TLS Working Group
INTERNET-DRAFT

Joo-won Jung
ChangHee Lee
INITECH, Inc.
12 July 2000

TLS Extension for SEED and HAS-160

draft-ietf-tls-seedhas-00.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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.

Introduction

This document proposes the addition of new cipher suites to the TLS protocol 1.0 [TLS] to support SEED and HAS-160.

The SEED algorithm is 128-bit symmetric block cipher algorithm. [SEED] The HAS-160 is 160-bit secure hash function, whose block size is 512 bit. [HAS] Both algorithms are developed in Korea since 1997 for stronger communication security. Currently, SEED is widely used and is the mandatory cipher in banking and stock applications in Korea.

HMAC of HAS-160

HMAC of HAS160 can be defined like HMAC_MD5 or HMAC_SHA1. Since HAS-160 is 512-bit block, 160-bit output secure hash algorithm, B=64 and L=20 as the notation of [HMAC].

The test values of HMAC_HAS160 is provided as appendix of this

document.

HMAC_HAS160 is used just for MAC of record layer. Adding HMAC_HAS160 does not affect the definitions of PRF, Finished message and other definitions using HMAC_MD5 or HMAC_SHA1.

Cipher Suites

In spite of the existence of Korean digital signature algorithm, KCDSA, RSA algorithm is more widely used in Korea. Therefore, we define cipher suites with RSA key exchange.

```
CipherSuite TLS_RSA_WITH_SEED_CBC_MD5          = { 0x00, 0x2C };
CipherSuite TLS_RSA_WITH_SEED_CBC_SHA          = { 0x00, 0x2D };
CipherSuite TLS_RSA_WITH_SEED_CBC_HAS160       = { 0x00, 0x2E };
```

Note: The above numeric definitions for Cipher Suites have not yet been registered. The numeric definitions are the following numbers of CipherSuite of TLS standard.[TLS]

References

[HAS] TTA.IS-10118, "Hash Function Standard - Part 2 : Hash Function Algorithm (HAS-160)", Telecommunications Technology Association, Republic of Korea, November, 1998.

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

[SEED] TTA.KO-12.0004, "128-bit Symmetric Block Cipher (SEED)", Telecommunications Technology Association, Republic of Korea, September 28, 1999.

[TLS] T. Dierks, and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, January 1999.

Test Values of HMAC_HAS160

```
test_case =     1
key =           0x0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b0b
key_len =       20
data =          "Hi There"
data_len =      8
digest =        0xf5b44115a53f716b6f488de1098ee7c251418623

test_case =     2
```

```
   key =          "Jefe"
   key_len =      4
   data =         "what do ya want for nothing?"
   data_len =     28
   digest =       0xa74547c1ef0aa147c7428ab7e71664549be2a412

   test_case =    3
   key =          0xaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
   key_len =      20
   data =         0xdd repeated 50 times
   data_len =     50
   digest =       0xe4c91bc71782fa44a56be1a34aae167e8ffc9734

   test_case =    4
   key =          0x0102030405060708090a0b0c0d0e0f10111213141516171819
   key_len =      25
   data =         0xcd repeated 50 times
   data_len =     50
   digest =       0x14d1055da875222053bf1180bbef8892eba3ac30

   test_case =    5
   key =          0x0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c0c
   key_len =      20
   data =         "Test With Truncation"
   data_len =     20
   digest =       0x63750d67af40e3fde33526545d300972a1527053

   test_case =    6
   key =          0xaa repeated 80 times
   key_len =      80
   data =         "Test Using Larger Than Block-Size Key - Hash Key First"
   data_len =     54
   digest =       0x63750d67af40e3fde33526545d300972a1527053

   test_case =    7
   key =          0xaa repeated 80 times
   key_len =      80
   data =         "Test Using Larger Than Block-Size Key and Larger
                  Than One Block-Size Data"
   data_len =     73
   digest =       0x1bdb821e399e208352c64f0655f6601e2a8a087c
```

Note: These values are not cross-verified with other organization.

Author's Address

Joo-won Jung

INITECH, Inc.
EMail: jwjung@initech.com

ChangHee Lee
INITECH, Inc.
EMail: chlee@initech.com

Phone: +82 2 3430 5700