address)

PAN: Personal Area Network

PDU: Protocol Data Unit

SAR: Segmentation And Reassembly

ULA: Unique Local Address

2. G.9959 parameters to use for IPv6 transport

   This chapter outlines properties applying to the PHY and MAC of
   G.9959 and how to use these for IPv6 transport.

2.1. Addressing mode

   G.9959 defines how a unique 32-bit HomeID network identifier is
   assigned by a network controller and how an 8-bit NodeID host
   identifier is allocated. NodeIDs are unique within the logical
   network identified by the HomeID. The logical network identified by
   the HomeID maps directly to an IPv6 subnet identified by one or more
   IPv6 prefixes.

   An IPv6 host MUST construct its link-local IPv6 address from the
   link-layer-derived IID in order to facilitate IP header compression
   as described in [RFC6282].

   A node interface MAY support the M flag of the RA message for the
   construction of routable IPv6 addresses. If the M flag is not
   supported, link-layer-derived addressing MUST be used. If the M flag
   is supported, link-layer-derived addressing MUST be used if the M
   flag is 0, while DHCPv6 address assignment MUST be used if the M flag
   is 1. Nodes using DHCPv6 assigned IPv6 addresses MUST comply with
   [RFC6775].

   A word of caution: since HomeIDs and NodeIDs are handed out by a
   network controller function during inclusion, identifier validity and
   uniqueness is limited by the lifetime of the logical network
   membership. This can be cut short by a mishap occurring to the
   network controller. Having a single point of failure at the network
   controller suggests that deployers of high-reliability applications
   should carefully consider adding redundancy to the network controller
   function.

   This warning applies to link-layer-derived addressing as well as to
   non-link-layer addressing deployments.

2.2. IPv6 Multicast support

   [RFC3819] recommends that IP subnetworks support (subnet-wide)
   multicast. G.9959 supports direct-range IPv6 multicast while subnet-
   wide multicast is not supported natively by G.9959. Subnet-wide
   multicast may be provided by an IP routing protocol or a mesh routing
   protocol operating below the 6LoWPAN layer. Routing protocol
   specifications are out of scope of this document.

IPv6 multicast packets MUST be carried via G.9959 broadcast.

As per [G.9959], this is accomplished as follows:

1. The destination HomeID of the G.9959 MAC PDU MUST be the HomeID
    of the logical network

2. The destination NodeID of the G.9959 MAC PDU MUST be the
    broadcast NodeID (0xff)

G.9959 broadcast MAC PDUs are only intercepted by nodes within the
logical network identified by the HomeID.

2.3. G.9959 MAC PDU size and IPv6 MTU

IPv6 packets MUST use G.9959 transmission profiles which support MAC
PDU payload sizes of 150 bytes or higher, i.e. profile R3 or higher.
(G.9959 profiles R1 and R2 only support MPDU payloads around 40 bytes
and the transmission speed is down to 9.6kbit/s)

[RFC2460] specifies that IPv6 packets may be up to 1280 octets.

G.9959 provides Segmentation And Reassembly for payloads up to 1350
octets. IPv6 Header Compression [RFC6282] improves the chances that
a short IPv6 packet can fit into a single G.9959 frame. Therefore,
section Section 3 specifies that [RFC6282] MUST be supported. With
the mandatory link-layer security enabled, a G.9959 R3 MAC PDU may
accommodate 6LoWPAN datagrams of up to 130 octets without triggering
G.9959 Segmentation and Reassembly. Longer 6LoWPAN datagrams will
lead to the transmission of multiple G.9959 PDUs.

2.4. Transmission status indications

The G.9959 MAC layer provides native acknowledgement and
retransmission of MAC PDUs. The G.9959 SAR layer does the same for
larger datagrams. A mesh routing layer may provide a similar feature
for routed communication. An IPv6 routing stack communicating over
G.9959 may utilize link-layer status indications such as delivery
confirmation and Ack timeout from the MAC layer.

2.5. Transmission security

Implementations claiming conformance with this document MUST enable
G.9959 shared network key security.

The shared network key is intended to address security requirements
in the home at the normal security requirements level. For
applications with high or very high requirements on confidentiality

and/or integrity, additional application layer security measures for
end-to-end authentication and encryption may need to be applied.
(The availability of the network relies on the security properties of
the network key in any case)

3. 6LoWPAN Adaptation Layer and Frame Format

   The 6LoWPAN encapsulation formats defined in this chapter are carried
   as payload in the G.9959 MAC PDU. IPv6 header compression [RFC6282]
   MUST be supported by implementations of this specification.

   All 6LoWPAN datagrams transported over G.9959 are prefixed by a
   6LoWPAN encapsulation header stack. The 6LoWPAN payload follows this
   encapsulation header stack. Each header in the header stack contains
   a header type followed by zero or more header fields. An IPv6 header
   stack may contain, in the following order, addressing, hop-by-hop
   options, routing, fragmentation, destination options, and finally
   payload [RFC2460]. The 6LoWPAN header format is structured the same
   way. Currently only one payload option is defined for the G.9959
   6LoWPAN header format.

   The definition of 6LoWPAN headers consists of the dispatch value, the
   definition of the header fields that follow, and their ordering
   constraints relative to all other headers. Although the header stack
   structure provides a mechanism to address future demands on the
   6LoWPAN adaptation layer, it is not intended to provide general
   purpose extensibility.

   An example of a complete G.9959 6LoWPAN datagram can be found in
   Appendix A.

3.1. Dispatch Header

   The dispatch header is shown below:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | 6LoWPAN CmdCls |    Dispatch    |   Type-specific header        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
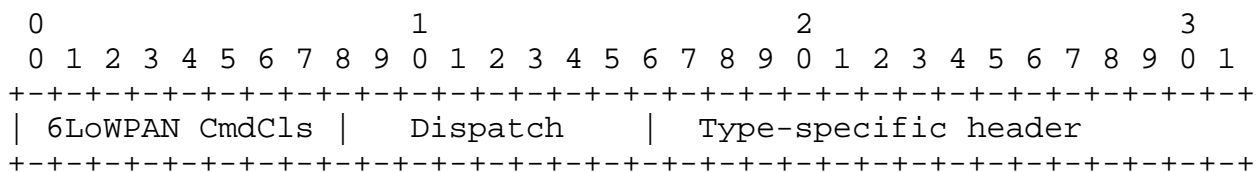
                    Figure 1: Dispatch Type and Header

   6LoWPAN CmdCls: 6LoWPAN Command Class identifier. This field MUST
   carry the value 0x4F [G.9959]. The value specifies that the
   following bits are a 6LoWPAN encapsulated datagram. Non-6LoWPAN

protocols MUST ignore the contents following the 6LoWPAN Command
Class identifier.

Dispatch: Identifies the header type immediately following the
Dispatch Header.

Type-specific header: A header determined by the Dispatch Header.

The dispatch value may be treated as an unstructured namespace. Only
a few symbols are required to represent current 6LoWPAN
functionality. Although some additional savings could be achieved by
encoding additional functionality into the dispatch byte, these
measures would tend to constrain the ability to address future
alternatives.

Dispatch values used in this specification are compatible with the
dispatch values defined by [RFC4944] and [RFC6282].

```
+------------+------------------------------------------+-----------+
| Pattern    | Header Type                              | Reference |
+------------+------------------------------------------+-----------+
| 01  1xxxxx | 6LoWPAN_IPHC - Compressed IPv6 Addresses | [RFC6282] |
+------------+------------------------------------------+-----------+
```
 All other Dispatch values are unassigned in this document.

                       Figure 2: Dispatch values

6LoWPAN_IPHC: IPv6 Header Compression. Refer to [RFC6282].

4. 6LoWPAN addressing

IPv6 addresses are autoconfigured from IIDs which are again
constructed from link-layer address information to save memory in
devices and to facilitate efficient IP header compression as per
[RFC6282].

A NodeID is mapped into an IEEE EUI-64 identifier as follows:

   IID = 0000:00ff:fe00:YYXX

                 Figure 3: Constructing a compressible IID

where XX carries the G.9959 NodeID and YY is a one byte value chosen
by the individual node. The default YY value MUST be zero. A node
MAY use other values of YY than zero to form additional IIDs in order
to instantiate multiple IPv6 interfaces. The YY value MUST be

ignored when computing the corresponding NodeID (the XX value) from
an IID.

The method of constructing IIDs from the link-layer address obviously
does not support addresses assigned or constructed by other means. A
node MUST NOT compute the NodeID from the IID if the first 6 bytes of
the IID do not comply with the format defined in Figure 3. In that
case, the address resolution mechanisms of RFC 6775 apply.

4.1. Stateless Address Autoconfiguration of routable IPv6 addresses

The IID defined above MUST be used whether autoconfiguring a ULA IPv6
address [RFC4193] or a globally routable IPv6 address [RFC3587] in
G.9959 subnets.

4.2. IPv6 Link Local Address

The IPv6 link-local address [RFC4291] for a G.9959 interface is
formed by appending the IID defined above to the IPv6 link local
prefix FE80::/64.

The "Universal/Local" (U/L) bit MUST be set to zero in keeping with
the fact that this is not a globally unique value [EUI64].

The resulting link local address is formed as follows:

```
      10 bits             54 bits                     64 bits
   +----------+-----------------------+---------------------------+
   |1111111010|        (zeros)        | Interface Identifier (IID) |
   +----------+-----------------------+---------------------------+
```
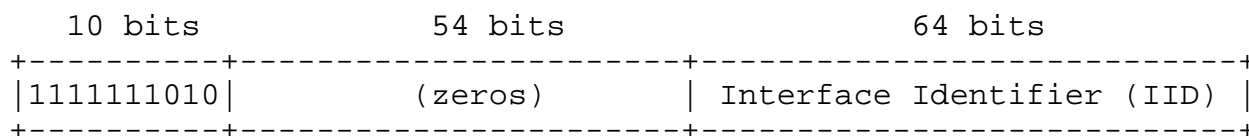
                  Figure 4: IPv6 Link Local Address

4.3. Unicast Address Mapping

The address resolution procedure for mapping IPv6 unicast addresses
into G.9959 link-layer addresses follows the general description in
Section 7.2 of [RFC4861]. The Source/Target Link-layer Address
option MUST have the following form when the link layer is G.9959.

```
                      0                   1
                      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
                     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                     |      Type     |    Length=1    |
                     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                     |      0x00     |     NodeID     |
                     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                     |             Padding            |
                     +-                              -+
                     |           (All zeros)          |
                     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
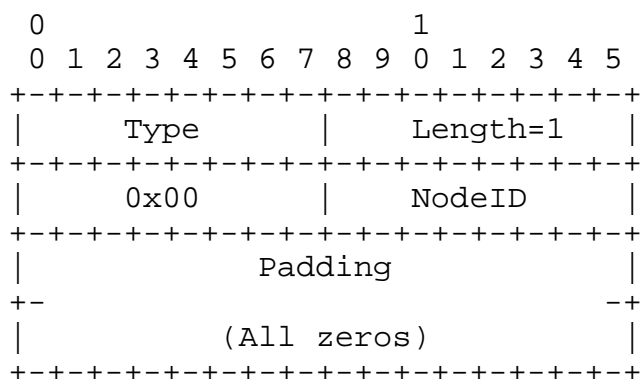
              Figure 5: IPv6 Unicast Address Mapping

   Option fields:

   Type: The value 1 signifies the Source Link-layer address. The value
   2 signifies the Destination Link-layer address.

   Length: This is the length of this option (including the type and
   length fields) in units of 8 octets. The value of this field is
   always 1 for G.9959 NodeIDs.

   NodeID: This is the G.9959 NodeID the actual interface currently
   responds to. The link-layer address may change if the interface
   joins another network at a later time.

4.4. On the use of Neighbor Discovery technologies

   [RFC4861] specifies how IPv6 nodes may resolve link layer addresses
   from IPv6 addresses via the use of link-local IPv6 multicast.
   [RFC6775] is an optimization of [RFC4861], specifically targeting
   6LoWPAN networks. [RFC6775] defines how a 6LoWPAN node may register
   IPv6 addresses with an authoritative border router (ABR). Mesh-under
   networks MUST NOT use [RFC6775] address registration. However,
   [RFC6775] address registration MUST be used if the first 6 bytes of
   the IID do not comply with the format defined in Figure 3.

4.4.1. Prefix and CID management (Route-over)

   In route-over environments, IPv6 hosts MUST use [RFC6775] address
   registration. A node implementation for route-over operation MAY use
   RFC6775 mechanisms for obtaining IPv6 prefixes and corresponding
   header compression context information [RFC6282]. RFC6775 Route-over
   requirements apply with no modifications.

4.4.2. Prefix and CID management (Mesh-under)

   An implementation for mesh-under operation MUST use [RFC6775]
   mechanisms for managing IPv6 prefixes and corresponding header
   compression context information [RFC6282]. [RFC6775] Duplicate
   Address Detection (DAD) MUST NOT be used, since the link-layer
   inclusion process of G.9959 ensures that a NodeID is unique for a
   given HomeID.

   With this exception and the specific redefinition of the RA Router
   Lifetime value 0xFFFF (refer to Section 4.4.2.3), the text of the
   following subsections is in compliance with [RFC6775].

4.4.2.1. Prefix assignment considerations

   As stated by [RFC6775], an ABR is responsible for managing
   prefix(es). Global routable prefixes may change over time. It is
   RECOMMENDED that a ULA prefix is assigned to the 6LoWPAN subnet to
   facilitate stable site-local application associations based on IPv6
   addresses. A node MAY support the M flag of the RA message. If the
   M flag is not supported, link-layer-derived addressing MUST be used.
   If the M flag is supported, link-layer-derived addressing MUST be
   used if the M flag is 0, while DHCPv6 address assignment MUST be used
   if the M flag is 1.

4.4.2.2. Robust and efficient CID management

   The 6LoWPAN Context Option (6CO) is used according to [RFC6775] in an
   RA to disseminate Context IDs (CID) to use for compressing prefixes.
   One or more prefixes and corresponding Context IDs MUST be assigned
   during initial node inclusion.

   When updating context information, a CID may have its lifetime set to
   zero to obsolete it. The CID MUST NOT be reused immediately; rather
   the next vacant CID should be assigned. Header compression based on
   CIDs MUST NOT be used for RA messages carrying Context Information.
   An expired CID and the associated prefix MUST NOT be reset but rather
   retained in receive-only mode if there is no other current need for
   the CID value. This will allow an ABR to detect if a sleeping node
   without clock uses an expired CID and in response, the ABR MUST
   return an RA with fresh Context Information to the originator.

4.4.2.3. Infinite prefix lifetime support for island-mode networks

   Nodes MUST renew the prefix and CID according to the lifetime
   signaled by the ABR. [RFC6775] specifies that the maximum value of
   the RA Router Lifetime field MAY be up to 0xFFFF. This document
   further specifies that the value 0xFFFF MUST be interpreted as

infinite lifetime. This value MUST NOT be used by ABRs. Its use is
only intended for a sleeping network controller; for instance a
battery powered remote control being master for a small island-mode
network of light modules.

5. Header Compression

IPv6 header compression [RFC6282] MUST be implemented according to
[RFC6282]. This section will simply identify substitutions that
should be made when interpreting the text of [RFC6282].

In general the following substitutions should be made:

o Replace "802.15.4" with "G.9959"

o Replace "802.15.4 short address" with "<Interface><G.9959 NodeID>"

o Replace "802.15.4 PAN ID" with "G.9959 HomeID"

When a 16-bit address is called for (i.e., an IEEE 802.15.4 "short
address") it MUST be formed by prepending an Interface label byte to
the G.9959 NodeID:

```
                    0                   1
                    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
                   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                   |    Interface    |    NodeID     |
                   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

A transmitting node may be sending to an IPv6 destination address
which can be reconstructed from the link-layer destination address.
If the Interface number is zero (the default value), all IPv6 address
bytes may be elided. Likewise, the Interface number of a fully
elided IPv6 address (i.e. SAM/DAM=11) may be reconstructed to the
value zero by a receiving node.

64 bit 802.15.4 address details do not apply.

6. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an
RFC.

7. Security Considerations

   The method of derivation of Interface Identifiers from 8-bit NodeIDs
   preserves uniqueness within the logical network. However, there is
   no protection from duplication through forgery. Neighbor Discovery
   in G.9959 links may be susceptible to threats as detailed in
   [RFC3756]. G.9959 networks may feature mesh routing. This implies
   additional threats due to ad hoc routing as per [KW03]. G.9959
   provides capability for link-layer security. G.9959 nodes MUST use
   link-layer security with a shared key. Doing so will alleviate the
   majority of threats stated above. A sizeable portion of G.9959
   devices is expected to always communicate within their PAN (i.e.,
   within their subnet, in IPv6 terms). In response to cost and power
   consumption considerations, these devices will typically implement
   the minimum set of features necessary. Accordingly, security for
   such devices may rely on the mechanisms defined at the link layer by
   G.9959. G.9959 relies on the Advanced Encryption Standard (AES) for
   authentication and encryption of G.9959 frames and further employs
   challenge-response handshaking to prevent replay attacks.

   It is also expected that some G.9959 devices (e.g. billing and/or
   safety critical products) will implement coordination or integration
   functions. These may communicate regularly with IPv6 peers outside
   the subnet. Such IPv6 devices are expected to secure their end-to-
   end communications with standard security mechanisms (e.g., IPsec,
   TLS, etc).

8. Acknowledgements

   Thanks to the authors of RFC 4944 and RFC 6282 and members of the
   IETF 6LoWPAN working group; this document borrows extensively from
   their work. Thanks to Erez Ben-Tovim, Kerry Lynn, Michael
   Richardson, Tommas Jess Christensen for useful comments. Thanks to
   Carsten Bormann for extensive feedback which improved this document
   significantly.

9. References

9.1. Normative References

   [EUI64]   IEEE, "communicationIDELINES FOR 64-BIT GLOBAL IDENTIFIER
               (EUI-64) REGISTRATION AUTHORITY", IEEE Std http://
               standards.ieee.org/regauth/oui/tutorials/EUI64.html,
               November 2012.

   [G.9959]   "G.9959 (02/12) + G.9959 Amendment 1 (10/13): Short range,
               narrow-band digital radiocommunication transceivers",
               February 2012.

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

[RFC2460]   Deering, S. and R. Hinden, "Internet Protocol, Version 6
            (IPv6) Specification", RFC 2460, December 1998.

[RFC2464]   Crawford, M., "Transmission of IPv6 Packets over Ethernet
            Networks", RFC 2464, December 1998.

[RFC3587]   Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global
            Unicast Address Format", RFC 3587, August 2003.

[RFC4193]   Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast
            Addresses", RFC 4193, October 2005.

[RFC4291]   Hinden, R. and S. Deering, "IP Version 6 Addressing
            Architecture", RFC 4291, February 2006.

[RFC4861]   Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
            "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,
            September 2007.

[RFC4941]   Narten, T., Draves, R., and S. Krishnan, "Privacy
            Extensions for Stateless Address Autoconfiguration in
            IPv6", RFC 4941, September 2007.

[RFC4944]   Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler,
            "Transmission of IPv6 Packets over IEEE 802.15.4
            Networks", RFC 4944, September 2007.

[RFC6282]   Hui, J. and P. Thubert, "Compression Format for IPv6
            Datagrams over IEEE 802.15.4-Based Networks", RFC 6282,
            September 2011.

[RFC6775]   Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann,
            "Neighbor Discovery Optimization for IPv6 over Low-Power
            Wireless Personal Area Networks (6LoWPANs)", RFC 6775,
            November 2012.

9.2. Informative References

[RFC3756]   Nikander, P., Kempf, J., and E. Nordmark, "IPv6 Neighbor
            Discovery (ND) Trust Models and Threats", RFC 3756, May
            2004.

   [RFC3819] Karn, P., Bormann, C., Fairhurst, G., Grossman, D.,
              Ludwig, R., Mahdavi, J., Montenegro, G., Touch, J., and L.
              Wood, "Advice for Internet Subnetwork Designers", BCP 89,
              RFC 3819, July 2004.

   [RFC6550] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R.,
              Levis, P., Pister, K., Struik, R., Vasseur, JP., and R.
              Alexander, "RPL: IPv6 Routing Protocol for Low-Power and
              Lossy Networks", RFC 6550, March 2012.

   [RFC6997] Goyal, M., Baccelli, E., Philipp, M., Brandt, A., and J.
              Martocci, "Reactive Discovery of Point-to-Point Routes in
              Low-Power and Lossy Networks", RFC 6997, August 2013.

Appendix A. G.9959 6LoWPAN datagram example

This example outlines each individual bit of a sample IPv6 UDP packet
arriving to a G.9959 node from a host in the Internet
via a PAN border router.

In the G.9959 PAN, the complete frame has the following fields.

G.9959:

```
   +------+---------+----------+---+-----+----------...
   |HomeID|SrcNodeID|FrmControl|Len|SeqNo|DestNodeID|
   +------+---------+----------+---+-----+----------+-...
```

6LoWPAN:

```
   ...+--------------+---------------+-----------------------...
      |6LoWPAN CmdCls|6LoWPAN_IPHC Hdr|Compressed IPv6 headers|
   ...--------------+---------------+-----------------------+-...
```

6LoWPAN, TCP/UDP, App payload:

```
   ...+-----------------------+-----------+-----------+
      |Uncompressed IPv6 headers|TCP/UDP/ICMP|App payload|
   ...-----------------------+-----------+-----------+
```

The frame comes from the source IPv6 address 2001:0db8:ac10:ef01::ff:fe00:12
06.
The source prefix 2001:0db8:ac10:ef01/64 is identified by the IPHC CID = 3.
The frame is delivered in direct range from the gateway which has source Nod
eID = 1.
The Interface Identifier (IID) ff:fe00:1206 is recognised as a link-layer-de
rived address
and is compressed to the 16 bit value 0x1206.

The frame is sent to the destination IPv6 address 2001:0db8:27ef:42ca::ff:fe
00:0004.
The destination prefix 2001:0db8:27ef:42ca/64 is identified by the IPHC CID
= 2.

The Interface Identifier (IID) ff:fe00:0004 is recognised as a link-layer-de
rived address.

Thanks to the link-layer-derived addressing rules, the sender knows that thi
s is to be sent
to G.9959 NodeID = 4; targeting the IPv6 interface instance number 0 (the de
fault).

To reach the 6LoWPAN stack of the G.9959 node, (skipping the G.9959 header f
ields) the first octet must be the 6LoWPAN Command Class (0x4F).

```
     0
     0 1 2 3 4 5 6 7 8
    +-+-+-+-+-+-+-+-+-...
    |      0x4F       |
    +-+-+-+-+-+-+-+-+-...
```

The Dispatch header bits '011' advertises a compressed IPv6 header to follow
.

```
     0                   1
     0 1 2 3 4 5 6 7 8 9 0
    +-+-+-+-+-+-+-+-+-+-+-...
    |      0x4F       |0 1 1
    +-+-+-+-+-+-+-+-+-+-+-...
```

The following bits encode the following:

TF = '11'   : Traffic Class and Flow Label are elided.
NH = '1'   : Next Header is elided
HLIM = '10' : Hop limit is 64

```
     0                   1
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
    |      0x4F       |0 1 1 1 1 1 0 1|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

CID = '1'   : CI data follows the DAM field
SAC = '1'   : Src addr uses stateful, context-based compression
SAM = '10' : Combine src CID and 16 bits to link-layer-derived addr
M = '0'   : Dest addr is not a multicast addr
DAC = '1'   : Dest addr uses stateful, context-based compression
DAM = '11' : Combine dest CID and dest NodeID to link-layer-derived addr

```
    0                   1                   2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
   |      0x4F      |0 1 1 1 1 1 0 1|1 1 1 0 0 1 1 1|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Address compression context identifiers:

SCI =   0x3
DCI =   0x2

```
       2           3
       4 5 6 7 8 9 0 1
    ...+-+-+-+-+-+-+-+-...
       | 0x3 | 0x2 |
    ...+-+-+-+-+-+-+-+-...
```

IPv6 header fields:
(skipping "version" field)
(skipping "Traffic Class")
(skipping "flow label")
(skipping "payload length")

IPv6 header address fields:

SrcIP = 0x1206 : 16 LS bits of link-layer-derived address to combine with SC
I
(skipping DestIP) - completely reconstructed from Dest NodeID and DCI

```
       2           3                   4
       4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
       | 0x3 | 0x2 |     0x12      |      0x06      |
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Hext header encoding for the UDP header:

Dispatch = '11110': Next Header dispatch code for UDP header
C =   '0'   : 16 bit checksupm carried inline
P =   '00'   : both src port and dest Port are carried in-line.
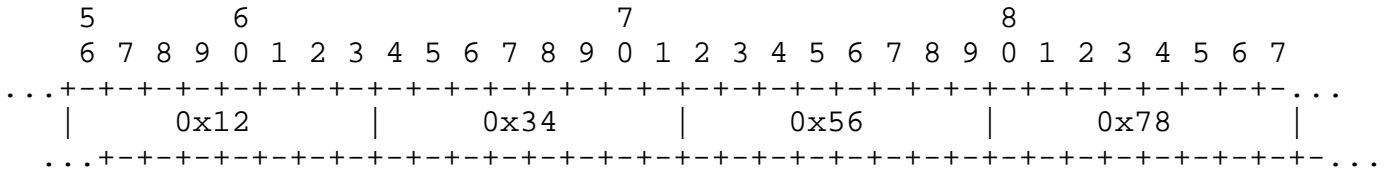
```
       4   5
       8 9 0 1 2 3 4 5
    ...+-+-+-+-+-+-+-+-...
       |1 1 1 1 0|0|0 0|
```
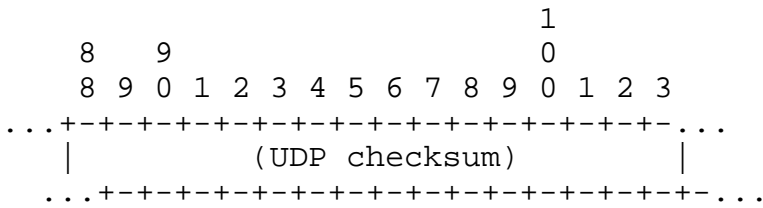
```
      ...+-+-+-+-+-+-+-+-...
```

UDP header fields:

src Port = 0x1234
dest port = 0x5678

```
      5           6                   7                   8
      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 2 3 4 5 6 7
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
      |      0x12      |      0x34     |      0x56     |      0x78     |
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

UDP header fields:

(skipping "length")
checksum = ... (actual checksum value depends on
                  the actual UDP payload)

```
                          1
      8   9               0
      8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
      |          (UDP checksum)       |
    ...+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
```

Add your own UDP payload here...

Appendix B. Change Log

B.1. Changes since -00

   o Clarified that mesh-under routing may take place below the 6LoWPAN
      layer but that specific mesh-under routing protocols are not
      within the scope of this doc.

   o Clarified that RFC6282 IPv6 Header Compression MUST be supported.

   o Clarified the text of section 5.4 on the use of RFC6775 address
      registration in mesh-under networks.

   o Split 5.4.2 into multiple paragraphs.

B.2. Changes since -01

   o Added this Change Log

   o Editorial nits.

   o Made IPv6 Header Compression mandatory. Therefore, the Dispatch
      value "01 000001 - Uncompressed IPv6 Addresses" was removed from
      figure 2.

   o Changed SHOULD to MUST: An IPv6 host SHOULD construct its link-
      local IPv6 address and routable IPv6 addresses from the NodeID in
      order to facilitate IP header compression as described in
      [RFC6282].

   o Changed SHOULD NOT to MUST NOT: Mesh-under networks MUST NOT use
      [RFC6775] address registration.

   o Changed SHOULD NOT to MUST NOT: [RFC6775] Duplicate Address
      Detection (DAD) MUST NOT be used.

   o Changed SHOULD NOT to MUST NOT: The CID MUST NOT be reused
      immediately;

   o Changed SHOULD NOT to MUST NOT: An expired CID and the associated
      prefix MUST NOT be reset but rather retained in receive-only mode

   o Changed LBR -> ABR

   o Changed SHOULD to MUST: , the ABR MUST return an RA with fresh
      Context Information to the originator.

   o Changed SHOULD NOT to MUST NOT: This value MUST NOT be used by
      ABRs. Its use is only intended for a sleeping network controller;

B.3. Changes since -02

   o Editorial nits.

   o Moved text to the right section so that it does not prohibit DAD
      for Route-Over deployments.

   o Introduced RA m flag support so that nodes may be instructed to
      use DHCPv6 for centralized address assignment.

Authors' Addresses

    Anders Brandt
    Sigma Designs
    Emdrupvej 26A, 1.
    Copenhagen O 2100
    Denmark

    Email: anders_brandt@sigmadesigns.com

    Jakob Buron
    Sigma Designs
    Emdrupvej 26A, 1.
    Copenhagen O 2100
    Denmark

    Email: jakob_buron@sigmadesigns.com