Network Working Group
Internet Draft
Intended status: Informational
Expires: May 2021

Y. Chen
J. Wang
B. Zhang
Z. Fan
X. Ma
Z. Li
J. Xie

November 30, 2020

China Academy of Information and Communications Technology

Use of the SM2 and SM3 Algorithms in Handle System

draft-chen-sm2-sm3-algorithms-00

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

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

This Internet-Draft will expire on May 30, 2021.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

The Handle System is a global name service that allows secured handle resolution and administration over the public Internet according to [1][5][3]. Handle System protocol [3] is designed to be transmitted as a byte stream via a TCP connection. In this document, SM2 and SM3 algorithms [4][5]are introduced into the handle system to enhance the security and compactivity. Trusted resolution and message credential are extended to support SM2 and SM3 algorithms.

Table of Contents

1. Introduction

RFC 3650-RFC 3652[1],[5][3] provide an open protocol, a general-purpose global name service, and a reference implementation of the protocol. RSA and DSA algorithms are commonly used when generating data signatures. With the development of cryptography and computer technology, the currently commonly used 1024-bit RSA algorithm faces security threats. In order to enhance data integrity protection based on digital signature and message verification, SM2 and SM3 algorithms are introduced to the Handle System.

2. SM2 and SM3 Algorithms Overview

Digital signature means that a signer generates a digital signature on data, and verification are performed to verify the authenticity of

the signature. Each signer has a unique pair of public key and private key, where the private key is used to generate the signature, and the verifier uses the signer's public key to verify the signature.

Verification of signature can be performed to confirm that the sender holds the corresponding private key through the public key and signature information without revealing the sender's private key. The signature can also bind the sender's identity and information to prevent imitation.

Both SM2 and SM3 are encryption algorithms approved by the Chinese National Cryptographic Bureau. The key length and the packet length are both 128 bits.

2.1. SM2 Algorithm

SM2 is an asymmetric cryptographic algorithm published by the Chinese State Cryptography Administration[4]. Its calculation speed and secret key generation speed are faster than RSA since the it is based on ECC. Currently, only SM3 can be used for data digest processing in the SM2 digital signature algorithm.

2.2. SM3 Algorithm

SM3 is a cryptographic hash algorithm independently designed by Chinese State Cryptography Administration[5]. The SM3 algorithm is normally used in data digest with a higher level of security than the MD5 algorithm and the SHA-1 algorithm.

3. Trusted Resolution with SM2 and SM3 Algorithms

Trusted resolution system is developed to achieve credibility and verification through data signing and issuing certificates based on the handle system proposed by [2]RFC3651.SM2 and SM3 algorithms could be used to generate signature and data digest during trusted resolution.

3.1. HS_CERT Extension

According to Handle System reference implementation, HS_CERT value is a handle value with a data field that consists of a JWT (json web token) structure[6] that are encoded with base64 respectively and concatenated with period. <Data> field of HS_CERT consists of the following entries:

<Header>

A JSON object with an "alg" member which suggests algorithms used in the HS_CERT value.

<Payload>

<Payload> is a JSON object with member "perms" objects ,"publicKey" objects, "iss" , "sub","exp","nbf" and "iat".

<Signature>

The <Signature> entry refers to signature information of the <Header> and <Payload> entries.

In this document, examples of HS_CERT data are illustrated considering the situation where SM2 and SM3 algorithms are used.

3.1.1. Header using SM2 and SM3

Example of the JWT Header in HS_CERT:

```
{
"alg":"SM2SM3"
}
```

The Header declares that the algorithms used in the certificate are SM2 and SM3 algorithms.

3.1.2. Payload using SM2 and SM3

The payload is where the valid information is stored. The payload is made up of two types of data: predefined claims and data extended by user according to the RFC7519[6]. After encrypted with base64, payload is stored in claim sets of JWT.

The following is an example of JWT Claims Set in HS_CERT data:

```
{
  "perms": [
    {
      "perm": "everything"
    }
  ],
  "publicKey": {
    "kty": "SM2",
    "x": "uwNiWxWtqh6TYfooxpxSpF3VEdOF0_NFrpMZu03nTVM",
    "y": "ODb6JqNLB8suvZtmccCJaTv0EpVcLGOuqPxMAM8faUw"
  },
  "iss": "100:88",
```

```
    "sub": "100:88.2045",

    "exp": 1632381987,

    "nbf": 1569309387,

    "iat": 1569309987
```

The "perms" (permission) claim indicates the user authority of the certificate.

There are four authority types defined:

"everything" means that issuer of the certificate can issue certificate to any handle in the system.

"thisHandle" means the issuer of the certificate can only issue certificate to itself.

"derivedPrefixes" indicates that the issuer can only issue certificates to its sub-naming authority.

"handlesUnderThisPrefix" means the issuer can only issue handle all the handles under this prefix.

The "publicKey" (public key) claim specifies algorithm of the public key, which is set to SM2 in this example.

3.1.3. Signatrue of the Header and Payload

The third part of <data> field in handle HS_CERT is a signature information, in which SM2 private key of the issuer is used to sign <Header> and <Payload>.

Generation of SM2 signature refers to the process of using the result of SM3 digest and the signer's private key to obtain the signature result.

3.2. HS_SIGNATRUE Extension

HS_SIGNATURE is a handle data structure where issuer signs the handle values to ensure authenticity of the information. HS_SIGNATRUE contains a data field that is a JWT (json web token) structure[6] RFC7519.

<Data> field of HS_SIGNATURE consists of the following entries:

<Header>

A JSON object with an "alg" member which suggests algorithms used in the HS_SIGNATURE value.

<Payload>

<Payload> is a JSON object with member objects "digests","iss" , "sub","exp","nbf","iat".

<Signature>

The <Signature> entry refers to signature information of the <Header>

and <Payload> entries.

Examples of HS_SIGNATRUE data using SM2 and SM3 algorithms are showing in following sections.

3.2.1. Header using SM2 and SM3

Example of the JWT Header in HS_SIGNATRUE:

```
  {
  "alg":"SM2SM3"
  }
```

The Header declares that the algorithms used in the certificate structure are SM2 and SM3 algorithms.

3.2.2. Payload using SM2 and SM3

The following is an example of JWT Claims Set in HS_SIGNATURE data:

```
{
"digests": {
          "alg": "SM3",
  "digests": [{
              "index": 100,
              "digest": "/7HpWmicaPFaMSePkbn+f/jcfAawEnieytM3qyJOha0="
            },
            {
              "index": 1,
              "digest": "300Bm9dCucz3vk+X71UWGuMe2FV62dEthRdb4iQvZzU="
            },
            {
```

```
                "index": 301,

                "digest": "aaV/sW/Eau00jtcDUzG7vqKVAc4mENJ2oZ+U4virnig="

              },

              {

                "index": 401,

                "digest": "XevUCYQfS+pucLkJA+vhVpC1lN4OVQzQugwfthpiaHk="

              }

            ]

   },

   "iss": "301:0.CR/20",

   "sub": "301:0.NA/20",

   "exp": 1640995200,

   "nbf": 1459815236,

   "iat": 1459815836

  }
```


The "digests" claim contains digests of handle values where SM3 algorithm is used to generate the digests.

The "iss" (issuer) claim identifies the handle that issued the

JWT.

The "sub" (subject) claim identifies the handle that is the

subject of the certificate.

The "exp" (expiration time) claim identifies the expiration time on

or after which the JWT MUST NOT be accepted for processing.

The "nbf" (not before) claim identifies the time before which the JWT MUST NOT be accepted for processing.

The "iat" (issued at) claim identifies the time at which the JWT was issued.

3.2.3. Signatrue of the Header and Payload

The third part of HS_SIGNATURE <data> field is a signature information, in which SM2 private key of the issuer is used to sign <Header> and <Payload> entries after base64 encoded and concatenated with period.

4. Message Credential with SM2 and SM3 Algorithms

4.1. Message Credential

The Handle system protocol provides support for the security protection of analytical data, including support for confidentiality protection based on symmetric encryption mechanism, and data integrity protection based on digital signature or message verification.

The Handle system protocol stipulates that the Handle message is made up of the following parts: message envelope, message header, message body, and message credential.

The client can request the Handle server to digitally sign the operation response message or generate a message authentication code by setting the authentication bit (CT) in the operation flag field (OpFlag) of the message header, and use the message credential field for delivery.

The message credential includes the following fields, according to RFC3652 [3]: Credential Length, Version, Reserved, Options, Signer, Type, <SignedInfo>.

4.2. Data Signing with SM2 Algorithm

<Type field> in the message credential is used to specify the type of algorithm used in the <SignedInfo> field, <SignedInfo> consists of the following fields:

SignedInfo: <Length>: 4-byte unsigned integer

DigestAlgorithm: <UTF8-String>

SignedData: <Length, Signature>

The following table lists the key type, and corresponding algorithm:

Type Name	Key Type	Algorithm
HS_SIGNED	DSA_PUB_KEY	DSA

HS_SIGNED RSA_PUB_KEY RSA-PSS

HS_SIGNED SM2_PUB_KEY SM2

4.3. SM3 Digest Algorithm in Message Credential

Where the <DigestAlgorithm> refers to the digest algorithm used to generate the digital signature and the <SignedData> contains digital signature over the Message Header and Message Body.

In this document, value of <DigestAlgorithm> could be set to "SM3" which means that SM3 algorithm is used to generate the data digest.

The following table lists the < DigestAlgorithm >s that could be supported:

Digest Algorithm Name	Algorithm
SHA-1	SHA-1
SHA-256/SHA256	SHA-256
SM3	SM3

5. Security Considerations

Data integrity under the protocol is achieved via the server's digital signature. Care must be taken to protect the server's private key from any impersonation attack.

6. IANA Considerations

7. References

7.1. Normative References

[1] Sun, S. and L. Lannom, "Handle System Overview", RFC 3650 November 2003.

[2] Sun, S., Reilly, S. and L. Lannom, "Handle System Namespace and Service Definition", RFC 3651, November 2003.

[3] Sun, S. and L. Lannom, "Handle System Overview", RFC 3652 November 2003.

[4] National Cryptography Administration, " Public Key Cryptographic Algorithm SM2 Based on Elliptic Curves", December 2010.

[5] National Cryptography Administration, "SM3 Cryptographic

Hash Algorithm", December 2010.

[6] Jones, et al., "JSON Web Token (JWT)", RFC 7519, May 2015.

8. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.The Trusted Resolution System described in this document relies on works and protocols put forward by RFC 3650, RFC 3651,and RFC 3652[1][5][3].

Author's Address

Yuying Chen
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 188 1008 2358
Email: chenyuying@caict.ac.cn

Jiahui Wang
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 186 0156 0021
Email: wangjiahui@caict.ac.cn

Bo Zhang
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191

China

Phone: +86 159 1112 3285
Email: zhangbo3@caict.ac.cn

Zhipeng Fan
CAICT

No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 159 1112 3285
Email: fanzhipeng@caict.ac.cn

Xufeng Ma
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 188 1143 3140
Email: maxufeng@caict.ac.cn

Zhiping Li
CAICT
No.52 Huayuan North Road, Haidian District

Beijing, Beijing, 100191
China

Phone: +86 185 1107 1386
Email: lizhiping@caict.ac.cn

Jiagui Xie
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 150 0138 5070
Email: xiejiagui@caict.ac.cn