|  |  |
| --- | --- |
| 3GPP TR 31.822 V18.1.0 (2023-09) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Core Network and Terminals;  Study on GBA\_U Based APIs;  (Release 18) | |
|  | |
|  |  |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2023, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 5

1 Scope 7

2 References 7

3 Definitions of terms, symbols and abbreviations 7

3.1 Abbreviations 7

4 Key Issues 8

4.1 Key Issues1: Support for GBA\_U\_APIs 8

4.2 Key Issues2: Support for Access Control to GBA\_U\_APIs 8

5 Solutions 8

5.1 Solution1: Classes and Interfaces of GBA\_U\_APIs 8

5.1.1 Description 8

5.1.1.1 General 8

5.1.1.2 Class cipher 8

5.1.1.2.1 Algorithm identifier 8

5.1.1.2.2 init 8

5.1.1.2.3 update 9

5.1.1.2.4 doFinal 9

5.1.1.3 Class signature 10

5.1.1.3.1 Algorithm identifier 10

5.1.1.3.2 init 10

5.1.1.3.3 update 10

5.1.1.3.4 sign 10

5.1.1.3.5 verify 11

5.1.2 Solution Evaluation 11

5.2 Solution2: Access Control to GBA\_U\_APIs 11

5.2.1 Description 11

5.2.2 Solution Evaluation 12

5.3 Solution 3: Access Control to GBA\_U\_APIs through install parameters 12

5.3.1 Description 12

5.3.2 Solution Evaluation 12

6 Overall Evaluation 12

6.1 Evaluation criteria 12

6.1.1 Overview 12

6.1.2 Key issues addressed 12

6.1.3 Impact on USIM and ISIM 12

6.1.4 Impact on UICC 12

6.1.5 Backward compatibility 13

6.1.6 Impact on core and RAN networks 13

6.2 Solutions evaluations 13

6.2.1 Overview 13

6.2.2 Solution 1 13

6.2.2.1 Key issues addressed 13

6.2.2.2 Impact on USIM and ISIM 13

6.2.2.3 Impact on UICC 13

6.2.2.4 Backward compatibility 13

6.2.2.5 Impact on core and RAN networks 13

6.2.3 Solution 2 13

6.2.3.1 Key issues addressed 13

6.2.3.2 Impact on USIM and ISIM 13

6.2.3.3 Impact on UICC 14

6.2.3.4 Backward compatibility 14

6.2.3.5 Impact on core and RAN networks 14

6.2.4 Solution 3 14

6.2.4.1 Key issues addressed 14

6.2.4.2 Impact on USIM and ISIM 14

6.2.4.3 Impact on UICC 14

6.2.4.4 Backward compatibility 14

6.2.4.5 Impact on core and RAN networks 14

7 Conclusion 14

Annex A (informative): Change history 15

# Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# 1 Scope

The present document studies the GBA\_U based APIs.

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic bootstrapping architecture".

[3] 3GPP TS 31.102: "Characteristics of the Universal Subscriber Identity Module (USIM) application".

[4] 3GPP TS 31.130: "(U)SIM Application Programming Interface (API); (U)SIM API for Java™ Card".

[5] ISO/IEC 18033-3: "2010/AMD1:2021 Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers – Amendment 1: SM4".

[6] ISO/IEC 10118-3: "2018 IT Security techniques – Hash-functions – Part 3: Dedicated hash-functions".

[7] 3GPP TS 31.116: "Remote APDU Structure for (U)SIM Toolkit applications".

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

3GPP 3rd Generation Partnership Project

AES Advanced Encryption Standard

AID Application Identifier

API Application Programming Interface

GBA Generic Bootstrapping Architecture

GBA\_U GBA with UICC-based enhancements

NAF Network Application Function

SM ShangMi

# 4 Key Issues

## 4.1 Key Issues1: Support for GBA\_U\_APIs

Since the Ks\_int\_NAF is derived from the Long Term Key of the UICC, and the GBA mechanism is highly recognized, the APIs for Ks\_int\_NAF should be defined, so other applications could use the Ks\_int\_NAF for security purpose.

## 4.2 Key Issues2: Support for Access Control to GBA\_U\_APIs

When an application calls GBA\_U\_APIs, the Ks\_int\_NAF is required. Since Ks\_int\_NAF is associated with B-TID and NAF\_ID, it is not allowed to use by all of applications. Therefore, access control should be supported to specify the AID that could call GBA\_U\_APIs with NAF\_ID

# 5 Solutions

## 5.1 Solution1: Classes and Interfaces of GBA\_U\_APIs

### 5.1.1 Description

#### 5.1.1.1 General

The USIM GBA\_U\_APIs consist of the package *uicc.usim.gba,* which include cipher and signature two classes.

Table 5.1.1.1: GBA\_U API

|  |  |  |
| --- | --- | --- |
| CLASS Summary | Method Summary | Description |
| cipher | init | Initializes the Cipher object with the appropriate Ks\_int\_NAF Key and algorithm specific parameters. |
| update | Generates encrypted/decrypted output from input data. |
| doFinal | Generates encrypted/decrypted output from all/last input data. |
| signature | init | Initializes the Signature object with the appropriate Ks\_int\_NAF Key and algorithm specific parameters. |
| update | Accumulates a signature of the input data. |
| sign | Generates the signature of all/last input data. |
| verify | Verifies the signature of all/last input data against the passed in signature. |

#### 5.1.1.2 Class cipher

##### 5.1.1.2.1 Algorithm identifier

Table 5.1.1.2.1: Algorithm identifier

|  |  |
| --- | --- |
| Identifier | Algorithm |
| 0x0D | AES-128-CBC-NOPAD |
| 0x0E | AES-128-ECB-NOPAD |
| 0x18 | SM4-ECB |
| 0x19 | SM4-CBC |

##### 5.1.1.2.2 init

public void init(byte algorithm, byte[] nafID, short nafOff, short nafLen, byte theMode, byte[] bArray, short bOff, short bLen)

Initializes the Cipher object with the appropriate Ks\_int\_NAF Key and algorithm specific parameters.

Parameters:

algorithm - see Table 5.1.1.2.1

nafID - the value of NAF ID

nafOff - offset of NAF ID

nafLen - length of NAF ID

theMode - one of MODE\_DECRYPT or MODE\_ENCRYPT

bArray - byte array containing algorithm specific initialization info

bOff - offset within bArray where the algorithm specific data begins

bLen - length of algorithm specific parameter data

##### 5.1.1.2.3 update

public short update(byte[] inBuff, short inOffset, short inLength, byte[] outBuff, short outOffset)

Generates encrypted/decrypted output from input data using Ks\_int\_NAF.This method is intended for multiple-part encryption/decryption operations.

This method requires temporary storage of intermediate results. In addition, if the input data length is not block aligned (multiple of block size) then additional internal storage may be allocated at this time to store a partial input data block. This may result in additional resource consumption and/or slow performance.

Parameters:

inBuff - the input buffer of data to be encrypted/decrypted

inOffset - the offset into the input buffer at which to begin encryption/decryption

inLength - the byte length to be encrypted/decrypted

outBuff - the output buffer, may be the same as the input buffer

outOffset - the offset into the output buffer where the resulting ciphertext/plaintext begins

Returns:

number of bytes output in outBuff.

##### 5.1.1.2.4 doFinal

public short doFinal(byte[] inBuff, short inOffset, short inLength, byte[] outBuff, short outOffset)

Generates encrypted/decrypted output from all/last input data using Ks\_int\_NAF. This method must be invoked to complete a cipher operation. This method processes any remaining input data buffered by one or more calls to the update() method as well as input data supplied in the inBuff parameter.

Parameters:

inBuff - the input buffer of data to be encrypted/decrypted

inOffset - the offset into the input buffer at which to begin encryption/decryption

inLength - the byte length to be encrypted/decrypted

outBuff - the output buffer, may be the same as the input buffer

outOffset - the offset into the output buffer where the resulting output data begins

Returns:

number of bytes output in outBuff.

#### 5.1.1.3 Class signature

##### 5.1.1.3.1 Algorithm identifier

Table 5.1.1.3.1: Algorithm identifier

|  |  |
| --- | --- |
| Identifier | Algorithm |
| 0x21 | [ALG\_HMAC\_SHA1](file:///C:\Users\huang\Documents\gprs\1_D\Spec\JAVA_CARD\JAVACard\JavaCard\javacard_specifications-3_0_1-RR\classic\api_classic\javacard\security\Signature.html#ALG_HMAC_SHA_256) |
| 0x22 | [ALG\_HMAC\_SHA\_256](file:///C:\Users\huang\Documents\gprs\1_D\Spec\JAVA_CARD\JAVACard\JavaCard\javacard_specifications-3_0_1-RR\classic\api_classic\javacard\security\Signature.html#ALG_HMAC_SHA_256) |
| 0x23 | [ALG\_HMAC\_SM3](file:///C:\Users\huang\Documents\gprs\1_D\Spec\JAVA_CARD\JAVACard\JavaCard\javacard_specifications-3_0_1-RR\classic\api_classic\javacard\security\Signature.html#ALG_HMAC_SHA_256) |

##### 5.1.1.3.2 init

public void init (byte algorithm, byte mode, byte[] nafID, short nafOff, short nafLen)

Initializes the Signature object with the appropriate Ks\_int\_NAF Key and algorithm specific parameters.

Parameters:

algorithm - see Table 5.1.1.3.1

mode - one of MODE\_SIGN or MODE\_VERIFY

nafID - the value of NAF ID

nafOff - offset of NAF ID

nafLen - length of NAF ID

##### 5.1.1.3.3 update

public void update(byte[] inBuff, short inOffset, short inLength)

Accumulates a signature of the input data. This method requires temporary storage of intermediate results.

Parameters:

inBuff - the input buffer of data to be encrypted/decrypted

inOffset - the offset into the input buffer at which to begin encryption/decryption

inLength - the byte length to be encrypted/decrypted

##### 5.1.1.3.4 sign

public short sign(byte[] inBuff, short inOffset, short inLength, byte[] sigBuff, short sigOffset)

Generates the signature of all/last input data using Ks\_int\_NAF.

Parameters:

inBuff - the input buffer of data to be signed

inOffset - the offset into the input buffer at which to begin signature generation

inLength - the byte length to sign

sigBuff - the output buffer to store signature data

sigOffset - the offset into sigBuff at which to begin signature data

Returns:

number of bytes of signature output in sigBuff.

##### 5.1.1.3.5 verify

public boolean verify (byte[] inBuff, short inOffset, short inLength, byte[] sigBuff, short sigOffset, short sigLength)

Verifies the signature of all/last input data against the passed in signature using Ks\_int\_NAF.

Parameters:

inBuff - the input buffer of data to be verified

inOffset - the offset into the input buffer at which to begin signature generation

inLength - the byte length to sign

sigBuff - the input buffer containing signature data

sigOffset - the offset into sigBuff where signature data begins

sigLength - the byte length of the signature data

Returns:

true if the signature verifies, false otherwise

Note: if sigLength is inconsistent with this Signature algorithm, false is returned.

### 5.1.2 Solution Evaluation

This solution addresses the key issue 1. It defines APIs for Ks\_int\_NAF derived from GBA, which include cipher and signature methods, and the APIs could be used by the applications on the USIM.

## 5.2 Solution2: Access Control to GBA\_U\_APIs

### 5.2.1 Description

This EF is available at the USIM ADF (Application DF) level.

This EF contains the list of AID associated with NAF\_ID, and it indicates the application which could uses the Ks\_int\_NAFcorresponding to NAF\_ID.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Identifier: '6Fxx' | | Structure: Linear fixed | | | Optional | |
| Record length: Z bytes | | | Update activity: low | | | |
| Access Conditions:  READ ADM  UPDATE ADM  DEACTIVATE ADM  ACTIVATE ADM | | | | | | |
| Bytes | Description | | | M/O | | Length |
| 1 to Z | AID of NAF Key LV objects | | | M | | Z bytes |

AID of NAF Key LV object

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Value | M/O | Length (bytes) |
| Length | Y | M | Note |
| AID value | -- | M | 5-16bytes |
| Length | Y | M | Note |
| NAF\_ID value | -- | M | Y |
| Note: The length is coded according to ISO/IEC 8825-1 | | | |

### 5.2.2 Solution Evaluation

This solution addresses the key issue 2. It provides a mechanism for access control to GBA\_U\_APIs, which defines an EF of USIM to store the list of AID associated with NAF\_ID.

## 5.3 Solution 3: Access Control to GBA\_U\_APIs through install parameters

### 5.3.1 Description

An alternative approach to Solution 2 for Access Control may be based on application installation parameters. Applications using APIs that require NAA provider or issuer provider – like file system access – are authorized to perform the intended operation by adding specific install parameters as specified in 3GPP TS 31.116 [7].

According to 3GPP TS 31.116 [7], depending on issuer configuration, such install parameters may be protected with Remote Applet Management credentials that identify the off-card entity performing the Remote Applet Management operation and also with additional credentials (ETSI DAP key to protect Access and Toolkit install parameters) that can be used as an additional authentication according to the card issuer’s security policy.

It is then proposed to add a new install parameter subject to additional authentication according to the card issuer’s security policy to authorize applications to use GBA\_U\_APIs.

As an example of the install parameter structure a new access domain parameter as specified in ETSI TS 102 226 § 8.2.1.3.2.5.1 can be requested to be added as “API Access mechanism” (e.g. value ‘03’).

### 5.3.2 Solution Evaluation

This solution addresses the key issue 2 in an alternative way than Solution 2. It provides a mechanism for access control to GBA\_U\_APIs that is managed by the RAM off-card entity potentially in conjunction with ETSI DAP mechanism according to the card issuer’s security policy.

# 6 Overall Evaluation

## 6.1 Evaluation criteria

### 6.1.1 Overview

In addition to the key issues described in the present document, the potential solutions also need to be evaluated against operational criteria when assessing their suitability. This clause details the evaluation criteria for the solutions.

### 6.1.2 Key issues addressed

Each solution clearly identifies the key issues addressed by the solution.

The evaluation assesses whether this key issue is met and any limitations on the way it is met (e.g. if the solution is only applicable to aspects of the key issue).

### 6.1.3 Impact on USIM and ISIM

The evaluation should clearly identify the types of USIM/ISIM that the solution works with, whether modification of existing USIMs/ISIMs is required and describe the impacts on USIM/ISIM.

### 6.1.4 Impact on UICC

The evaluation should clearly identify the UICC that the solution works with, whether modification of the existing UICC is required and describe the impacts on the UICC.

### 6.1.5 Backward compatibility

The evaluation should clearly identify that the solution is backward compatible.

### 6.1.6 Impact on core and RAN networks

The evaluation should clearly identify any changes required in the core network and/or RAN.

## 6.2 Solutions evaluations

### 6.2.1 Overview

This Technical Report describes Classes and Interfaces and Access Control for GBA\_U\_APIs. In this document, 2 Key Issues are derived, and all of them are addressed by solutions.

### 6.2.2 Solution 1

#### 6.2.2.1 Key issues addressed

This solution addresses key issue 1.

Table 6.1: Key issue 1 and solution evaluation

|  |  |  |
| --- | --- | --- |
| Key Issue | Solution | Evaluation  (subclause reference) |
| Key Issues1: Support for GBA\_U\_APIs | Solution1:Classes and Interfaces of GBA\_U\_APIs | 5.1.2 |

#### 6.2.2.2 Impact on USIM and ISIM

The solution has no impact on the USIM neither on the ISIM.

#### 6.2.2.3 Impact on UICC

UICC is impacted with new API to be supported.

#### 6.2.2.4 Backward compatibility

This solution is backward compatible.

#### 6.2.2.5 Impact on core and RAN networks

This solution has no impact on the core network or RAN.

### 6.2.3 Solution 2

#### 6.2.3.1 Key issues addressed

This solution addresses key issue 2.

Table 6.2: Key issue 2 and solution evaluation

|  |  |  |
| --- | --- | --- |
| Key Issue | Solution | Evaluation  (subclause reference) |
| Key Issues2: Support for Access Control to GBA\_U\_APIs | Solution2: Access Control to GBA\_U\_APIs | 5.2.2 |

#### 6.2.3.2 Impact on USIM and ISIM

USIM is impacted with new file to be defined.

#### 6.2.3.3 Impact on UICC

This solution has no impact on the UICC.

#### 6.2.3.4 Backward compatibility

This solution is backward compatible.

#### 6.2.3.5 Impact on core and RAN networks

This solution has no impact on the core network or RAN.

### 6.2.4 Solution 3

#### 6.2.4.1 Key issues addressed

This solution addresses key issue 2.

Table 6.3: Key issue 2 solution evaluation

|  |  |  |
| --- | --- | --- |
| Key Issue | Solution | Evaluation  (subclause reference) |
| Key Issues2: Support for Access Control to GBA\_U\_APIs | Solution3: Access Control to GBA\_U\_APIs | 5.3.2 |

#### 6.2.4.2 Impact on USIM and ISIM

None.

#### 6.2.4.3 Impact on UICC

A new install parameter needs to be defined at UICC level.

#### 6.2.4.4 Backward compatibility

This solution is backward compatible.

#### 6.2.4.5 Impact on core and RAN networks

This solution has no impact on the core network or RAN.

# 7 Conclusion

This technical report fulfills the objectives of the study on GBA\_U Based APIs, all of the key issues have been addressed and the solutions have been evaluated, normative work in 3GPP CT6 WG should be based on solutions in the present document.

Annex A (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2022-08 | CT6#112-e | C6-220493 |  |  |  | Initial version of TR - Skeleton document | 0.1.0 |
| 2022-08 | CT6#112-e | C6-220493 |  |  |  | Implement the pCRs C6-220515, C6-220516, and C6-220517. | 0.1.0 |
| 2023-03 | CT6#114 | C6-230102 |  |  |  | Implement the pCRs C6-230098, C6-230099, C6-230100, and C6-230101. | 0.2.0 |
| 2023-03 | CT6#114 |  |  |  |  | Implement the pCR C6-230113. Update the change history. | 0.3.0 |
| 2023-03 | CT#99 | CP-230028 |  |  |  | Presented for information and approval | 1.0.0 |
| 2023-03 | CT#99 |  |  |  |  | TR Approved | 18.0.0 |
| 2023-09 | CT#101 | CP-232144 | 0001 | 1 | F | Remove SM1 Algorithm form the Class cipher | 18.1.0 |