

**The Use of HMAC-RIPEMD-160-96 within ESP and AH**  
<[draft-ietf-ipsec-auth-hmac-ripemd-160-96-03.txt](#)>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

This document is a submission to the IETF Internet Protocol Security (IPSEC) Working Group. Comments are solicited and should be addressed to the working group mailing list ([ipsec@tis.com](mailto:ipsec@tis.com)) or to the editor.

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 draft documents are 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>.

Distribution of this memo is unlimited.

Abstract

This draft describes the use of the HMAC algorithm [[RFC-2104](#)] in conjunction with the RIPEMD-160 algorithm [[RIPEMD-160](#)] as an authentication mechanism within the revised IPSEC Encapsulating Security Payload [[ESP](#)] and the revised IPSEC Authentication Header [[AH](#)]. HMAC with RIPEMD-160 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 draft specifies the use of RIPEMD-160 [[RIPEMD-160](#)] 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-RIPEMD-160-96 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 draft, HMAC-RIPEMD-160-96 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**

[[RIPEMD-160](#)] describes the underlying RIPEMD-160 algorithm, while [[RFC-2104](#)] describes the HMAC algorithm. The HMAC algorithm provides a framework for inserting various hashing algorithms such as RIPEMD-160.

HMAC-RIPEMD-160-96 operates on 64-byte blocks of data. Padding requirements are specified in [[RIPEMD-160](#)] and are part of the RIPEMD-160 algorithm. Padding bits are only necessary in computing the HMAC-RIPEMD-160 authenticator value and MUST NOT be included in the packet.

HMAC-RIPEMD-160-96 produces a 160-bit authenticator value. This 160-bit value can be truncated as described in [RFC2104](#). 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 160-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-RIPEMD-160-96.



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". [RIPEMD-160] provides some performance analysis. As of this writing no detailed performance analysis has been done of HMAC or HMAC combined with RIPEMD-160.

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

### **3. Keying Material**

HMAC-RIPEMD-160-96 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 160-bits MUST be supported. Key lengths other than 160-bits SHALL NOT be supported. A key length of 160-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 160-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.

[ESP] describes the general mechanism to obtain keying material for the ESP transform. The derivation of the key from some amount of keying material does not differ between the manual and automatic key management mechanisms.

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] states that for "minimally reasonable hash functions" the "birthday attack" is impractical. For a 64-byte block hash such as HMAC-RIPEMD-160-96, 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.) No time-based attacks are discussed in the document.

While it is still cryptographically prudent to perform frequent rekeying, current literature does not include any recommended key lifetimes for HMAC-RIPEMD. When recommendations for HMAC-RIPEMD key lifetimes become available they will be included in a revised version of this document.

#### **4. Interaction with the ESP Cipher Mechanism**

As of this writing, there are no known issues which preclude the use of the HMAC-RIPEMD-160-96 algorithm with any specific cipher algorithm.

#### **5. Security Considerations**

The security provided by HMAC-RIPEMD-160-96 is based upon the strength of HMAC, and to a lesser degree, the strength of RIPEMD-160. At the time of this writing there are no known cryptographic attacks against RIPEMD-160.

It is also important to consider that while RIPEMD-160 was never developed to be used as a keyed hash algorithm, HMAC had that criteria from the onset.

[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 RIPEMD-160 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. [Kapp97] contains test vectors and example code to assist in verifying the correctness of HMAC-RIPEMD-160-96 code.





## 6. Acknowledgments

This document is derived from work by C. Madson and R. Glenn and 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.

## 7. References

- [RIPEMD-160] Dobbertin, H., Bosselaers A., and Preneel, B.  
"RIPEMD-160: A Strengthened Version of RIPEMD"  
April 1996, ftp.esat.kuleuven.ac.be:  
/pub/COSIC/bosselaer/ripemd/ripemd160.ps.gz
- [RFC-2104] Krawczyk, H., Bellare, M., Canetti, R., "HMAC: Keyed-  
Hashing for Message Authentication", [RFC-2104](#),  
February, 1997
- [Bellare96a] Bellare, M., Canetti, R., Krawczyk, H., "Keying  
Hash Functions for Message Authentication", Advances  
in Cryptography, Crypto96 Proceeding, June 1996.
- [ESP] Kent, S., Atkinson, R., "IP Encapsulating Security  
Payload", [draft-ietf-ipsec-esp-v2-01.txt](#),  
work in progress, October, 1997
- [AH] Kent, S., Atkinson, R., "IP Authentication Header",  
[draft-ietf-ipsec-auth-header-02.txt](#), work in progress,  
October 1997
- [Thayer97a] Thayer, R., Doraswamy, N., Glenn, R., "IP Security  
Document Framework",  
[draft-ietf-ipsec-doc-framework-01.txt](#), work in  
progress, July 1997.
- [Kapp97] Kapp, J.S., "Test Cases for HMAC-RIPEMD160 and  
HMAC-RIPEMD128", [RFC-2286](#), March 1998
- [RFC-2119] Bradner, S., "Key words for use in RFCs to Indicate  
Requirement Levels", [RFC-2119](#), March 1997.



**8. Authors' Address**

Angelos D. Keromytis  
Distributed Systems Lab  
Computer and Information Science Department  
University of Pennsylvania  
200 S. 33rd Street  
Philadelphia, PA 19104 - 6389  
angelos@dsl.cis.upenn.edu

Niels Provos  
Center for Information Technology Integration  
University of Michigan  
519 W. William  
Ann Arbor, Michigan 48103 USA  
provos@citi.umich.edu

The IPsec working group can be contacted through the chairs:

Robert Moskowitz  
rgm@icsa.net  
International Computer Security Association

Ted T'so  
tytso@mit.edu  
Massachusetts Institute of Technology

