                    IP Authentication using Keyed SHA

Status of this Memo

    This document defines an Experimental Protocol for the Internet
    community. This does not specify an Internet standard of any kind.
    Discussion and suggestions for improvement are requested.
    Distribution of this memo is unlimited.

Abstract

    This document describes the use of keyed SHA with the IP
    Authentication Header.

Table of Contents

1. Introduction

   The Authentication Header (AH) [RFC-1826] provides integrity and
   authentication for IP datagrams. This specification describes the AH
   use of keys with the Secure Hash Algorithm (SHA) [FIPS-180-1].

      It should be noted that this document specifies a newer version of
      the SHA than that described in [FIPS-180], which was flawed. The
      older version is not interoperable with the newer version.

   This document assumes that the reader is familiar with the related
   document "Security Architecture for the Internet Protocol" [RFC-
   1825], which defines the overall security plan for IP, and provides
   important background for this specification.

1.1. Keys

   The secret authentication key shared between the communicating
   parties SHOULD be a cryptographically strong random number, not a
   guessable string of any sort.

   The shared key is not constrained by this transform to any particular
   size. Lengths of up to 160 bits MUST be supported by the
   implementation, although any particular key may be shorter. Longer
   keys are encouraged.

1.2. Data Size

   SHA's 160-bit output is naturally 32-bit aligned. However, many
   implementations require 64-bit alignment of the following headers, in
   which case an additional 32 bits of padding is added, either before
   or after the SHA output.

   The size and position of this padding are negotiated as part of the
   key management. Padding bits are filled with unspecified
   implementation dependent (random) values, which are ignored on
   receipt.

1.3. Performance

   Preliminary results indicate that SHA is 62% as fast as MD5, and 80%
   as fast as DES hashing. That is,

                            SHA < DES < MD5

Nota Bene:
    Suggestions are sought on alternative authentication algorithms
    that have significantly faster throughput, are not patent-
    encumbered, and still retain adequate cryptographic strength.

2. Calculation

The 160-bit digest is calculated as described in [FIPS-180-1]. At
the time of writing, a portable C language implementation of SHA is
available via FTP from ftp://rand.org/pub/jim/sha.tar.gz.

The form of the authenticated message is

        key, keyfill, datagram, key, SHAfill

First, the variable length secret authentication key is filled to the
next 512-bit boundary, using the same pad with length technique
defined for SHA.

Then, the filled key is concatenated with (immediately followed by)
the invariant fields of the entire IP datagram (variant fields are
zeroed), concatenated with (immediately followed by) the original
variable length key again.

A trailing pad with length to the next 512-bit boundary for the
entire message is added by SHA itself. The 160-bit SHA digest is
calculated, and the result is inserted into the Authentication Data
field.

Discussion:
    The leading copy of the key is padded in order to facilitate
    copying of the key at machine boundaries without requiring re-
    alignment of the following datagram. The padding technique
    includes a length which protects arbitrary length keys. Filling
    to the SHA block size also allows the key to be prehashed to avoid
    the physical copy in some implementations.

    The trailing copy of the key is not necessary to protect against
    appending attacks, as the IP datagram already includes a total
    length field. It reintroduces mixing of the entire key, providing
    minimal protection for very long and very short datagrams, and
    marginal robustness against possible attacks on the IP length
    field itself.

When the implementation adds the keys and padding in place before and after the IP datagram, care must be taken that the keys and/or padding are not sent over the link by the link driver.

Security Considerations

Users need to understand that the quality of the security provided by this specification depends completely on the strength of the SHA hash function, the correctness of that algorithm's implementation, the security of the key management mechanism and its implementation, the strength of the key [CN94], and upon the correctness of the implementations in all of the participating nodes.

The SHA algorithm was originally derived from the MD4 algorithm [RFC-1320]. A flaw was apparently found in the original specification of SHA [FIPS-180], and this document specifies the use of a corrected version [FIPS-180-1].

At the time of writing of this document, there are no known flaws in the SHA algorithm. That is, there are no known attacks on SHA or any of its components that are better than brute force, and the 160-bit hash output by SHA is substantially more resistant to brute force attacks than the 128-bit hash size of MD4 and MD5.

However, as the flaw in the original SHA algorithm shows, cryptographers are fallible, and there may be substantial deficiencies yet to be discovered in the algorithm.

Acknowledgements

Some of the text of this specification was derived from work by Randall Atkinson for the SIP, SIPP, and IPv6 Working Groups.

Preliminary performance analysis was provided by Joe Touch.

Comments should be submitted to the ipsec@ans.net mailing list.

References

   [CN94]   John M. Carroll & Sri Nudiati, "On Weak Keys and Weak Data:
             Foiling the Two Nemeses", Cryptologia, Vol. 18 No. 23 pp.
             253-280, July 1994.

   [FIPS-180]
             "Secure Hash Standard", Computer Systems Laboratory,
             National Institute of Standards and Technology, U.S.
             Department Of Commerce, May 1993.

             Also known as: 58 Fed Reg 27712 (1993).

   [FIPS-180-1]
             "Secure Hash Standard", National Institute of Standards and
             Technology, U.S. Department Of Commerce, April 1995.

             Also known as: 59 Fed Reg 35317 (1994).

   [RFC-1320]
             Ronald Rivest, "The MD4 Message-Digest Algorithm", RFC-1320,
             April 1992.

   [RFC-1700]
             Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC
             1700, USC/Information Sciences Institute, October 1994.

   [RFC-1825]
             Atkinson, R., "Security Architecture for the Internet
             Protocol", RFC-1825, Naval Research Laboratory, July 1995.

   [RFC-1826]
             Atkinson, R., "IP Authentication Header", RFC-1826, Naval
             Research Laboratory, July 1995.

Author's Address

   Questions about this memo can also be directed to:

      Perry Metzger
      Piermont Information Systems Inc.
      160 Cabrini Blvd., Suite #2
      New York, NY 10033

      perry@piermont.com

      William Allen Simpson
      Daydreamer
      Computer Systems Consulting Services
      1384 Fontaine
      Madison Heights, Michigan 48071

      Bill.Simpson@um.cc.umich.edu
          bsimpson@MorningStar.com