Network Working Group
Internet-Draft
Updates: 4944 (if approved)
Intended status: Standards Track
Expires: June 11, 2009

J. Hui, Ed.
Arch Rock Corporation
P. Thubert
Cisco

December 8, 2008

Compression Format for IPv6 Datagrams in 6LoWPAN Networks
draft-ietf-6lowpan-hc-04

Status of this Memo

Abstract

This document specifies an IPv6 header compression format for IPv6 packet delivery in 6LoWPAN networks. The compression format relies on shared context information to allow compression of arbitrary prefixes and addresses. This document specifies an interface to an abstract context database but the content and the management of the database are out of scope. This document specifies compression of multicast addresses and a framework for compressing next headers. This framework specifies UDP compression and is prepared for additional transports.

Table of Contents

1. Introduction

The [IEEE 802.15.4] standard specifies an MTU of 128 bytes, yielding about 80 octets of actual MAC payload once security is turned on, on a wireless link with a link throughput of 250 kbps or less. The 6LoWPAN adaptation format [RFC4944] was specified to carry IPv6 datagrams over such constrained links, taking into account limited bandwidth, memory, or energy resources that are expected in applications such as Wireless Sensor Networks. [RFC4944] defines a Mesh Addressing header to support sub-IP forwarding, a Fragmentation header to support the IPv6 minimum MTU requirement [RFC2460], and stateless header compression for IPv6 datagrams (LOWPAN_HC1 and LOWPAN_HC2) to reduce the relatively large IPv6 and UDP headers down to (in the best case) several bytes.

LOWPAN_HC1 and LOWPAN_HC2 are insufficient for most practical uses of 6LoWPAN networks. LOWPAN_HC1 is most effective for link-local unicast communication, where IPv6 addresses carry the link-local prefix and Interface Identifiers (IID) directly derived from IEEE 802.15.4 addresses. In this case, both addresses may be completely elided. However, though link-local addresses are commonly used for local protocol interactions such as IPv6 ND [RFC4861], DHCPv6 [RFC3315] or routing protocols, they are not normally used for application layer data traffic, so the actual value of this compression mechanism is limited.

Routable addresses must be used when communicating with devices external to the LoWPAN or in a route-over configuration where IP forwarding occurs within the LoWPAN. For routable addresses, LOWPAN_HC1 requires both IPv6 source and destination addresses to carry the prefix in-line. In cases where the Mesh Addressing header is not used, the IID of a routable address must be carried in-line. However, LOWPAN_HC1 requires 64-bits for the IID when carried in-line and cannot be shortened even when it is derived from the IEEE 802.15.4 16-bit short address.

When the destination is an IPv6 multicast address, LOWPAN_HC1 requires the full 128-bit address to be carried in-line. This specification provides an additional mechanism to compress Unique Local, Global and multicast IPv6 Addresses based on shared states within contexts. It also introduces a number of additional improvements over [RFC4944].

LOWPAN_HC1 cannot elide the IPv6 Hop Limit in the IPv6 header, even though a limited set of values are useful in many practical cases. For instance, if the LoWPAN is a mesh-under stub, a Hop Limit of 1 for inbound and a default value such as 64 for outbound are usually enough for application layer data traffic. Compressing that field

enables saving one octet per packet.

LOWPAN_HC1 can be extended to include a LOWPAN_HC2 octet to support compression of UDP, TCP, or ICMPv6; that LOWPAN_HC2 octet is placed right after the LOWPAN_HC1 octet and before the uncompressed IP fields. This specification moves the transport control octet after the uncompressed IP fields for a more properly layered structure.

[RFC4944] defines a compression mechanism for UDP, but that mechanism does not enable checksum compression when rendered possible by additional upper layer mechanisms such as upper layer Message Integrity Check (MIC). This specification adds the capability to elide the UDP checksum over the LoWPAN, which allows savings of two additional octets.

Finally, LOWPAN_HC1 lacks the flexibility to support the compression of additional transport mechanisms that could be introduced in the future.

This document specifies a header compression format for IPv6 datagrams. This format improves on the header compression format defined in [RFC4944] by generalizing it to support a broader range of communication paradigms, including both mesh-under and route-over configurations; communication to nodes internal and external to the 6LoWPAN network; and multicast communication. This document also defines a flexible framework for compressing arbitrary next headers and defines UDP header compression within this framework. This compression format carries forward the design concepts in RFC 4944 [RFC4944], minimizing compression state and state maintenance by relying on shared context among all nodes in a 6LoWPAN network.

1.1. 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 RFC 2119 [RFC2119].

2. IPv6 Header Compression

In this section, we define the LOWPAN_IPHC encoding format for compressing the IPv6 header. To enable effective compression LOWPAN_IPHC relies on information pertaining to the entire 6LoWPAN network. LOWPAN_IPHC assumes the following will be the common case for 6LoWPAN communication: Version is 6; Traffic Class and Flow Label are both zero; Payload Length can be inferred from lower layers from either the 6LoWPAN Fragmentation header or the IEEE 802.15.4 header; Hop Limit will be set to a well-known value by the source; addresses

assigned to 6LoWPAN interfaces will be formed using the link-local prefix or a single routable prefix assigned to the entire 6LoWPAN network; addresses assigned to 6LoWPAN interfaces are formed with an IID derived directly from either the 64-bit extended or 16-bit short IEEE 802.15.4 addresses.

```
+-------------------------------------+-----------------------
| Dispatch + LOWPAN_IPHC (2-3 octets) | Compressed IPv6 Header
+-------------------------------------+-----------------------
```

Figure 1: LOWPAN_IPHC Header

The LOWPAN_IPHC encoding utilizes 13 bits, 5 of which are taken from the rightmost bits of the dispatch type. The encoding may be extended by another octet to support additional contexts. Uncompressed IPv6 header fields follow the LOWPAN_IPHC encoding, as shown in Figure 1. With the above scenario, the LOWPAN_IPHC can compress the IPv6 header down to two octets (the dispatch octet and the LOWPAN_IPHC encoding) with link-local communication. When routing over multiple IP hops, LOWPAN_IPHC can compress the IPv6 header down to 7 octets (2-octets dispatch/LOWPAN_IPHC, 1-octet Hop Limit, 2-octet Source Address, and 2-octet Destination Address).

2.1. LOWPAN_IPHC Encoding Format

2.1.1. Base Format

```
  0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 0 | 1 | 1 |  TF   |NH | HLIM  |CID|SAC|  SAM  | M |DAC|  DAM  |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Figure 2: LOWPAN_IPHC Encoding

TF: Traffic Class, Flow Label:
   00: Traffic Class + 4-bit Pad + Flow Label (4 bytes)
   01: ECN + 2-bit Pad + Flow Label (3 bytes)
   10: Traffic Class (1 byte)
   11: Version, Traffic Class, and Flow Label are compressed.

NH: Next Header:
   0: Full 8 bits for Next Header are carried in-line.
   1: The Next Header field is compressed and the next header is compressed using LOWPAN_NHC, which is discussed in Section 3.

HLIM: Hop Limit:
   00: The Hop Limit field is carried in-line.
   01: The Hop Limit field is elided and the the hop limit is 1.
   10: The Hop Limit field is elided and the the hop limit is 64.
   11: The Hop Limit field is elided and the hop limit is 255.

CID: Context Identifier Extension:
   0: No additional 8-bit Context Identifier Extension is used. If context-based compression is specified in either SC or DC, the default context is used.
   1: An additional 8-bit Context Identifier Extension field immediately follows the DAM field.

SAC: Source Address Compression
   0: Source address compression uses stateless compression.
   1: Source address compression uses stateful, context-based compression.

SAM: Source Address Mode:
   If SAC=0:
      00: 0 bits. The address is the unspecified address.
      01: 64 bits. The first 64-bits of the address are elided. The value of those bits is the link-local prefix padded with zeros. The remaining 64 bits are carried inline.
      10: 16 bits. The first 112 bits of the address are elided. The value of those bits is the link-local prefix padded with zeros. The remaining 16 bits are carried inline.
      11: 0 bits. The address is fully elided. The first 64 bits of the address are elided. The remaining 64 bits are computed from the link-layer address as defined in [RFC4944].
   If SAC=1:
      00: 128 bits. The full address is carried in-line.
      01: 64 bits. The first 64-bits of the address are elided. The value of those bits is taken from the context and padded with zeros. The remaining 64 bits are carried inline.
      10: 16 bits. The first 112 bits of the address are elided. The value of those bits is taken from the context and padded with zeros. The remaining 16 bits are carried inline.

11: 0 bits. The address is fully elided. The first 64 bits are taken from the context. The remaining 64 bits are computed from the link-layer address as defined in [RFC4944].

M: Multicast Compression
- 0: Destination address does not use multicast compression.
- 1: Destination address uses multicast compression.

DAC: Destination Address Compression
- 0: Destination address compression uses stateless compression.
- 1: Destination address compression uses stateful, context-based compression.

DAM: Destination Address Mode:

If M=0: When DAC=0, any elided prefix bits are the link-local prefix padded by zeros. When DAC=1, any elided prefix bits are taken from the context and padded by zeros.
- 00: 128 bits. The full address is carried in-line.
- 01: 64 bits. The first 64-bits of the address are elided. The remaining 64 bits are carried inline.
- 10: 16 bits. The first 112 bits of the address are elided. The remaining 16 bits are carried inline.
- 11: 0 bits. The address is fully elided. The first 64 bits of the address are elided. The remaining 64 bits are computed from the link-layer address as defined in [RFC4944].

If M=1 and DAC=0:
- 00: 48 bits. The address takes the form FFXX::00XX:XXXX:XXXX.
- 01: 32 bits. The address takes the form FFXX::00XX:XXXX.
- 10: 16 bits. The address takes the form FF0X::0XXX.
- 11: 8 bits. The address takes the form FF02::00XX.

If M=1 and DAC=1:
- 00: 128 bits. The full address is carried in-line.
- 01: 48 bits. The address takes the form FFXX::XX[plen]:[prefix]:XXXX:XXXX. The values of plen and prefix are taken from the specified context.
- 10: reserved
- 11: reserved

2.1.2. Context Identifier Extension

This specification expects that an abstract set of states called contexts is shared between the node that compresses a packet and the node(s) that need to expand it. The specification enables the transport of an opaque index that is used to lookup the abstract context database. The index in encoded with 4 bits enabling to address up to 16 contexts.

This specification requires that services associated to the abstract context database implement an interface to the 6LoWPAN compressor to help compress and uncompress an address based on the parameters passed by the compressor and the information in the abstract context database.

The interface MUST provide the methods to lookup a context ID from a prefix and a prefix length for encoding, and reversely lookup a prefix and a prefix length from a context ID for decoding.

How the contexts are shared and maintained is out of scope. The actual context information is out of scope. Actions in response to unknown and/or invalid contexts are out of scope.

The interface might be extended to allow for further stateful compression, for instance for SAC = 11, additional context information might be used to store the full IPv6 address using the Link layer Address as an additional index.

If the CID field is set to '1' in the LOWPAN_HC encoding, then an additional octet extends the LOWPAN_HC encoding following the DAM bits but before the IPv6 header fields that are carried in-line. The additional octet identifies the prefix when the IPv6 source and/or destination address is compressed. The context identifier is 4 bits for each address, supporting up to 16 contexts. The encoding is shown in Figure 3.

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
|      SCI      |      DCI      |
+---+---+---+---+---+---+---+---+
```

Figure 3: LOWPAN_IPHC Encoding

SCI: Source Context Identifier Identifies the prefix that is used when the IPv6 source address is compressed.

DCI: Destination Context Identifier Identifies the prefix that is used when the IPv6 destination address is compressed.

2.2. IPv6 Header Encoding

Fields carried in-line (in part or in whole) appear in the same order as they do in the IPv6 header format [RFC2460]. The Version field is always elided. The IPv6 Payload Length field MUST always be elided and inferred from lower layers using the 6LoWPAN Fragmentation header or the IEEE 802.15.4 header. Unicast IPv6 addresses may be

compressed to 64 or 16 bits or completely elided. Multicast IPv6 addresses may be compressed to 8, 16, or 24 bits.

2.2.1. Traffic Class and Flow Label Compression

The Traffic Class field in the IPv6 header comprises 6 bits of diffserv extension [RFC2474] and 2 bits of Explicit Congestion Notification (ECN) [RFC3168]. If the ECN information is carried by the Lower Layers in a compatible fashion then it can be elided from the 6LoWPAN header. Otherwise, it has to be transported in one of the following encodings.

The TF field in the LOWPAN_HC encoding indicate whether the Traffic Class and Flow Label are carried in-line in the compressed IPv6 header. When Flow Label is included while the Traffic Class is compressed, an additional 4 bits are included to maintain byte-alignment. Two of the 4 bits contain the ECN bits from the Traffic Class field.

To ensure that the ECN bits appear in the same location for all encodings that include them, the Traffic Class field is rotated right by 2 bits in the compressed IPv6 header. The encodings are shown below:

```
                      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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |ECN|   DSCP    |  rsv  |             Flow Label                |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

TF = 00: Traffic Class and Flow Label carried in-line.

```
                      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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |ECN|rsv|             Flow Label                |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

TF = 01: Flow Label carried in-line.

```
  0 1 2 3 4 5 6 7
 +-+-+-+-+-+-+-+-+
 |ECN|   DSCP    |
 +-+-+-+-+-+-+-+-+
```

TF = 10: Traffic Class carried in-line.

2.2.2. Stateless Multicast Addresses Compression

LOWPAN_HC supports stateless compression of multicast address when M = 1 and SAC = 0. An IPv6 multicast address may be compressed down to 48, 32, 16, or 8 bits using stateless compression. The format supports compression of the Solicited-Node Multicast Address (FF02::1:FFXX:XXXX) as well as any IPv6 multicast address where the upper bits of the multicast group identifier are zero. The compressed forms only carry the least-significant bits of the multicast group identifier.

```
                     1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Flags | Scope |              Group Identifier                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |       Group Identifier        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

DAM = 00. 48-bit Compressed Multicast Address (FFfs::00gg:gggg:gggg)

```
                     1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Flags | Scope |              Group Identifier                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

DAM = 01. 32-bit Compressed Multicast Address (FFfs:00gg:gggg).

```
                     1
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Scope |   Group Identifier    |
```

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

        DAM = 10. 16-bit Compressed Multicast Address (FF0s::0ggg).



      0 1 2 3 4 5 6 7
     +-+-+-+-+-+-+-+-+
     |   Group ID    |
     +-+-+-+-+-+-+-+-+

         DAM = 11. 8-bit Compressed Multicast Address (FF02::gg).
```

2.2.3. Stateful Multicast Addresses Compression

LOWPAN_HC supports stateful compression of multicast addresses when M = 1 and SAC = 1. This document currently defines SAM = 01: context-based compression of Unicast-Prefix-based IPv6 Multicast Addresses [RFC3306][RFC3956]. In particular, the Prefix Length and Network Prefix can be taken from a context. As a result, LOWPAN_HC can compress a Unicast-Prefix-based IPv6 Multicast Address down to 6 octets by only carrying the 4-bit Flags, 4-bit Scope, 8-bit RIID, and 32-bit Group Identifier in-line.

```
                          1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     | Flags | Scope |    Reserved    |       Group Identifier        |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |        Group Identifier        |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

     DAM = 01. Unicast-Prefix-based IPv6 Multicast Address Compression
```

The Reserved field MUST carry the reserved bits from the multicast address format as described in [RFC3306]. When a Rendezvous Point is encoded in the multicast address as described in [RFC3956], the Reserved field carries the RIID bits in-line.

3. IPv6 Next Header Compression

LOWPAN_IPHC elides the IPv6 Next Header field when the NH bit is set

to 1. It also indicates the use of 6LoWPAN next header compression, LOWPAN_NHC. The value of IPv6 Next Header is recovered from the first bits in the LOWPAN_NHC encoding. The following bits are specific to the IPv6 Next Header value. Figure 4 shows the structure of an IPv6 datagram compressed using LOWPAN_IPHC and LOWPAN_NHC.

```
+-------------+-------------+-------------+-----------------+--------
| LOWPAN_IPHC | In-line     | LOWPAN_NHC  | In-line Next    | Payload
|   Encoding  |   IP Fields |   Encoding  |   Header Fields |
+-------------+-------------+-------------+-----------------+--------
```

Figure 4: Typical LOWPAN_IPHC/LOWPAN_NHC Header Configuration

3.1. LOWPAN_NHC Format

Compression formats for different next headers are identified by a variable length bit-pattern immediately following the LOWPAN_IPHC compressed header. When defining a next header compression format, the number of bits used SHOULD be determined by the perceived frequency of using that format. However, the number of bits and any remaining encoding bits SHOULD respect octet alignment. The following bits are specific to the next header compression format. In this document, we define a compression format for UDP headers.

```
+----------------+--------------------------
| var-len NHC ID | compressed next header...
+----------------+--------------------------
```

Figure 5: LOWPAN_NHC Encoding

3.2. UDP Header Compression

This document defines a compression format for UDP headers using LOWPAN_NHC. The UDP compression format is shown in Figure 6. Bits 0 through 4 represent the NHC ID and '11110' indicates the specific UDP header compression encoding defined in this section.

3.2.1. Compressing UDP ports

This specification introduces a range of well-known port (0xF0Bx) that can be compressed to 4 bits. Considering that this represents only 16 contiguous ports, it can be expected that many incompatible applications will use the same port numbers of their own end-to-end needs.

The overloading of the 0xF0Bx ports increases the risk of getting the wrong type of payload and misinterpreting the content compared to ports that reserved at IANA. It is thus recommended that the use of those ports be associated with a mechanism such as a Transport Layer Security (TLS) Message Integrity Check (MIC) that validates that the content is expected and checked for integrity.

3.2.2. Compressing UDP checksum

The UDP checksum operation is mandatory with IPv6 [RFC2460] for all packets. For that reason [RFC4944] disallows the compression of the UDP checksum.

With this specification, a compressor in the source transport endpoint MAY elide the UDP checksum in certain cases for instance:

Upper Layer Message Integrity Check: In this case, there is some other form of integrity check in the UDP payload that covers at least the same information as the UDP checksum (pseudo-header, data) and has at least the same strength.

Tunneling: In this case, 6LoWPAN is deployed as a wireless pseudo-fieldbus by tunneling existing field protocols over UDP. If the tunneled PDU possesses its own addressing, security and integrity check, the tunneling mechanism MAY authorize to elide the UDP checksum in order to save on the encapsulation overhead.

This elision is indicated by setting the 'C' bit in the LOWPAN_NHC header.

A 6LoWPAN endpoint that compresses the LOWPAN_NHC header MUST NOT elide the UDP checksum (set the C bit) unless it has been authorized to do so by the source of the packet. In the source transport endpoint, this authorization can come from upper layer transport or application protocol instance that originated the packet. In a forwarding node, this authorization is implied when the incoming packet had the optimization applied (had the C bit set).

A 6LoWPAN endpoint that expands the LOWPAN_NHC header MUST reconstitute the UDP checksum by computing the valid value for the datagram as specified in [RFC0768] and [RFC2460], and place the result of that computation in the restored UDP header, unless it has been authorized to ignore the checksum operation. In the destination transport endpoint this authorization can come from upper layer transport that will receive the packet and would ignore the UDP checksum should it be restored.

3.2.3. UDP LOWPAN_NHC Format

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| 1 | 1 | 1 | 1 | 0 | C |   P   |
+---+---+---+---+---+---+---+---+
```

Figure 6: UDP Header Encoding

C: Checksum:
   0: All 16 bits of Checksum are carried in-line.
   1: All 16 bits of Checksum are elided. The Checksum is recovered by recomputing it on the 6LoWPAN termination point.

P: Ports:
   00: All 16 bits for both Source Port and Destination Port are carried in-line.
   01: All 16 bits for Source Port are carried in-line. First 8 bits of Destination Port is 0xF0 and elided. The remaining 8 bits of Destination Port are carried in-line.
   10: First 8 bits of Source Port are 0xF0 and elided. The remaining 8 bits of Source Port are carried in-line. All 16 bits for Destination Port are carried in-line.
   11: First 12 bits of both Source Port and Destination Port are 0xF0B and elided. The remaining 4 bits for each are carried in-line.

Fields carried in-line (in part or in whole) appear in the same order as they do in the IPv6 header format [RFC2460]. IPv6 addresses may be compressed to 64 or 16 bits or completely elided. The UDP Length field MUST always be elided and is inferred from lower layers using the 6LoWPAN Fragmentation header or the IEEE 802.15.4 header.

4. IANA Considerations

This document defines a new IPv6 header compression format for 6LoWPAN networks. The document allocates Dispatch type values of 0x08-0x0F (TBD) for LOWPAN_IPHC.

5. Security Considerations

The definition of LOWPAN_IPHC permits the compression of header information on communication that could take place in its absence, albeit in a less efficient form. It recognizes that a IEEE 802.15.4

PAN may have associated with it a number of prefixes through shared context. How the shared context is assigned and managed is beyond the scope of this document.

The overloading of the 0xF0Bx ports increases the risk of getting the wrong type of payload and misinterpreting the content compared to ports that reserved at IANA. It is thus recommended that the use of those ports be associated with a mechanism such as a Transport Layer Security (TLS) Message Integrity Check (MIC) that validates that the content is expected and checked for integrity.

6. Acknowledgements

Thanks to Julien Abeille, Carsten Bormann, Christos Polyzois, Erik Nordmark, Robert Assimiti, Shoishi Sakane, Zach Shelby, Stephen Dawson-Haggerty, Jay Werb, and Mathilde Durvy for useful design consideration and implementation feedback.

7. Changes

Draft 04:
- Fixed typos leftover from the changes in 03.
- Gave more details on UDP checksum compression.
- Greater discussion on the use of context information and clarification that its details are out of scope.
- Added security concern on 0xF0Bx port overloading.

Draft 03:
- Decoupled meaning of SAM bits from the destination address.
- Have separate bit to indicate multicast address compression.
- More extensive support for multicast address compression, including Unicast-Prefix-based Multicast Addresses.

Draft 02:
- Updated wording with compression mode to clarify that a compression mode does not enforce what kind of destination address is being used. Specifically changed Destination Dependent Field to Compression Mode.
- Specify that the configuration and management of contexts is out of scope of this document.

Draft 01:

- HC back to 1 byte by default by stealing a few bits from the dispatch field.
- Added better support for multicast address compression.
- Fixed alignment for UDP port compression.
- Better support for Traffic Class and Flow Label compression.
- Pascal joined as an author.

8. References

8.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[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.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[RFC4007] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., and B. Zill, "IPv6 Scoped Address Architecture", RFC 4007, March 2005.

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

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

8.2. Informative References

[IEEE 802.15.4]
IEEE Computer Society, "IEEE Std. 802.15.4-2006", October 2006.

[RFC3306] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6

Multicast Addresses", RFC 3306, August 2002.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3956] Savola, P. and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", RFC 3956, November 2004.

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

Authors' Addresses

Jonathan W. Hui (editor)
Arch Rock Corporation
501 2nd St. Ste. 410
San Francisco, California 94107
USA

Phone: +415 692 0828
Email: jhui@archrock.com

Pascal Thubert
Cisco Systems
Village d'Entreprises Green Side
400, Avenue de Roumanille
Batiment T3
Biot - Sophia Antipolis 06410
FRANCE

Phone: +33 4 97 23 26 34
Email: pthubert@cisco.com

Full Copyright Statement

Intellectual Property