Network Working Group S. Chang (NIST)
R. Glenn (NIST)
March 20, 1997

Internet Draft

HMAC-SHA-1-96 IP Authentication with Replay Prevention

<draft-ietf-ipsec-ah-hmac-sha-1-96-00.txt>

Status of This Memo

Distribution of this memo is unlimited.

This document is an Internet-Draft. 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 not appropriate to use Internet Drafts as reference material, or to cite them other than as a ``working draft'' or ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the internet-drafts Shadow Directories on:

ftp.is.co.za (Africa)
nic.nordu.net (Europe)
ds.internic.net (US East Coast)
ftp.isi.edu (US West Coast)
munnari.oz.au (Pacific Rim)

Abstract

This document describes a keyed-SHA transform to be used in conjunction with the IP Authentication Header [RFC-1826]. The particular transform is based on [RFC-2104]. A replay prevention field is also specified.

Contents

1. Introduction

The IP Authentication Header (AH) provides integrity and authentication for IP datagrams [RFC-1826]. The transform specified in this document uses a keyed-SHA mechanism based on [RFC-2104]. The mechanism uses the (key-less) SHA hash function [FIPS-180-1] which produces a message digest. When combined with an AH Key, Authentication Data is produced. This value is placed in the Authentication Data field of the AH [RFC-1826]. This value is also the basis for the data integrity service offered by the AH protocol.

To provide protection against replay attacks, a Replay Prevention field is specified as a transform option. This field is used to help prevent attacks in which a message is stored and re-used later, replacing or repeating the original. The Security Parameters Index (SPI) [RFC-1825] is used to determine whether this option is included in the AH.

Familiarity with the following documents is assumed: "Security Architecture for the Internet Protocol" [RFC-1825], "IP Authentication Header" [RFC-1826], and "HMAC: Keyed Hashing for Message Authentication" [RFC-2104].

All implementations that claim conformance or compliance with the IP Authentication Header specification [RFC-1826] SHOULD implement this HMAC-SHA-1-96 transform.

1.1 Terminology

In this document, the words that are used to define the significance of each particular requirement are usually capitalized. These words are:

- MUST

This word or the adjective "REQUIRED" means that the item is an absolute requirement of the specification.

- SHOULD

This word or the adjective "RECOMMENDED" means that there might exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before taking a different course.

- MAY

This word or the adjective "OPTIONAL" means that this item is truly

optional. One vendor might choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

For the purpose of this specification, the terms conformance and compliance are synonymous.

1.2 Keys

The "AH Key" is used as a shared secret between two communicating parties. The Key is not a "cryptographic key" as used in a traditional sense. Instead, the AH key (shared secret) is hashed with the transmitted data and thus, assures that an intervening party cannot duplicate the Authentication Data.

Even though an AH key is not a cryptographic key, the rudimentary concerns of cryptographic keys still apply. Consider that the algorithm and most of the data used to produce the output is known. The strength of the transform lies in the singular mapping of the key (which needs to be strong) and the IP datagram (which is known) to the Authentication Data. Thus, implementations should, and as frequently as possible, change the AH key. Keys need to be chosen at random, or generated using a cryptographically strong pseudo-random generator seeded with a random seed. [RFC-2104]

All conforming and compliant implementations MUST support a key length of 160 bits or less. Implementations SHOULD support longer key lengths as well. It is advised that the key length be chosen to be the length of the hash algorithm output, which is 160 bits for SHA. For other key lengths the following concerns MUST be considered.

A key length of zero is prohibited and implementations MUST prevent key lengths of zero from being used with this transform, since no effective authentication could be provided by a zero-length key. SHA operates on 64-byte blocks. Keys longer than 64-bytes are first hashed using SHA. The resulting hash is then used to calculate the Authentication Data.

1.3 Data Size

HMAC-SHA-1 generates a message digest of 160 bits. HMAC-SHA-1-96 uses the first or left most 96 bits as the Authentication Data. This procedure is known as truncation. In the case of this transform, truncation is used to help maintain 64-bit packet header alignment, eliminate unnecessary overhead, and potentially provided stronger authentication. [RFC-2104] provides more information on the advantages and disadvantages of truncation.

2. Packet Format

```
+---------------+---------------+---------------+---------------+
| Next Header   | Length        |            RESERVED           |
+---------------+---------------+---------------+---------------+
|                              SPI                              |
+---------------+---------------+---------------+---------------+
|                     Replay Prevention                         |
+---------------+---------------+---------------+---------------+
|                                                               |
+                                                               +
|                     Authentication Data                       |
+                                                               +
|                                                               |
+---------------+---------------+---------------+---------------+
 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8
```

The Next Header, RESERVED, and SPI fields are specified in [RFC-1826]. The Length field is the length of the Replay Prevention field and the Authentication Data in 32-bit words. The Length field will always be set to 4 (128 bits) for HMAC-SHA-1-96.

2.1 Replay Prevention

The Replay Prevention field is a 32-bit value used to guarantee that each packet exchanged between two parties is different. Each IPsec Security Association specifies whether Replay Prevention is used for that Security Association. The Replay Prevention field is always included in the calculation of the Authentication Data. If Replay Prevention is NOT in use, the 32-bit value is set to 0, included in the calculation of the Authentication Data, and ignored upon receipt with regard to checking for replay. This field is used to help prevent attacks in which a message is stored and re-used later, replacing or repeating the original. Replay Prevention SHOULD be implemented.

Replay Prevention SHOULD be implemented. If Replay Prevention is not implemented, the 32-bit field remains are part of the AH and is treated as if Replay Prevention is NOT in use (i.e. the 32-bit value is set to 0, included in the calculation of the Authentication Data, and ignored upon receipt with regard to checking for replay.

The 32-bit field is an up counter starting at a value of 1.

The secret shared key MUST NOT be used for a period of time that allows the counter to wrap, that is, to transmit more than 2^32 packets using a single key.

Upon receipt, the replay value is assured to be increasing. An implementation MAY accept out of order packets. If an "out of order window" is supported, an implementation MUST guarantee that any and all packets accepted out of order have not arrived before. That is, an implementation MUST accept any packet, at most, once. The size of the window is a negotiated value specified by the IPsec Security Association.

[ESP-DES-MD5] provides more information on negotiated windows sizes, example code that implements a 32 packet replay window, and a test routine to show how it could be implemented.

When the destination address is a multicast address and more than one sender is sharing the same IPsec Security Association to that multicast destination address, then Replay Prevention SHOULD NOT be enabled. When Replay Prevention is desired for a multicast session having multiple senders to the same multicast destination address, each sender SHOULD have its own IPsec Security Association.

2.2 Authentication Data Calculation

The Authentication Data is the output of the SHA authentication algorithm as described in [FIPS-180-1]. The digest is calculated over the entire IP datagram. Fields within the datagram that are variant during transit and the Authentication Data field itself must contain all zeros prior to the computation [RFC-1826]. The Replay Prevention field, used or not, is included in the calculation.

To compute HMAC-SHA-1 over the data 'text', the following is calculated:

```
SHA (K XOR opad, SHA (K XOR ipad, text))
```

The result of which is truncated to 96 bits (retaining the left most bits) to produce HMAC-SHA-1-96.

K denotes the secret key shared by the parties. If K is longer than 64-bytes, it MUST first be hashed using SHA. In this case, K is the resulting hash. The variables 'ipad', 'opad' denote fixed strings for inner and outer padding respectively. The two strings are:

```
ipad = the byte 0x36 repeated 64 times,
opad = the byte 0x5C repeated 64 times.
```

The calculation of the Authentication Data consists of the following steps:

(1) append zeros to the end of K to create a 64 byte string (e.g., if K is of length 16 bytes it will be appended with 48 zero bytes 0x00)
(2) XOR (bitwise exclusive-OR) the 64 byte string computed in step (1) with ipad
(3) concatenate to the 64 byte string resulting from step (2) the data stream 'text'
(4) apply SHA to the stream generated in step (3)
(5) XOR the 64 byte string computed in step (1) with opad
(6) concatenate to the 64 byte string resulting from step (5) the SHA result of step (4)
(7) apply SHA to the stream generated in step (6)
(8) use the left most 96 bits of the result obtained in (7) as the final result

A similar computation is described in more detail, along with example code and performance improvements, in [RFC-2104]. Implementers should consult [RFC-2104] for more information on the HMAC technique for keying a cryptographic hash function.

3. Security Considerations

The security provided by this transform is based on the strength of SHA, the correctness of the algorithm's implementation, the security of the key management mechanism and its implementation, the strength of the associated secret key, and upon the correctness of the implementations in all of the participating systems. [RFC-2104] contains a detailed discussion on the strengths and weaknesses of HMAC algorithms. [HMAC-TESTS] contains test vectors and example code to assist in verifying the correctness of HMAC-SHA-1 code.

Acknowledgments

This document is largely based on text written by Hugo Krawczyk. The format used was derived from work by William Simpson and Perry Metzger. The text on replay prevention is derived from work by Jim Hughes.

References

[RFC-1825] R. Atkinson, "Security Architecture for the Internet Protocol", RFC-1825, August 1995.

[RFC-1826] R. Atkinson, "IP Authentication Header", RFC-1826, August 1995.

[RFC-1828] P. Metzger, W. A. Simpson, "IP Authentication using Keyed MD5", RFC-1828, August 1995.

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

[FIPS-180-1] NIST, FIPS PUB 180-1: Secure Hash Standard, April 1995.
[URL] http://csrc.nist.gov/fips/fip180-1.txt (ascii)
[URL] http://csrc.nist.gov/fips/fip180-1.ps (postscript)

[ESP-DES-MD5] J. Hughes, "Combined DES-CBC, MD5, and Replay Prevention Security Transform", Internet Draft, September 1996.

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

Authors' Addresses

Shu-jen Chang
NIST
Building 820, Room 456
Gaithersburg, MD 20899

shu-jen.chang@nist.gov

Robert Glenn
NIST
Building 820, Room 455
Gaithersburg, MD 20899

rob.glenn@nist.gov