GM

CRYPTOGRAPHY INDUSTRY STANDARD

OF THE PEOPLE'S REPUBLIC OF CHINA

ICS 35.040

L 80

File No.: 44634-2014

GM/T 0033-2014

Interface specifications of time stamp

时间戳接口规范

Issued on: February 13, 2014 **Implemented on: February 13, 2014**

Issued by: State Cryptography Administration

Table of Contents

Interface specifications of time stamp

1 Scope

This Standard specifies the time stamp service interface for application systems and time stamp authority systems, including the format of the time stamp requests and response messages, transmission mode, and time stamp service interface function.

This Standard is applicable to the specifications of the products related to time stamp service based on the public key cryptographic infrastructure application technology framework as well as the integration and application of time stamp services.

2 Normative references

The following documents are essential to the application of this document. For dated references, only the editions with the dates indicated are applicable to this document. For undated references, only the latest editions (including all the amendments) are applicable to this document.

GB/T 20520 *Information security technology - Public key infrastructure - Time stamp specification*

GM/T 0006 *Cryptographic application identifier criterion specification*

GM/T 0010 *SM2 cryptography message syntax specification*

GM/T 0019 *Universal cryptography service interface specification*

RFC 3066 *Tags for the Identification of Languages*

RFC 3161 *Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)*

RFC 3369 *Cryptographic Message Syntax (CMS)*

3 Terms and definitions

The following terms and definitions are applicable to this document.

3.1 Certification authority; CA

An entity that performs full life-cycle management of a digital certificate, also known as an electronic certification authority.

3.2 Cryptographic hash algorithm

It is also known as hash algorithm, cryptographic hash algorithm or Hash algorithm. The algorithm maps an arbitrarily long bit string to a fixed-length bit string, satisfying the following three properties:

(1) It is computationally difficult to find an input that maps to the output for a given output;
(2) It is computationally difficult to find another input that maps to the same output for a given input;
(3) It is computationally difficult to find that different inputs map to the same output.

3.3 Digital signature

The result obtained by the signer performing crypto-operation on the hash value of the data to be signed by using the private key. The result can only be verified by the signer's public key for verifying the integrity of the data to be signed, the authenticity of the signer's identity and the non-repudiation of the signature.

3.4 SM2 algorithm

A public key cryptographic algorithm based on elliptic curves, with a key length of 256 bits.

3.5 Time stamp; TS

Data that is obtained by signing time and other data to be signed, for indicating the time attribute of the data.

3.6 Time stamp authority system

Management system used to generate and manage the time stamps.

3.7 Time stamp service

The time stamp authority system provides the user with the time stamp service. The file is provided by the user. The time stamp authority system issues a time stamp to this file.

4 Abbreviations

The following abbreviations are applicable to this document:

DER Distinguished Encoding Rules

algorithm approved by the State Cryptography Administration. If the TSA does not identify the given cryptographic hash algorithm or if the cryptographic hash algorithm does not comply with the relevant requirements of the State Cryptography Administration, the TSA shall refuse to provide the time stamp service and set the badAlg's pkiStatusInfo structure in the return message.

— The reqPolicy field represents the security policy. The security policy is provided by the TSA. The user is able to select the required security policy to set this field. The type of reqPolicy is TSAPolicyId, which is defined as follows:

TSAPolicyId ::= OBJECT IDENTIFIER

— The nonce field is a random number that is used for verifying the legitimacy of response messages and prevent replay attacks without a reliable local clock.

— The certReq field is used to request the TSA public key certificate. In case of true, the TSA shall provide its public key certificate in the response message. The certificate is pointed out by the SigningCertificate attribute ESSCertID in the response message, and is stored in the Certificates field of the SignedData structure in the response message.

— Extension is an extension field that is used for adding additional information to the application message. For an extension, whether it is a critical extension or not, as long as it appears in the request message and cannot be identified by the TSA, the TSA shall not generate a time stamp and return the failure information (unacceptedExtension).

The time stamp request message does not need to provide the requester's identity. If the TSA needs to identify the requester's identity, a separate two-way identity authentication shall be carried out. The realization of two-way identity authentication is not specified in this Standard.

7.2 Response format

After receiving the application message, the TSA shall return a response message to the requester whether the application succeeds or fails. The response message is a correct time stamp or a time stamp that contains the failure information.

The ASN.1 data format that defines the time stamp response message is as follows:

— The version field indicates the version number of the time stamp.

— The policy field shall indicate which policy of the TSA the response message is generated from. If similar fields appear in the Time Stamp Req [Translator note: TimeStampReq?], there shall be the same value herein, otherwise the error (unacceptedPolicy) shall be returned. This policy may include, but is not limited to, the following similar information:

- Under what conditions is this time stamp used;
- The validity of the time stamp log so that it can be verified later that the time stamp is trustworthy.

— The messageImprint shall have the same value as a similar field in the TimeStampReq, provided that the digest value has the same length as expected by the hashAlgorithm tag's algorithm.

— The serialNumber field is an integer assigned by the TSA. For each time stamp issued by a given TSA, the serialNumber shall be unique (that is, the TSA's name and serial number can identify a time stamp flag). It shall be noted that this feature shall also be retained even after a possible service interruption (such as crash).

— The genTime is the time when TSA creates a time stamp, expressed in UTC time to reduce the confusion caused by the usage of local time zone.

— The accuracy indicates the maximum error that may occur in time. The sum of genTime and accuracy values can be used to obtain the upper time limit for TSA to create the time stamp. Similarly, the lower time limit for TSA to create the time stamp can be obtained by subtracting the accuracy. The specific definition is as follows:

```
Accuracy ::= SEQUENCE{
    seconds         INTEGER                 OPTIONAL,--s
    millis          INTEGER (1..999)        OPTIONAL,--ms
    micros          [1]INTEGER (1..999)     OPTIONAL--μs
    }
```

If the seconds, millis or micros does not appear, the values of these fields that do not appear shall be assigned 0. When the option of accuracy does not appear, the accuracy may be obtained from other ways, such as TSAPolicyId.

— The ordering represents the sorting conditions of the time stamp. If the ordering field does not appear, or if the ordering field appears but is set to false, the genTime field will only indicate the time when TSA creates a time stamp. In this case, only when the difference between the first genTime and the second genTime in two time-stamps is greater than the sum of the precision of these two genTime, the time stamp flag issued by the same

TSA or different TSAs may be sorted. If the ordering field appears and is set to true, each time stamp issued by the same TSA may be sorted by genTime, regardless of the precision of genTime.

— If the nonce field appears in the TimeStampReq, it shall also appear here, and the value of which shall also be equal to that in the TimeStampReq.

— The purpose of the tsa field is to provide a clue to identify the name of the TSA. If present, it shall be the same as one of the subject names in the certificate that validates the time stamp.

— The extensions field is a common practice for adding additional information in the future. Special extension types may be defined by the organization or group and declared for registration.

8 Communication modes of the time stamp service and the time stamp authority system

8.1 E-mail mode

In e-mail mode, the user sends a time stamp application to a TSA-specified e-mail address by e-mail. The TSA returns the issued time stamp to the customer by e-mail. The following MIME objects shall be used for application and issuance:

a) Application message:

Content-Type:application/timestamp-query
Content-Transfer-Encoding:base64

DER encoding of the application message, followed by base64 encoding.

b) Response message:

Content-Type:application/timestamp-reply
Content-Transfer-Encoding:base64

DER encoding of the response message, followed by base64 encoding.

8.2 File mode

In file mode, the user sends the application message to the TSA in a file with the extension ".tsq". The TSA also returns the generated response message to the user in a file with the extension ".tsr". The request and response files contain only the DER encoding of the message. File transfer shall be trusted.

CREATE a time stamp request	STF_CreateTSRequest
CREATE a time stamp response	STF_CreateTSResponse
VERIFY the time stamp validity	STF_VerifyTSValidity
GET the main time stamp information	STF_GetTSInfo
PARSE the time stamp details	STF_GetTSDetail

The return values of the above functions are shown in Annex A.

The communication modes used in this Standard for time stamp request and response are based on Chapter 8. The definition of the communication interface is not within the scope of this Standard.

9.2 InitEnvironment function

Prototype: SGD_UINT32 STF_InitEnvironment(void** phTSHandle)

Description: CREATE a time stamp environment handle.

Parameter: phTSHandle[OUT]: Time stamp environment handle pointer.

Return value: 0: Success;

Others: Failed.

9.3 ClearEnvironment function

Prototype: SGD_UINT32 STF_ ClearEnvironment(void* hTSHandle)

Description: CLEAR a time stamp environment handle.

Parameter: hTSHandle[IN]: Time stamp environment handle.

Return value: 0: Success;

Others: Failed.

9.4 Create TS request

Prototype: SGD_UINT32 STF_CreateTSRequest (void* hTSHandle,

This is an excerpt of the PDF (Some pages are marked off intentionally)

------ The End ------