TLS
Internet-Draft
Intended status: Informational
Expires: March 22, 2020

P. Yang
Ant Financial

September 19, 2019

# SM Cipher Suites for Transport Layer Security (TLS) Protocol Version 1.3

draft-yang-tls-tls13-sm-suites-01

## Abstract

This draft specifies a set of cipher suites for the Transport Layer Security (TLS) protocol version 1.3 to support SM cryptographic algorithms.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on March 22, 2020.

Table of Contents

1. Introduction . . . . . 2
1.1. The SM Algorithms . . . . . 3
1.2. Terminology . . . . . 3
2. Proposed Cipher Suites . . . . . 3
3. Cipher Suites Definitions . . . . . 4
3.1. TLS Versions . . . . . 4
3.2. Authentication . . . . . 4
3.2.1. SM2 Signature Scheme . . . . . 4
3.3. Key Exchange . . . . . 5
3.3.1. Hello Messages . . . . . 5
3.3.2. CertificateRequest . . . . . 6
3.3.3. Certificate . . . . . 7
3.3.4. CertificateVerify . . . . . 7
3.4. Key Scheduling . . . . . 7
3.5. Cipher . . . . . 7
3.5.1. AEAD_SM4_GCM . . . . . 7
3.5.2. AEAD_SM4_CCM . . . . . 8
3.6. Hash . . . . . 9
4. IANA Considerations . . . . . 9
5. Security Considerations . . . . . 10
6. References . . . . . 10
6.1. Normative References . . . . . 10
6.2. Informative References . . . . . 11
Appendix A. Contributors . . . . . 12
Appendix B. Acknowledgments . . . . . 12
Author's Address . . . . . 12

## 1. Introduction

This document describes two new cipher suites for the Transport Layer Security (TLS) protocol version 1.3 (a.k.a TLSv1.3, [RFC8446]). The new cipher suites are listed as follows (or Section 2):

```
CipherSuite TLS_SM4_GCM_SM3 = { 0x00, 0xC6 };
CipherSuite TLS_SM4_CCM_SM3 = { 0x00, 0xC7 };
```

These new cipher suites contains several SM cryptographic algorithms that provide both authentication and confidentiality. For the more detailed introduction to SM cryptographic algorithms, please read Section 1.1. These cipher suites follow what TLSv1.3 requires. For instance, all the cipher suites mentioned in this draft use ECDHE as the key exchange scheme and use SM4 in either GCM mode or CCM mode to meet the need of TLSv1.3 to have an AEAD capable encryption algorithm.

For the details about how these new cipher suites negotiate shared encryption key and protect the record structure, please read Section 3.

### 1.1. The SM Algorithms

The new cipher suites defined in this draft use several different SM cryptographic algorithms including SM2 for authentication, SM4 for encryption and SM3 as the hash function.

SM2 is a set of elliptic curve based cryptographic algorithms including digital signature, public key encryption and key exchange scheme. In this draft, only the SM2 digital signature algorithm is involved, which has now already been added to ISO/IEC 14888-3:2018 [ISO-SM2] (as well as in [GBT.32918.2-2016]). SM4 is a block cipher defined in [GBT.32907-2016] and now is being standardized by ISO to ISO/IEC 18033-3:2010 [ISO-SM4]. SM3 is a hash function which produces an output of 256 bits. SM3 has already been accepted by ISO in ISO/IEC 10118-3:2018 [ISO-SM3], and also been described by [GBT.32905-2016].

### 1.2. Terminology

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, BCP 14 [RFC2119] and indicate requirement levels for compliant TLSv1.3 implementations.

## 2. Proposed Cipher Suites

The cipher suites defined here have the following identifiers:

```
CipherSuite TLS_SM4_GCM_SM3 = { 0x00, 0xC6 };
CipherSuite TLS_SM4_CCM_SM3 = { 0x00, 0xC7 };
```

To accomplish a TLSv1.3 handshake, more objects have been introduced along with the cipher suites as follows.

The SM2 signature algorithm and SM3 hash function used in the Signature Algorithm extension defined in appendix-B.3.1.3 of [RFC8446]:

```
SignatureScheme sm2sig_sm3 = { 0x0708 };
```

The SM2 elliptic curve ID used in the Supported Groups extension defined in appendix-B.3.1.4 of [RFC8446]:

```
NamedGroup curveSM2 = { 41 };
```

## 3. Cipher Suites Definitions

### 3.1. TLS Versions

The only capable version for the new cipher suites defined in this document is TLSv1.3. Implementations of this document MUST NOT apply these cipher suites into any TLS protocols that have an older version than 1.3.

### 3.2. Authentication

#### 3.2.1. SM2 Signature Scheme

All cipher suites defined in this document use SM2 signature algorithm as the authentication method when doing a TLSv1.3 handshake.

SM2 signature is defined in [ISO-SM2]. In general, SM2 is a signature algorithm based on elliptic curves. SM2 signature algorithm uses a fixed elliptic curve parameter set defined in [GBT.32918.5-2016]. This curve has the name curveSM2 and IANA is requested to assign a value for it. Unlike other elliptic curve based public key algorithm like ECDSA, SM2 cannot select other elliptic curves in practice, but it's allowed to write test cases by using other elliptic curve parameter sets for SM2, take Annex F.14 of [ISO-SM2] as a reference.

Implementations of the cipher suites defined in this document SHOULD conform to what [GBT.32918.5-2016] requires, that is to say, the only valid elliptic curve parameter for SM2 signature algorithm (a.k.a curveSM2) is defined as follows:

```
curveSM2: a prime field of 256 bits

y^2 = x^3 + ax + b

p  = FFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF
     FFFFFFFF 00000000 FFFFFFFF FFFFFFFF
a  = FFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF
     FFFFFFFF 00000000 FFFFFFFF FFFFFFFC
b  = 28E9FA9E 9D9F5E34 4D5A9E4B CF6509A7
     F39789F5 15AB8F92 DDBCBD41 4D940E93
n  = FFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFF
     7203DF6B 21C6052B 53BBF409 39D54123
Gx = 32C4AE2C 1F198119 5F990446 6A39C994
     8FE30BBF F2660BE1 715A4589 334C74C7
Gy = BC3736A2 F4F6779C 59BDCEE3 6B692153
     D0A9877C C62A4740 02DF32E5 2139F0A0
```

SM2 signature algorithm requests an identifier value when generate the signature, as well as when verifying an SM2 signature. Implementations of this document MUST use the following ASCII string value as the SM2 identifier when doing a TLSv1.3 key exchange:

```
TLSv1.3+GM+Cipher+Suite
```

Except if either a client or a server needs to verify the peer's SM2 certificate contained in the Certificate message, the following ASCII string value SHOULD be used as the SM2 identifier according to [GMT.0009-2012]:

```
1234567812345678
```

In the octet presentation, it should be:

```
0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38,
0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38
```

In practice, the SM2 identifier used in a certificate signature depends on the CA who signs that certificate. CAs may choose other values rather than the one mentioned above. Implementations of this document SHOULD confirm this information by themselves.

## 3.3. Key Exchange

### 3.3.1. Hello Messages

The new cipher suites defined in this document update the key exchange information in the Hello messages. Implementations of these new ciphers suites MUST conform to the new requirements.

#### 3.3.1.1. ClientHello

A TLSv1.3 client is REQUIRED to include the new cipher suites in its 'cipher_suites' array of the ClientHello structure defined in Section 4.1.2 of [RFC8446].

Other requirements on the extensions of ClientHello message are:

- For supported_groups extension, 'curveSM2' MUST be included;
- For signature_algorithms extension, 'sm2sig_sm3' MUST be included;
- For signature_algorithms_cert extension (if presented), 'sm2sig_sm3' MUST be included;
- For key_share extension, a KeyShareEntry with SM2 related values MUST be added if the client wants to start a TLSv1.3 key negotiation using SM cipher suites.

#### 3.3.1.2. ServerHello

If a TLSv1.3 server receives a ClientHello message containing the new cipher suites defined in this document, it MAY choose to use the new cipher suites. If so, then the server MUST put one of the new cipher suites defined in this document into its ServerHello's 'cipher_suites' array and eventually sends it to the client side.

The following extensions MUST conform to the new requirements:

- For key_share extension, a KeyShareEntry with SM2 related values MUST be added if the server wants to start a TLSv1.3 key negotiation using SM cipher suites.

### 3.3.2. CertificateRequest

If a CertificateRequest message is sent by the server to require the client to send its certificate for authentication purpose, the following requirements MUST be fulfilled:

- The only valid signature algorithm present in 'signature_algorithms' extension MUST be 'sm2sig_sm3'. That is to say, if server finally chooses to use a SM cipher suite, the signature algorithm for client's certificate SHOULD only be SM2 and SM3 capable ones.

#### 3.3.3. Certificate

When server sends the Certificate message which contains the server certificate to the client side, several new rules are added that will affect the certificate selection:

- The public key in the certificate MUST be a valid SM2 public key.
- The signature algorithm used by the CA to sign current certificate MUST be sm2sig_sm3.
- The certificate MUST be capable for signing, e.g., the digitalSignature bit of X.509's Key Usage extension is set.

#### 3.3.4. CertificateVerify

In the certificateVerify message, the signature algorithm MUST be sm2sig_sm3, indicating the hash function MUST be SM3 and the signature algorithm MUST be SM2 signature algorithm.

### 3.4. Key Scheduling

As described in Section 1.1, SM2 is actually a set of cryptographic algorithms including one key exchange protocol which defines methods such as key derivation function, etc. In this document, SM2 key exchange protocol is not introduced and SHALL NOT be used in the key exchange steps defined in Section 3.3. Implementations of this document SHOULD always conform to what TLSv1.3 [RFC8446] and its successors require about the key derivation and related methods.

### 3.5. Cipher

The new cipher suites introduced in this document add two new AEAD encryption algorithms, AEAD_SM4_GCM and AEAD_SM4_CCM, which stand for SM4 cipher in Galois/Counter mode and SM4 cipher [GBT.32907-2016] in Counter with CBC-MAC mode, respectively.

This section defines the AEAD_SM4_GCM and AEAD_SM4_CCM AEAD algorithms in a style of what [RFC5116] has used to define AEAD ciphers based on AES cipher.

#### 3.5.1. AEAD_SM4_GCM

The AEAD_SM4_GCM authenticated encryption algorithm works as specified in [GCM], using SM4 as the block cipher, by providing the key, nonce, and plaintext, and associated data to that mode of operation. An authentication tag conformed to what Section 5.2 of TLSv1.3 [RFC8446] requires is used, which in details SHOULD be

constructed by the TLS record header. The AEAD_SM4_GCM ciphertext is formed by appending the authentication tag provided as an output to the GCM encryption operation to the ciphertext that is output by that operation. The input and output lengths are as follows:

K_LEN is 16 octets,

P_MAX is 2^36 - 31 octets,

A_MAX is 2^61 - 1 octets,

N_MIN and N_MAX are both 12 octets, and

C_MAX is 2^36 - 15 octets.

To generate the nonce, implementations of this document MUST conform to what TLSv1.3 specifies (See [RFC8446], Section 5.3).

A security analysis of GCM is available in [MV04].

### 3.5.2. AEAD_SM4_CCM

The AEAD_SM4_CCM authenticated encryption algorithm works as specified in [CCM], using SM4 as the block cipher, by providing the key, nonce, associated data, and plaintext to that mode of operation. The formatting and counter generation function are as specified in Appendix A of that reference, and the values of the parameters identified in that appendix are as follows:

the nonce length n is 12,

the tag length t is 16, and

the value of q is 3.

An authentication tag conformed to what Section 5.2 of TLSv1.3 [RFC8446] requires is used, which in details SHOULD be constructed by the TLS record header. The AEAD_SM4_CCM ciphertext is formed by appending the authentication tag provided as an output to the CCM encryption operation to the ciphertext that is output by that operation. The input and output lengths are as follows:

K_LEN is 16 octets,

P_MAX is 2^24 - 1 octets,

A_MAX is 2^64 - 1 octets,

N_MIN and N_MAX are both 12 octets, and

C_MAX is 2^24 + 15 octets.

To generate the nonce, implementations of this document MUST conform to what TLSv1.3 specifies (See [RFC8446], Section 5.3).

A security analysis of CCM is available in [J02].

### 3.6. Hash

SM3 is defined by ISO as [ISO-SM3]. During a TLSv1.3 handshake with SM cipher suites, the hash function is REQUIRED to be SM3. Implementations MUST use SM3 for digest, key derivation, Transcript-Hash and other purposes during a TLSv1.3 key exchange process.

## 4. IANA Considerations

IANA has assigned the values {0x00, 0xC6} and {0x00, 0xC7} with the names TLS_SM4_GCM_SM3, TLS_SM4_CCM_SM3, to the "TLS Cipher Suite" registry with this document as reference, as shown below.

| Value | Description | DTLS-OK | Recommended | Reference |
|---|---|---|---|---|
| 0x00,0xC6 | TLS_SM4_GCM_SM3 | No | No | this RFC |
| 0x00,0xC7 | TLS_SM4_CCM_SM3 | No | No | this RFC |

IANA has assigned the value 0x0708 with the name sm2sig_sm3, to the "TLS SignatureScheme" registry, as shown below.

| Value | Description | DTLS-OK | Recommended | Reference |
|---|---|---|---|---|
| 0x0708 | sm2sig_sm3 | No | No | this RFC |

IANA has assigned the value 41 with the name curveSM2, to the "TLS Supported Groups" registry, as shown below.

| Value | Description | DTLS-OK | Recommended | Reference |
|---|---|---|---|---|
| 41 | curveSM2 | No | No | this RFC |

## 5. Security Considerations

At the time of writing this draft, there are no known weak keys for SM cryptographic algorithms SM2, SM3 and SM4, and no security problem has been found on those algorithms.

- The cipher suites described in this document _MUST NOT_ be used with TLSv1.2 or earlier.

## 6. References

### 6.1. Normative References

[CCM] Dworkin, M, ., "NIST Special Publication 800-38C: The CCM Mode for Authentication and Confidentiality", May 2004, <http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C.pdf>.

[GCM] Dworkin, M, ., "NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC.", November 2007, <http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf>.

[ISO-SM2] International Organization for Standardization, "IT Security techniques -- Digital signatures with appendix -- Part 3: Discrete logarithm based mechanisms", ISO ISO/IEC 14888-3:2018, November 2018, <https://www.iso.org/standard/76382.html>.

[ISO-SM3] International Organization for Standardization, "IT Security techniques -- Hash-functions -- Part 3: Dedicated hash-functions", ISO ISO/IEC 10118-3:2018, October 2018, <https://www.iso.org/standard/67116.html>.

[ISO-SM4] International Organization for Standardization, "IT Security techniques -- Encryption algorithms -- Part 3: Block ciphers", ISO ISO/IEC 18038-3:2010, December 2010, <https://www.iso.org/standard/54531.html>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5116] McGrew, D., "An Interface and Algorithms for Authenticated Encryption", RFC 5116, DOI 10.17487/RFC5116, January 2008, <https://www.rfc-editor.org/info/rfc5116>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

## 6.2. Informative References

[GBT.32905-2016] Standardization Administration of China, "Information security technology --- SM3 cryptographic hash algorithm", GB/T 32905-2016, March 2017, <http://www.gmbz.org.cn/upload/2018-07-24/1532401392982079739.pdf>.

[GBT.32907-2016] Standardization Administration of China, "Information security technology --- SM4 block cipher algorithm", GB/T 32907-2016, March 2017, <http://www.gmbz.org.cn/upload/2018-04-04/1522788048733065051.pdf>.

[GBT.32918.2-2016] Standardization Administration of China, "Information security technology --- Public key cryptographic algorithm SM2 based on elliptic curves --- Part 2: Digital signature algorithm", GB/T 32918.2-2016, March 2017, <http://www.gmbz.org.cn/upload/2018-07-24/1532401673138056311.pdf>.

[GBT.32918.5-2016] Standardization Administration of China, "Information security technology --- Public key cryptographic algorithm SM2 based on elliptic curves --- Part 5: Parameter definition", GB/T 32918.5-2016, March 2017, <http://www.gmbz.org.cn/upload/2018-07-24/1532401863206085511.pdf>.

[GMT.0009-2012] State Cryptography Administration of China, "SM2 cryptography algorithm application specification", GM/T 0009-2016, November 2012, <http://www.gmbz.org.cn/main/viewfile/2018011001400692565.html>.

[J02] Jonsson, J, ., "On the Security of CTR + CBC-MAC", 2002, <http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/proposedmodes/ccm/ccm-ad1.pdf>.

[MV04] Viega, McGrew,., "The Security and Performance of the Galois/Counter Mode (GCM)", December 2004, <http://eprint.iacr.org/2004/193>.

## Appendix A. Contributors

Wuqiong Pan
Ant Financial
wuqiong.pwq@antfin.com

Qin Long
Ant Financial
zhuolong.lq@antfin.com

Kepeng Li
Ant Financial
kepeng.lkp@antfin.com

## Appendix B. Acknowledgments

To be determined.

## Author's Address

Paul Yang
Ant Financial
No. 77 Xueyuan Road
Hangzhou 310000
China

Phone: +86-571-2688-8888
Fax: +86-571-8643-2811
Email: kaishen.yy@antfin.com