

**Source:** SA WG3 (Security)

**Title:** CR to 33.220: Introduction of a UICC-based Generic Bootstrapping Architecture (Rel-6)

**Document for:** Approval

**Agenda Item:** 7.3.3

---

SA Doc number	Spec	CR	Rev	Phase	Subject	Cat	Version-Current	SA WG3 Doc number	Workitem
SP-040378	33.220	004	-	Rel-6	Introduction of a UICC-based Generic Bootstrapping Architecture	B	6.0.0	S3-040413	SEC1-SC

## CHANGE REQUEST

⌘ **33.220 CR 004** ⌘ rev **-** ⌘ Current version: **6.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** UICC apps  ME  Radio Access Network  Core Network

<b>Title:</b>	⌘ Introduction of a UICC-based Generic Bootstrapping Architecture		
<b>Source:</b>	⌘ SA WG3		
<b>Work item code:</b>	⌘ SEC1-SC	<b>Date:</b>	⌘ 13/05/2004
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ Rel-6
	<i>Use one of the following categories:</i> <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP <a href="#">TR 21.900</a> .		<i>Use one of the following releases:</i> <b>2</b> (GSM Phase 2) <b>R96</b> (Release 1996) <b>R97</b> (Release 1997) <b>R98</b> (Release 1998) <b>R99</b> (Release 1999) <b>Rel-4</b> (Release 4) <b>Rel-5</b> (Release 5) <b>Rel-6</b> (Release 6)

<b>Reason for change:</b>	⌘ Addition of a UICC-based Generic Bootstrapping mechanism		
<b>Summary of change:</b>	⌘		
<b>Consequences if not approved:</b>	⌘ The feature cannot be used i.e. for MBMS Key Management which would restrict the storage for MBMS keys to the ME only.		

<b>Clauses affected:</b>	⌘ - Introduction of new section 5 of TS 33.220 v6.0.0 - Addition of new definitions and abbreviation to section 3										
<b>Other specs affected:</b>	<table border="1" style="display: inline-table; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">Y</td> <td style="width: 20px; text-align: center;">N</td> </tr> <tr> <td style="text-align: center;">X</td> <td style="text-align: center;"></td> </tr> <tr> <td style="text-align: center;"></td> <td style="text-align: center;">X</td> </tr> <tr> <td style="text-align: center;"></td> <td style="text-align: center;">X</td> </tr> </table> Other core specifications Test specifications O&M Specifications	Y	N	X			X		X	⌘ TS 24.109, 29.229, 31.102	
Y	N										
X											
	X										
	X										
<b>Other comments:</b>	⌘ -										

BEGIN OF CHANGE

---

# 1 Scope

The present document describes the security features and a mechanism to bootstrap authentication and key agreement for application security from the 3GPP AKA mechanism. Candidate applications to use this bootstrapping mechanism include but are not restricted to subscriber certificate distribution TS 33.221 [5]. Subscriber certificates support services whose provision mobile operator assists, as well as services that mobile operator provides.

The scope of this specification includes a generic AKA bootstrapping function, an architecture overview and the detailed procedure how to bootstrap the credential.

[Section 4 of this specification describes a mechanism, called GBA\\_ME, to bootstrap authentication and key agreement, which does not require any changes to the UICC. Section 5 of this specification describes a mechanism, called GBA\\_U, to bootstrap authentication and key agreement, which does require changes to the UICC, but provides enhanced security by storing certain derived keys on the UICC.](#)

NOTE: The specification objects are scheduled currently in phases. For this specification release, only the case is considered where bootstrapping server functionality and network application function are located in the same network as the HSS. In further specification release, other configurations may be considered.

---

END OF CHANGE

BEGIN OF CHANGE

---

---

# 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

**Bootstrapping Server Function:** BSF is hosted in a network element under the control of an MNO.

Editor's note: Definition to be completed.

**Network Application Function:** NAF is hosted in a network element under the control of an MNO.

Editor's note: Definition to be completed.

**Transaction Identifier:**

Editor's note: Definition to be completed.

[ME-based GBA: in GBA\\_ME, all GBA-specific functions are carried out in the ME. The UICC is GBA-unaware. If the term GBA is used in this document without any further qualification then always GBA\\_ME is meant, cf. section 4 of this specification.](#)

UICC-based GBA: this is a GBA with UICC-based enhancement. In GBA U, the GBA-specific functions are split between ME and UICC, cf. section 5 of this specification.

## 3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AK	Anonymity Key
AKA	Authentication and Key Agreement
BSF	Bootstrapping Server Function
CA	Certificate Authority
GAA	Generic Authentication Architecture
GBA	Generic Bootstrapping Architecture
<u>GBA ME</u>	<u>ME-based GBA</u>
<u>GBA U</u>	<u>GBA with UICC-based enhancements</u>
HSS	Home Subscriber System
IK	Integrity Key
KDF	Key Derivation Function
<u>Ks_int</u>	<u>Derived key in GBA U which remains on UICC</u>
<u>Ks_ext</u>	<u>Derived key in GBA U</u>
MNO	Mobile Network Operator
NAF	Network Application Function
PKI	Public Key Infrastructure

---

END OF CHANGE

BEGIN OF CHANGE

---

---

## 5 UICC-based enhancements to Generic Bootstrapping Architecture (GBA U)

It is assumed that the UICC, BSF, and HSS involved in the procedures specified in this section are capable of handling the GBA U specific enhancements. For issues of migration from UICC, BSF, and HSS, which are not GBA U - aware, see Annex C of this specification. The procedures specified in this section also apply if NAF is not GBA U aware, but, of course, in that case there are no benefits of the GBA U specific enhancements.

### 5.1 Architecture and reference points for bootstrapping with UICC-based enhancements

The text from section 4.3 of this specification applies also here, with the addition that the interface between the ME and the UICC, as specified in TS 31.102 [1], needs to be enhanced with GBA U specific commands. The requirements on these commands can be found in section 5.2.1, details on the procedures in section 5.3.

## 5.2 Requirements and principles for bootstrapping with UICC-based enhancements

The requirements and principles from section 4.4 also apply here with the following addition:

### 5.2.1 Requirements on UE

The 3G AKA keys CK and IK resulting from a run of the protocol over the Ub reference point shall not leave the UICC.

The UICC shall be able to distinguish between authentication requests for GBA\_U, and authentication requests for other 3G authentication domains.

Upon an authentication request from the ME, which the UICC recognises as related to GBA\_U, the UICC shall derive two keys from CK and IK. All 3G MEs are capable of such a request.

Upon request from the ME, the UICC shall be able to derive further NAF-specific keys from the derived key stored on the UICC. Only GBA\_U aware 3G MEs are capable of such a request.

Editor's Note: The location (whether in the UICC or in the ME) of the storage of Ks\_ext is ffs.

## 5.3 Procedures for bootstrapping with UICC-based enhancements

### 5.3.1 Initiation of bootstrapping

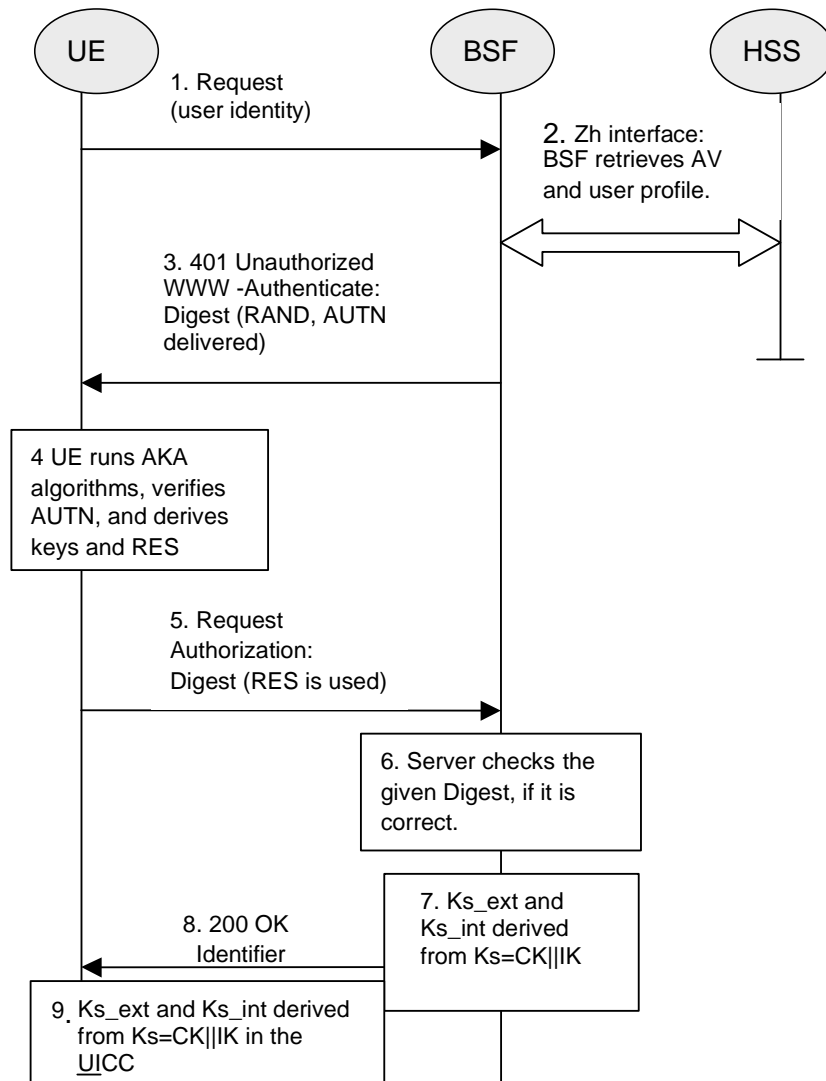
The text from section 4.5.1 of this document applies also here.

### 5.3.2 Bootstrapping procedure

The procedure specified in this section differs from the procedure specified section 4.5.2 in the generation of the Authentication Vector in the HSS and the local handling of keys in the UE and the BSF. The messages exchanged over the Ub reference point are identical for both procedures.

When a UE wants to interact with a NAF, and it knows that the bootstrapping procedure is needed, it shall first perform a bootstrapping authentication (see Figure 5.1). Otherwise, the UE shall perform a bootstrapping authentication only when it has received bootstrapping initiation required message or a bootstrapping renegotiation indication from the NAF, or when the lifetime of the key in UE has expired (cf. subclause 5.3.3).

NOTE: The main steps from the specifications of the AKA protocol in TS 33.102 [2] and the HTTP digest AKA protocol in RFC 3310 [4] are repeated in Figure 5.1 for the convenience of the reader. In case of any potential conflict, the specifications in TS 33.102 [2] and RFC 3310 [4] take precedence.



**Figure 5.1: The bootstrapping procedure with UICC-based enhancements**

1. [The ME sends an HTTP request towards the BSF.](#)
  2. [The BSF retrieves the user profile and one or a whole batch of Authentication Vectors \(AV, AV = RAND||AUTN||XRES||CK||IK\) over the Zh reference point from the HSS. The HSS recognises that the UICC is GBA U aware and that the request for AVs came from a GBA U aware BSF, and generates a GBA U-AV.. If the BSF received GBA U-AVs then it stores the XRES after flipping the least significant bit.](#)
- [Editor's Note: The GBA U-AV will be described within Annex D of this specification](#)
3. [Then BSF forwards the RAND and AUTN to the UE in the 401 message \(without the CK, IK and XRES\). This is to demand the UE to authenticate itself.](#)
  4. [The ME sends RAND and AUTN to the UICC. The UICC checks AUTN to verify that the challenge is from an authorised network; the UICC also calculates CK, IK and RES. This will result in session keys CK and IK in both BSF and UICC.](#)

5. The UICC checks if a GBA U-AV was received as specified in step 2 of this clause. If this is not the case, the UICC transfers RES, CK and IK to the ME, and the ME proceeds according to the procedures specified in section 4 of this document, without involving the UICC any further. If a GBA U-AV was received, the UICC then applies a suitable key derivation function h1 to Ks, which is the concatenation of CK and IK, and possibly further h1-key derivation parameters to obtain two keys, Ks\_ext and Ks\_int, each of length 128 bit, i.e.  $h1(Ks, h1 \text{ key derivation parameters}) = Ks\_ext || Ks\_int$  (cf. also Figure 5.2). The UICC then transfers RES (after flipping the least significant bit) and Ks\_ext to the ME and stores Ks\_int/Ks\_ext on the UICC.

Editor's Note: The definition of the h1 is left to ETSI SAGE and is to be included in the Annex B of the present specification

Editor's Note: The location (whether in the UICC or in the ME) of the storage of Ks\_ext is ffs.

6. The ME sends another HTTP request, containing the Digest AKA response (calculated using RES), to the BSF.

7. The BSF authenticates the UE by verifying the Digest AKA response.

8. The BSF generates the key Ks by concatenating CK and IK. The BSF checks if the AV was a GBA U-AV as specified in step 2 of this clause. If this is not the case, the BSF applies the procedures specified in section 4 of this document. If the GBA U-AV was recognized then the BSF applies the key derivation function h1 to Ks and possibly further h1-key derivation parameters to obtain two keys, Ks\_ext and Ks\_int, in the same way as the UICC did in step 5. The Transaction Identifier value shall be also generated in format of NAI by taking the RAND value from step 3, and the BSF server name, i.e. RAND@BSF\_servers\_domain\_name.

9. The BSF shall send a 200 OK message, including the Transaction Identifier, to the UE to indicate the success of the authentication. In addition, in the 200 OK message, the BSF shall supply the lifetime of the keys Ks\_ext and Ks\_int. The lifetimes of the keys Ks\_ext and Ks\_int shall be the same.

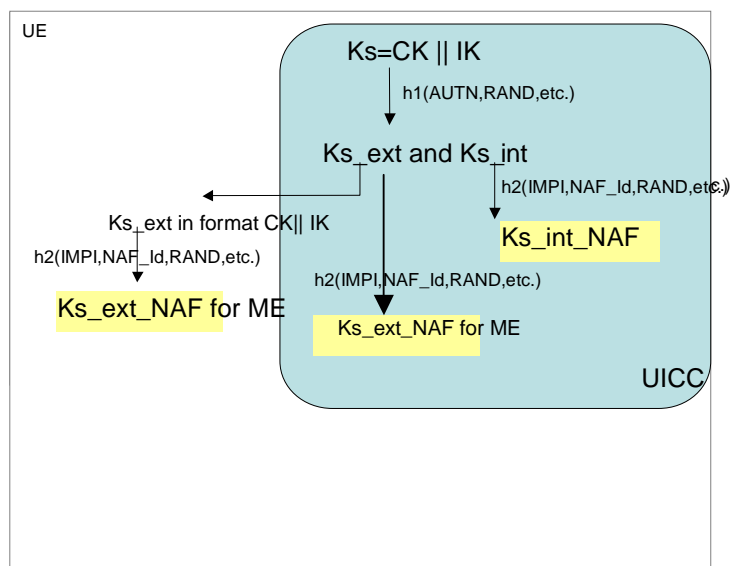
10. The BSF shall use the keys Ks\_ext and Ks\_int to derive the NAF-specific keys Ks\_ext\_NAF and Ks\_int\_NAF, if requested by a NAF over the Zn reference point. Ks\_ext\_NAF and Ks\_int\_NAF are used for securing the Ua reference point. The UE shall use the key Ks\_ext to derive the NAF-specific key Ks\_ext\_NAF, if applicable. The UICC shall use the key Ks\_int to derive the NAF-specific key Ks\_int\_NAF, if applicable.

Ks\_ext\_NAF is computed as  $Ks\_ext\_NAF = h2(Ks\_ext, h2\text{-key derivation parameters})$ , and Ks\_int\_NAF is computed in the UICC as  $Ks\_int\_NAF = h2(Ks\_int, h2\text{-key derivation parameters})$ , where h2 is a suitable key derivation function, and the h2-key derivation parameters include the user's IMPI, the NAF\_Id and RAND. The NAF\_Id consists of the full DNS name of the NAF.

Editor's Note: The definition of the h2 is left to ETSI SAGE and is to be included in the Annex B of the present specification

NOTE: The NOTE2 of clause 4.5.2 also applies here.

The ME, the UICC and the BSF store the keys Ks\_ext and Ks\_int together with the associated Transaction Identifier for further use, until the lifetime of Ks\_ext and Ks\_int has expired, or until the keys Ks\_ext and Ks\_int are updated.



**Figure 5.2: Key derivation for GBA-aware UICC when GBA-run was triggered**

### 5.3.3 Procedures using bootstrapped Security Association

Before communication between the UE and the NAF can start, the UE and the NAF first have to agree whether to use shared keys obtained by means of the GBA. If the UE does not know whether to use GBA with this NAF, it uses the Initiation of Bootstrapping procedure described in section 5.3.1.

Once the UE and the NAF have established that they want to use GBA then every time the UE wants to interact with a NAF the following steps are executed as depicted in figure 5.3.

Next, the UE and the NAF have to agree, which type of keys to use,  $Ks\_ext\_NAF$  or  $Ks\_int\_NAF$ , or both. The default is the use of  $Ks\_ext\_NAF$  only. This use is also supported by MEs and NAFs, which are GBA\_U unaware. If  $Ks\_int\_NAF$ , or both, are to be used, this use has to be agreed between UE and NAF prior to the execution of the procedure described in the remainder of this clause 5.3.3. How this agreement is reached is application-specific and is not within the scope of this document.

**NOTE:** Such an agreement could e.g. be reached by manual configuration, or by an application-specific protocol step.

**Editor's Note:** The support of unaware GBA\_U MEs, which are GBA\_ME aware only is FFS

In general, UE and NAF will not yet share the key(s) required to protect the  $U_a$  reference point. If they do not, the UE proceeds as follows:

- if  $Ks\_ext\_NAF$  is required and a key  $Ks\_ext$  is available in the UE, the UE derives the key  $Ks\_ext\_NAF$  from  $Ks\_ext$ , as specified in clause 5.3.2;
- if  $Ks\_int\_NAF$  is required and a key  $Ks\_int$  is available in the UICC, the ME requests the UICC to derive the key  $Ks\_int\_NAF$  from  $Ks\_int$ , as specified in clause 5.3.2;

**NOTE:** if it is not desired by the UE to use the same  $Ks\_ext/int$  to derive more than one  $Ks\_ext/int\_NAF$  then the UE should first agree on new keys  $Ks\_ext$  and  $Ks\_int$  with the BSF over the  $U_b$  reference point, as specified in clause 5.3.2, and then proceeds to derive  $Ks\_ext\_NAF$  or  $Ks\_int\_NAF$ , or both, as required.



- if Ks\_ext and Ks\_int are not available in the UE, the UE first agrees on new keys Ks\_ext and Ks\_int with the BSF over the Ub reference point, as specified in clause 5.3.2, and then proceeds to derive Ks\_ext\_NAF or Ks\_int\_NAF, or both, as required;
- if the NAF shares a key with the UE, but the NAF requires an update of that key, it shall send a suitable bootstrapping renegotiation request to the UE and terminate the protocol used over Ua reference point. The form of this indication depends on the particular protocol used over Ua reference point. If the UE receives a bootstrapping renegotiation request, it starts a run of the protocol over Ub, as specified in section 5.3.2, in order to obtain new keys.

NOTE: If the shared keys between UE and NAF become invalid, the NAF can set deletion conditions to the corresponding security association for subsequent removal.

NOTE: If it is not desired by the NAF to use the same Ks to derive more than one Ks\_int/ext\_NAF then the NAF should always reply to the first request sent by a UE by sending a key update request to the UE.

UE and NAF can now start the communication over Ua reference point using the keys Ks\_ext\_NAF or Ks\_int\_NAF, or both, as required. They proceed as follows:

- The UE supplies the Transaction Identifier to the NAF, as specified in clause 5.3.2, to allow the NAF to retrieve the corresponding keys from the BSF

NOTE: To allow for consistent key derivation in BSF and UE, both have to use the same FQDN for derivation (cf. NOTE2 in section 4.5.2). For each protocol used over Ua it shall be specified if only cases (1) and (2) of NOTE2 are allowed for the NAF or if the protocol used over Ua shall transfer also the FQDN used for key derivation by UE to NAF.

NOTE: The UE may adapt the keys Ks\_ext\_NAF or Ks\_int\_NAF to the specific needs of the Ua reference point. This adaptation is outside the scope of this specification.

- when the UE is powered down, or when the UICC is removed, any GBA\_U keys shall be deleted from storage in the ME. There is no need to delete keys Ks\_int and Ks\_int\_NAF from storage in the UICC;

NOTE: After each run of the protocol over the Ub reference point, new keys Ks\_ext and Ks\_int, associated with a new transaction identifier, are derived in the UE according to clause 5.3.2, so that it can never happen, that keys Ks\_ext and Ks\_int with different transaction identifiers simultaneously exist in the UE.

- When new keys Ks\_ext and Ks\_int are agreed over the Ub reference point and new NAF-specific keys need to be derived for one NAF\_Id, then both, Ks\_ext\_NAF and Ks\_int\_NAF (if present), shall be updated for this NAF\_Id, but further keys Ks\_ext\_NAF or Ks\_int\_NAF relating to other NAF\_Ids, which may be stored on the UE, shall not be affected;

NOTE: This rule ensures that the keys Ks\_ext\_NAF and Ks\_int\_NAF are always in synch at the UE and the NAF.

NAF now starts communication over the Zn reference point with the BSF.

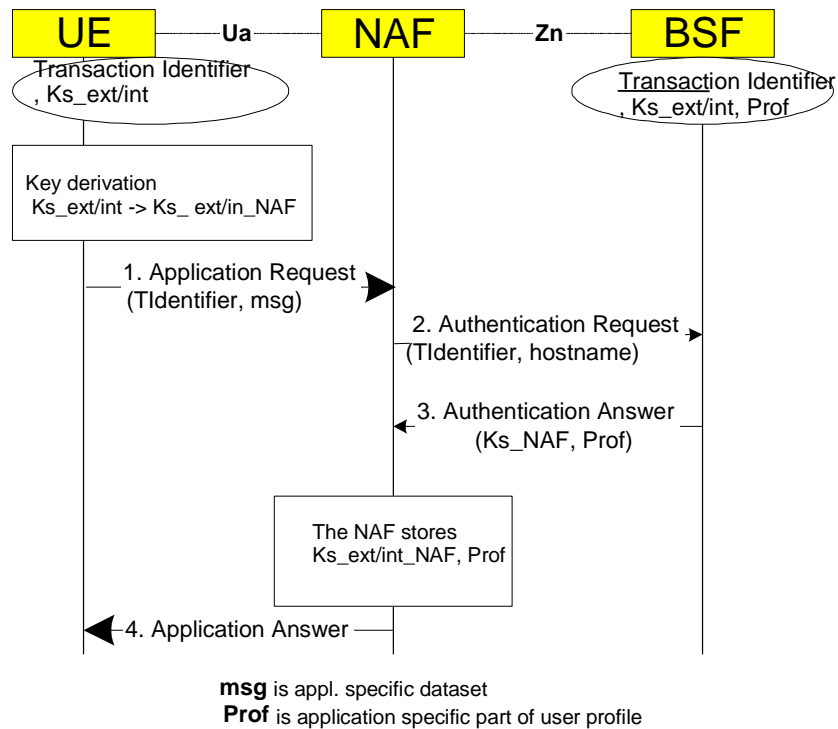
- The NAF requests from the BSF the keys corresponding to the Transaction Identifier, which was supplied by the UE to the NAF over the Ua reference point. If the NAF is GBA\_U aware it indicates this by including a corresponding flag in the request. If the NAF has several FQDNs, which may be used in conjunction with this specification, then the NAF shall transfer in the request over Zn the same FQDN, which was used over Ua (cf. note above on key derivation in this section).
- With the keys request over the Zn reference point, the NAF shall supply NAF's public hostname that UE has used to access NAF to BSF, and BSF shall be able to verify that NAF is authorized to use that hostname.
- The BSF derives the keys Ks\_ext\_NAF, and Ks\_int\_NAF (if additionally required), as specified in clause 5.3.2. If the NAF indicated in its request that it is GBA\_U aware, the BSF supplies to NAF both keys, Ks\_ext\_NAF, and Ks\_int\_NAF, otherwise the BSF supplies only Ks\_ext\_NAF. In addition, the BSF supplies the lifetime time of these keys. If the key identified by the Transaction Identifier supplied by the

NAF is not available at the BSF, the BSF shall indicate this in the reply to the NAF. The NAF then indicates a bootstrapping renegotiation request (See Figure 4.5) to the UE.

NOTE: The NAF may adapt the keys  $Ks\_ext\_NAF$  and  $Ks\_int\_NAF$  to the specific needs of the Ua reference point in the same way as the UE did. This adaptation is outside the scope of this specification.

The NAF now continues with the protocol used over the Ua reference point with the UE.

Once the run of the protocol used over Ua reference point is completed the purpose of bootstrapping is fulfilled as it enabled the UE and NAF to use Ua reference point in a secure way.



**Figure 5.3: The bootstrapping usage procedure with UICC-based enhancements**

### 5.3.4 Procedure related to service discovery

The text from section 4.5.4 of this document applies also here.

END OF CHANGE

BEGIN OF CHANGE

## [Annex C \(informative\): Issues regarding migration from GBA\\_ME to GBA\\_U](#)

[This Annex contains a few rules which should be heeded when upgrading from GBA\\_ME to GBA\\_U in order to avoid incompatibilities.](#)

[The HSS \(AuC\) shall be upgraded first before NAFs are introduced in the network that uses the GBA\\_U services and the GBA-aware UICC has to be administrated within the HSS so that the HSS \(AuC\) can generate the GBA\\_U-AV.](#)

[The HSS \(AuC\) does NOT need to be upgraded in case GBA-aware UICC's are introduced within the network, but no NAFs make use of it.](#)

[The upgrade of the BSF to support GBA\\_U shall occur no later than that of the HSS.](#)

BEGIN OF CHANGE

---

## Annex **D**~~E~~ (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
2004-03	SP-23	SP-040175	-	-	Presented for approval at TSG SA #23	1.2.1	2.0.0
2004-03	SP-23	-	-	-	Approved and placed under Change Control (Rel-6)	2.0.0	6.0.0

---

END OF CHANGE