                  The Use of HMAC-MD5-96 within ESP and AH

Status of this Memo

Copyright Notice

Abstract

   This memo describes the use of the HMAC algorithm [RFC-2104] in
   conjunction with the MD5 algorithm [RFC-1321] as an authentication
   mechanism within the revised IPSEC Encapsulating Security Payload
   [ESP] and the revised IPSEC Authentication Header [AH]. HMAC with MD5
   provides data origin authentication and integrity protection.

   Further information on the other components necessary for ESP and AH
   implementations is provided by [Thayer97a].

1. Introduction

   This memo specifies the use of MD5 [RFC-1321] combined with HMAC
   [RFC-2104] as a keyed authentication mechanism within the context of
   the Encapsulating Security Payload and the Authentication Header.
   The goal of HMAC-MD5-96 is to ensure that the packet is authentic and
   cannot be modified in transit.

   HMAC is a secret key authentication algorithm. Data integrity and
   data origin authentication as provided by HMAC are dependent upon the
   scope of the distribution of the secret key. If only the source and
   destination know the HMAC key, this provides both data origin
   authentication and data integrity for packets sent between the two
   parties; if the HMAC is correct, this proves that it must have been
   added by the source.

In this memo, HMAC-MD5-96 is used within the context of ESP and AH.
For further information on how the various pieces of ESP - including
the confidentiality mechanism -- fit together to provide security
services, refer to [ESP] and [Thayer97a]. For further information on
AH, refer to [AH] and [Thayer97a].

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].

2. Algorithm and Mode

[RFC-1321] describes the underlying MD5 algorithm, while [RFC-2104]
describes the HMAC algorithm. The HMAC algorithm provides a framework
for inserting various hashing algorithms such as MD5.

HMAC-MD5-96 operates on 64-byte blocks of data. Padding requirements
are specified in [RFC-1321] and are part of the MD5 algorithm. If
MD5 is built according to [RFC-1321], there is no need to add any
additional padding as far as HMAC-MD5-96 is concerned. With regard
to "implicit packet padding" as defined in [AH], no implicit packet
padding is required.

HMAC-MD5-96 produces a 128-bit authenticator value. This 128-bit
value can be truncated as described in RFC 2104. For use with either
ESP or AH, a truncated value using the first 96 bits MUST be
supported. Upon sending, the truncated value is stored within the
authenticator field. Upon receipt, the entire 128-bit value is
computed and the first 96 bits are compared to the value stored in
the authenticator field. No other authenticator value lengths are
supported by HMAC-MD5-96.

The length of 96 bits was selected because it is the default
authenticator length as specified in [AH] and meets the security
requirements described in [RFC-2104].

2.1 Performance

[Bellare96a] states that "(HMAC) performance is essentially that of
the underlying hash function". [RFC-1810] provides some performance
analysis and recommendations of the use of MD5 with Internet
protocols. As of this writing no performance analysis has been done
of HMAC or HMAC combined with MD5.

[RFC-2104] outlines an implementation modification which can improve
per-packet performance without affecting interoperability.

3. Keying Material

   HMAC-MD5-96 is a secret key algorithm. While no fixed key length is
   specified in [RFC-2104], for use with either ESP or AH a fixed key
   length of 128-bits MUST be supported. Key lengths other than 128-
   bits MUST NOT be supported (i.e. only 128-bit keys are to be used by
   HMAC-MD5-96). A key length of 128-bits was chosen based on the
   recommendations in [RFC-2104] (i.e. key lengths less than the
   authenticator length decrease security strength and keys longer than
   the authenticator length do not significantly increase security
   strength).

   [RFC-2104] discusses requirements for key material, which includes a
   discussion on requirements for strong randomness. A strong pseudo-
   random function MUST be used to generate the required 128-bit key.

   At the time of this writing there are no specified weak keys for use
   with HMAC. This does not mean to imply that weak keys do not exist.
   If, at some point, a set of weak keys for HMAC are identified, the
   use of these weak keys must be rejected followed by a request for
   replacement keys or a newly negotiated Security Association.

   [ARCH] describes the general mechanism for obtaining keying material
   when multiple keys are required for a single SA (e.g. when an ESP SA
   requires a key for confidentiality and a key for authentication).

   In order to provide data origin authentication, the key distribution
   mechanism must ensure that unique keys are allocated and that they
   are distributed only to the parties participating in the
   communication.

   [RFC-2104] makes the following recommendation with regard to
   rekeying. Current attacks do not indicate a specific recommended
   frequency for key changes as these attacks are practically
   infeasible. However, periodic key refreshment is a fundamental
   security practice that helps against potential weaknesses of the
   function and keys, reduces the information avaliable to a
   cryptanalyst, and limits the damage of an exposed key.

4. Interaction with the ESP Cipher Mechanism

   As of this writing, there are no known issues which preclude the use
   of the HMAC-MD5-96 algorithm with any specific cipher algorithm.

5. Security Considerations

   The security provided by HMAC-MD5-96 is based upon the strength of
   HMAC, and to a lesser degree, the strength of MD5. [RFC-2104] claims
   that HMAC does not depend upon the property of strong collision
   resistance, which is important to consider when evaluating the use of
   MD5, an algorithm which has, under recent scrutiny, been shown to be
   much less collision-resistant than was first thought. At the time of
   this writing there are no practical cryptographic attacks against
   HMAC-MD5-96.

   [RFC-2104] states that for "minimally reasonable hash functions" the
   "birthday attack", the strongest attack know against HMAC, is
   impractical. For a 64-byte block hash such as HMAC-MD5-96, an attack
   involving the successful processing of 2**64 blocks would be
   infeasible unless it were discovered that the underlying hash had
   collisions after processing 2**30 blocks. A hash with such weak
   collision-resistance characteristics would generally be considered to
   be unusable.

   It is also important to consider that while MD5 was never developed
   to be used as a keyed hash algorithm, HMAC had that criteria from the
   onset. While the use of MD5 in the context of data security is
   undergoing reevaluation, the combined HMAC with MD5 algorithm has
   held up to cryptographic scrutiny.

   [RFC-2104] also discusses the potential additional security which is
   provided by the truncation of the resulting hash. Specifications
   which include HMAC are strongly encouraged to perform this hash
   truncation.

   As [RFC-2104] provides a framework for incorporating various hash
   algorithms with HMAC, it is possible to replace MD5 with other
   algorithms such as SHA-1. [RFC-2104] contains a detailed discussion
   on the strengths and weaknesses of HMAC algorithms.

   As is true with any cryptographic algorithm, part of its strength
   lies in the correctness of the algorithm implementation, the security
   of the key management mechanism and its implementation, the strength
   of the associated secret key, and upon the correctness of the
   implementation in all of the participating systems. [RFC-2202]
   contains test vectors and example code to assist in verifying the
   correctness of HMAC-MD5-96 code.

6. Acknowledgments

   This document is derived in part from previous works by Jim Hughes,
   those people that worked with Jim on the combined DES/CBC+HMAC-MD5
   ESP transforms, the ANX bakeoff participants, and the members of the
   IPsec working group.

   We would also like to thank Hugo Krawczyk for his comments and
   recommendations regarding some of the cryptographic specific text in
   this document.

7. References

   [RFC-1321]   Rivest, R., "MD5 Digest Algorithm", RFC 1321, April
                1992.

   [RFC-2104]   Krawczyk, H., Bellare, M., and R. Canetti, "HMAC:
                Keyed-Hashing for Message Authentication", RFC 2104,
                February 1997.

   [RFC-1810]   Touch, J., "Report on MD5 Performance", RFC 1810, June
                1995.

   [Bellare96a] Bellare, M., Canetti, R., and H. Krawczyk, "Keying Hash
                Functions for Message Authentication", Advances in
                Cryptography, Crypto96 Proceeding, June 1996.

   [ARCH]   Kent, S., and R. Atkinson, "Security Architecture for
                the Internet Protocol", RFC 2401, November 1998.

   [ESP]   Kent, S., and R. Atkinson, "IP Encapsulating Security
                Payload", RFC 2406, November 1998.

   [AH]   Kent, S., and R. Atkinson, "IP Authentication Header",
                RFC 2402, November 1998.

   [Thayer97a] Thayer, R., Doraswamy, N., and R. Glenn, "IP Security
                Document Roadmap", RFC 2411, November 1998.

   [RFC-2202]   Cheng, P., and R. Glenn, "Test Cases for HMAC-MD5 and
                HMAC-SHA-1", RFC 2202, March 1997.

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

8. Editors' Address

   Cheryl Madson
   Cisco Systems, Inc.

   EMail: cmadson@cisco.com

   Rob Glenn
   NIST

   EMail: <rob.glenn@nist.gov>

 The IPsec working group can be contacted through the chairs:

   Robert Moskowitz
   ICSA

   EMail: rgm@icsa.net

   Ted T'so
   Massachusetts Institute of Technology

   EMail: tytso@mit.edu

9. Full Copyright Statement

   Copyright (C) The Internet Society (1998). All Rights Reserved.

   This document and translations of it may be copied and furnished to
   others, and derivative works that comment on or otherwise explain it
   or assist in its implementation may be prepared, copied, published
   and distributed, in whole or in part, without restriction of any
   kind, provided that the above copyright notice and this paragraph are
   included on all such copies and derivative works. However, this
   document itself may not be modified in any way, such as by removing
   the copyright notice or references to the Internet Society or other
   Internet organizations, except as needed for the purpose of
   developing Internet standards in which case the procedures for
   copyrights defined in the Internet Standards process must be
   followed, or as required to translate it into languages other than
   English.

   The limited permissions granted above are perpetual and will not be
   revoked by the Internet Society or its successors or assigns.

   This document and the information contained herein is provided on an
   "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
   TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
   BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
   HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
   MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.