Network Working Group
Internet-Draft
Expires: June 7, 2010

E. Ertekin
C. Christou
R. Jasani
Booz Allen Hamilton
T. Kivinen
Safenet, Inc.
C. Bormann
Universitaet Bremen TZI

December 4, 2009

IKEv2 Extensions to Support Robust Header Compression over IPsec (ROHCoIPsec)
draft-ietf-rohc-ikev2-extensions-hcoipsec-10

Abstract

In order to integrate Robust Header Compression (ROHC) with IPsec, a mechanism is needed to signal ROHC channel parameters between end-points. Internet Key Exchange (IKE) is a mechanism which can be leveraged to exchange these parameters. This document specifies extensions to IKEv2 that will allow ROHC and its associated channel parameters to be signaled for IPsec security associations (SAs).

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 June 7, 2010.

Copyright Notice

Table of Contents

1. Introduction

Increased packet header overhead due to IPsec [IPSEC] can result in the inefficient utilization of bandwidth. Coupling ROHC [ROHC] with IPsec offers an efficient way to transfer protected IP traffic.

ROHCoIPsec [ROHCOIPSEC] requires configuration parameters to be initialized at the compressor and decompressor. Current specifications for hop-by-hop ROHC negotiate these parameters through a link-layer protocol such as Point-to-Point Protocol (PPP) (i.e. ROHC over PPP [ROHC-PPP]). Since key exchange protocols (e.g. IKEv2 [IKEV2]) can be used to dynamically establish parameters between IPsec peers, this document defines extensions to IKEv2 to signal ROHC parameters for ROHCoIPsec.

2. Terminology

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 [BRA97].

3. ROHC Channel Initialization for ROHCoIPsec

The following subsections define extensions to IKEv2 which enables an initiator and a responder to signal parameters required to establish a ROHC channel for a ROHCoIPsec session.

3.1. ROHC_SUPPORTED Notify Message

ROHC channel parameters MUST be signaled at either the establishment or rekeying of a Child SA. Specifically, a new Notify message type is used during the IKE_AUTH and CREATE_CHILD_SA exchanges to convey these parameters.

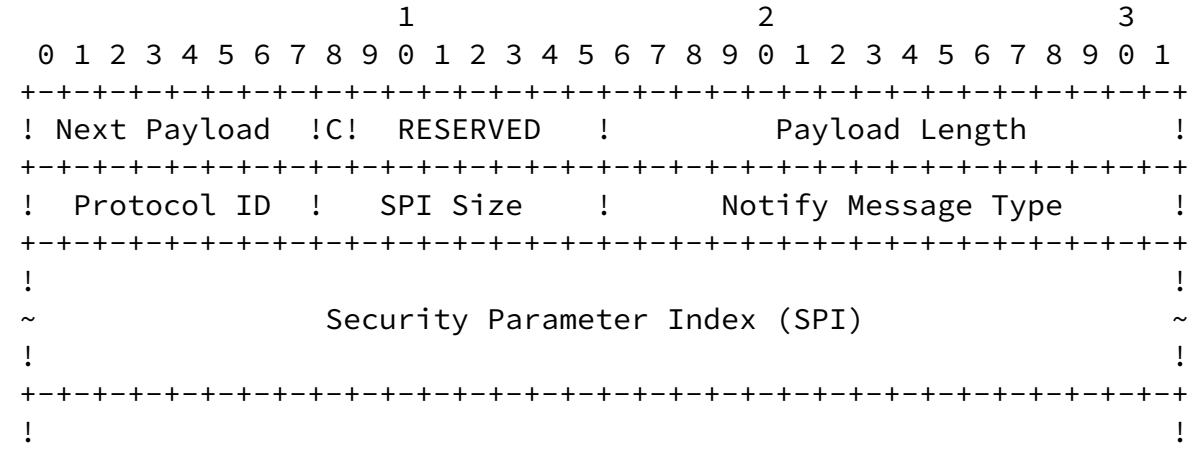
The Notify payload sent by the initiator MUST contain the channel parameters for the ROHC session. These parameters indicate the capabilities of the ROHC decompressor at the initiator. Upon receipt of the initiator's request, the responder will either ignore the payload (if it doesn't support ROHC or the proposed parameters) or respond with a Notify payload that contains its own ROHC channel parameters.

Note that only one Notify payload is used to convey ROHC parameters. If multiple Notify payloads containing ROHC parameters are received, all but the first such Notify payload MUST be dropped. If the initiator does not receive a Notify Payload with the responder's ROHC

channel parameters, ROHC MUST NOT be enabled on the Child SA.

A new Notify Message Type value, denoted ROHC_SUPPORTED, indicates that the Notify payload is conveying ROHC channel parameters (Section 4).

The Notify Payload (defined in RFC 4306 [IKEV2]) is illustrated in Figure 1.



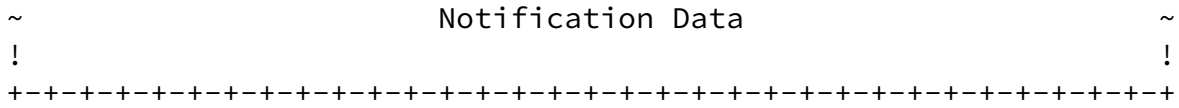


Figure 1. Notify Payload format.

The fields of the Notify Payload are set as follows:

Next Payload (1 octet)
Identifier for the payload type of the next payload in the message. Further details can be found in RFC 4306 [IKEV2].

Critical (1 bit)
Since all IKEv2 implementations support the Notify Payload, this value MUST be set to zero.

Payload Length (2 octets)
As defined in RFC 4306 [IKEV2], this field indicates the length of the current payload, including the generic payload header.

Protocol ID (1 octet)
Since this Notification message is used during the creation of a Child SA, this field MUST be set to zero.

SPI Size (1 octet)

This value MUST be set to zero, since no SPI is applicable (ROHC parameters are set at SA creation, thus the SPI has not been defined).

Notify Message Type (2 octets)
This field MUST be set to ROHC_SUPPORTED.

Security Parameter Index (SPI)
Since the SPI Size field is 0, this field MUST NOT be transmitted.

Notification Data (variable)
This field MUST contain at least three ROHC Attributes (Section 3.1.1).

3.1.1. ROHC Attributes

The ROHC_SUPPORTED Notify message is used to signal channel parameters between ROHCoIPsec compressor and decompressor. The message contains a list of "ROHC Attributes" which contain the parameters required for the ROHCoIPsec session.

The format for signaling ROHC Attributes takes a similar format to the Transform Attributes described in Section 3.3.5 of RFC 4306[IKEV2]. The ROHC Attribute is shown in Figure 2.

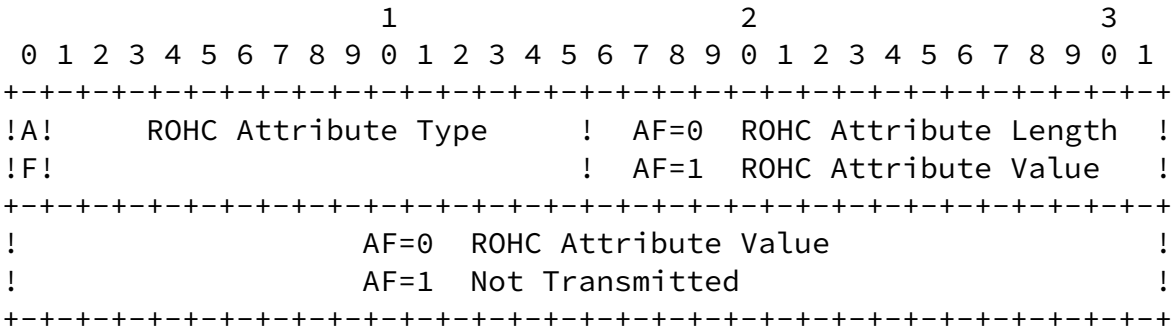


Figure 2. Format of the ROHC Attribute.

- Attribute Format (AF) (1 bit) - If the AF bit is a zero (0), then the ROHC Attribute is expressed in a Type/Length/Value format. If the AF bit is a one (1), then the ROHC attribute is expressed in a Type/Value (TV) format.
- ROHC Attribute Type (15 bits) - Unique identifier for each type of ROHC attribute (Section 3.1.2).
- ROHC Attribute Length (2 octets) - Length (in octets) of the Attribute Value. When the AF bit is a one (1), the ROHC Attribute Value is 2 octets and the ROHC Attribute Length field is not present.
- ROHC Attribute Value (variable length) - Value of the ROHC Attribute associated with the ROHC Attribute Type. If the AF bit is a zero (0), this field's length is defined by the ROHC Attribute Length field. If the AF bit is a one (1), the length of the ROHC Attribute Value is 2 octets.

3.1.2. ROHC Attribute Types

This section describes five ROHC Attribute Types: MAX_CID, ROHC_PROFILE, ROHC_INTEG, ROHC_ICV_LEN, and MRRU. The value allocated for each ROHC Attribute Type is specified in Section 4.

MAX_CID (Maximum Context Identifier, AF = 1)

The MAX_CID attribute is a mandatory attribute. Exactly one MAX_CID attribute MUST be sent. The MAX_CID field indicates the maximum value of a context Identifier supported by the ROHCoIPsec decompressor. This attribute value is two octets in length. The range of values for MAX_CID MUST be at least 0 and at most 16383 (the value 0 implies having one context). The recipient of the MAX_CID Attribute MUST only use up to MAX_CID context identifiers for compression.

Note that the MAX_CID parameter is a one-way notification (i.e., the sender of the attribute indicates what it can handle to the other end); therefore, different values for MAX_CID may be announced in each direction.

ROHC_PROFILE (ROHC Profile, AF = 1)

The ROHC_PROFILE attribute is a mandatory attribute. Each ROHC_PROFILE attribute has a fixed length of 4 octets, and its attribute value is a two-octet long ROHC Profile Identifier [ROHCPROF]. There MUST be at least one ROHC_PROFILE attribute included in the ROHC_SUPPORTED Notify Message. If multiple ROHC_PROFILE attributes are sent, the order is arbitrary. The recipient of a ROHC_PROFILE attribute(s) MUST only use the profile(s) proposed for compression.

Several common profiles are defined in RFC 3095 [ROHCV1] and 5225 [ROHCV2]. Note, however, that two versions of the same profile MUST NOT be signaled. For example, if a ROHCoIPsec decompressor supports both ROHCv1 UDP (0x0002) and ROHCv2 UDP (0x0102), both profiles MUST NOT be signaled. This restriction is needed, as packets compressed by ROHC express only the 8 least significant bits of the profile identifier; since the 8 least significant bits for corresponding profiles in ROHCv1 and ROHCv2 are identical, the decompressor is not capable of determining the ROHC version that was used to compress the packet.

Note that the ROHC_PROFILE attribute is a one-way notification;

therefore, different values for ROHC_PROFILE may be announced in each direction.

ROHC_INTEG (Integrity Algorithm for Verification of Decompressed Headers, AF = 1)

The ROHC_INTEG attribute is a mandatory attribute. There MUST be at least one ROHC_INTEG attribute contained within the ROHC_SUPPORTED Notify message. The attribute value contains the identifier of an integrity algorithm that is used to ensure the integrity of the decompressed packets (i.e. ensure that the decompressed packet headers are identical to the original packet headers prior to compression).

Authentication algorithms that MUST be supported are specified in the "Authentication Algorithms" table in section 3.1.1 ("ESP Encryption and Authentication Algorithms") of RFC 4835 [CRYPTO-ALG] (or its successor).
The integrity algorithm is represented by a two octet value that corresponds to the value listed in the IKEv2 Parameters registry [IKEV2-PARA] "For Transform Type 3 (Integrity Algorithm)" section.
Upon receipt of the ROHC_INTEG attribute(s), the responder MUST select exactly one of the proposed algorithms; the chosen value is sent back in the ROHC_SUPPORTED Notify message returned by the responder to the initiator. The selected integrity algorithm MUST be used in both directions. If the responder does not accept any of the algorithms proposed by the initiator, ROHC MUST NOT be enabled on the SA.

It is noted that:

1. The keys (one for each direction) for this Integrity Algorithm are derived from the IKEv2 KEYMAT (see [IKEV2], Section 2.17). For the purposes of this key derivation, ROHC is considered to be an IPsec protocol. When a ROHC-enabled CHILD_SA is rekeyed, the key associated with this integrity algorithm is rekeyed as well.
2. A ROHCoIPsec initiator MAY signal a value of zero (0x0000) in a ROHC_INTEG attribute. This corresponds to "NONE" in the Integrity Algorithm Transform ID registry. The ROHCoIPsec responder MAY select this value by responding to the initiator with a ROHC_INTEG attribute of zero (0x0000). In this scenario, no integrity algorithm is applied in either direction.
3. The ROHC_INTEG attribute is a parameter that is negotiated between two ends. In other words, the initiator indicates what it supports; the responder selects one of the ROHC_INTEG values proposed, and sends the selected value to the initiator.

ROHC_ICV_LEN (Integrity Algorithm Length, AF = 1)
The ROHC_ICV_LEN attribute is an optional attribute. There MAY be zero or one ROHC_ICV_LEN attribute contained within the ROHC_SUPPORTED Notify message. The attribute specifies the number of Integrity Check Value (ICV) octets the sender expects to receive on incoming ROHC packets. The ICV of the negotiated ROHC_INTEG algorithms MUST be truncated to ROHC_ICV_LEN bytes by taking the first ROHC_ICV_LEN bytes of the output. Both the initiator and responder announce a single value for their own ICV length. The recipient of the ROHC_ICV_LEN attribute MUST truncate the ICV to the length contained in the message. If the value of the ROHC_ICV_LEN attribute is zero, then an ICV MUST NOT be sent. If no ROHC_ICV_LEN attribute is sent at all or the ROHC_ICV_LEN is larger than the length of the ICV of selected algorithm, then the full ICV length as specified by the ROHC_INTEG algorithm MUST be sent.

Note that the ROHC_ICV_LEN attribute is a one-way notification; therefore, different values for ROHC_ICV_LEN may be announced in each direction.

MRRU (Maximum Reconstructed Reception Unit, AF = 1)
The MRRU attribute is an optional attribute. There MAY be zero or one MRRU attribute contained within the ROHC_SUPPORTED Notify message. The attribute value is two octets in length. The attribute specifies the size of the largest reconstructed unit in octets that the ROHCoIPsec decompressor is expected to reassemble from ROHC segments (see Section 5.2.5 of [ROHCV1]). This size includes the CRC, and the ROHC ICV. If MRRU is 0 or if no MRRU attribute is sent, segment headers MUST NOT be transmitted on the ROHCoIPsec channel.

Note that the MRRU attribute is a one-way notification; therefore, different values for MRRU may be announced in each direction.

If an unknown ROHC Attribute Type Value is received, it MUST be silently ignored.

3.2. ROHC Channel Parameters that are Implicitly Set

The following ROHC channel parameters MUST NOT be signaled:

- LARGE_CIDS: This value is implicitly determined by the value of MAX_CID (e.g. if MAX_CID is <= 15, LARGE_CIDS is assumed to be 0).
- FEEDBACK_FOR: When a pair of SAs are created (one in each direction), the ROHC channel parameter FEEDBACK_FOR MUST be set implicitly to the other SA of the pair (i.e. the SA pointing in

the reverse direction).

4. Security Considerations

The ability to negotiate the length of the ROHC ICV may introduce vulnerabilities to the ROHCoIPsec protocol. Specifically, the capability to signal a short ICV length may result in scenarios where erroneous packets are forwarded into the protected domain. This security consideration is documented in further detail in Section 6.1.4 of [ROHCOIPSEC] and Section 5 of [IPSEC-ROHC].

This security consideration can be mitigated by using longer ICVs, but this comes at the cost of additional overhead, which reduces the overall benefits offered by ROHCoIPsec.

5. IANA Considerations

This document defines a new Notify Message (Status Type). Therefore, IANA is requested to allocate one value from the IKEv2 Notify Message registry to indicate ROHC_SUPPORTED. Note that, since this Notify Message is a Status Type, values ranging from 0 to 16383 must not be allocated for ROHC_SUPPORTED.

In addition, IANA is requested to create a new "ROHC Attribute Types" registry in the IKEv2 Parameters Registry [IKEV2-PARA]. Within the "ROHC Attribute Types" registry, this document allocates the following values:

Registry:

Value	ROHC Attribute Type	Format	Reference
0	RESERVED		[rfcThis]
1	Maximum Context Identifier (MAX_CID)	TV	[rfcThis]
2	ROHC Profile (ROHC_PROFILE)	TV	[rfcThis]
3	ROHC Integrity Algorithm (ROHC_INTEG)	TV	[rfcThis]
4	ROHC ICV Length in bytes (ROHC_ICV_LEN)	TV	[rfcThis]
5	Maximum Reconstructed Reception Unit (MRRU)	TV	[rfcThis]
6-16383	Unassigned		[rfcThis]
16384-32767	Private use		[rfcThis]

Following the policies outlined in [IANA-CONSIDERATIONS], the IANA policy for assigning new values for the ROHC Attribute Types registry shall be Designated Expert.

For registration requests, the responsible IESG area director will appoint the Designated Expert during the IETF last call. The intention is that any allocation will be accompanied by a published RFC. The Designated expert will post a request to both the rohc and ipsec mailing lists (or a successor designated by the Area Director)

for comment and review. Before expiration of the IETF last call, the Designated Expert will either approve or deny the registration request and publish a notice of the decision to both mailing lists (or their successors), as well as informing IANA. A denial notice must be justified by an explanation.

6. Acknowledgments

The authors would like to thank Mr. Sean O'Keeffe, Mr. James Kohler, and Ms. Linda Noone of the Department of Defense, as well as Mr. Rich Espy of OPnet for their contributions and support in the development of this document.

The authors would also like to thank Mr. Yoav Nir, and Mr. Robert A Stangarone Jr.: both served as committed document reviewers for this specification.

In addition, the authors would like to thank the following for their numerous reviews and comments to this document:

- Mr. Magnus Westerlund
- Dr. Stephen Kent
- Mr. Lars-Erik Jonsson
- Mr. Pasi Eronen
- Dr. Jonah Pezeshki
- Mr. Carl Knutsson
- Dr. Joseph Touch

Finally, the authors would also like to thank Mr. Tom Conkle, Ms. Michele Casey, and Mr. Etzel Brower.

7. References

7.1. Normative References

[IPSEC] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

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

[IKEV2] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[BRA97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[ROHCV1] Bormann, C., Burmeister, C., Degermark, M., Fukushima, H., Hannu, H., Jonsson, L., 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.

[ROHCV2] Pelletier, G. and K. Sandlund, "RObust Header Compression Version 2 (ROHCv2): Profiles for RTP, UDP, IP, ESP and UDP Lite", RFC 5225, April 2008.

[IPSEC-ROHC] Ertekin, E., Christou, C., and C. Bormann, "IPsec Extensions to Support ROHCoIPsec", work in progress , December 2009.

7.2. Informative References

[ROHCOIPSEC] Ertekin, E., Jasani, R., Christou, C., and C. Bormann, "Integration of Header Compression over IPsec Security Associations", work in progress , December 2009.

[ROHC-PPP] Bormann, C., "Robust Header Compression (ROHC) over PPP",

RFC 3241, April 2002.

[ROHCPROF]
"RObust Header Compression (ROHC) Profile Identifiers", www.iana.org/assignments/rohc-pro-ids , May 2008.

[CRYPTO-ALG]
Manral, V., "Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 4835, April 2007.

[IKEV2-PARA]
IANA, "IKEv2 Parameters, http://www.iana.org/assignments/ikev2-parameters", November 2009.

[IANA-CONSIDERATIONS]
Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, October 1998.

Authors' Addresses

Emre Ertekin
Booz Allen Hamilton
5220 Pacific Concourse Drive, Suite 200
Los Angeles, CA 90045
US

Email: ertekin_emre@bah.com

Chris Christou
Booz Allen Hamilton
13200 Woodland Park Dr.
Herndon, VA 20171
US

Email: christou_chris@bah.com

Rohan Jasani
Booz Allen Hamilton
13200 Woodland Park Dr.
Herndon, VA 20171
US

Email: ro@breakcheck.com

Tero Kivinen
Safenet, Inc.
Fredrikinkatu 47
HELSINKI
FI

Email: kivinen@iki.fi

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

Email: cabo@tzi.org