6lo Working Group   C. Gomez
Internet-Draft   UPC
Intended status: Standards Track   A. Minaburo
Expires: 31 August 2024   Consultant
                                                          February 2024

         Transmission of SCHC-compressed packets over IEEE 802.15.4 networks
                        draft-ietf-6lo-schc-15dot4-05

Abstract

   A framework called Static Context Header Compression and
   fragmentation (SCHC) has been designed with the primary goal of
   supporting IPv6 over Low Power Wide Area Network (LPWAN) technologies
   [RFC8724]. One of the SCHC components is a header compression
   mechanism. If used properly, SCHC header compression allows a
   greater compression ratio than that achievable with traditional
   6LoWPAN header compression [RFC6282]. For this reason, it may make
   sense to use SCHC header compression in some 6LoWPAN environments,
   including IEEE 802.15.4 networks. This document specifies how a
   SCHC-compressed packet can be carried over IEEE 802.15.4 networks.
   The document also enables the transmission of SCHC-compressed UDP/
   CoAP headers over 6LoWPAN-compressed IPv6 packets.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   RFC 6282 is the main specification for IPv6 over Low power Wireless
   Personal Area Network (6LoWPAN) IPv6 header compression [RFC6282].
   That RFC was designed assuming IEEE 802.15.4 as the layer below the
   6LoWPAN adaptation layer, and it has also been reused (with proper
   adaptations) for IPv6 header compression over many other technologies
   relatively similar to IEEE 802.15.4 in terms of characteristics such
   as physical layer bit rate, layer 2 maximum payload size, etc.
   Examples of such technologies comprise BLE, DECT-ULE, ITU G.9959, MS/
   TP, NFC, and PLC. RFC 6282 provides additional functionality, such
   as a mechanism for UDP header compression.

   In the best cases, RFC 6282 allows to compress a 40-byte IPv6 header
   down to a 2-byte compressed header (for link-local interactions) or a
   3-byte compressed header (when global IPv6 addresses are used). On
   the other hand, RFC 6282 typically compresses a UDP header to a size
   of 2 to 4 bytes. Therefore, in advantageous conditions, a 48-byte
   uncompressed IPv6/UDP header may be compressed down to a 4- to 6-byte
   format (when using link-local addresses) or a 5- to 7-byte format
   (for global interactions) by using RFC 6282.

   Recently, a framework called Static Context Header Compression (SCHC)
   has been designed with the primary goal of supporting IPv6 over Low
   Power Wide Area Network (LPWAN) technologies [RFC8724]. SCHC
   comprises header compression and fragmentation functionality tailored
   to the extraordinary constraints of LPWAN technologies, which are
   more severe than those exhibited by IEEE 802.15.4 or other relatively
   similar technologies. SCHC header compression allows a greater
   compression ratio than that of RFC 6282. If used properly, SCHC
   allows to compress an IPv6/UDP header down to e.g. a single byte. In
   addition, SCHC can be used to compress Constrained Application
   Protocol (CoAP) headers [RFC7252][RFC8824], which further increases
   the achievable performance improvement of using SCHC header

compression, since there is no 6LoWPAN header compression mechanism defined for CoAP. Therefore, it may make sense to use SCHC header compression in some 6LoWPAN environments, including IEEE 802.15.4 networks, considering its greater efficiency.

This document specifies how a SCHC-compressed packet can be carried over IEEE 802.15.4 networks. In order to ease a transition from existing 6LoWPAN/6Lo implementations to support SCHC header compression, the document also enables the transmission of SCHC-compressed UDP/CoAP headers over 6LoWPAN-compressed IPv6 packets. Further transition approaches are also described.

The mechanism to be used to provide the SCHC header compression context to the nodes in an IEEE 802.15.4 network is out of the scope of this document.

Note that, as per this document, and while SCHC defines fragmentation mechanisms as well, 6LoWPAN/6Lo fragmentation is used when necessary to transport SCHC-compressed packets over IEEE 802.15.4 networks [RFC4944][RFC8930][RFC8931].

This specification updates RFC 8138 and RFC 9008.

2. Terminology

2.1. Requirements language

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 BCP14 [RFC2119], [RFC8174], when, and only when, they appear in all capitals, as shown here.

2.2. Background on previous specifications

The reader is expected to be familiar with the terms and concepts defined in specifications of 6LoWPAN frame formats [RFC4944], RPL [RFC6550] and companion documents [RFC6553][RFC6554][RFC9008], 6LoWPAN Routing Header [RFC8138], SCHC [RFC8724], and SCHC for CoAP [RFC8824].

RFC 8724 defines the Rule concept, whereby a Rule may be used to support header compression or fragmentation functionality. In the present document, Rules are only used for header compression.

RFC 6775 defines the term 6LoWPAN Node (6LN) as the following: "A 6LoWPAN node is any host or router participating in a LoWPAN. This term is used when referring to situations in which either a host or router can play the role described." In this document, as in RFC 9008, 6LN acts as a leaf.

3. Architecture

3.1. Protocol stacks

3.1.1. Main protocol stack

The traditional 6LoWPAN-based protocol stack for constrained devices (Figure 1, left) places the 6LoWPAN adaptation layer between IPv6 and an underlying technology such as IEEE 802.15.4. Suitable upper layer protocols include CoAP [RFC7252] and UDP. (Note that, while CoAP has also been specified over TCP, and TCP may play a significant role in IoT environments [RFC9006], 6LoWPAN header compression has not been defined for TCP, as of the writing.)

6LoWPAN can be envisioned as a set of two main sublayers, where the upper one provides header compression, while the lower one offers fragmentation.

This document defines an alternative approach for packet header compression over IEEE 802.15.4, which leads to a modified protocol stack (Figure 1, right). Fragmentation functionality remains the one defined by 6LoWPAN [RFC4944] and 6Lo [RFC8930][RFC8931].

```
    +-----------+           +-----------+
    | CoAP, other|          | CoAP, other|
    +-----------+           +-----------+
    | UDP, other |          | UDP, other |
    +-----------+           +-----------+
    |    IPv6    |          |    IPv6    |
    +-----------+           +-----------+
    | 6LoWPAN HC |          |  SCHC HC   |  <-- NEW
    +-----------+           +-----------+
    |6LoWPAN Frag|          |6LoWPAN Frag|
    +-----------+           +-----------+
    |  802.15.4  |          |  802.15.4  |
    +-----------+           +-----------+
```

Figure 1: Traditional 6LoWPAN-based protocol stack over IEEE
802.15.4 (left) and alternative protocol stack using SCHC for
header compression (right). HC and Frag stand for Header
Compression and Fragmentation, respectively.

SCHC header compression may be applied to the headers of different
protocols or sets of protocols. Some examples include: i) IPv6
packet headers, ii) joint IPv6 and UDP packet headers, iii) joint
IPv6, UDP and CoAP packet headers, etc.

3.1.2. Transition protocol stacks

In order to ease a transition from existing 6LoWPAN implementations
to support SCHC header compression, the present document also: i)
illustrates two possible protocol stacks, where 6LoWPAN header
compression is used to compress IPv6/UDP headers while SCHC
compresses CoAP headers (see Section 5.1), and ii) enables the
transmission of SCHC-compressed UDP/CoAP headers over 6LoWPAN-
compressed IPv6 packets (see Section 5.2). However, note that the
greatest header compression performance can be achieved by using SCHC
to also compress the UDP header.

RFC 8824 defines how SCHC can be used to compress CoAP headers,
including Object Security for Constrained RESTful Environments
(OSCORE)-protected messages [RFC8613]. On the other hand, it is
possible to carry SCHC-compressed CoAP headers over UDP by means of
using SCHC UDP ports [I-D.ietf-intarea-schc-protocol-numbers].
Figure 2 (left) shows the resulting protocol stack, where 6LoWPAN
header compression is applied to UDP and IPv6. When Datagram
Transport Layer Security (DTLS) [RFC9147] is preferred to protect
SCHC-compressed CoAP messages, the DTLS layer sits between the SCHC
and UDP layers (Figure 2, right).

```
                                        +------------+
                                        |    CoAP    |
                  +------------+         +------------+
                  |    CoAP    |         |    SCHC    |
                  +------------+         +------------+
                  |    SCHC    |         |    DTLS    |
                  +------------+         +------------+
                  |    UDP     |         |    UDP     |
                  +------------+         +------------+
                  |    IPv6    |         |    IPv6    |
                  +------------+         +------------+
                  | 6LoWPAN HC |         | 6LoWPAN HC |
                  +------------+         +------------+
                  |6LoWPAN Frag|         |6LoWPAN Frag|
                  +------------+         +------------+
                  |  802.15.4  |         |  802.15.4  |
                  +------------+         +------------+
```
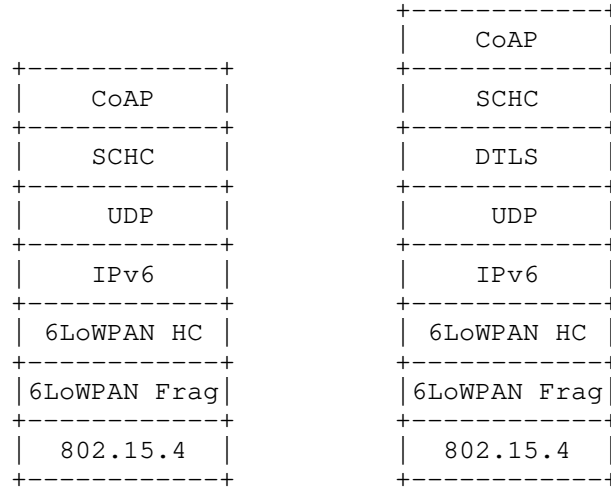
Figure 2: Transition protocol stacks where 6LoWPAN header
compression is applied to UDP and IPv6. The leftmost protocol
stack supports the use of OSCORE, whereas the rightmost one
corresponds to the use of DTLS to protect SCHC-compressed CoAP
messages.

Finally, the "transition" protocol stack enabled by this document,
which allows the transmission of 6LoWPAN-compressed IPv6 packets
containing SCHC-compressed UDP/CoAP data units, is shown in Figure 3
(rightmost).

```
                                                +------------+
                                                |    CoAP    |
   +------------+     +------------+             +------------+
   | CoAP, other|     | CoAP, other|            |    UDP     |
   +------------+     +------------+             +------------+
   | UDP, other |     | UDP, other |            |   SCHC HC  |   <-- NEW
   +------------+     +------------+             +------------+
   |    IPv6    |     |    IPv6    |             |    IPv6    |
   +------------+     +------------+             +------------+
   | 6LoWPAN HC |     |   SCHC HC  |  <-- NEW    | 6LoWPAN HC |
   +------------+     +------------+             +------------+
   |6LoWPAN Frag|     |6LoWPAN Frag|            |6LoWPAN Frag|
   +------------+     +------------+             +------------+
   |  802.15.4  |     |  802.15.4  |             |  802.15.4  |
   +------------+     +------------+             +------------+
```
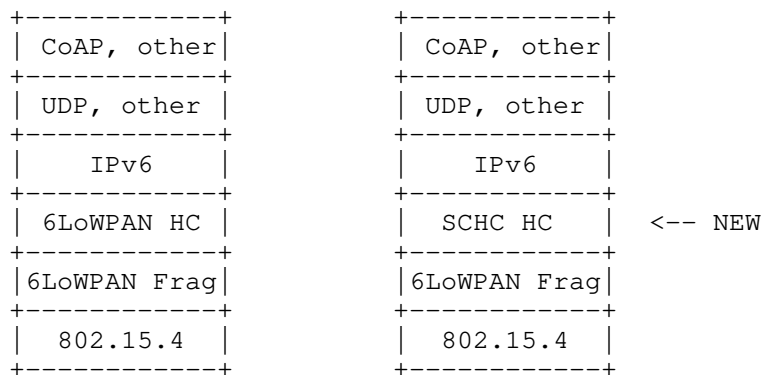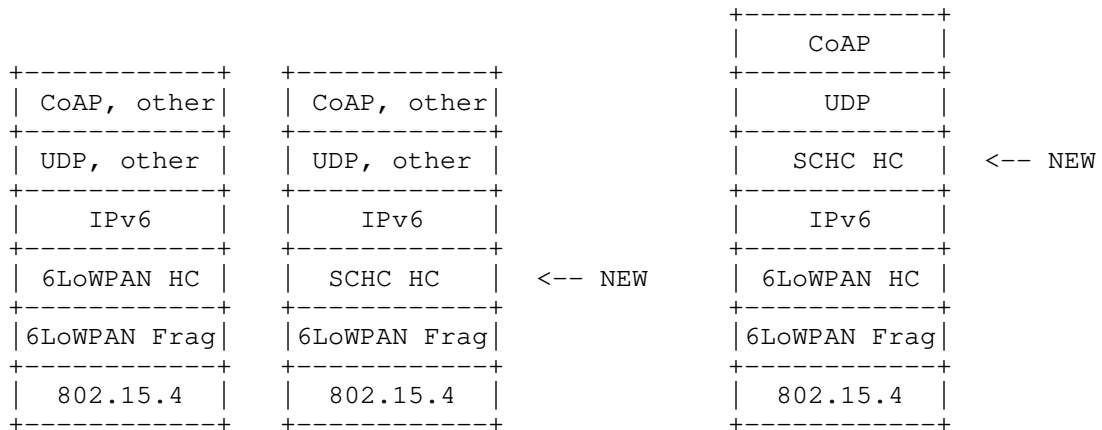
Figure 3: Traditional 6LoWPAN-based protocol stack over IEEE 802.15.4 (left), alternative protocol stack using SCHC for header compression (middle), and transition protocol stack using SCHC for header compression of UDP/CoAP headers (right). HC and Frag stand for Header Compression and Fragmentation, respectively.

3.2. Network topologies

IEEE 802.15.4 supports two main network topologies: the star topology, and the peer-to-peer (i.e., mesh) topology.

SCHC has been designed for LPWAN technologies, which are typically based on a star topology where constrained devices (e.g., sensors) communicate with a less constrained, central network gateway [RFC 8376]. However, as stated in [draft-ietf-schc-architecture], SCHC is generic and it can also be used in networking environments beyond the ones originally considered for SCHC.

SCHC compression is applicable to both star topology and mesh topology IEEE 802.15.4 networks. The mechanism to be used to provide the SCHC header compression context to the nodes in an IEEE 802.15.4 network is out of the scope of this document.

3.3. Single-hop communication

In order to support the transmission of SCHC-compressed packets between two endpoints that are single-hop neighbors, both endpoints MUST store the Rules intended for the communication between those two endpoints.

The frame format to be used to carry a SCHC-compressed packet in single-hop communication is described in Section 4.1.

3.4. Multihop communication

6LoWPAN defines two approaches for multihop communication: Route-Over and Mesh-Under [RFC6606]. In Route-Over, routing is performed at the IP layer. In Mesh-Under, routing functionality is located at the adaptation layer, below IP. This section describes how SCHC-compressed packets are transmitted over a multihop IEEE 802.15.4 network, for both Route-Over and Mesh-Under.

3.4.1. Straightforward Route-Over (SRO)

   SCHC header compression MAY be used in a Route-Over network in a
   straightforward approach, whereby all routers (i.e., all 6LRs and
   6LBRs) MUST store all the Rules in use by any nodes in the network,
   whereas a host MUST store the Rules defined for its communication
   with other endpoints. This approach is called Straightforward Route-
   Over (SRO). In this case, 6LoWPAN routers are able to decompress (if
   needed) received packet headers and compress packet headers before
   being forwarded. In SRO, a RuleID MUST NOT be reused across disjoint
   pairs of endpoints.

   Figure 4 illustrates an example network with the Rules that need to
   be stored by the nodes in SRO. In this example, RuleID 1 is intended
   for communication between Host A and Host B, RuleID 2 is intended for
   communication between Host A and Host C, and RuleID 3 is used for the
   communication between Host A and an external node called Host E.


```
                                     Host E
                                    /
                 (RuleID 1)       +--------+
                 (RuleID 2)   --- |Internet|
                 (RuleID 3)    /  +--------+
               6LBR ---------
              /       \
             /         \
          6LR          6LR ------------+              Pair of endpoints
    (RuleID 1) |         | (RuleID 1)   |        RuleID 1:      A, B
    (RuleID 2) |         | (RuleID 2)   |        RuleID 2:      A, C
    (RuleID 3) |         | (RuleID 3)   |        RuleID 3:      A, E
               |         |              |
          Host A     Host B        Host C
           (RuleID 1)  (RuleID 1)    (RuleID 2)
           (RuleID 2)
           (RuleID 3)
```
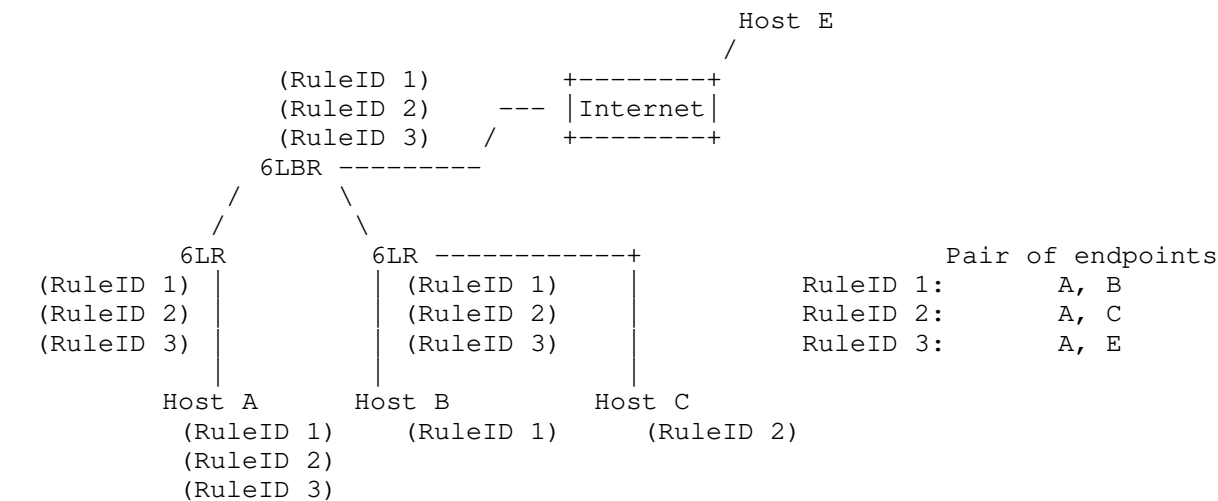
 Figure 4: Rules stored by each node in an example network using SRO.

   The frame format to be used to carry a SCHC-compressed packet in SRO
   is described in Section 4.1.

3.4.2. Tunneled, RPL-based Route-Over (TRO)

   In a Route-Over network that uses the IPv6 Routing Protocol for Low-
   Power and Lossy Networks (RPL) [RFC6550], the RPL non-storing mode
   [RFC6550, RFC 6554] and [RFC8138] MAY be exploited in order to
   efficiently transmit SCHC-compressed packets. In this approach,
   packets sent by a 6LN are tunneled to the root, and packets intended
   for 6LNs are tunneled from the root (note: a tunnel is not needed
   when the root itself is the source). Traffic between two 6LNs
   traverses an Upward tunnel to the root and a Downward tunnel from the
   root. The present document defines the described approach as
   Tunneled, RPL-based Route-Over approach (TRO).

   In TRO, each 6LoWPAN node (i.e., a host, a 6LR or a 6LBR) MUST store
   the Rules defined for its communication with other endpoints. A 6LR
   is thus relieved to store Rules used by pairs of endpoints that do
   not include the 6LR itself. A 6LBR MUST store all the Rules used by
   all nodes in the network.

   If all 6LNs in the 6LoWPAN network are RALs, a RuleID MAY be reused
   across disjoint pairs of endpoints, to identify different Rules used
   by such disjoint pairs of endpoints, at the expense of increased
   RuleID management complexity. Else, RuleIDs MUST NOT be reused
   across disjoint pairs of endpoints.

   Figure 5 illustrates the Rules that need to be stored by the nodes in
   TRO, based on the same example network and endpoint pairs shown in
   Figure 4.


```
                                        Host E
                                       /
               (RuleID 1)         +--------+
               (RuleID 2)    --- |Internet|
               (RuleID 3)   /     +--------+
            6LBR ---------
           /      \
          /        \
        6LR          6LR ------------+              Pair of endpoints
    (no Rules) |        | (no Rules)  |         RuleID 1:      A, B
              |        |             |          RuleID 2:      A, C
              |        |             |          RuleID 3:      A, E
              |        |             |
        Host A       Host B        Host C
         (RuleID 1)   (RuleID 1)    (RuleID 2)
         (RuleID 2)
         (RuleID 3)
```
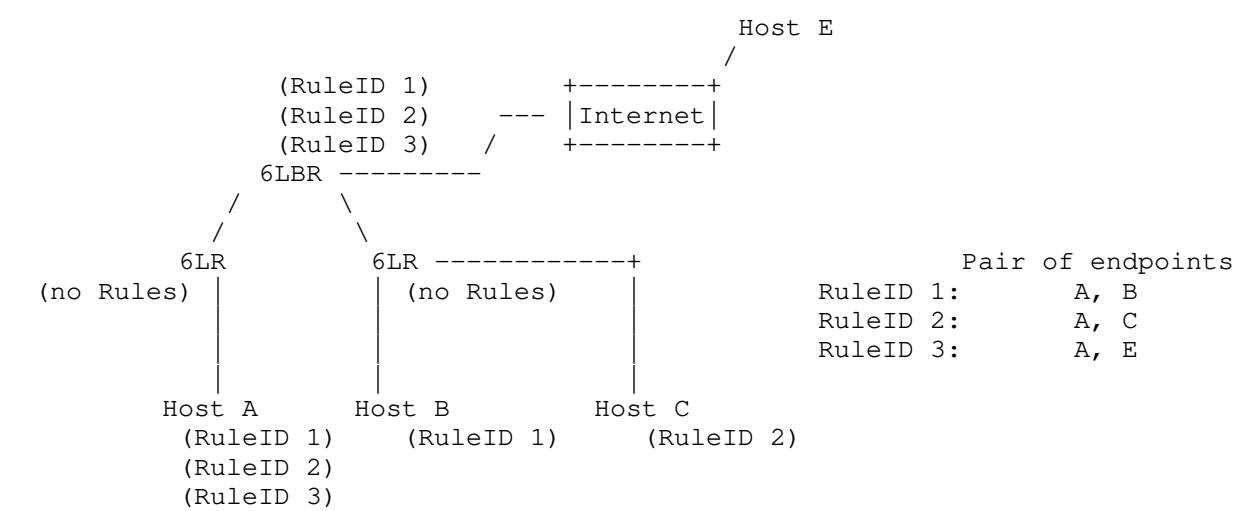
Figure 5: Rules stored by each node in an example network using TRO.

RFC 9008 describes how the communication between a 6LN and another endpoint (another 6LN or the root of the same RPL domain, or an external node, e.g., on the Internet) is performed. For the sake of description clarity, Figure 6 (adapted from Figure 3 in RFC 9008) provides a reference topology including nodes referred to in the remainder of this subsection.

```
                     +------------+
                     |  INTERNET  |---------+
                     +------------+         |
                                          Z |
                                   +-------+
                                   | 6LBR  |
                     +-----------|(root) |--------+
                     |             +-------+        |
                     |                              |
                   Y |                            X |
           +---|---+                      +---|---+
           |  6LR  |                      |  6LR  |
     +-------|       |--+           +--|       |--+
     |     +-------+   |           |   +-------+   |
     |               |           |               |
   W |             V |           |               |
 +---|---+       +---|---+       |               |
 |  6LR  |       |  6LR  |       |               |
 |       |       |       |       |               |
 +---|---+       +-|---|-+       |               |
     |               |           |               |
     |           +----+          |               |
   U |         T |    |     S     R |           Q |
 +-----+-+     +-------+   +---|--+     +---|---+     +---|---+
 |  RAL  |     |  RUL  |   |  RAL  |     |  RAL  |     |  RUL  |
 |  6LN  |     |  6LN  |   |  6LN  |     |  6LN  |     |  6LN  |
 +-------+     +-------+   +------+     +-------+     +-------+
```
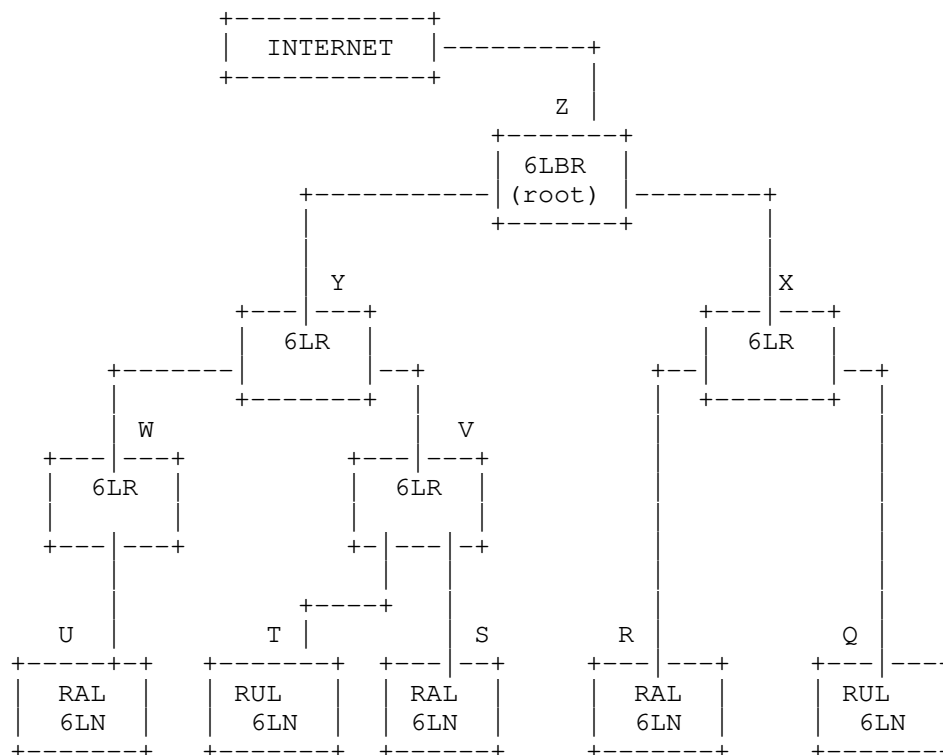
Figure 6: Reference topology to support the description of TRO.

In RPL non- storing mode, for Downward traffic, the root adds a source-routing header. The root also performs IPv6-in-IPv6 encapsulation, except when the root itself is the packet source. The IPv6-in-IPv6 encapsulation terminates at the 6LN (if it is a RAL, e.g., U, S or R) or at the last 6LR, e.g., V or X, (if the 6LN is a RUL, e.g., T or Q). For Upward traffic, IPv6-in-IPv6 encapsulation is performed by the first 6LR, e.g. V or X, when the 6LN is a RUL, e.g., T or Q, that sends a packet to an external node or to another 6LN in the same RPL domain, but not to the root. When the 6LN is a

RAL (e.g., U, S or R) that sends packets to the same destinations, IPv6-in-IPv6 encapsulation may be performed (by the RAL itself). The destination in the outer header of the IPv6-in-IPv6 encapsulation for Upward traffic is the root.

This document updates RFC 9008 by specifying that, in TRO, when a 6LN transmits an IPv6 packet whose header is compressed by means of SCHC instead of 6LoWPAN header compression (RFC 6282), the SCHC-compressed packet MUST be tunneled by means of IPv6-in-IPv6 encapsulation up to the root. This applies regardless of the inner, SCHC-compressed packet destination.

For Upward traffic, when the 6LN is a RAL (e.g., U, S or R), the 6LN itself performs the IPv6-in-IPv6 encapsulation. However, if the 6LN is a RUL (e.g., T or Q), IPv6-in-IPv6 encapsulation is performed by the first 6LR (e.g., E or C, respectively). In the latter case, in order to enable efficient packet transmission in the first hop from the 6LN, the first 6LR SHOULD be provided with SCHC Rules allowing efficient header compression of packets sent by that 6LN.

For Downward traffic, when the 6LN is a RUL (e.g., G or J), in order to enable efficient packet transmission in the last hop to the 6LN, the last 6LR (e.g., V or X, respectively) SHOULD be provided with SCHC Rules allowing efficient header compression of packets sent to that 6LN.

Not providing such SCHC Rules to the first or last 6LR (for Upward or Downward traffic, respectively) should only happen if it is not practical or possible to do so (e.g., due to lack of available memory at the 6LR).

For the sake of efficiency, RFC 8138 MUST be used to compress IPv6-in-IPv6 headers, the RPL Option (RFC 6553) and the source routing header (RPL Routing Header type 3, RFC 6554).

The frame format to be used to carry a SCHC-compressed packet in TRO is described in Section 4.3.

3.4.3. Pointer-based Route-Over (PRO)

In the previous approach, TRO, intermediate nodes do not have to know the IPv6 destination address of a SCHC-compressed IPv6 packet to be able to forward it. Another approach where intermediate nodes do not have to store the compression/decompression Rules used by the endpoints, which in addition does not require IPv6-in-IPv6 encapsulation, non-storing mode RPL and RFC 8138 compression, is called Pointer-based Route-Over (PRO).

In PRO, a pointer (called "SCHC Pointer") is prepended to the SCHC-compressed packet, in order to indicate the location and length of the Hop Limit and the destination address residues in the SCHC-compressed header. Therefore, a 6LR is able to determine the IPv6 destination address of a SCHC-compressed packet, decrement its Hop Limit and route the packet, without the need to store the corresponding Rules. Note that, in PRO, each 6LoWPAN node (i.e., a host, a 6LR, or a 6LBR) MUST store the Rules defined for its communication as an endpoint with other endpoints. A 6LBR MUST store the Rules used by any network node for communication with external nodes.

In PRO, a RuleID MAY be reused across disjoint pairs of endpoints. To identify different Rules used by such disjoint pairs of endpoints, the endpoint nodes store an additional identifier along with each RuleID and its corresponding Rule. This identifier may be the IPv6 address of the other endpoint or a SCHC Header session ID [I-D.ietf-schc-architecture].

Figure 7 illustrates the Rules that are stored by the nodes in an example network based on PRO. Note that, in this example, the network exploits the fact that PRO allows a given RuleID to be reused by disjoint pairs of endpoints.

```
                                            Host E
                                           /
                                +--------+- Host F
                 (RuleID 3, AE)  --- |Internet|
                 (RuleID 3, BF)  /    +--------+
                 6LBR -----------
               /       \
              /         \
           6LR           6LR ------------+               Pair of endpoints
   (no Rules)/|          | (no Rules)    |          RuleID 1:      A, B
           / |          |               |          RuleID 2:      A, C
          / |          |               |          RuleID 2:      D, B
         / |          |               |          RuleID 3:      A, E
    Host D  Host A     Host B          Host C      RuleID 3:      B, F
  (RuleID 2)  (RuleID 1)  (RuleID 1)     (RuleID 2)
            (RuleID 2)  (RuleID 2)
            (RuleID 3)  (RuleID 3)
```
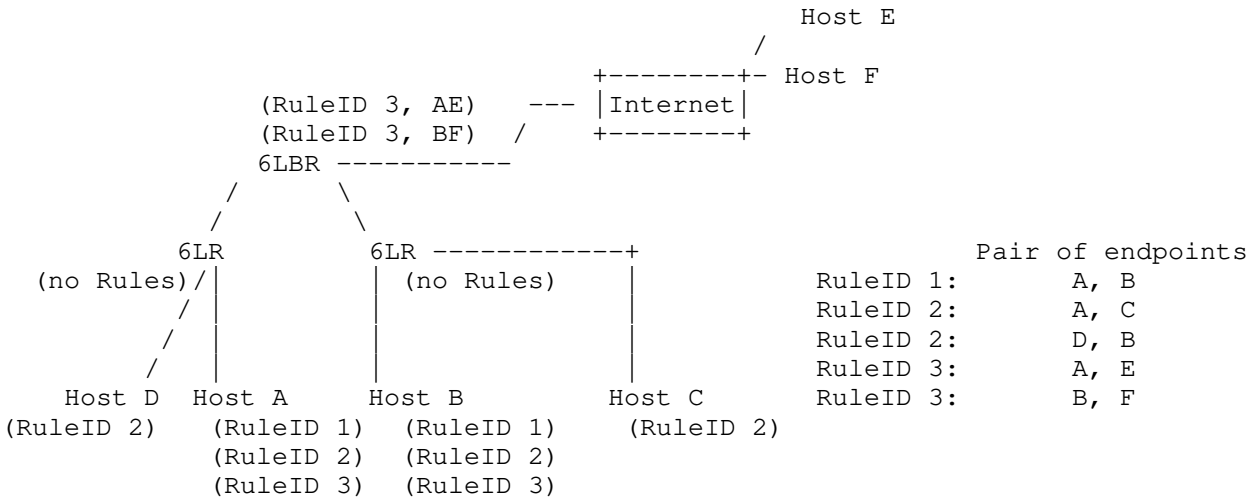
Figure 7: In this example, RuleID 2 and RuleID 3 are used by two
             disjoint pairs of endpoints.

PRO is compatible with RPL storing mode, as well as with other routing protocols.

3.4.4. Mesh-Under

   When SCHC header compression is used in a Mesh-Under network, Mesh-
   Under operates as described in RFC 4944. The frame format to be used
   to carry a SCHC-compressed packet in the Mesh-Under approach is
   described in Section 4.3.

   For header compression in a Mesh-Under network, a network node MUST
   store the Rules defined for its communication with other endpoints.

   In this case, a RuleID MAY be reused across disjoint pairs of
   endpoints, to identify different Rules used by such disjoint pairs of
   endpoints.

   Figure 8 illustrates the Rules that need to be stored by the nodes
   when SCHC is used for header compression in a Mesh-under network,
   based on the same example network and endpoint pairs shown in
   Figure 7. Note that, in this example, the network exploits the fact
   that Mesh-under allows a given RuleID to be reused by disjoint pairs
   of endpoints, even if the Rules sharing the same RuleID are
   different. As in PRO, these Rules are distinguished by an
   identifier, which may be the IPv6 address of the other endpoint or a
   SCHC Header session ID [I-D.ietf-schc-architecture]. Nodes denoted
   "m" in Figure 8 correspond to Mesh-Under forwarders [RFC 6606].

```
                                            Host E
                                             /
                              +--------+- Host F
               (RuleID 3, AE)   --- |Internet|
               (RuleID 3, BF)  /    +--------+
               6LBR -----------
              /      \
             /        \
            m          m --------------+              Pair of endpoints
    (no Rules)/|        | (no Rules)   |          RuleID 1:      A, B
            /  |        |              |          RuleID 2:      A, C
           /   |        |              |          RuleID 2:      D, B
          /    |        |              |          RuleID 3:      A, E
       Host D  Host A    Host B        Host C     RuleID 3:      B, F
      (RuleID 2)  (RuleID 1)  (RuleID 1)  (RuleID 2)
                  (RuleID 2)  (RuleID 2)
                  (RuleID 3)  (RuleID 3)
```
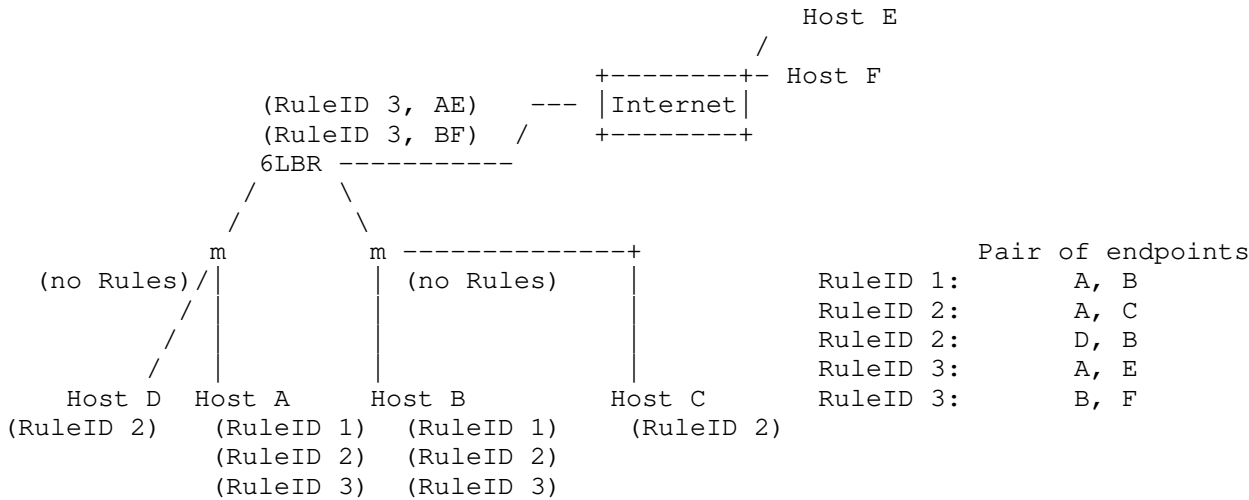
   Figure 8: Rules stored by each node in an example network using
   Mesh-Under. In this example, RuleID 2 and RuleID 3 are used by
                     disjoint pairs of endpoints.

4. Frame Format

   This section defines the frame formats that can be used when a SCHC-
   compressed packet is carried over IEEE 802.15.4. Such formats are
   carried as IEEE 802.15.4 frame payload.

   TO-DO: align, if needed, with current SCHC WG discussion regarding
   SCHC headers.

4.1. Single-hop or SRO frame format

   This subsection defines the frame format for carrying SCHC-compressed
   packets over IEEE 802.15.4 for single-hop communication (see 3.3) or
   when SRO is used for multihop communication (see 3.4.1). This format
   comprises a SCHC Dispatch Type, a SCHC Packet (i.e. a SCHC-compressed
   packet (RFC 8724), and Padding bits, if any). Figure 9 illustrates
   the described frame format.

```
          <---------- IEEE 802.15.4 frame payload ---------->

                        <------ SCHC Packet ----->
          +--------------+--------------+---------+ - - - - +
          | SCHC Dispatch | Cmprd Header | Payload | Padding |
          +--------------+--------------+---------+ - - - - +
```
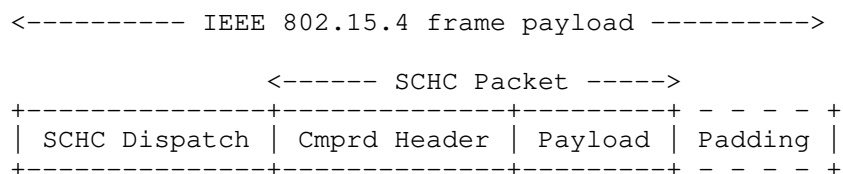
   Figure 9: Encapsulated, SCHC-compressed packet, for single-hop or SRO
             transmission. Padding bits are added if needed.

4.1.1. SCHC Dispatch

   Adding SCHC header compression to the panoply of header compression
   mechanisms used in 6LoWPAN/6Lo environments creates the need to
   signal when a packet header has been compressed by using SCHC. To
   this end, the present document specifies the SCHC Dispatch. The SCHC
   Dispatch indicates that the next field in the frame format is a SCHC-
   compressed header (SCHC Header in Figure 9, see 4.1.2)).

   This document defines the SCHC Dispatch as a 6LoWPAN Dispatch Type
   for SCHC header compression [RFC4944]. With the aim to minimize
   overhead, the present document allocates a 1-byte pattern in Page 0
   [RFC8025] for the SCHC Dispatch Type:

   SCHC Dispatch Type bit pattern: 01000100 (Page 0) (Note: to be
   confirmed by IANA))

4.1.2. SCHC-compressed Header

   The SCHC-compressed Header ("Cmprd Header" in Figure 9) corresponds
   to a packet header that has been compressed by using SCHC. As
   defined in [RFC8724], a SCHC-compressed header comprises a RuleID,
   and a compression residue. As per the present specification, a
   RuleID size between 1 and 16 bits is RECOMMENDED. In order to decide
   the RuleID size to be used in a network, the trade-off between
   (compressed) header overhead and the number of Rules needs to be
   carefully assessed.

4.1.3. Padding

   If SCHC header compression leads to a SCHC Packet size of a non-
   integer number of bytes, padding bits of value equal to zero MUST be
   appended to the SCHC Packet as appropriate to align to an octet
   boundary.

4.2. TRO frame format

   This subsection defines the frame formats for carrying SCHC-
   compressed packets over IEEE 802.15.4 in TRO (see 3.3.2). Such
   formats are based on RFC 8138; however, instead of RFC 6282 header
   compression, this specification uses SCHC header compression.
   Accordingly, this specification updates RFC 8138 by stating that a
   6LoRH header MUST always be placed before the LOWPAN_IPHC as defined
   in RFC 6282 [RFC6282] or the SCHC Dispatch, followed by the SCHC-
   compressed packet, as defined in the present specification.

   Since 6LoRH uses Dispatch Types in Page 1, the present specification
   also defines a SCHC Dispatch Type in Page 1, with the same bit
   pattern as the one in Page 0: 01000100 (to be confirmed by IANA).

   In the TRO frame formats, the SCHC-compressed header is preceded by
   the SCHC Dispatch (in this case, in Page 1).

   The frame format for Downward transmission, except when the SCHC-
   compressed packet source is a RPL root, is shown in Figure 10:

```
        <---------------- IEEE 802.15.4 frame payload -------------------->

                                            <- SCHC pkt ->
   +-- ... -+-- ... --+- ... -+--- ... --+---- ... -+-----+------+ - - +
   |11110001|SRH-6LoRH| RPI-  | IP-in-IP | 01000100 |Cmprd|payload| pad |
   |Page 1  |         | 6LoRH |  6LoRH   |SCHCDsptch| Hdr |       |     |
   +-- ... -+-- ... --+- ... -+--- ... --+---- ... -+-----+------+ - - +
                                         (Page 1)

                                    <----- This specification ----->
```
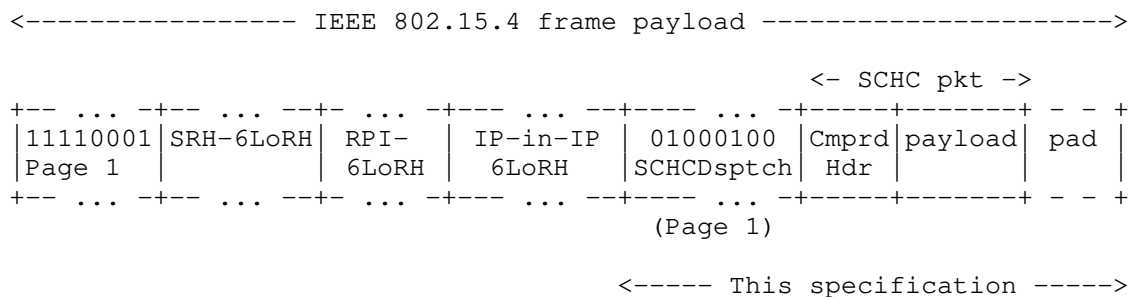
        Figure 10: Downward frame format for SCHC-compressed packets in
                  TRO, when the source is not a RPL root.

   The frame format for Downward transmission, when the SCHC-compressed
   packet source is a RPL root, is shown in Figure 11:

```
         <-------------- IEEE 802.15.4 frame payload -------------->

                                       <- SCHC pkt ->
    +-- ... -+-- ... --+- ... -+---- ... -+-----+------+ - - +
    |11110001|SRH-6LoRH| RPI-  | 01000100 |Cmprd|payload| pad |
    |Page 1  |         | 6LoRH |SCHCDsptch| Hdr |       |     |
    +-- ... -+-- ... --+- ... -+---- ... -+-----+------+ - - +
                        (Page 1)

                              <----- This specification ----->
```
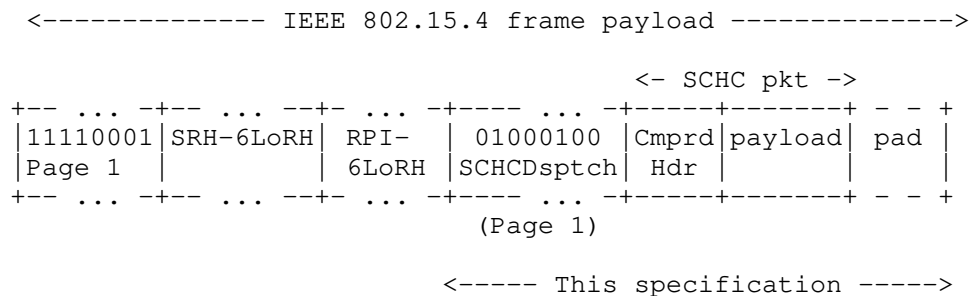
         Figure 11: Downward frame format for SCHC-compressed packets in
                  TRO, when the source is a RPL root.

   The frame format for Upward transmission is shown in Figure 12 (note
   that it does not include the source routing header that is present in
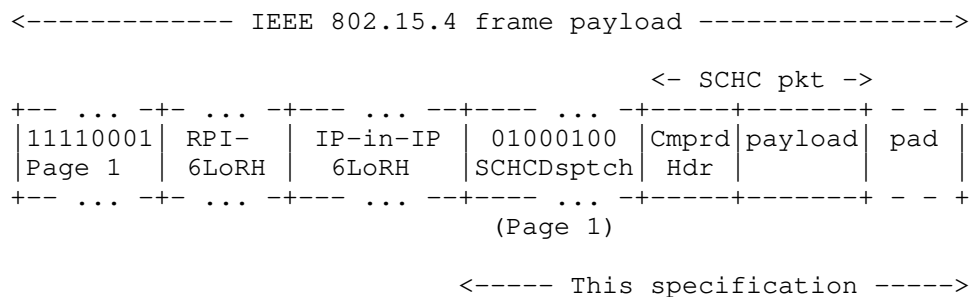   the Downward frame format):

```
         <------------- IEEE 802.15.4 frame payload --------------->

                                            <- SCHC pkt ->
      +-- ... -+- ... -+--- ... --+---- ... -+-----+-------+ - - +
      |11110001| RPI-  | IP-in-IP | 01000100 |Cmprd|payload| pad |
      |Page 1  | 6LoRH |   6LoRH  |SCHCDsptch| Hdr |       |     |
      +-- ... -+- ... -+--- ... --+---- ... -+-----+-------+ - - +
                                    (Page 1)

                                    <----- This specification ----->
```

     Figure 12: Upward frame format for SCHC-compressed packets in TRO.

4.3. PRO frame format

   This subsection describes the frame format for carrying SCHC-
   compressed packets over IEEE 802.15.4 in PRO (see 3.3.3). Such
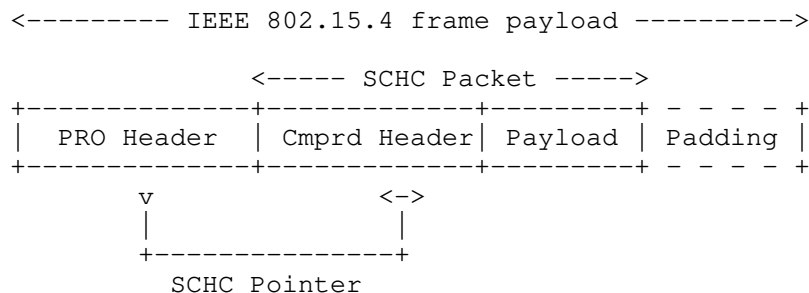   format is shown in Figure 13:

```
         <--------- IEEE 802.15.4 frame payload ---------->

                    <----- SCHC Packet ----->
         +--------------+-------------+---------+ - - - - +
         | PRO Header   | Cmprd Header| Payload | Padding |
         +--------------+-------------+---------+ - - - - +
                 v                 <->
                 |                  |
                 +--------------+
                   SCHC Pointer
```

     Figure 13: frame format for SCHC-compressed packets in PRO.

   The PRO Header format is shown in Figure 14:

```
         0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3
        +--------------+-+ - - - +-------------+-+-------------+
        |     SCHC     |C|       |             |H|             |
        |   Pointer    |I|  DCI  | Bit Pointer |L|   Address   |
        |   Dispatch   |D|       |             |M|   Length    |
        +--------------+-+ - - - +-----------+-+-+-----------+
```
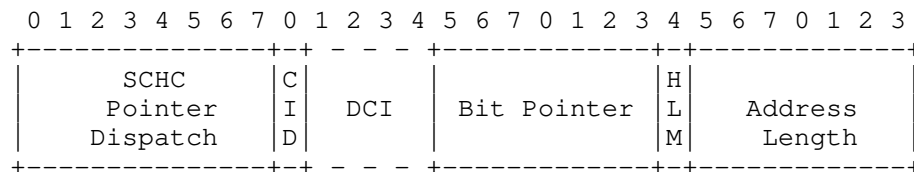
     Figure 14: PRO Header format.
```

The first field in Figure 14 is defined as the SCHC Pointer Dispatch, which signals the start of a PRO Header format.  This document defines the SCHC Pointer Dispatch as a 6LoWPAN Dispatch Type [RFC4944] for SCHC header compression.

With the aim to minimize header overhead, the present document allocates a 1-byte pattern in the 6LoWPAN Dispatch Type Page 0 [RFC8025] for the SCHC Pointer Dispatch Type:

SCHC Pointer Dispatch Type bit pattern: 01000101 (Page 0) (Note: to be confirmed by IANA))

The next field in the PRO Header is the Context IDentifier (CID) flag, which is set to 1 to signal that the Destination Context Identifier (DCI) field (see PRO_header_format) is present in the frame.  When CID is set to 0, the DCI field is not present.

The DCI field is optional.  When present, it has a size of 4 bits. Similarly to RFC 6282, this field identifies the prefix of the IPv6 destination address.  How such prefix context is distributed and maintained is out of the scope of the present document.

The Bit pointer gives the starting position of the Hop Limit followed by the IPv6 destination address in the SCHC residue of the SCHC-compressed IPv6 header (in bits), starting after the Address Length field and before the first field of the SCHC-compressed IPv6 header (i.e., the RuleID).  For example, if the Hop Limit and the IPv6 destination address residue are the only residues in a SCHC-compressed IPv6 packet header (i.e., such residue starts right after the RuleID in the SCHC-compressed header), then the Bit pointer will have a value of RuleID length in bits.

The Hop Limit (HLM) flag is 1 bit that indicates the length of the Hop Limit field residue in the SCHC-compressed IPv6 header.  When HLM equals 0, the Hop Limit compression residue has a size of 4 bits.  In this case, the 4 most significant bits of the uncompressed Hop Limit field are equal to 0.  Therefore, Hop Limit compression applies only to Hop Limit values between 15 and 0.  When HLM is set to 1, the Hop Limit compression residue has a size of 8 bits (i.e., it is uncompressed).

Address Length indicates the size of the IPv6 destination address residue (in bits).  It can be up to 128 bits to allow representing the complete destination address, if needed.

PRO requires a special SCHC Rule design where the FIDs of the IPv6 Destination and Source addresses are swapped (see 6.1.1).

## 4.4.  Mesh-Under frame format

This subsection describes the frame formats for carrying SCHC-compressed packets over IEEE 802.15.4 in the Mesh-Under approach (see 3.3.3).  Note that the formats are provided in this section for the sake of clarity and completeness, since they are the same as those in RFC 4944, except for the fact that SCHC-compressed packets are carried.

The frame format for a SCHC-compressed packet to be sent by means of Mesh-Under, when fragmentation is not needed, is shown in Figure 15:

```
  <------------------- IEEE 802.15.4 frame payload -------------------->

                                          <----- SCHC Packet ----->
  +----------+------------+--------------+------------+--------+ - - +
  | Mesh Type | Mesh Header | SCHC Dispatch | Cmprd Header| Payload | pad |
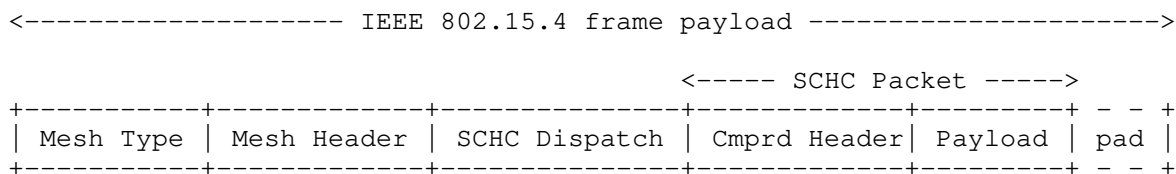  +----------+------------+--------------+------------+--------+ - - +
```

Figure 15: Encapsulated, SCHC-compressed packet, for Mesh-Under transmission (without fragmentation). Padding bits are added if needed.

The frame format for a SCHC-compressed packet to be sent by means of Mesh-Under, which also requires fragmentation, is shown in Figure 16:

```
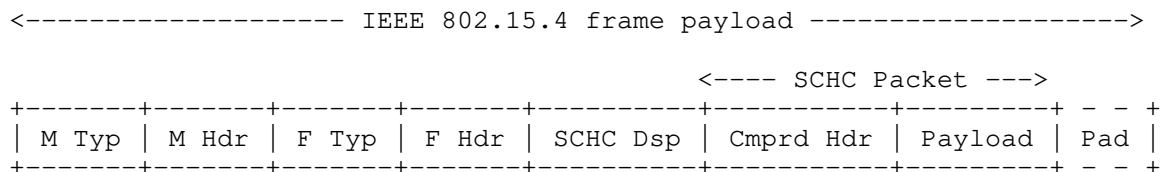  <------------------- IEEE 802.15.4 frame payload ------------------->

                                        <---- SCHC Packet --->
  +-------+-------+-------+-------+----------+----------+--------+ - - +
  | M Typ | M Hdr | F Typ | F Hdr | SCHC Dsp | Cmprd Hdr | Payload | Pad |
  +-------+-------+-------+-------+----------+----------+--------+ - - +
```

Figure 16: Encapsulated, SCHC-compressed packet, for Mesh-Under transmission (with fragmentation). Padding bits are added if needed.

The frame format for a SCHC-compressed packet to be sent by means of Mesh-Under, which also requires a broadcast header to support mesh broadcast/multicast, is shown in Figure 17:

```
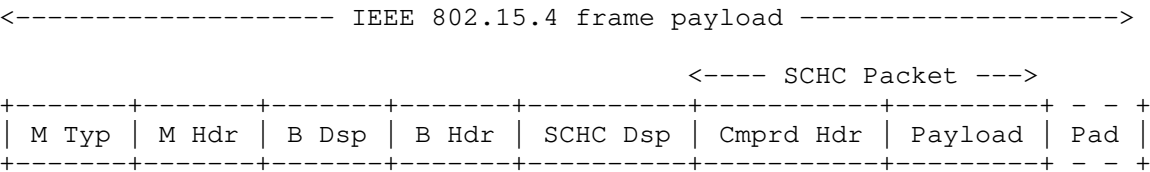<------------------- IEEE 802.15.4 frame payload ------------------->

                                            <---- SCHC Packet --->
+-------+-------+-------+-------+----------+-----------+---------+ - - +
| M Typ | M Hdr | B Dsp | B Hdr | SCHC Dsp | Cmprd Hdr | Payload | Pad |
+-------+-------+-------+-------+----------+-----------+---------+ - - +
```

    Figure 17: Encapsulated, SCHC-compressed packet, for mesh
      broadcast/multicast in Mesh-Under transmission (without
fragmentation). Padding bits are added if needed. 'B Dsp' and
 'B Hdr' stand for 'Broadcast Dispatch' and 'Broadcast Header',
                          respectively.

   As in RFC 4944, when more than one LoWPAN header is used in the same
   packet, they MUST appear in the following order: Mesh Addressing
   Header, Broadcast Header, Fragmentation Header.

4.5. Summary

   The different transmission alternatives enabled by the present
   document are shown in Figure 18:

```
+-------------+-----------------------------------------------------------+
| Single-hop  |                     Multihop                              |
+-------------+-------------------------------------------+---------------+
|             |                 Route-Over                |               |
|             +-----------+---------------+---------------+ Mesh-Under    |
|             |    SRO    |     TRO       |     PRO       |               |
+-------------+-----------+---------------+---------------+---------------+
|SCHC Dispatch| SCHC Disp |IP-in-IP, 6LoRH,|SCHC Ptr Disp,| Mesh Headers, |
|             |           | SCHC Dispatch | SCHC Pointer  | SCHC Dispatch |
+-------------+-----------+---------------+---------------+---------------+
|   see 4.1   |  see 4.1  |    see 4.2    |    see 4.3    |    see 4.4    |
+-------------+-----------+---------------+---------------+---------------+
```

   Figure 18: Summary of alternatives for the transmission of SCHC-
     compressed packets over IEEE 802.15.4 enabled by the present
                  document, and corresponding artifacts

5. Enabling the transition protocol stack

   In order to enable the transition protocol stack, (i.e., supporting
   SCHC-compressed UDP/CoAP headers over 6LoWPAN-compressed IPv6
   packets), the present document exploits the work that is being done
   by the INTAREA WG, to define a new Internet Protocol Number for SCHC
   [I-D.ietf-intarea-schc-protocol-numbers]. In this approach, the NH
   field of the RFC 6282-compressed IPv6 header format is set to 0. The
   Next Header field of the IPv6 header remains an 8-bit (uncompressed)
   field carrying the SCHC Internet Protocol Number. The resulting
   protocol encapsulation and corresponding format for an unfragmented
   packet, which is carried as IEEE 802.15.4 frame payload, is shown in
   Figure 19. Padding is added as needed to align the format to an
   octet boundary.

```
     <---------------- IEEE 802.15.4 frame payload ------------------>
     +---------------------+----------------+-------------+ - - +
     | RFC6282-compressed  |                |             |     |
     |      IPv6 header     | SCHC-compressed | CoAP Payload | Pad |
     |(NH=0,Next Header=SCHC)| UDP/CoAP headers |             |     |
     +---------------------+----------------+-------------+- - -+
```

     Figure 19: Protocol data unit encapsulation and format for the
     transition protocol stack using a SCHC Internet Protocol Number

   For networks using the transition protocol stack based on RPL
   routing, the formats defined in RFC 8138 may also be used for the
   sake of efficiency, as shown in Figure 20. In this figure, the first
   field is the Page switch with value 1, followed by RFC
   8138-compressed routing artifacts, then followed by the RFC
   6282-compressed IPv6 header (which indicates that the next header
   data unit is a SCHC Packet).

```
    <----------------------- IEEE 802.15.4 frame payload ---------------------
 ->
    +--------+-----------+-----------------+--------------+-------------+ - -
 +
    |11110001|8138-cmprssd|  6282-compressed |              |             |
 |
    |(Page 1)|  routing   |   IPv6 header    |SCHC-compressed| CoAP Payload | Pad
 |
    |        | artifacts  |(NH=0,NxtHdr=SCHC)| UDP/CoAP hdrs |             |
 |
    +--------+-----------+-----------------+--------------+-------------+ - -
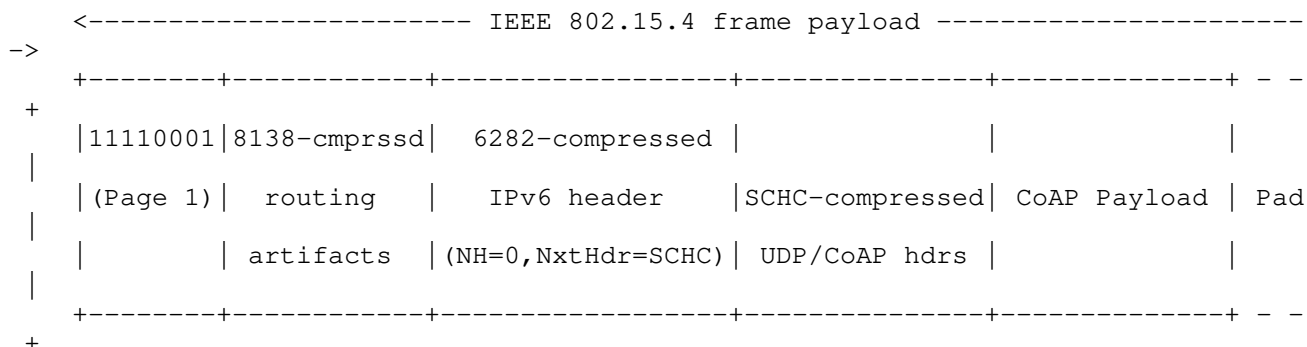 +
```

     Figure 20: Protocol data unit encapsulation and format for the
     transition protocol stack using a SCHC Internet Protocol Number
              and RFC 8138-compressed routing artifacts

6. SCHC compression for IPv6, UDP, and CoAP headers

   SCHC header compression may be applied to the headers of different
   protocols or sets of protocols. Some examples include: i) IPv6
   packet headers, ii) joint IPv6 and UDP packet headers, iii) joint
   IPv6, UDP and CoAP packet headers, etc.

   Each Rule defines the set of protocols whose headers are compressed.
   For example, in a given deployment, RuleIDs 1 to 3 may be defined for
   IPv6 header compression only, RuleIDs 4 to 7 may be used for IPv6/UDP
   header compression, and RuleIDs 8 to 15 may be used for IPv6/UDP/CoAP
   header compression.

   This section describes how IPv6, UDP, and CoAP header fields are
   compressed.

6.1. SCHC compression for IPv6 and UDP headers

   IPv6 and UDP header fields MUST be compressed as per Section 10 of
   RFC 8724.

   IPv6 addresses are split into two 64-bit-long fields; one for the
   prefix and one for the Interface Identifier (IID).

   To allow for a single Rule being used for both directions, RFC 8724
   identifies IPv6 addresses and UDP ports by their role (Dev or App)
   and not by their position in the header (source or destination).
   This optimization can be used as is in some IEEE 802.15.4 networks
   (e.g., an IEEE 802.15.4 star topology where the peripheral devices
   (Devs) send/receive packets to/from a network-side entity (App)).

   However, in some types of 6LoWPAN environments (e.g., when a sender
   and its destination are both peer nodes in a mesh topology network),
   additional functionality is needed to allow use of the Dev and App
   roles for C/D. In this case, each SCHC C/D entity needs to know its
   role (Dev or App) in addition to the Rule(s), and corresponding
   RuleIDs, for each endpoint it communicates with before such
   communication occurs [I-D.ietf-schc-architecture]. In such cases,
   the terms Uplink and Downlink that have been defined in RFC 8724 need
   to be understood in the context of each specific pair of endpoints.

RFC 8724 (Section 7.1) states that "In a Rule, the Field Descriptors are listed in the order in which the fields appear in the packet header". The present specification updates RFC 8724 to state that, in order to allow IPv6 header compression in PRO, the Field Descriptors of the IPv6 destination address (i.e., IPv6 DevPrefix and IPv6 DevIID) MUST appear before the Field Descriptors of the IPv6 source address (i.e., IPv6 AppPrefix and IPv6 AppIID), while the rest of fields appear in the same order as in the IPv6 packet header.

In PRO, in order to support IPv6 header compression, one Rule MUST be defined for each direction between the two involved C/D endpoints. In such a Rule, the IPv6 DevPrefix and IPv6 DevIID FIDs MUST refer to the destination address (i.e., the destination endpoint takes the "Dev" role) of the SCHC-compressed IPv6 header. This allows a 6LR to read the compression residue of the Hop Limit and IPv6 destination address fields of the SCHC-compressed header by means of the Bit Pointer.

6.1.1. Compression of IPv6 addresses

Compression of IPv6 source and destination prefixes MUST be performed as per Section 10.7.1 of RFC 8724. Additional guidance is given in the present section.

Compression of IPv6 source and destination IIDs MUST be performed as per Section 10.7.2 of RFC 8724. One particular consideration when SCHC C/D is used in IEEE 802.15.4 networks is that, in contrast with some LPWAN technologies, IEEE 802.15.4 data frame headers include both source and destination fields. If the Dev or App IID are based on an L2 address, in some cases the IID can be reconstructed with information coming from the L2 header. Therefore, in those cases, DevIID and AppIID CDAs can be used.

RFC 8724 states that "If the Rule is intended to compress packets with different prefix values, match-mapping SHOULD be used" (Section 10.7.1 of RFC 8724) and "If several IIDs are possible, then the TV contains the list of possible IIDs, the MO is set to "match-mapping" and the CDA is set to "mapping-sent"" (Section 10.7.2 of RFC 8724). However, in PRO, a source node MUST NOT use the match-mapping operator or the "mapping-sent" CDA to compress the IPv6 destination address prefix or the IPv6 destination IID, because 6LRs do not store SCHC context, and therefore do not have the match-mapping index meaning information.

6.1.2. UDP checksum field

   RFC 8724 states that "a SCHC compressor MAY elide the UDP checksum
   when another layer guarantees at least equal integrity protection for
   the UDP payload and the pseudo-header".

   IEEE 802.15.4 frames carry a 16-bit Frame Check Sequence (FCS), which
   is computed by means of a 16-bit ITU-T CRC algorithm. Considering
   the FCS size, the greater error detection capabilities of CRC
   compared with checksum, and the fact that the IEEE 802.15.4 FCS will
   be checked at each hop in an IEEE 802.15.4 multihop network, the UDP
   checksum MUST be elided when using SCHC to compress UDP headers.

6.2. SCHC compression for CoAP headers

   CoAP header fields MUST be compressed as per Sections 4 to 6 of RFC
   8824. Additional guidance is given in this section.

   For CoAP header compression/decompression, the SCHC Rules description
   uses direction information in order to reduce the number of Rules
   needed to compress headers.

   As stated in 5.1, in some types of 6LoWPAN environments (e.g., when a
   sender and its destination are both peer nodes in a mesh topology
   network), each SCHC C/D entity needs to know its role (Dev or App),
   in addition to the Rule(s), and corresponding RuleIDs, for each
   endpoint it communicates with before such communication occurs
   [I-D.ietf-schc-architecture]. Therefore, in such cases, direction
   information will be specific to each pair of endpoints.

7. Neighbor Discovery

   A number of optimizations have been developed in order to efficiently
   support IPv6 Neighbor Discovery (ND) in 6LoWPAN environments (6LoWPAN
   ND) [RFC 6775][RFC 8505]. SCHC can also be used to compress 6LoWPAN
   ND packets. At the time of this writing, compression of ICMPv6 or
   ICMPv6-based protocols has not been specified. Therefore, currently,
   only the IPv6 header of a packet carrying a 6LoWPAN ND message can be
   compressed. Nevertheless, future specifications may define how
   ICMPv6 and 6LoWPAN ND messages can be compressed. (Note: the charter
   of the new IETF SCHC WG includes the development of "ICMPv6-based
   protocols" over SCHC as a potential work item.)

8. Fragmentation and reassembly

   After applying SCHC header compression to a packet intended for
   transmission, if the size of the resulting SCHC Packet (Section 4)
   exceeds the IEEE 802.15.4 frame payload space available, such SCHC
   Packet MUST be fragmented, carried and reassembled by means of the
   fragmentation and reassembly functionality defined by 6LoWPAN
   [RFC4944] or 6Lo [RFC8930][RFC8931].

   In a Route-Over multihop network, the 6LoWPAN fragment forwarding
   technique called Virtual Reassembly Buffer (VRB) [RFC8930] SHOULD be
   used. However, VRB might not be the best approach for a particular
   network, e.g., if at least one of the caveats described in Section 6
   of RFC 8930 is unacceptable or cannot be addressed.

9. IANA Considerations

   This document requests the allocation of the 6LoWPAN Dispatch Type
   Field bit pattern 01000100 (in Pages 0 and 1) as SCHC Dispatch Type.

   This document also requests the allocation of the 6LoWPAN Dispatch
   Type Field bit pattern 01000101 (in Page 0) as SCHC Pointer Dispatch
   Type.

10. Security Considerations

   This document does not define SCHC header compression functionality
   beyond the one defined in RFC 8724. Therefore, the security
   considerations in section 12.1 of RFC 8724 and in section 9 of RFC
   8824 apply.

   As a safety measure, a SCHC decompressor implementing the present
   specification MUST NOT reconstruct a packet larger than 1500 bytes
   [RFC8724].

   IEEE 802.15.4 networks support link-layer security mechanisms such as
   encryption and authentication. As in RFC 8824, the use of a
   cryptographic integrity-protection mechanism to protect the SCHC
   headers is REQUIRED.

11. Acknowledgments

   Ana Minaburo and Laurent Toutain suggested for the first time the use
   of SCHC in environments where 6LoWPAN has traditionally been used.
   Flavien Moullec is a contributor to this document. Laurent Toutain,
   Pascal Thubert, Dominique Barthel, Guangpeng Li, Carsten Bormann,
   Nathan Lecorchet, Stuart Cheshire, Kiran Makhijani, and Georgios Z.
   Papadopoulos made comments that helped shape this document.

12. References

12.1. Normative References

[I-D.ietf-intarea-schc-protocol-numbers]
          Moskowitz, R., Card, S. W., Wiethuechter, A., and P.
          Thubert, "Protocol Numbers for SCHC", Work in Progress,
          Internet-Draft, draft-ietf-intarea-schc-protocol-numbers-
          01, 12 October 2023,
          <https://datatracker.ietf.org/doc/html/draft-ietf-intarea-
          schc-protocol-numbers-01>.

[I-D.ietf-schc-architecture]
          Pelov, A., Thubert, P., and A. Minaburo, "Static Context
          Header Compression (SCHC) Architecture", Work in Progress,
          Internet-Draft, draft-ietf-schc-architecture-01, 6 October
          2023, <https://datatracker.ietf.org/doc/html/draft-ietf-
          schc-architecture-01>.

[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>.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler,
          "Transmission of IPv6 Packets over IEEE 802.15.4
          Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007,
          <https://www.rfc-editor.org/info/rfc4944>.

[RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6
          Datagrams over IEEE 802.15.4-Based Networks", RFC 6282,
          DOI 10.17487/RFC6282, September 2011,
          <https://www.rfc-editor.org/info/rfc6282>.

[RFC6550] Winter, T., Ed., Thubert, P., Ed., 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,
          DOI 10.17487/RFC6550, March 2012,
          <https://www.rfc-editor.org/info/rfc6550>.

   [RFC6553] Hui, J. and JP. Vasseur, "The Routing Protocol for Low-
              Power and Lossy Networks (RPL) Option for Carrying RPL
              Information in Data-Plane Datagrams", RFC 6553,
              DOI 10.17487/RFC6553, March 2012,
              <https://www.rfc-editor.org/info/rfc6553>.

   [RFC6554] Hui, J., Vasseur, JP., Culler, D., and V. Manral, "An IPv6
              Routing Header for Source Routes with the Routing Protocol
              for Low-Power and Lossy Networks (RPL)", RFC 6554,
              DOI 10.17487/RFC6554, March 2012,
              <https://www.rfc-editor.org/info/rfc6554>.

   [RFC6606] Kim, E., Kaspar, D., Gomez, C., and C. Bormann, "Problem
              Statement and Requirements for IPv6 over Low-Power
              Wireless Personal Area Network (6LoWPAN) Routing",
              RFC 6606, DOI 10.17487/RFC6606, May 2012,
              <https://www.rfc-editor.org/info/rfc6606>.

   [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained
              Application Protocol (CoAP)", RFC 7252,
              DOI 10.17487/RFC7252, June 2014,
              <https://www.rfc-editor.org/info/rfc7252>.

   [RFC8025] Thubert, P., Ed. and R. Cragie, "IPv6 over Low-Power
              Wireless Personal Area Network (6LoWPAN) Paging Dispatch",
              RFC 8025, DOI 10.17487/RFC8025, November 2016,
              <https://www.rfc-editor.org/info/rfc8025>.

   [RFC8065] Thaler, D., "Privacy Considerations for IPv6 Adaptation-
              Layer Mechanisms", RFC 8065, DOI 10.17487/RFC8065,
              February 2017, <https://www.rfc-editor.org/info/rfc8065>.

   [RFC8138] Thubert, P., Ed., Bormann, C., Toutain, L., and R. Cragie,
              "IPv6 over Low-Power Wireless Personal Area Network
              (6LoWPAN) Routing Header", RFC 8138, DOI 10.17487/RFC8138,
              April 2017, <https://www.rfc-editor.org/info/rfc8138>.

   [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>.

   [RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz,
              "Object Security for Constrained RESTful Environments
              (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019,
              <https://www.rfc-editor.org/info/rfc8613>.

   [RFC8724] Minaburo, A., Toutain, L., Gomez, C., Barthel, D., and JC.
              Zuniga, "SCHC: Generic Framework for Static Context Header
              Compression and Fragmentation", RFC 8724,
              DOI 10.17487/RFC8724, April 2020,
              <https://www.rfc-editor.org/info/rfc8724>.

   [RFC8824] Minaburo, A., Toutain, L., and R. Andreasen, "Static
              Context Header Compression (SCHC) for the Constrained
              Application Protocol (CoAP)", RFC 8824,
              DOI 10.17487/RFC8824, June 2021,
              <https://www.rfc-editor.org/info/rfc8824>.

   [RFC8930] Watteyne, T., Ed., Thubert, P., Ed., and C. Bormann, "On
              Forwarding 6LoWPAN Fragments over a Multi-Hop IPv6
              Network", RFC 8930, DOI 10.17487/RFC8930, November 2020,
              <https://www.rfc-editor.org/info/rfc8930>.

   [RFC8931] Thubert, P., Ed., "IPv6 over Low-Power Wireless Personal
              Area Network (6LoWPAN) Selective Fragment Recovery",
              RFC 8931, DOI 10.17487/RFC8931, November 2020,
              <https://www.rfc-editor.org/info/rfc8931>.

   [RFC9008] Robles, M.I., Richardson, M., and P. Thubert, "Using RPI
              Option Type, Routing Header for Source Routes, and IPv6-
              in-IPv6 Encapsulation in the RPL Data Plane", RFC 9008,
              DOI 10.17487/RFC9008, April 2021,
              <https://www.rfc-editor.org/info/rfc9008>.

   [RFC9147] Rescorla, E., Tschofenig, H., and N. Modadugu, "The
              Datagram Transport Layer Security (DTLS) Protocol Version
              1.3", RFC 9147, DOI 10.17487/RFC9147, April 2022,
              <https://www.rfc-editor.org/info/rfc9147>.

12.2. Informative References

   [RFC9006] Gomez, C., Crowcroft, J., and M. Scharf, "TCP Usage
              Guidance in the Internet of Things (IoT)", RFC 9006,
              DOI 10.17487/RFC9006, March 2021,
              <https://www.rfc-editor.org/info/rfc9006>.

Appendix A. Header compression examples

   Uplink packet

   Source address: fd00::202:2:2:2 with port 8765
   Destination address: 2001::1 with port 5678
   Payload: "Hello 1" 68 65 6C 6C 6F 20 31

```
Uncompressed IPv6/UDP packet:
60 00 00 00 00 17 00 40    FD 00 00 00 00 00 00 00
02 02 00 02 00 02 00 02    20 01 00 00 00 00 00 00
00 00 00 00 00 00 00 01    22 3D 16 2E 00 0F 33 68
68 65 6C 6C 6F 20 31

IPv6/UDP header length: 48 bytes
Total length:          55 bytes
```

In this example, for SCHC compression of IPv6/UDP headers, RuleID
0x20 is used. The Rule corresponding to RuleID 0x20 is shown in
Figure 21.

FID	FL	FP	DI	TV	MO	CDA		Sent [bits]
IPv6 Version	4	1	Bi	6	ignore	not-sent		
IPv6 Diffserv	8	1	Bi	0	equal	not-sent		
IPv6 Flow Label	20	1	Bi	0	equal	not-sent		
IPv6 Length	16	1	Bi		ignore	compute-*		
IPv6 Next Header	8	1	Bi	17	equal	not-sent		
IPv6 Hop Limit	8	1	Bi	64	ignore	not-sent		
IPv6 DevPrefix	64	1	Bi	FD00::/64	equal	not-sent		
IPv6 DevIID	64	1	Bi		ignore	value-sent		64
IPv6 AppPrefix	64	1	Bi	2001::/64	equal	not-sent		
IPv6 AppIID	64	1	Bi	::1	equal	not-sent		
UDP DevPort	16	1	Bi	8765	equal	not-sent		
UDP AppPort	16	1	Bi	5678	equal	not-sent		
UDP Length	16	1	Bi		ignore	compute-*		
UDP checksum	16	1	Bi		ignore	compute-*		

Figure 21: Illustration of an example Rule with RuleID 0x20

A.1. Single-hop or SRO frame format

```
SCHC-compressed packet:
44 20 02 02 00 02 00 02
00 02 68 65 6C 6C 6F 20
31
```

      Header length: 10 bytes
      SCHC Dispatch: 44 (01000100)
      SCHC RuleID: 0x20 (1 byte)
      SCHC residue: 02 02 00 02 00 02 00 02
      Payload: 68 65 6C 6C 6F 20 31
      Total length: 17 bytes

A.2. TRO frame format

      TO-DO

A.3. PRO frame format

      SCHC-compressed packet:
      45 88 40 20 02 02 00 02
      00 02 00 02 68 65 6C 6C
      6F 20 31

      Header length: 12 bytes
      SCHC Pointer Dispatch: 45 (01000101)
      SCHC Pointer: 88 40
      SCHC Pointer P: 1
      SCHC Pointer Bit Pointer: 8
      SCHC Address length: 64 bits
      SCHC RuleID: 0x20 (1 byte)
      SCHC residue: 02 02 00 02 00 02 00 02
      Payload: 68 65 6C 6C 6F 20 31
      Total length: 19 bytes

A.4. Mesh-Under frame format

      TO-DO

A.5. Enabling the transition protocol stack

      Uplink packet

      Source address: fe80::201:1:1:1 with port 46487
      Destination address: fe80::1 with port 5683
      Payload (Temperature value): DA 8C E8 75 15 66 3B 00 1B 37
      SCHC protocol number: 145 (0x91)

      Uncompressed IPv6/UDP/CoAP packet:
      60 0D 4E 65 00 25 11 40   FE 80 00 00 00 00 00 00
      02 01 00 01 00 01 00 01   FE 80 00 00 00 00 00 00
      00 00 00 00 00 00 00 01   B5 97 16 33 00 25 00 38
      50 02 B6 F7 BA 74 65 6D   70 65 72 61 74 75 72 D1
      EA 00 FF DA 8C E8 75 15   66 3B 00 1B 37

IPv6/UDP/CoAP header length: 67 bytes
Total length: 77 bytes

In this example, for SCHC compression of UDP/CoAP headers, RuleID
0x22 is used. The Rule corresponding to RuleID 0x22 is shown in
Figure 22.

FID	FL	FP	DI	TV	MO	CDA	Sent [HEX]
UDP DevPort	16	1	Bi		ignore	value-sent	B5 97
UDP AppPort	16	1	Bi	5683	equal	not-sent	
UDP Length	16	1	Bi		ignore	compute-*	
UDP checksum	16	1	Bi		ignore	compute-*	
CoAP Version	16	1	Bi	1	equal	not sent	
CoAP Type	16	1	Up	01	equal	not sent	
CoAP TKL	32	1	Bi	0x00	equal	not sent	
CoAP Code	8	1	Up	0.02	equal	not-sent	
CoAP MID	16	1	Bi		ignore	value-sent	B6 F7
CoAP OptUri-Path	10	1	Up	/temperature	equal	not-sent	
CoAP Opt No-Resp	1	1	Up	00	equal	not-sent	
CoAP Opt EndOpt	8	1	Up	0xFF	equal	not-sent	

Figure 22: Illustration of an example Rule with RuleID 0x22

IPv6 packet (with uncompressed header) carrying the SCHC-compressed UDP/CoAP head
ers:
60 0D 4E 65 00 25 91 40   FE 80 00 00 00 00 00 00
02 01 00 01 00 01 00 01   FE 80 00 00 00 00 00 00
00 00 00 00 00 00 00 01   22 B5 97 B6 F7 DA 8C E8
75 15 66 3B 00 1B 37

    Compressed packet (IPv6 using 6LoWPAN + UDP/CoAP using SCHC):
    6A 11 0D 4E 65 91 02 01   00 01 00 01 00 01 00 00
    00 00 00 00 00 01 22 B5   97 B6 F7 DA 8C E8 75 15
    66 3B 00 1B 37

```
Header length: 27 bytes
IPHC: 6A 11
  Dispatch: 011
  TF: 01
  NH: 0
  HLIM: 10
  CID: 0
  SAC: 0
  SAM: 01
  M: 0
  DAC: 0
  DAM: 01

  Traffic Class: 0D4E65
  Next Header: 91
  Src. Address: 201:1:1:1
  Dst. Address: ::1

Next Header: 91 (SCHC)
SCHC RuleID: 0x22
SCHC Residue:
  UDP Dev Port: B5 97 (46487)
  CoAP MID: B6 F7 (46839)

Total length: 37 bytes
```

Appendix B. Analysis of route-over multihop approaches

   This section provides an analysis of the features, pros and cons of
   the route-over multihop approaches defined in this document: i) SRO,
   ii) TRO, and iii) PRO.

   TO-DO: align with latest descriptions of SRO, TRO and PRO.

B.1. SRO

   SRO incurs the lowest header overhead among the considered Route-Over
   approaches, as it only requires the SCHC Dispatch (1 byte). However,
   it is the most demanding approach in terms of memory usage, since all
   network nodes (including intermediate nodes) need to store all the
   Rules in use in the network. Therefore, it will be suitable for
   rather small networks and/or where nodes have sufficient memory.
   Also, SCHC context should be ideally and actually be as static as
   possible, in order to avoid frequent network- wide stored SCHC
   context updates.

B.2. TRO

   TRO incurs a header overhead that includes a fixed part (a Page
   Switch plus the SCHC Dispatch, of 1 byte each), plus a variable part
   that comprises RFC 8138-compressed routing artifacts.

   Regarding the latter, in a Downward transmission, it would include
   the SRH-6LoRH (of variable size, of 4 bytes in the best case, or
   e.g., 8 bytes as in Fig. 20 of RFC 8138), the RPI-6LoRH (3 bytes in
   the best case) and the IP-in-IP header (not present if the source is
   the Root, at least 3 bytes otherwise). In the cases considered, and
   when the Root is not the packet source, the total header overhead of
   this approach would be of at least 12-16 bytes.

   For upward transmission, the variable part of the header overhead for
   this approach would include only the RPI-6LoRH (at least, 3 bytes)
   and the IP-in-IP header (at least, 3 bytes). Therefore, in the cases
   considered, the total header overhead of this approach would be of at
   least 8 bytes.

   An advantage of this approach is that a node only has to store the
   Rules for the communications it is involved in as an endpoint, which
   minimizes memory requirements and the impact of potential SCHC
   context updates. For example, pure intermediate nodes do not have to
   store SCHC context.

   Note that this approach requires the network to use RPL, non-storing
   mode. Furthermore, the paths for communication between two nodes in
   the same network or with external nodes will need to traverse the
   Root. For communication with external nodes, traversing the Root
   will be needed anyway, therefore this feature does not pose any
   issue. However, this constraint will preclude the usage of optimal
   routes (when they do not include the Root node).

B.3. PRO

   PRO incurs a header overhead that includes a 2-byte fixed part (the
   SCHC Pointer Dispatach plus the SCHC Pointer itself) and a variable
   part (i.e., the destination address compression residue). The size
   of the latter will depend on and will need to be planned for the
   intended use case of the network:

   A.- In special cases (e.g., if there is only one possible destination
   that is known beforehand), there will not be a destination address
   residue.

B.- If interactions are always intranetwork (i.e., the prefix is known by intermediate nodes), and there can be several possible destinations in the network, the destination address residue will be up to 8 bytes (it could be less depending on how the addresses in that network are built, for example, it could be just 2 bytes).

C.- If interactions can occur with various external networks (i.e., the destination prefix is not known beforehand), the destination address residue will have to be the whole address (16 bytes), since an intermediate node does not know which is the destination prefix.

An advantage of this approach, as in TRO, is that a node only has to store the Rules for the communications it is involved in as an endpoint, which minimizes memory requirements and the impact of potential SCHC context updates. For example, pure intermediate nodes do not have to store SCHC context.

A potential advantage of PRO is that, in contrast with TRO, paths for intranetwork communication are not necessarily constrained to traversing a root node. Therefore, for intranetwork communication, the chances of using optimal paths are greater. Another feature is that the routing solution to be used is not tied to RPL non-storing mode.

B.4. Summary

Assessing the suitability of the different approaches requires considering the following dimensions: network size, node memory capabilities, header overhead, routing constraints / path optimality, intra- or inter-network communication.

TO-DO: to be completed.

Authors' Addresses

Carles Gomez
UPC
C/Esteve Terradas, 7
08860 Castelldefels
Spain
Email: carles.gomez@upc.edu

Ana Minaburo
Consultant
Rue de Rennes
35510 Cesson-Sevigne
France

Email: anaminaburo@gmail.com