GM

CRYPTOGRAPHIC INDUSTRY STANDARD

OF THE PEOPLE'S REPUBLIC OF CHINA

ICS 35.040

L 80

GM/T 0067-2019

Interface specifications of authentication based on digital certificate

基于数字证书的身份鉴别接口规范

Issued on: July 12, 2019 **Implemented on: July 12, 2019**

Issued by: State Cryptography Administration

Table of Contents

Interface specifications of authentication based on digital certificate

1 Scope

This standard specifies the digital certificate-based identity authentication interface in the upper application of the public key cryptographic infrastructure system.

This standard applies to the development of identity authentication services in the upper application of the public key cryptographic infrastructure system, the R&D and testing of the identity authentication system of the certificate application support platform; it can also be used to guide the application system to standardize the use of certificates for identity authentication.

2 Normative references

The following documents are essential to the application of this document. For the dated documents, only the versions with the dates indicated are applicable to this document; for the undated documents, only the latest version (including all the amendments) are applicable to this standard.

GB/T 15843.1-2017 Information technology - Security techniques - Entity authentication - Part 1: General

GB/T 15843.3-2016 Information technology - Security techniques - Entity authentication - Part 3: Mechanisms using digital signature techniques

3 Terms and definitions

The following terms and definitions apply to this document.

3.1

Certificate authentication system

A system that manages the entire life cycle of digital certificates such as the sign-off, issuance, renewal, revocation of digital certificates.

3.2

An elliptic curve public key cryptographic algorithm, the key length of which is 256 bits.

3.9

SM3 algorithm

A cryptographic hash algorithm, the output of which is 256 bits.

4 Abbreviations

The following abbreviations apply to this document.

CA: Certificate authority

CN: Common name

CRL: Certificate revocation list

DN: Distinguished name

LDAP: Lightweight directory access protocol

OID: Object identifier

PKI: Public key infrastructure

5 Implementation method

5.1 Overview

The realization of identity authentication includes proxy identity authentication mode and call mode. Identity authentication T and application B are a mutually trusted whole. The identity authentication mechanism used in these two modes follows GB/T 15843.3-2016.

5.2 Proxy authentication mode

In this mode, the identity of user A is authenticated by the proxy identity authentication service T; then the result of the authentication is passed to application B. This identity authentication mode is called proxy identity mode, which is generally implemented by message.

The authentication protocol is carried out between the user A and the proxy

b) When the proxy identity authentication service T receives a message containing TokenAT, it performs the following steps:

1) Verify the validity of A's certificate, including the validity period, whether it is issued by a trusted organization, the status of the certificate, verification of the certificate key usage;

2) Verify TokenAT.

c) The proxy identity authentication service T sends T's certificate and TokenTA to A (see the form of TokenTA in 5.3.2 of GB 15843.3-2016);

d) When receiving a message containing TokenTA, user A performs the following steps:

1) Verify the validity of T's certificate, including the validity period, whether it is issued by a trusted organization, whether it is in the blacklist, verification of the certificate key usage;

2) Verify TokenTA.

e) The proxy identity authentication service T passes the verified identity of A to application B.

5.3 Call mode

After the application obtains the user's identity, it actively calls the external service interface of the identity authentication service to perform identity authentication to obtain the identity authentication result, which is called the call mode. It is generally implemented by interface functions.

In this mode, application B starts the verification process and authenticates user A. It controls the uniqueness and timeliness of the authentication protocol by generating and verifying random numbers R_B (see Appendix B of GB/T 15843.1-2017). The verification mechanism is as shown in Figure 3:

Figure 5 -- Structure of identity authentication interface system

The identity authentication service module on which the identity authentication interface specification is based on is located between the application system and the cryptographic service interface. It provides identity authentication service for the application system through this interface. The cryptographic operations required by the identity authentication module are implemented by invoking cryptographic services through the cryptographic service interface specification.

The identity authentication interface is logically divided into two parts, namely: environment function and identity authentication function.

7.2.2 Environmental functions

The environment function is responsible for creating and managing the secure program space, responsible for creating and managing the various resources and signals required in the secure program space, ensuring that the secure program space will not be illegally accessed during the running of the application program, thereby causing information leakage. The environment function is responsible for completing the secure connection with the identity authentication service, ensuring that the subsequent security operations are carried out in a secured and trusted program space.

When an application uses the identity authentication interface, it must first call the initialization environment function (SIF_Initialize) to create and initialize a secure application space; complete the connection and initialization with the identity authentication service. Before the application program is terminated, it shall call the clear environment function (SIF_Finalize) to terminate the connection with the identity authentication service, destroy the created security program space, prevent the security risks caused by memory residue.

7.2.3 Identity authentication function

The identity authentication function realizes the acquisition of user information and the verification of user identity (the main means are through certificate verification and analysis of the certificate revocation list). The application program realizes the identity authentication based on the digital certificate by calling the identity authentication function.

7.3 Message definition

7.3.1 Message format definition

The message includes two parts: the message header and the message body,

<msg>

<msg_head>

<msg_type>0</msg_type>

<msg_id>0100</msg_id>

<version>1</version>

</msg_head>

<msg_body>

<connectid> Connect ID </connectid>

</msg_body>

</msg>

b) User identity gets response

<? xmlversion = "1.0" encoding = "UTF-8"?>

<msg>

<msg_head>

<msg_type>1 or 2</msg_type>

<msg_id>0100</msg_id>

<version>1</version>

</msg_head>

<msg_body>

<connectid> Connect ID </connectid>

<userinfo> Identity information </userinfo>

<error_no> Error code </error_no>

</msg_body>

</msg>

7.3.4 User credential generation message

<msg>

<msg_head>

<msg_type>0</msg_type>

<msg_id>1000</msg_id>

<version>1</version>

</msg_head>

<msg_body>

<userseed> Random information (Base64 encoding) </userseed>

<cert> Certificate (Base64 encoded) for generating user credentials </cert>

</msg_body>

</msg>

d) User credential generation response

<? xmlversion = "1.0" encoding = "UTF-8"?>

<msg>

<msg_head>

<msg_type>1 or 2</msg_type>

<msg_id>1000</msg_id>

<version>1</version>

</msg_head>

<msg_body>

<usertoken> Generated user credentials (Base64 encoding) </usertoken>

<error_no> Error code </error_no>

</msg_body>

</msg>

7.3.5 User credential verification message

identity authentication service (Base64 encoding) </resultsign>

<error_no> Error code </error_no>

</msg_body>

</msg>

7.4 Function interface definition

7.4.1 Overview

Interface functions include the following specific functions. For the return value of each function, please refer to Appendix A for the definition of error codes:

a) Initialization: SIF_Initialize

b) Termination: SIF_Finalize

c) Get interface version: SIF_GetVersion

d) Random information needed to generate user credentials: SIF_GenRandom

e) Generate user credentials: SIF_GenUserToken

f) Verify user credentials: SIF_VerifyUserToken

g) Confirm the authenticity of the verification result: SIF_VerifyResult

h) Get user identity: SIF_GetUserInfo

7.4.2 Initialization function

Prototype:

SGD_INT32SIF_Initialize(SGD _CHAR* pucIpAddr,

SGD_INTiPort,SGD_VOID* phHandle);

Description: Initialize the identity authentication service and create an identity authentication service handle

Parameter:

pucIpAddr [in]: The address of the identity authentication server; it may be NULL, which means that the remote service is not connected

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

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