                    IP Header Compression over PPP

Status of this Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

Abstract

   This document describes an option for negotiating the use of header
   compression on IP datagrams transmitted over the Point-to-Point
   Protocol (RFC 1661). It defines extensions to the PPP Control
   Protocols for IPv4 and IPv6 (RFC 1332, RFC 2472). Header compression
   may be applied to IPv4 and IPv6 datagrams in combination with TCP,
   UDP and RTP transport protocols as specified in RFC 2507, RFC 2508
   and RFC 3545.

1. Introduction

   The IP Header Compression (IPHC) defined in [RFC2507] may be used for
   compression of both IPv4 and IPv6 datagrams or packets encapsulated
   with multiple IP headers. IPHC is also capable of compressing both
   TCP and UDP transport protocol headers. The IP/UDP/RTP header
   compression defined in [RFC2508] and [RFC3545] fits within the
   framework defined by IPHC so that it may also be applied to both IPv4
   and IPv6 packets.

In order to establish compression of IP datagrams sent over a PPP
link each end of the link must agree on a set of configuration
parameters for the compression. The process of negotiating link
parameters for network layer protocols is handled in PPP by a family
of network control protocols (NCPs). Since there are separate NCPs
for IPv4 and IPv6, this document defines configuration options to be
used in both NCPs to negotiate parameters for the compression scheme.

This document obsoletes RFC 2509, adding two new suboptions to the IP
header compression configuration option. One suboption negotiates
usage of Enhanced RTP-Compression (specified in [RFC3545]), and the
other suboption negotiates header compression for only TCP or only
non-TCP packets.

IPHC relies on the link layer's ability to indicate the types of
datagrams carried in the link layer frames. In this document nine
new types for the PPP Data Link Layer Protocol Field are defined
along with their meaning.

In general, header compression schemes that use delta encoding of
compressed packets require that the lower layer does not reorder
packets between compressor and decompressor. IPHC uses delta
encoding of compressed packets for TCP and RTP. The IPHC
specification [RFC2507] includes methods that allow link layers that
may reorder packets to be used with IPHC. Since PPP does not reorder
packets these mechanisms are disabled by default. 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.

2. Configuration Option

   This document specifies a new compression protocol value for the IPCP
   IP-Compression-Protocol option as specified in [RFC1332]. The new
   value and the associated option format are described in section 2.1.

   The option format is structured to allow future extensions to the
   IPHC scheme.

   NOTE: The specification of link and network layer parameter
      negotiation for PPP [RFC1661], [RFC1331], [RFC1332] does not
      prohibit multiple instances of one configuration option but states
      that the specification of a configuration option must explicitly
      allow multiple instances. [RFC3241] updates RFC 1332 by
      explicitly allowing the sending of multiple instances of the IP-
      Compression-Protocol configuration option, each with a different
      value for IP-Compression-Protocol. Each type of compression
      protocol may independently establish its own parameters.

      NOTE: [RFC1332] is not explicit about whether the option
         negotiates the capabilities of the receiver or of the sender. In
         keeping with current practice, we assume that the option describes
         the capabilities of the decompressor (receiving side) of the peer
         that sends the Config-Req.

2.1. Configuration Option Format

      Both the network control protocol for IPv4, IPCP [RFC1332] and the
      IPv6 NCP, IPV6CP [RFC2472] may be used to negotiate IP Header
      Compression parameters for their respective protocols. The format of
      the configuration option is the same for both IPCP and IPV6CP.

      Description

         This NCP configuration option is used to negotiate parameters for
         IP Header Compression. Successful negotiation of parameters
         enables the use of Protocol Identifiers FULL_HEADER,
         COMPRESSED_TCP, COMPRESSED_TCP_NODELTA, COMPRESSED_NON_TCP and
         CONTEXT_STATE as specified in [RFC2507]. The option format is
         summarized below. The fields are transmitted from left to right.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |     Type      |     Length    |    IP-Compression-Protocol    |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |           TCP_SPACE           |         NON_TCP_SPACE         |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |         F_MAX_PERIOD          |          F_MAX_TIME           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          MAX_HEADER           |         suboptions...         |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
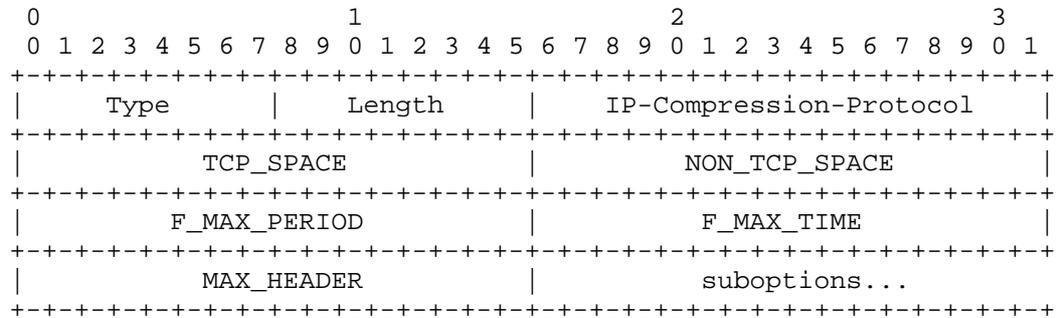
      Type
         2

      Length
         >= 14

      The length may be increased if the presence of additional
      parameters is indicated by additional suboptions.

      IP-Compression-Protocol
         0061 (hex)

   TCP_SPACE
      The TCP_SPACE field is two octets and indicates the maximum value
      of a context identifier in the space of context identifiers
      allocated for TCP.

         Suggested value: 15

      TCP_SPACE must be at least 0 and at most 255 (the value 0 implies
      having one context).

   NON_TCP_SPACE
      The NON_TCP_SPACE field is two octets and indicates the maximum
      value of a context identifier in the space of context identifiers
      allocated for non-TCP. These context identifiers are carried in
      COMPRESSED_NON_TCP, COMPRESSED_UDP and COMPRESSED_RTP packet
      headers.

         Suggested value: 15

      NON_TCP_SPACE must be at least 0 and at most 65535 (the value 0
      implies having one context).

   F_MAX_PERIOD
      Maximum interval between full headers. No more than F_MAX_PERIOD
      COMPRESSED_NON_TCP headers may be sent between FULL_HEADER
      headers.

         Suggested value: 256

      A value of zero implies infinity, i.e. there is no limit to the
      number of consecutive COMPRESSED_NON_TCP headers.

   F_MAX_TIME
      Maximum time interval between full headers. COMPRESSED_NON_TCP
      headers may not be sent more than F_MAX_TIME seconds after sending
      the last FULL_HEADER header.

         Suggested value: 5 seconds

      A value of zero implies infinity.

   MAX_HEADER
      The largest header size in octets that may be compressed.

         Suggested value: 168 octets

The value of MAX_HEADER should be large enough so that at least
the outer network layer header can be compressed. To increase
compression efficiency MAX_HEADER should be set to a value large
enough to cover common combinations of network and transport layer
headers.

suboptions
    The suboptions field consists of zero or more suboptions. Each
    suboption consists of a type field, a length field and zero or
    more parameter octets, as defined by the suboption type. The
    value of the length field indicates the length of the suboption in
    its entirety, including the lengths of the type and length fields.

```
    0                   1                   2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Length    |  Parameters...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
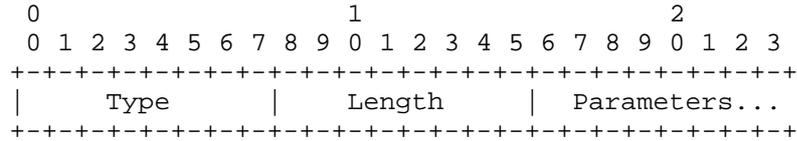
2.2. RTP-Compression Suboption

The RTP-Compression suboption is included in the NCP IP-Compression-
Protocol option for IPHC if IP/UDP/RTP compression is to be enabled.

Inclusion of the RTP-Compression suboption enables use of additional
Protocol Identifiers COMPRESSED_RTP and COMPRESSED_UDP along with
additional forms of CONTEXT_STATE as specified in [RFC2508].

Description

    Enable use of Protocol Identifiers COMPRESSED_RTP, COMPRESSED_UDP
    and CONTEXT_STATE as specified in [RFC2508].

```
        0                   1
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |     Type      |     Length    |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
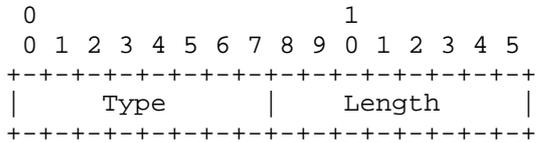
    Type
        1

    Length
        2

2.3. Enhanced RTP-Compression Suboption

   To use the enhanced RTP header compression defined in [RFC3545], a
   new sub-option 2 is added. Sub-option 2 is negotiated instead of,
   not in addition to, sub-option 1.

   Description

      Enable use of Protocol Identifiers COMPRESSED_RTP and
      CONTEXT_STATE as specified in [RFC2508]. In addition, enable use
      of [RFC3545] compliant compression including the use of Protocol
      Identifier COMPRESSED_UDP with additional flags and use of the C
      flag with the FULL_HEADER Protocol Identifier to indicate use of
      HDRCKSUM with COMPRESSED_RTP and COMPRESSED_UDP packets.

```
         0                   1
         0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |      Type      |     Length    |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
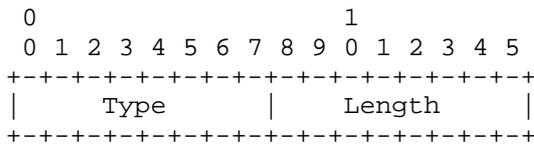
      Type
         2

      Length
         2

2.4. Negotiating header compression for only TCP or only non-TCP
      packets

   In RFC 2509 it was not possible to negotiate only TCP header
   compression or only non-TCP header compression because a value of 0
   in the TCP_SPACE or the NON_TCP_SPACE fields actually means that 1
   context is negotiated.

   A new suboption 3 is added to allow specifying that the number of
   contexts for TCP_SPACE or NON_TCP_SPACE is zero, disabling use of the
   corresponding compression.

   Description

   Enable header compression for only TCP or only non-TCP packets.

```
    0                   1                   2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |   Parameter   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

      Type
         3

      Length
         3

      Parameter

      The parameter is 1 byte with one of the following values:

      1 = the number of contexts for TCP_SPACE is 0
      2 = the number of contexts for NON_TCP_SPACE is 0

   This suboption overrides the values that were previously assigned to
   TCP_SPACE and NON_TCP_SPACE in the IP Header Compression option.

   If suboption 3 is included multiple times with parameter 1 and 2,
   compression is disabled for all packets.

3. Multiple Network Control Protocols

   The IPHC protocol is able to compress both IPv6 and IPv4 datagrams.
   Both IPCP and IPV6CP are able to negotiate option parameter values
   for IPHC. These values apply to the compression of packets where the
   outer header is an IPv4 header and an IPv6 header, respectively.

3.1. Sharing Context Identifier Space

   For the compression and decompression of IPv4 and IPv6 datagram
   headers the context identifier space is shared. While the parameter
   values are independently negotiated, sharing the context identifier
   spaces becomes more complex when the parameter values differ. Since
   the compressed packets share context identifier space, the
   compression engine must allocate context identifiers out of a common
   pool; for compressed packets, the decompressor has to examine the
   context state to determine what parameters to use for decompression.

Context identifier spaces are not shared between TCP and non-
TCP/UDP/RTP. Doing so would require additional mechanisms to ensure
that no error can occur when switching from using a context
identifier for TCP to non-TCP.

4. Demultiplexing of Datagrams

The IPHC specification [RFC2507] defines four header formats for
different types of compressed headers. They are compressed TCP,
compressed TCP with no delta encoding, compressed non-TCP with 8 bit
CID and compressed non-TCP with 16 bit CID. The two non-TCP formats
may be distinguished by their contents so both may use the same
link-level identifier. A fifth header format, the full header is
distinct from a regular header in that it carries additional
information to establish shared context between the compressor and
decompressor.

The specification of IP/UDP/RTP Header Compression [RFC2508] defines
four additional formats of compressed headers. They are for
compressed UDP and compressed RTP (on top of UDP), both with either
8- or 16-bit CIDs. In addition, there is an explicit error message
from the decompressor to the compressor.

The link layer must be able to indicate these header formats with
distinct values. Nine PPP Data Link Layer Protocol Field values are
specified below.

FULL_HEADER
    The frame contains a full header as specified in [RFC2508] Section
    3.3.1. This is the same as the FULL_HEADER specified in [RFC2507]
    Section 5.3.
        Value: 0061 (hex)

COMPRESSED_TCP
    The frame contains a datagram with a compressed header with the
    format as specified in [RFC2507] Section 6a.
        Value: 0063 (hex)

COMPRESSED_TCP_NODELTA
    The frame contains a datagram with a compressed header with the
    format as specified in [RFC2507] Section 6b.
        Value: 2063 (hex)

COMPRESSED_NON_TCP
    The frame contains a datagram with a compressed header with the
    format as specified in either Section 6c or Section 6d of
    [RFC2507].
        Value: 0065 (hex)

COMPRESSED_RTP_8
   The frame contains a datagram with a compressed header with the
   format as specified in [RFC2508] Section 3.3.2, using 8-bit CIDs.
      Value: 0069 (hex)

COMPRESSED_RTP_16
   The frame contains a datagram with a compressed header with the
   format as specified in [RFC2508] Section 3.3.2, using 16-bit CIDs.
      Value: 2069 (hex)

COMPRESSED_UDP_8
   The frame contains a datagram with a compressed header with the
   format as specified in [RFC2508] Section 3.3.3 or as specified in
   [RFC3545] Section 2.1, using 8-bit CIDs.
      Value: 0067 (hex)

COMPRESSED_UDP_16
   The frame contains a datagram with a compressed header with the
   format as specified in [RFC2508] Section 3.3.3 or as specified in
   [RFC3545] Section 2.1, using 16-bit CIDs.
      Value: 2067 (hex)

CONTEXT_STATE
   The frame is a link-level message sent from the decompressor to
   the compressor as specified in [RFC2508] Section 3.3.5.
      Value: 2065 (hex)

5. Changes from RFC 2509

   Two new suboptions are specified. See Sections 2.3 and 2.4.

6. References

6.1. Normative References

   [RFC1144] Jacobson, V., "Compressing TCP/IP Headers for low-speed
              serial links", RFC 1144, February 1990.

   [RFC1332] McGregor, G., "The PPP Internet Protocol Control Protocol
              (IPCP)", RFC 1332, May 1992.

   [RFC2472] Haskin, D. and E. Allen, "IP Version 6 over PPP", RFC
              2472, December 1998.

   [RFC2507] Degermark, M., Nordgren, B. and S. Pink, "Header
              Compression for IP", RFC 2507, February 1999.

   [RFC2508]   Casner, S. and V. Jacobson, "Compressing IP/UDP/RTP
               Headers for Low-Speed Serial Links", RFC 2508, February
               1999.

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

   [RFC3545]   Koren, T., Casner, S., Geevarghese, J., Thompson, B. and
               P. Ruddy, "Enhanced Compressed RTP (CRTP) for Links with
               High Delay, Packet Loss and Reordering", RFC 3545, July
               2003.

6.2. Informative References

   [RFC1661]   Simpson, W., Ed., "The Point-To-Point Protocol (PPP)", STD
               51, RFC 1661, July 1994.

   [RFC2434]   Narten, T. and H. Alvestrand, "Guidelines for Writing an
               IANA Considerations Section in RFCs", BCP 26, RFC 2434,
               October 1998.

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

   [RFC3550]   Schulzrinne, H., Casner, S., Frederick, R. and V.
               Jacobson, "RTP: A Transport Protocol for Real-Time
               Applications", RFC 3550, July 2003.

7. IANA Considerations

   This document does not require any additional allocations from
   existing namespaces in the IANA Point-to-Point Protocol Field
   Assignments registry. However, there are three namespaces that were
   defined by RFC 1332, RFC 2472, and RFC 2509 but not created in the
   registry. Those three namespaces, described below, have been added
   to the PPP registry. This document specifies two additional
   allocations in the third one.

   Section 3.2 of RFC 1332 specifies an IP-Compression-Protocol
   Configuration Option for the PPP IP Control Protocol and defines one
   value for the IP-Compression-Protocol type field in that option. An
   IANA registry has been created to allocate additional values for that
   type field. As stated in RFC 1332, the values for the IP-
   Compression-Protocol type field are always the same as the (primary)
   PPP DLL Protocol Number assigned to packets of the particular
   compression protocol. Assignment of additional IP-Compression-
   Protocol type values is through the IETF consensus procedure as
   specified in [RFC2434].

Section 4.2 of RFC 2472 specifies an IPv6-Compression-Protocol
Configuration Option for the PPP IPv6 Control Protocol and defines
one value for the IPv6-Compression-Protocol type field in that
option. An IANA registry has been created to allocate additional
values for that type field. The IPv6-Compression-Protocol
Configuration Option has the same structure as the IP-Compression-
Protocol Configuration Option defined in RFC 1332, but the set of
values defined for the type field may be different. As stated in RFC
2472, the values for the IPv6-Compression-Protocol type field are
always the same as the (primary) PPP DLL Protocol Number assigned to
packets of the particular compression protocol. Assignment of
additional IPv6-Compression-Protocol type values is through the IETF
consensus procedure as specified in [RFC2434].

Section 2.1 of RFC 2509 specifies an additional type value to be
registered for both the IP-Compression-Protocol Configuration Option
and the IPv6-Compression-Protocol Configuration Option to indicate
use of the "IP Header Compression" protocol. The specification of
that type value is repeated in Section 2.1 of this document which
obsoletes RFC 2509. In conjunction with the additional type value,
the format for the variable-length option is specified. The format
includes a suboption field that may contain one or more suboptions.
Each suboption begins with a suboption type value. An IANA registry
has been created for the suboption type values; and is titled, "IP
Header Compression Configuration Option Suboption Types".

Section 2.2 of RFC 2509 (and this document) defines one suboption
type. Sections 2.3 and 2.4 of this document define two additional
suboption types. It is expected that the number of additional
suboptions that will need to be defined is small. Therefore, anyone
wishing to define new suboptions is required to produce a revision of
this document to be vetted through the normal Internet Standards
process, as specified in [RFC2434].

RFC 2509 also defines nine PPP Data Link Layer Protocol Field values
which are already listed in the IANA registry of Point-to-Point
Protocol Field Assignments. Section 4 of this document repeats the
specification of those values without change.

8. Security Considerations

Negotiation of the option defined here imposes no additional security
considerations beyond those that otherwise apply to PPP [RFC1661].

The use of header compression can, in rare cases, cause the
misdelivery of packets. If necessary, confidentiality of packet
contents should be assured by encryption.

Encryption applied at the IP layer (e.g., using IPSEC mechanisms)
precludes header compression of the encrypted headers, though
compression of the outer IP header and authentication/security
headers is still possible as described in [RFC2507]. For RTP
packets, full header compression is possible if the RTP payload is
encrypted by itself without encrypting the UDP or RTP headers, as
described in [RFC3550]. This method is appropriate when the UDP and
RTP header information need not be kept confidential.

9. Intellectual Property Rights Notice

The IETF takes no position regarding the validity or scope of any
intellectual property or other rights that might be claimed to
pertain to the implementation or use of the technology described in
this document or the extent to which any license under such rights
might or might not be available; neither does it represent that it
has made any effort to identify any such rights. Information on the
IETF's procedures with respect to rights in standards-track and
standards-related documentation can be found in BCP-11. Copies of
claims of rights made available for publication and any assurances of
licenses to be made available, or the result of an attempt made to
obtain a general license or permission for the use of such
proprietary rights by implementors or users of this specification can
be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any
copyrights, patents or patent applications, or other proprietary
rights which may cover technology that may be required to practice
this standard. Please address the information to the IETF Executive
Director.

10. Acknowledgements

   Mathias Engan was the primary author of RFC 2509, of which this
   document is a revision.

11. Authors' Addresses

   Tmima Koren
   Cisco Systems, Inc.
   170 West Tasman Drive
   San Jose, CA 95134-1706
   United States

   EMail: tmima@cisco.com

   Stephen L. Casner
   Packet Design
   3400 Hillview Avenue, Building 3
   Palo Alto, CA 94304
   United States

   EMail: casner@packetdesign.com

   Carsten Bormann
   Universitaet Bremen FB3 TZI
   Postfach 330440
   D-28334 Bremen, GERMANY

   Phone: +49.421.218-7024
   Fax: +49.421.218-7000
   EMail: cabo@tzi.org

12. Full Copyright Statement

Acknowledgement