           Robust Header Compression (ROHC) over 802 networks
                    draft-bormann-rohc-over-802-02

Status of this Memo

   This Internet-Draft is submitted to IETF in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF), its areas, and its working groups. Note that
   other groups may also distribute working documents as Internet-
   Drafts.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   The list of current Internet-Drafts can be accessed at
   http://www.ietf.org/ietf/1id-abstracts.txt.

   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html.

   This Internet-Draft will expire on January 14, 2010.

Copyright Notice

Abstract

   Various proposals have been submitted to the ROHC working group for
   enabling the use of ROHC [RFC3095] header compression over Ethernet,
   802.11 and other 802-based links.

Previous proposals generally suffered from a lack of systems
perspective on 802 networks. The present document attempts to supply
some systems perspective and provides a rough outline for a solution.

This is a submission to the IETF ROHC WG. Please direct discussion
to its mailing list, rohc@ietf.org

$Revision: 1.9 $

Table of Contents

1. Introduction

RFC 3095 [RFC3095] defines four ROHC profiles for the header
compression of IP, UDP, RTP, ESP, and related protocol headers, as
well as a framework that has been used to define a number of related
profiles (such as IP ROHC [RFC3843] and UDP-lite based RTP ROHC
[RFC4019]). Since, the framework has been extracted into RFC 4995
[RFC4995] and several "version 2" ROHC profiles have been defined
[RFC4996] [RFC5225]. ROHC as a framework is also useful for
transporting legacy compression formats where this is desirable
[I-D.bormann-rohc-avt-crtp-profile].

To enable robust header compression over a specific link layer, the
ROHC profile specifications have to be complemented by a link-layer
specific specification, typically called "ROHC-over-X". One such
specification has been defined in the IETF, ROHC over PPP [RFC3241].
Other ROHC-over-X specifications have been defined by the
organizations defining specific link layers, such as 3GPP.

No specification currently exists for applying robust header
compression to IEEE 802 networks such as Ethernet, 802.11, or 802.16.
A number of proposals have been made to the IETF ROHC WG , but it
became obvious quickly that the solutions that seem to suggest
themselves do not work at the desirable level of efficiency.

The lack of a specification for IEEE 802 networks also impacts
related IETF standards, such as IP over DVB [RFC4326]. While IP over
DVB is not by itself an IEEE 802 network, actual implementations
often are closely tied to Ethernets by technologies related to
bridging, making some form of interoperability at the compressed
level desirable.

This document first discusses some issues about ROHC-over-802, then
lists some potential non-issues, defines an encapsulation format for
ROHC-over-802, and finally discusses an appropriate negotiation
mechanism.

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

2. Discussion

2.1. Overall Requirements

There is little need for robust header compression in a classical
Ethernet (802.3) environment, which is both relatively high-speed and

(at least at the segment level) virtually error-free. However, WLAN
(802.11) and WPAN (802.15) links are often bandwidth limited; the
same will hold for WMAN (802.16) links. They also (depending on link
quality and load) can exhibit loss and delay patterns that would
motivate the use of ROHC in such scenarios. Since voice over IP is
and will be commonly used in these networks, header compression will
continue to be useful.

In the ROHC framework, header compression is performed at the
boundary between Layer 3 (IP) and Layer 2 (802, in the case of ROHC-
over-802). 802 networks are often bridged, i.e. multiple 802
technologies may contribute to a Layer 2 path that constitutes what
is considered to be "the link" from a ROHC framework point of view.
In practical implementation, nodes such as routers (often one end of
a ROHC channel) in many cases don't connect directly to 802.11 links,
but send packets on 802.3 (Ethernet) links that then are bridged by
"Access Points" to 802.11 links. System architectures for other 802
technologies also often make use of bridging.

(In this document, we use the term "bridging" for any kind of
interconnection of IEEE 802 LANs above the physical layer but below
the MAC service boundary, i.e. whenever an L3-visible hop is built
from multiple L2 constituents by interconnection with bridge-like
devices -- even if not all these L2 intermediate systems are
completely compliant to the definitions of the term "bridge" in
section 6.3.2 of IEEE 802 and in IEEE 802.1D.)

One can conclude: It is not sufficient to just look at the wireless
links -- ROHC-over-802 also needs to work on 802.3 (and other fixed-
line 802) networks. In effect, a single solution for applying ROHC
to all 802 (and related) environments needs to be defined independent
of the physical layer technology. By staying above the MAC service
interface, the solution can be largely oblivious of the specifics of
the 802 technologies employed. A nice side effect is that this will
simplify both standardization and implementation.

2.2. Elements of a Solution

A ROHC-over-X specification needs to define two elements:
o An encapsulation for ROHC framework packets, and
o a negotiation mechanism for agreeing on the use of ROHC and on the
   parameters of the ROHC channel.

(While a negotiation mechanism is not strictly needed for every ROHC-
over-X document, it is clearly too late for the alternative, i.e.
making ROHC mandatory and defining fixed channel parameter values for
any use of IP over 802.)

2.3. Who Should Standardize?

(This section is not intended to become part of any standards document resulting from this work.)

In previous discussions, the question was raised which body should standardize ROHC-over-802. As mentioned in the introduction, one ROHC-over-X protocol has been defined in the IETF, other ones have been defined in the standards bodies defining the link layers under consideration.

In the view of the author, a good test would be to see who has defined the IP-over-X specification. The ROHC-over-PPP specification clearly fits in the IETF as both the IP-over-PPP specification and the PPP specification itself are IETF specifications. For 802 networks, the IETF also has specified the link layer mapping of IP, including a number of ancillary protocols (ARP and ND) necessary for these mappings. If these protocols need to be extended, it would be more appropriate for the IETF to do so. The system issues of complex 802 networks do have a bearing on ROHC-over-802 and are in the domain of the IEEE; on the other hand, no good arguments exist currently that would call for an extension to the 802 protocols for ROHC. In summary, the author believes that IETF is the right body to work on ROHC-over-802.

Related questions are: a) Who is the community of interest? Which standards meetings do they attend? b) Which body has the required expertise? c) What existing work is underway? Are there conflicts between that work and the proposed work?

For question a) the answer appears to be the group of people who participate within the IETF ROHC WG. This group also has demonstrated expertise in header compression issues, but not necessarily in the details of link layer capabilities negotiation that may need to be part of a solution.

(If work is required on the subject of link layer capabilities negotiation, e.g. use of ROHC, this would fit within the charter of existing IEEE 802 groups; however, staying above the MAC service interface would avoid the architectural need for this. Otherwise, while the ROHC over 802 component seems best suited to IETF, there may be link layer components to the work that are best done in IEEE 802.)

In any case, close review of this work by IEEE 802 experts is advisable.

2.4. Why not just use PPPoE?

   An informational RFC specifies a widely deployed specification for
   PPP over Ethernet (PPPoE [RFC2516]), and, as mentioned there is a
   specification for ROHC over PPP [RFC3241]. For a number of reasons,
   just combining these as a ROHC-over-802 solution would be suboptimal:

   1. PPPoE's encapsulation together with the PPP encapsulation has a
       fixed overhead of eight bytes per packet, negating some of the
       savings provided by header compression.
   2. PPPoE does not solve the minimum-size padding problem (see
       below).
   3. PPPoE has a different model than the usual IP-over-802 model,
       with discovery and session stages, and possibly multiple PPP
       sessions. This complexity is often not required.

   On the other hand, if PPPoE is in active use on an 802 link, adding
   ROHC-over-PPP may be a simple way to add robust header compression.

3. Issues

3.1. Ethernet Minimum Frame Size

   Due to its roots in the CSMA/CD protocol, Ethernet (IEEE 802.3)
   defines a minimum frame size of 64 bytes. Of these, 14 bytes are
   used for the MAC header and 4 are used for the MAC trailer (frame
   check sequence), which means that the minimum payload of an Ethernet
   packet is 46 bytes.

   The existing IPv4-over-802 [RFC1042] specification uses the "total
   size" field in the IP packet to indicate how much of the 802 packet
   payload is actually an IP packet; this indirectly indicates the
   presence of padding, if any.

   ROHC compresses away the "total size" field. Instead, it relies on
   the link layer (or the ROHC-over-X protocol) to provide a packet
   size. A ROHC-over-802 encapsulation could use a number of ways to
   provide this packet size:

   1. It still could rely on the link layer size and use ROHC padding
       schemes to always inflate the size to at least 46 bytes.
   2. It could add a length field.
   3. It could make use of the length-field variant of the 802 MAC
       header format; this requires a different way of demultiplexing
       ROHC packets from other LLC packets.

   Note that solutions 1 and 2 mean that ROHC-compressed packets shorter

than 46 bytes will be padded out to this length if they ever go over
an 802.3 link. Worse, there will be no way for an 802.3-to-802.x
bridge to identify this padding and remove it, so the padding will
remain on any wireless segments of the link layer path. Given that
many voice over IP packets will have payloads of 10 to 20 bytes and
headers often can be compressed down to 3 bytes or less, this entails
a significant overhead.

So, apart from the issue of properly indicating padding, a more
interesting property of a ROHC-over-802 encapsulation is whether it
allows 802.3-to-802.x bridges to remove any padding inserted on the
802.3 segments.

3.2. Negotiation and the existing IP-over-802 model

In the existing IP-over-802 model (as exemplified by IPv4-over-802
[RFC1042]) assumes that once the MAC (link layer) address of a node
is known, packets can be sent to it. No channel setup/teardown is
provided for. In particular, a node can lose its state (be rebooted)
and packets can still be sent to it based on the knowledge of the MAC
address.

(Note that channel setup/teardown procedures that may be available
with specific 802 technologies such as 802.11 are often not end-to-
end with respect to the L2 path. E.g., a router connected to an
802.3 segment connected to an 802.11 AP may not notice when the
802.11 station goes away and comes back.)

The ROHC channel model [RFC3759] assumes that channel state is
maintained explicitly, at least if the more advanced O- and R-modes
are to be used. In addition, this channel setup is used to negotiate
parameters of the channel (such as variants of the encapsulation
format or the maximum number of compression contexts supported).

Also, while there is a ROHC channel for each direction, each ROHC
channel itself is bidirectional in the sense that (at least if not
just U-mode is to be used) there needs to be a way to return
feedback.

Finally, only the receiving end of a packet flow may be aware that
there is a benefit in using header compression (for illustration,
consider a VoWLAN phone that is receiving packets from a router that
is different than the router it chose as its default router and thus
for the reverse packet flow). Therefore, there should be a way to
initiate the setup of a ROHC channel from the receiving end.

4. Non-Issues

4.1. Reordering

Fortunately, 802 links are sequence-preserving, so there is no need
to re-sequence packets to avoid reordering (as would be required by
unmodified use of the current ROHC framework and profiles).

(The sequence preservation property holds as long as all packets of a
context are sent on the same 802.1p priority group. The author is
unable to imagine good reasons for using multiple 802.1p priority
groups for one ROHC context.)

(See also ROHC over PPP [RFC3241], section 1, which says:) ROHC
assumes that the link layer delivers packets in sequence. PPP
normally does not reorder packets. When using reordering mechanisms
such as multiclass multilink PPP [RFC2686], care must be taken so
that packets that share the same compression context are not
reordered. (Note that in certain cases, reordering may be acceptable
to ROHC, such as within a sequence of packets that all do not change
the decompression context.)

4.2. Padding a non-issue?

One argument could be that the padding issue outlined in Section 3.1
can be ignored for most 802 networks, either because the payloads
will be larger than for the most heavily compressing voice codecs or
because the header overhead is already rather high (e.g., for
802.11b, the link-layer header overhead in typical configurations is
about as large as that of three-digit numbers of bytes in the
payload).

The author takes the viewpoint that a solution that is intended to be
used universally through the 802 space does need to address padding.

5. Encapsulation

Based on the considerations above, this document proposes to use LLC
encapsulation of ROHC packets. Several approaches would have been
possible:

1. An SAP value (Logical Link Control Address) is allocated to ROHC.
    The per-packet overhead is reduced to three bytes. Note that
    this means the negotiation protocol needs to fix the small-CID
    vs. large-CID choice (alternatively, ROHC-over-802 could simply
    always use large CIDs, or even a pair of SAP values could be
    allocated).

2. SAP 0xAA is used. By setting the first byte of the OUI to a
    value illegal for an OUI (multicast/private), the rest of the
    frame can be used for the ROHC packet, reducing the overhead to
    four bytes. The first (illegal OUI) byte can be used to
    demultiplex variants, e.g. small-CID and large-CID ROHC packets
    as well as possible negotiation protocol packets (see below).
    What would be the second and third OUI bytes are already used for
    the ROHC packet.

3. A full SNAP header is used. (From an overhead perspective, for
    802.3 networks this is not better than the PPPoE case, but, like
    the previous proposals, it does allow the removal of padding by
    bridges.) Note that, to maintain reliable padding removal even
    over multiple header conversions between 802.3 and other types of
    802 links, this could NOT be a basic ethertype-carrying SNAP
    header -- this would be converted to an 802.3 header on the first
    conversion to 802.3 and would lose its padding-removal property
    on any further conversions. To prevent this, a non-zero OUI
    could be used.

Of these theoretically possible approaches, this document chooses
variant 1. The actual SAP (SSAP/DSAP) value (Logical Link Control
Address) to be used is to be defined (preferably one allocated by the
registration authority [refauth]); for testing until the SAP value is
assigned, the unreserved value of AC (hex) should be used.

In summary, the frame format including an Ethernet MAC header could
look like in Figure 1 (the CRC in the MAC trailer is not shown). The
Ethernet MAC header includes the length field, which is the length of
the ROHC header and payload plus the static LLC header. This means
the total per-packet overhead is 21 bytes, 18 bytes for the Ethernet
MAC header and trailer and three bytes for the LLC header carrying
the ROHC identification.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Destination MAC Address                     |
+                           +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           |                                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                                   +
|                     Source MAC Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       ROHC Length + 3     |   ROHC SAP    |   ROHC SAP    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   0x03 (UI)   |   ROHC header and payload...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

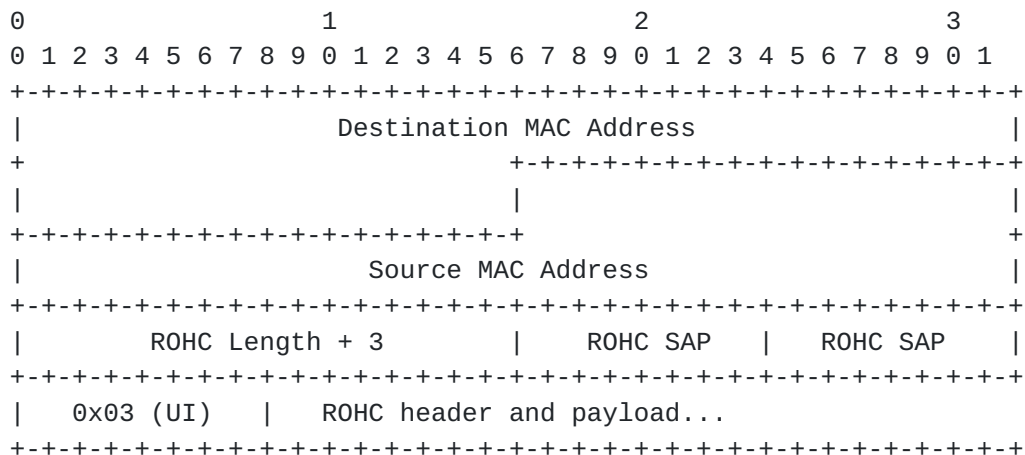   Figure 1: ROHC packet including Ethernet MAC header

5.1. ULE encapsulation

For IP over DVB, bridged frame encapsulation (type 0x0001) can be
used unchanged. If a more compact encoding (more like the ethertype
compatible formats) is required, the encapsulation as defined in
Figure 2 and Figure 3 can be used. (The type value is provisional
and needs to be defined in the ULE type registry.)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|       Length  (15b)        |          Type = 0x00AC          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Receiver Destination NPA Address  (6B)               |
+                             +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             |                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                                 +
|                                                               |
=                      ROHC header and payload...               =
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          (CRC-32)                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: ROHC packet in ULE (D=0)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|       Length  (15b)        |          Type = 0x00AC          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
=                      ROHC header and payload...               =
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          (CRC-32)                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
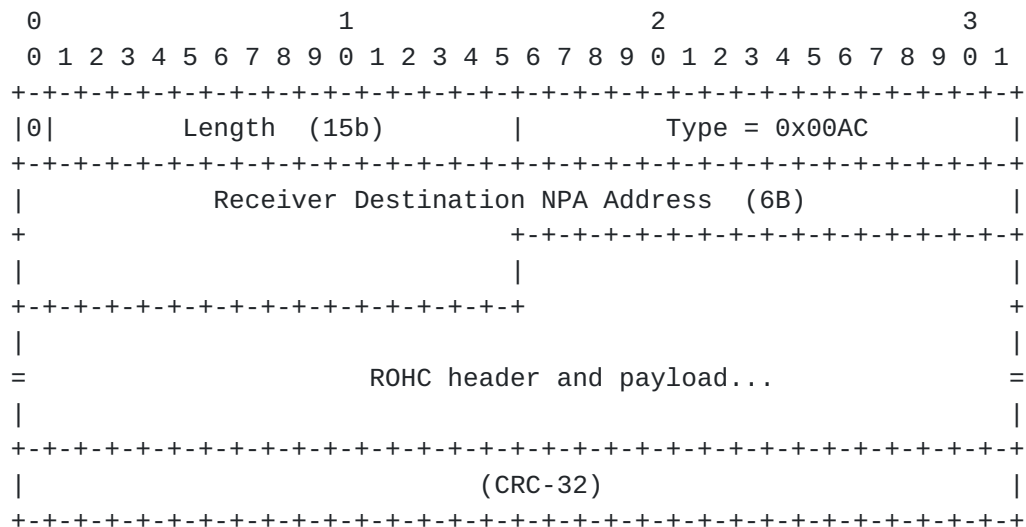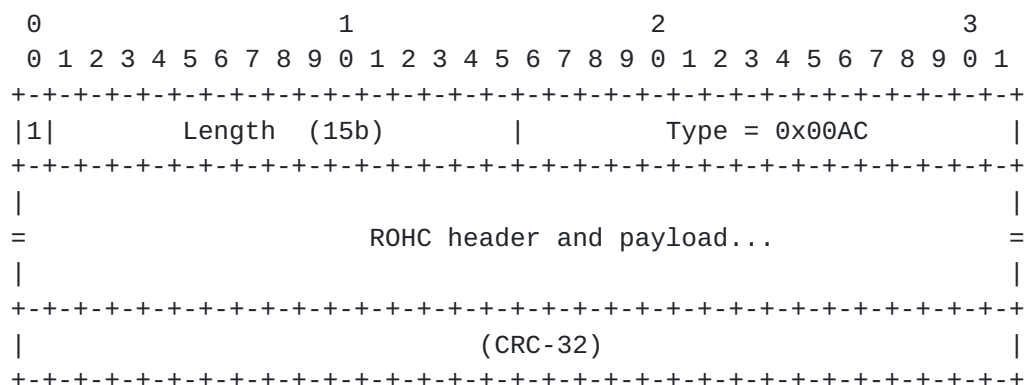
Figure 3: ROHC packet in ULE (D=1)

6. Negotiation

Negotiation of ROHC channels can either be piggy-backed on the
existing address resolution/neighbor discovery protocols or a
completely separate negotiation protocol can be used.

For IPv4, extending ARP sounds rather difficult at this point in the
evolution of this protocol. For IPv6, while ND is probably a more
extensible protocol, it is not clear that it is the right place for
negotiating link-layer characteristics.

Instead, a simple negotiation protocol should be defined that is
based on regularly probing the peer node for ROHC capability and
offering a capability set. A magic number scheme can be used both to
ensure liveness of the peer state assumed and as a basic security
measure.

The negotiation protocol should preferably share its encapsulation
with the ROHC encapsulation itself to ensure probes only arrive if
there is no obstacle to LLC-style frames. Additional checking could
be made part of the protocol that would detect common mistakes when
implementing IEEE 802 framing.

This specification therefore uses the ROHC encapsulation for also
carrying the negotiation payload. This is achieved by hijacking the
ROHC Add-CID packet types 11100001 to 11101111, see Figure 4; note
that R cannot be 0 (this would indicate a ROHC Padding byte).
Remaining_length gives the length of the negotiation payload and thus
echoes the LLC length (ROHC Length + 3) minus the 6 bytes consumed;
this serves as a check that the length was not damaged by a faulty
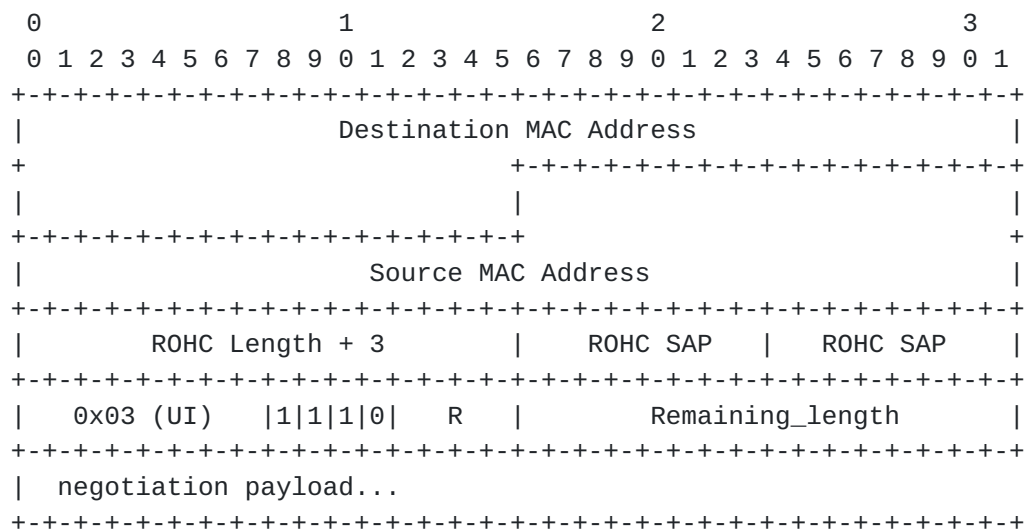IEEE 802 implementation. (The ULE encapsulations are defined
analogously.)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Destination MAC Address                     |
+                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                    Source MAC Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       ROHC Length + 3         |   ROHC SAP    |   ROHC SAP    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   0x03 (UI)   |1|1|1|0|   R   |        Remaining_length       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  negotiation payload...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Figure 4: ROHC negotiation packet including Ethernet MAC header

The negotiation payload defined in this specification looks exactly
like an RFC 3241 Robust Header Compression (ROHC) Option [RFC3241].
If R is odd, it indicates the receiving capability of the originator

of the negotiation payload. If R is even, it indicates the actual
values that will be used in sending ROHC packets by the originator.

For unicast packets and where bidirectional connectivity is
available, a ROHC capable receiver SHOULD occasionally send
negotiation solicitation packets with R=1 to known neighbors, e.g.
triggered by the reception of ARP or ND packets or of actual data
packets (these negotiation solicitation packets MUST be strictly rate
limited and MUST NOT be sent unless activity is detected from a
neighbor). A ROHC capable sender MAY then send negotiation
advertisement packets with R=2; once bidirectional advertisement has
been achieved, the ROHC capable receiver SHOULD answer with its own
actual values used in sending ROHC packets in negotiation
advertisement packets with R=4 (no longer sending any R=2 packets).
Only when the negotiation advertisement packet exchange has been
completed SHOULD the ROHC capable sender start sending actual ROHC
packets instead of IP packets encapsulated the usual way. In the
spirit of IPv6 Neighbor Unreachability Detection (NUD [RFC4861]), the
negotiation exchanges should be repeated whenever it is unclear
whether the ROHC packets are successfully decompressed.

For multicast packets or for unidirectional connectivity, a ROHC
capable sender SHOULD send packets with R=6 to the MAC-layer
multicast address. ROHC receivers MUST NOT answer. There is no way
for a multicast/unidirectional sender to ascertain its receivers
indeed all support ROHC and are reached by ROHC packets. (Extensions
to IGMP/MLD could be defined to remedy this.)

For ROHC over 802, LARGE_CIDS is always set. ROHC capability is
always indicated for both IPv4 and IPv6.

7. Security Considerations

Making a node believe its peer node is ROHC capable when it isn't is
one way to stage a denial of service attack, as is maliciously
tearing down ROHC state. The ROHC negotiation protocol probably
needs to have security that is commensurate to that of the address
resolution/neighbor discovery protocol in use. (Extensions to
ICMPv6/SEND could be defined to make ROHC negotiation more secure.)

The ROHC protocol itself is quite susceptible to attacks. To quote
RFC 3095 [RFC3095]:

   Denial-of-service attacks are possible if an intruder can
   introduce (for example) bogus STATIC, DYNAMIC or FEEDBACK packets
   onto the link and thereby cause compression efficiency to be
   reduced. However, an intruder having the ability to inject

arbitrary packets at the link layer in this manner raises
additional security issues that dwarf those related to the use of
header compression.

8. References

8.1. Normative References

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

[RFC3241]   Bormann, C., "Robust Header Compression (ROHC) over PPP",
            RFC 3241, April 2002.

[RFC4995]   Jonsson, L-E., Pelletier, G., and K. Sandlund, "The RObust
            Header Compression (ROHC) Framework", RFC 4995, July 2007.

8.2. Informative References

[RFC3095]   Bormann, C., Burmeister, C., Degermark, M., Fukushima, H.,
            Hannu, H., Jonsson, L-E., Hakenberg, R., Koren, T., Le,
            K., Liu, Z., Martensson, A., Miyazaki, A., Svanbro, K.,
            Wiebke, T., Yoshimura, T., and H. Zheng, "RObust Header
            Compression (ROHC): Framework and four profiles: RTP, UDP,
            ESP, and uncompressed", RFC 3095, July 2001.

[RFC3242]   Jonsson, L-E. and G. Pelletier, "RObust Header Compression
            (ROHC): A Link-Layer Assisted Profile for IP/UDP/RTP",
            RFC 3242, April 2002.

[RFC3408]   Liu, Z. and K. Le, "Zero-byte Support for Bidirectional
            Reliable Mode (R-mode) in Extended Link-Layer Assisted
            RObust Header Compression (ROHC) Profile", RFC 3408,
            December 2002.

[RFC3843]   Jonsson, L-E. and G. Pelletier, "RObust Header Compression
            (ROHC): A Compression Profile for IP", RFC 3843,
            June 2004.

[RFC4019]   Pelletier, G., "RObust Header Compression (ROHC): Profiles
            for User Datagram Protocol (UDP) Lite", RFC 4019,
            April 2005.

[RFC2516]   Mamakos, L., Lidl, K., Evarts, J., Carrel, D., Simone, D.,
            and R. Wheeler, "A Method for Transmitting PPP Over
            Ethernet (PPPoE)", RFC 2516, February 1999.

[RFC1042]   Postel, J. and J. Reynolds, "Standard for the transmission
            of IP datagrams over IEEE 802 networks", STD 43, RFC 1042,
            February 1988.

[RFC3759]   Jonsson, L-E., "RObust Header Compression (ROHC):
            Terminology and Channel Mapping Examples", RFC 3759,
            April 2004.

[RFC5225]   Pelletier, G. and K. Sandlund, "RObust Header Compression
            Version 2 (ROHCv2): Profiles for RTP, UDP, IP, ESP and
            UDP-Lite", RFC 5225, April 2008.

[RFC4996]   Pelletier, G., Sandlund, K., Jonsson, L-E., and M. West,
            "RObust Header Compression (ROHC): A Profile for TCP/IP
            (ROHC-TCP)", RFC 4996, July 2007.

[RFC4326]   Fairhurst, G. and B. Collini-Nocker, "Unidirectional
            Lightweight Encapsulation (ULE) for Transmission of IP
            Datagrams over an MPEG-2 Transport Stream (TS)", RFC 4326,
            December 2005.

[RFC2686]   Bormann, C., "The Multi-Class Extension to Multi-Link
            PPP", RFC 2686, September 1999.

[I-D.bormann-rohc-avt-crtp-profile]
            Bormann, C., "A ROHC Profile for CRTP (ROHC-CRTP)",
            draft-bormann-rohc-avt-crtp-profile-00 (work in progress),
            March 2007.

[refauth]   "IEEE Logical Link Control Address & Standard Group MAC
            Address Registration Authority",
            <http://standards.ieee.org/regauth/llc/index.html>.

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

Appendix A. Acknowledgements

The issue of enabling robust header compression over 802 networks has
been brought up repeatedly, e.g. by Kris Fleming in a contribution
called ROHCoWEM (draft-fleming-rohc-wireless-ethernet-med).   The
participant comments at the Atlanta IETF ROHC WG meeting (November
2002) provided the basis for the analytical part of this document.

I would like to thank Bernard Aboba, Pedro Fortuna, Stephen McCann,
and Mike Morton for reviewing earlier versions of this document and

suppyling extremely useful comments. In particular, Bernard provided
a number of comments that proved useful for fleshing out Section 2.3.
(All errors, however, are mine.)

Author's Address

   Carsten Bormann
   Universitaet Bremen TZI
   Postfach 330440
   Bremen D-28334
   Germany

   Phone: +49 421 218 63921
   Fax: +49 421 218 7000
   Email: cabo@tzi.org