3GPP TSG SA WG3 Security — S3#34 **S3-040490**

July 6 - 9, 2004, Acapulco, Mexico

CR-Form-v7

CHANGE REQUEST

⌘ **TS 33.220** **CR** **CRNum** ⌘ **rev** ⌘ Current version: **6.1.0** ⌘

*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⌘ [X] ME [ ] Radio Access Network [ ] Core Network [X]

Title:	⌘	Introducing the Special-RAND mechanism for GBA_U
Source:	⌘	Siemens
Work item code:	⌘	SSC-GBA
Date:	⌘	28/06/2004
Category:	⌘	**B**
Release:	⌘	Rel-6

Use one of the following categories:
- ***F*** *(correction)*
- ***A*** *(corresponds to a correction in an earlier release)*
- ***B*** *(addition of feature),*
- ***C*** *(functional modification of feature)*
- ***D*** *(editorial modification)*

Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Introducing a special RAND format that tells the UICC to run the key derivation procedures as described by GBA_U (section 5 of TS 33.220). Removing the Editor's Note on GBA_U AV.
Summary of change:	⌘	Introducing a special RAND format to securely execute GBA on the UICC
Consequences if not approved:	⌘	The UICC has no indication whether to run GBA_U or a UMTS authentication (as for GBA). The GBA_U concept cannot be used.

Clauses affected:	⌘	5.3.2, New normative Annex D

Other specs affected: ⌘	Y	N		⌘
		X	Other core specifications	
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

***** BEGIN OF CHANGE *****

5.3.2 Bootstrapping procedure

The procedure specified in this clause differs from the procedure specified clause 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 (see clause 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.

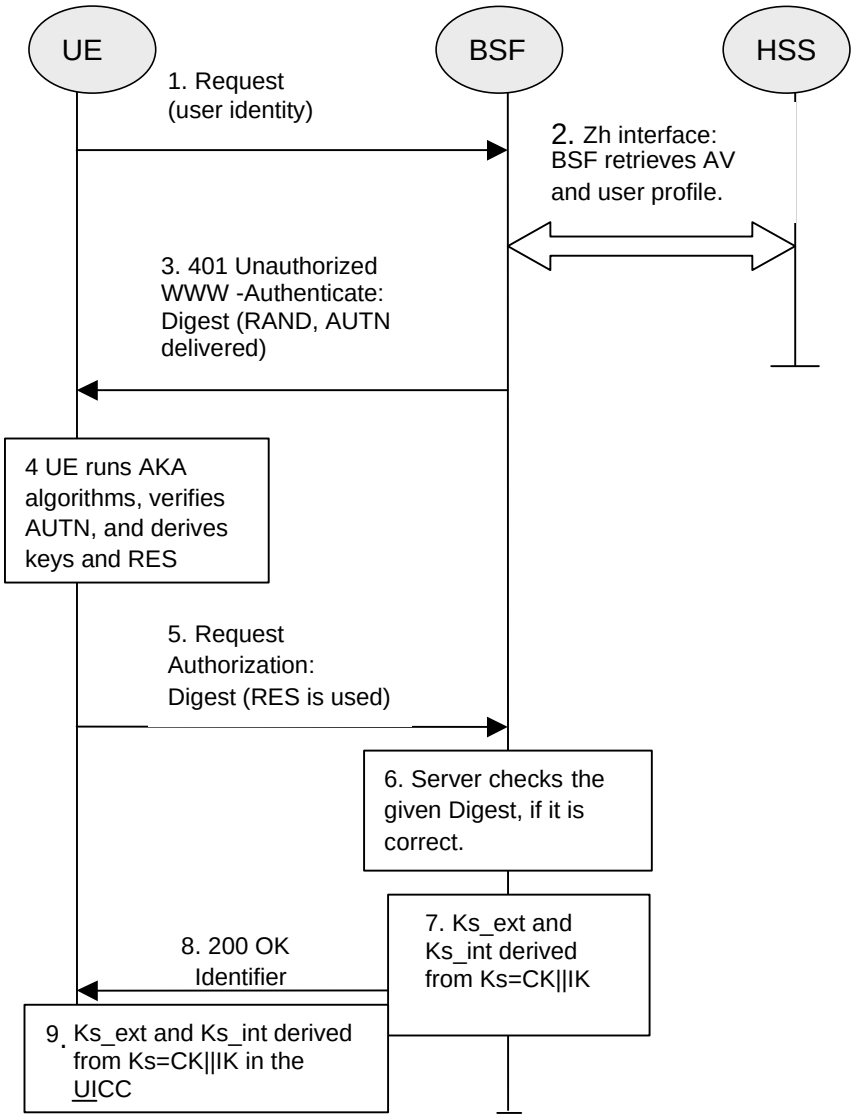


Figure 5.1: The bootstrapping procedure with UICC-based enhancements

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.

~~Editors' Note~~NOTE: The GBA_U-AV is described within ~~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 key derivation parameters) = Ks_ext || Ks_int (see 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.

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

Editors' 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 clause 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-key derivation parameters), and Ks_int_NAF is computed in the UICC as Ks_int_NAF = h2 (Ks_int, h2-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.

Editors' 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 NOTE 2 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.

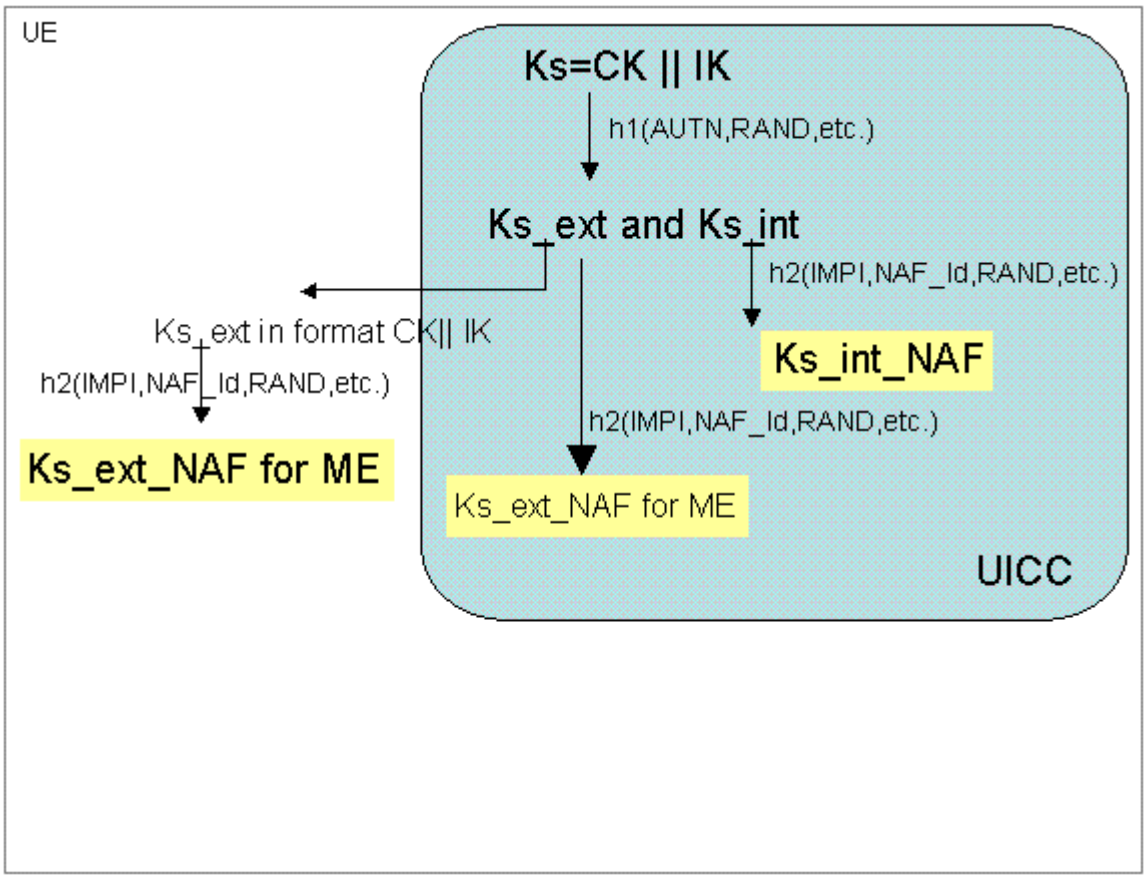


Figure 5.2: Key derivation for GBA-aware UICC when GBA-run was triggered

***** End of change *****

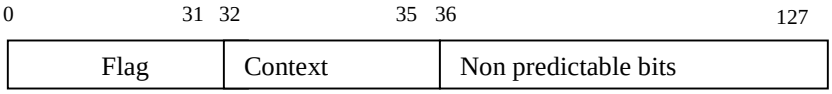
***** Next change *****

Annex D (normative): Structure of the RAND for GBA_U

This section specifies the structure of the special-RAND for GBA_U. A GBA-aware UICC shall recognize the GBA_U special-RAND and perform the key derivations that are described within section 5 and Annex B. An HSS (AuC) that supports GBA_U shall only generate the special-RANDs defined within this Annex C when, for a GBA-aware UICC, an Authentication Vector Request originates from a GBA_U aware BSF.

The ME takes the received RAND unmodified as the input to the authentication and ciphering key generation algorithms A3 and A8.

The structure of special RAND values is the following for GBA_U:



Bit 0 is the most significant bit of RAND and bit 127 is the least significant bit of RAND.

- length of Flag: 32 bits;
- length of Context: 4 bits;
- length of Non predictable bits: 92 bits.

Flag:

In special-RAND values, the flag is set to a particular binary pattern (all 32 bits set to 1) to indicate that bits 32-35 (Context bits) shall be interpreted by the UICC.

Context:

The value 0000 is used for GBA_U.

***** End of change *****