Network Working Group   Pete Chown
INTERNET DRAFT   Skygate Technology
<draft-ietf-tls-ciphersuite-04.txt>   5 July 2001

AES Ciphersuites for TLS

Status of this Memo

Overview

At present, the symmetric ciphers supported by TLS are RC2, RC4, IDEA, DES and triple DES. The protocol would be enhanced by the addition of AES [AES] ciphersuites, for the following reasons:

1. RC2, RC4 and IDEA are all subject to intellectual property claims. RSA Security Inc has trademark rights in the names RC2 and RC4, and claims that the RC4 algorithm itself is a trade secret. Ascom Systec Ltd owns a patent on the IDEA algorithm.

2. Triple DES is much less efficient than more modern ciphers.

3. Now the AES process is completed there will be commercial pres¡ sure to use the selected cipher. The AES is efficient and has withstood extensive cryptanalytic efforts. The AES is

therefore a desirable choice.

4. Currently the DHE ciphersuites only allow triple DES (along with some ``export'' variants which do not use a satisfactory key length). At the same time the DHE ciphersuites are the only ones to offer forward secrecy.

This document proposes several new ciphersuites, with the aim of overcoming these problems.

Cipher Usage

The new ciphersuites proposed here are very similar to the follow¡ ing, defined in [TLS]:

TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA
TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA

All the ciphersuites described here use the AES in cipher block chaining (CBC) mode. Furthermore, they use SHA-1 [SHA-1] in an HMAC construction as described in section 5 of [TLS]. (Although the TLS ciphersuite names include the text ``SHA'', this actually refers to the modified SHA-1 version of the algorithm.)

The ciphersuites differ in the type of certificate and key exchange method. The ciphersuites defined here use the following options for this part of the protocol:

CipherSuite	Certificate type (if applicable) and key exchange algorithm
TLS_RSA_WITH_AES_128_CBC_SHA	RSA
TLS_DH_DSS_WITH_AES_128_CBC_SHA	DH_DSS
TLS_DH_RSA_WITH_AES_128_CBC_SHA	DH_RSA
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	DHE_DSS
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	DHE_RSA
TLS_DH_anon_WITH_AES_128_CBC_SHA	DH_anon
TLS_RSA_WITH_AES_256_CBC_SHA	RSA
TLS_DH_DSS_WITH_AES_256_CBC_SHA	DH_DSS
TLS_DH_RSA_WITH_AES_256_CBC_SHA	DH_RSA
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	DHE_DSS
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	DHE_RSA
TLS_DH_anon_WITH_AES_256_CBC_SHA	DH_anon

For the meanings of the terms RSA, DH_DSS, DH_RSA, DHE_DSS, DHE_RSA and DH_anon, please refer to sections 7.4.2 and 7.4.3 of [TLS].

The AES supports key lengths of 128, 192 and 256 bits. However, this document only defines ciphersuites for 128- and 256-bit keys. This is to avoid unnecessary proliferation of ciphersuites. Rijn¡ dael actually allows for 192- and 256-bit block sizes as well as the 128-bit blocks mandated by the AES process. The ciphersuites defined here all use 128-bit blocks.

The new ciphersuites will have the following definitions:

```
CipherSuite TLS_RSA_WITH_AES_128_CBC_SHA      = { 0x00, 0x2F };
CipherSuite TLS_DH_DSS_WITH_AES_128_CBC_SHA   = { 0x00, 0x30 };
CipherSuite TLS_DH_RSA_WITH_AES_128_CBC_SHA   = { 0x00, 0x31 };
CipherSuite TLS_DHE_DSS_WITH_AES_128_CBC_SHA  = { 0x00, 0x32 };
CipherSuite TLS_DHE_RSA_WITH_AES_128_CBC_SHA  = { 0x00, 0x33 };
CipherSuite TLS_DH_anon_WITH_AES_128_CBC_SHA  = { 0x00, 0x34 };

CipherSuite TLS_RSA_WITH_AES_256_CBC_SHA      = { 0x00, 0x35 };
CipherSuite TLS_DH_DSS_WITH_AES_256_CBC_SHA   = { 0x00, 0x36 };
CipherSuite TLS_DH_RSA_WITH_AES_256_CBC_SHA   = { 0x00, 0x37 };
CipherSuite TLS_DHE_DSS_WITH_AES_256_CBC_SHA  = { 0x00, 0x38 };
CipherSuite TLS_DHE_RSA_WITH_AES_256_CBC_SHA  = { 0x00, 0x39 };
CipherSuite TLS_DH_anon_WITH_AES_256_CBC_SHA  = { 0x00, 0x3A };
```

Padding

In section 4.7 of [TLS], the padding algorithm for RSA encryption is defined to be PKCS #1 block type 2 [PKCS1-1.5]. When the AES

ciphersuites are used, however, RSA encryption operations are padded using OAEP [PKCS1-2.0].

The specific padding algorithm is described in [PKCS1-2.0] section 9.1.1 and is denoted EME-OAEP. A hash function and a mask genera¡ tion function must be selected in order for EME-OAEP to be com¡ pletely defined. For the purposes of the AES ciphersuites, the hash function is SHA-1 and the mask generation function is MGF1, described in [PKCS1-2.0] section 10.2.1.

Security Considerations

It is not believed that the new ciphersuites are ever less secure than the corresponding older ones. The AES is believed to be secure, and it has withstood extensive cryptanalytic attack.

The ephemeral Diffie-Hellman ciphersuites provide forward secrecy without any known reduction in security in other areas. To obtain the maximum benefit from these ciphersuites:

1. The ephemeral keys should only be used once. With the TLS pro¡ tocol as currently defined there is no significant efficiency gain from reusing ephemeral keys.

2. Ephemeral keys should be destroyed securely when they are no longer required.

3. The random number generator used to create ephemeral keys must not reveal past output even when its internal state is compro¡ mised.

[TLS] describes the anonymous Diffie-Hellman (ADH) ciphersuites as deprecated. The ADH ciphersuites defined here are not deprecated. However, when they are used, particular care must be taken:

1. ADH provides confidentiality but not authentication. This means that (if authentication is required) the communicating parties must authenticate to each other by some means other than TLS.

2. ADH is vulnerable to man-in-the-middle attacks, as a conse¡ quence of the lack of authentication. The parties must have a way of determining whether they are participating in the same TLS connection. If they are not, they can deduce that they are under attack, and presumably abort the connection.

   For example, if the parties share a secret, it is possible to compute a MAC of the TLS Finished message. An attacker would

have to negotiate two different TLS connections; one with each communicating party. The Finished messages would be different in each case, because they depend on the parties' public keys (among other things). For this reason, the MACs computed by each party would be different.

It is important to note that authentication techniques which do not use the Finished message do not usually provide protection from this attack. For example, the client could authenticate to the server with a password, but it would still be vulnerable to man-in-the-middle attacks.

Copyright

Intellectual Property

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to per¡ tain to the implementation or use other technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the

IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such pro¡ prietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

During the development of the AES, NIST published the following statement on intellectual property:

> SPECIAL NOTE - Intellectual Property
>
> NIST reminds all interested parties that the adoption of AES is being conducted as an open standards-setting activity. Specifically, NIST has requested that all interested parties identify to NIST any patents or inven¡ tions that may be required for the use of AES. NIST hereby gives public notice that it may seek redress under the antitrust laws of the United States against any party in the future who might seek to exercise patent rights against any user of AES that have not been disclosed to NIST in response to this request for information.

One of the authors of Rijndael signed the following disclaimer when submitting the algorithm to NIST for consideration in the AES pro¡ cess:

> I, Joan Daemen, do hereby declare that to the best of my knowledge the practice of the algorithm, reference imple¡ mentation, and mathematically optimized implementations, I have submitted, known as Rijndael may be covered by the following U.S. and/or foreign patents:
>
> none
>
> I do hereby declare that I am aware of no patent applica¡ tions which may cover the practice of my submitted algo¡ rithm, reference implementation or mathematically opti¡ mized implementations.

I do hereby understand that my submitted algorithm may not be selected for inclusion in the Advanced Encryption Standard. I also understand and agree that after the close of the submission period, my submission may not be withdrawn from public consideration for inclusion in the Federal Information Processing Standard (FIPS) for Advanced Encryption Standard (AES). I further understand that I will not receive financial compensation from the government for my submission. I certify that, to the best of my knowledge, I have fully disclosed all patents and patent applications relating to my algorithm. I also understand that the U.S. Government may, during the course of the lifetime of the AES or during the FIPS pub¡ lic review process, modify the algorithm's specifications (e.g., to protect against a newly discovered vulnerabil¡ ity). Should my submission be selected for inclusion in the AES, I hereby agree not to place any restrictions on the use of the algorithm intending it to be available on a worldwide, non-exclusive, royalty-free basis.

I do hereby agree to provide the statements for any patent or patent application identified to cover practice of my algorithm, reference implementation or mathemati¡ cally optimized implementations and the right to use such implementations for the purposes of the AES evaluation process.

I understand that NIST will announce the selected algo¡ rithm(s) and proceed to publish the draft FIPS for public comment. If my algorithm (or the derived algorithm) is not selected for inclusion in the FIPS (including those not selected for second round of public evaluation), I understand that all rights, including use rights of the reference and mathematically optimized implementations, revert back to the submitter (and other owner[s] as appropriate). Additionally, should the U.S. Government not select my algorithm for inclusion in the AES after a period of four years from the close of the submission date for candidate algorithms, all rights revert to the submitter (and other owner[s] as appropriate).

[signed]

Title: Cryptographer
Dated: 10-6-98
Place: Brussels

The following disclaimer was signed at the start of the second "round" of the AES process:

Dear Mr Foti [of NIST],

Hereby we confirm that the original patent and patent application information, as provided to NIST with our original submission in June 1998, has not changed. To the best of our knowledge, there are no patents or patent applications covering the practice of the algorithm, ref¡ erence implementation or the mathematically optimized implementations.

[signed]
Joan Daemen, Vincent Rijmen

Acknowledgements

I would like to thank the ietf-tls mailing list contributors who have made helpful suggestions for this document.

References

[AES] J. Daemen, V. Rijmen, "The Rijndael Block Cipher" http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf 3rd September 1999.

[PKCS1-1.5] RSA Laboratories, "PKCS #1: RSA Encryption Standard" version 1.5, November 1993.

[PKCS1-2.0] B. Kaliski, J. Staddon, "PKCS #1: RSA Cryptography Specifications Version 2.0" RFC 2437.

[SHA-1] FIPS PUB 180-1, "Secure Hash Standard," National Institute of Standards and Technology, U.S. Department of Commerce, April 17, 1995.

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

Author's Address

Pete Chown
Skygate Technology Ltd
8 Lombard Road
London
SW19 3TZ

United Kingdom

Phone: +44 20 8542 7856
Email: pc@skygate.co.uk