3GPP TSG SA WG3 Security — S3#34 **S3-040654**
6 - 9 July 2004, Acapulco, Mexico

CR-Form-v7

CHANGE REQUEST

⌘ **33.220** **CR** ⌘ **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 [X] Radio Access Network [ ] Core Network [X]

Title:	⌘	Creation of GBA_U AV in the BSF
Source:	⌘	Axalto, Siemens
Work item code:	⌘	SSC-GBA
Date:	⌘	9/07/2004
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:	⌘	Procedure to produce GBA_U specific AV was missing and required by the GBA_U functionality.
Summary of change:	⌘	-Addtion of mandatory support of GBA_U in BSF -Addition of GBA UICC capabilities in GBA user security settings -Addition of MAC modification by the BSF for GBA_U -Addition of specific AKA procedure in the UICC for GBA_U -Removal of some GBA_U impacts to HSS
Consequences if not approved:	⌘	GBA_U feature is incomplete.

Clauses affected:	⌘	5, 5.2.X (New), 5.3.2, Annex C (Deleted)

		Y	N			
Other specs affected:	⌘	X		Other core specifications	⌘	TS 31.102, TS 31.103, TS 29.109
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	-

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] and TS 31.103 [ ], 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.

5.2.x Requirements on BSF

BSF shall support both GBA_U and GBA_ME bootstrapping procedures. The decision on running one or the other shall be based on subscription information (i.e. UICC capabilities)

The BSF shall be able to acquire the UICC capabilities related to GBA as part of the GBA user security settings received from the HSS.

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 and Authentication Vectors 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.

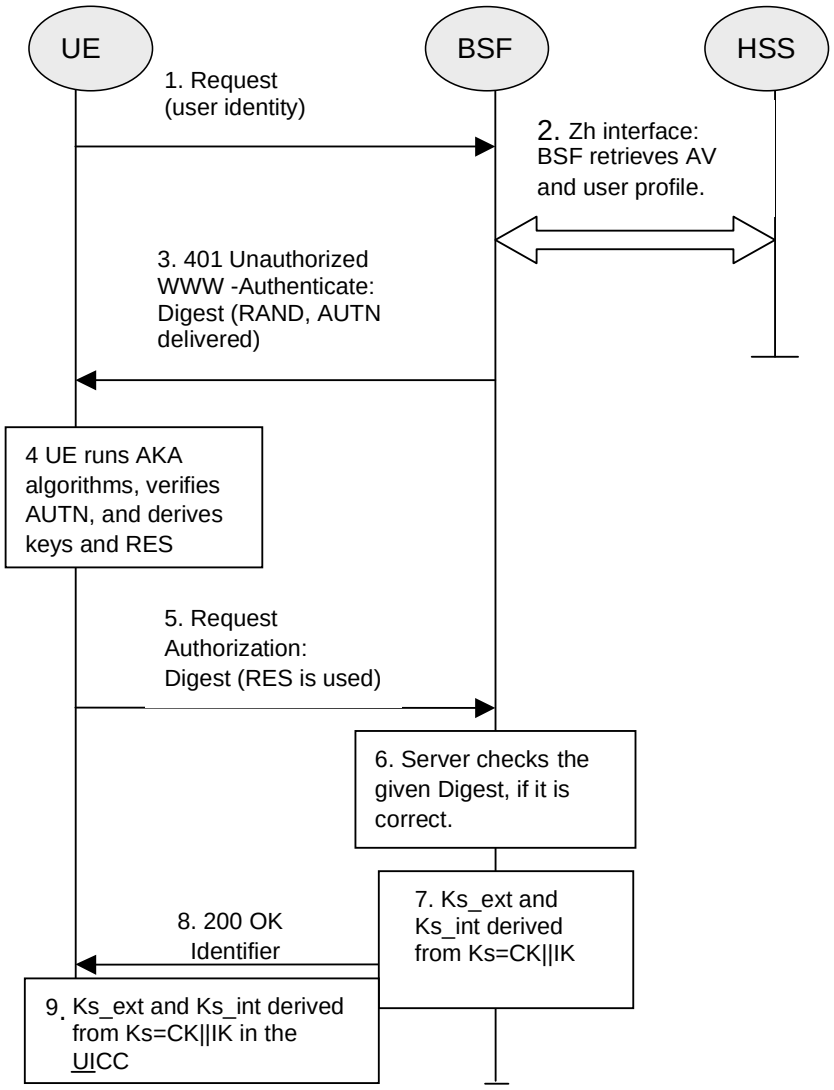

Figure 5.1: The bootstrapping procedure with UICC-based enhancements

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.~~ The BSF can then decide to perform GBA_U, based on the user security settings (USSs). In this case, the BSF proceeds in the following way:

   -BSF computes MAC* = MAC ⊕ SHA-1(IK_1) (where IK= IK_1|| IK_2 and ⊕ is a exclusive or as described in 33.102 [2])

   Editor's note: The exact format of the MAC modification function is to be reviewed. The output of SHA-1 needs to be truncated to exact amount of bits needed (64 bits).

   ~~- If the BSF received GBA_U-AVs then it~~ BSF 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* (where AUTN* = SQN ⊕ AK || AMF || MAC*) 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 calculates I~~C~~K and MAC (by performing MAC= MAC* ⊕ SHA-1(IK_1)). Then the UICC checks AUTN (i.e. SQN ⊕ AK || AMF || MAC) 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, t~~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 (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 t~~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.

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.

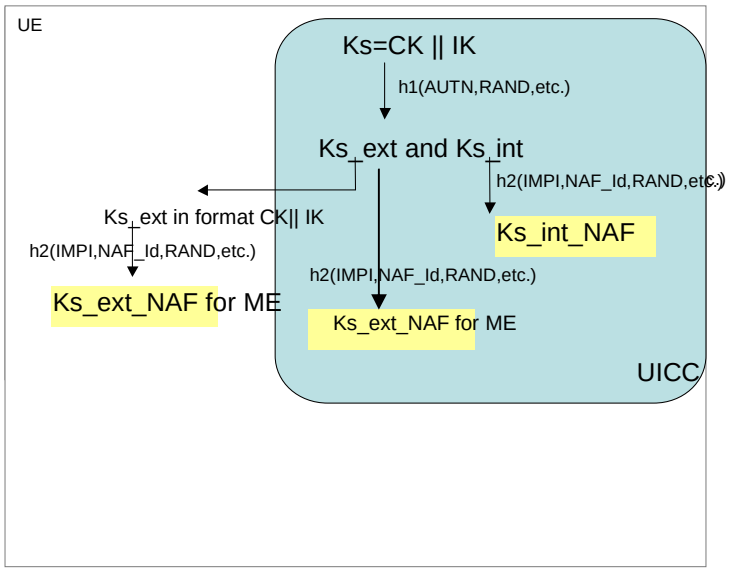

Figure 5.2: Key derivation for GBA-aware UICC when GBA-run was triggered

---

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.~~

END OF CHANGE