3GPP TSG SA WG3 Security — S3#34 **S3-040515**
6 - 9 July 2004
Acapulco, Mexico

Agenda item: 7.9.2 GBA

Title: Clarification of Ks_ext

Source: Huawei

Document for: Discussion and Decision

1 Introduction

According to the enhanced Generic Bootstrapping Architecture(UICC-based GBA), the two key Ks_int and Ks_ext derived from Ks will be used to different model. Ks and Ks_int store in UICC and never leave, when ME need the Ks_int_NAF , ME ask UICC to generate it; the Ks_ext can be sent to ME, when ME need the Ks_ext_NAF , ME can generate it directly. This contribution discuss the storage and usage of Ks_ext.

2 Discussion

The capability of ME:

When the UICC, BSF and HSS all are GBA-U-aware entity, the UICC-based GBA can start work regardless whether the ME is GBA-U-aware. There may be two kind of ME : one is GBA-U-aware, and another is GBA-ME-aware only.

From the view of storage, the Ks_ext can be stored in different location for those two different capability of ME.

GBA-U-aware ME: the Ks_ext can be stored in UICC and ME or only in UICC. When the Ks_ext is only in UICC, the ME may request the Ks_ext_NAF from UICC.

GBA-ME-aware ME: the Ks_ext must be stored in ME. Because the ME don't know he should request Ks_ext_NAF from UICC.

Security of Ks_ext: .It is obvious the UICC is the safer storage for Ks_ext than ME, so if it is possible , the Ks_ext also should remain in UICC.

If the GBA-U-aware ME store the Ks_ext, the UICC must keep the copy of Ks_ext. Because of reason of security , when the UE is powered down, or when the UICC is removed, any GBA_U keys shall be deleted from storage in the ME, in this case the UICC have to resend the Ks_ext to ME. To the GBA-U-aware ME, if the Ks_ext remain in UICC , not only the security can be enhanced but also the duplicated resource in ME can be saved.

UICC determine the Ks_ext remain in UICC or send to ME base on the capability of ME.

There is a profile information in ME，and it is sent by the ME to the UICC as part of the UICC initialization

procedure(described in TS 31.111), so the UICC can get the capability information of ME and know the ME is the GBA-U-aware ME or the GBA-ME-aware ME only. If the ME is the GBA-ME-aware only, then UICC send the Ks_ext to ME, otherwise the UICC remain the Ks_ext in UICC. When ME need the Ks_ext_NAF, ME ask UICC to generate it. Because the ME may request Ks_int_NAF, the ME should indicate the Ks_ext_NAF or Ks_int_NAF is needed in the request message.

3 Conclusion

According to the above analysis, ask SA3 endorse the follow summarize:

1 The Ks_ext should be remained in UICC as possible.

2 The UICC can determine remain the Ks_ext in UICC or send it to ME.

3 GBA-U-aware ME can request Ks_ext_NAF or Ks_int_NAF from UICC explicitly

If the above summarize are endorsed , approval the attached CR.

CR-Form-v7

CHANGE REQUEST

⌘	**TS 33.220**	**CR**	**CRNum**	⌘ **rev**	**-**	⌘	Current version:	**V 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⌘ ☒ ME ☒ Radio Access Network ☐ Core Network ☐

Title:	⌘	Clarification of Ks_ext		
Source:	⌘	Huawei		
Work item code:	⌘	SSC-GBA	***Date:*** ⌘	23-06-2004
Category:	⌘	**F**	***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:	⌘	The usage and storage of Ks_ext are not clear now
Summary of change:	⌘	The UICC shall be able to determine to remain Ks_ext in UICC or send it to ME. GBA_U-aware ME can request Ks_ext_NAF or Ks_int_NAF from UICC explicitly. Delete a Editors' Note that the corresponding issue is closed.
Consequences if not approved:	⌘	The Ks_ext cann't work well with unclear description.

Clauses affected:	⌘	5.2.1 , 5.3.2, 5.3.3,				
		Y	**N**			
Other specs Affected:	⌘	**X**		Other core specifications	⌘	31.102 , 31.111
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘					

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

***********************************Begin of change *****************************

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.

The UICC shall be able to determine to remain Ks_ext in UICC or send it to ME.

~~Editors' 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 clause 4.5.1 of this document applies also here.

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.

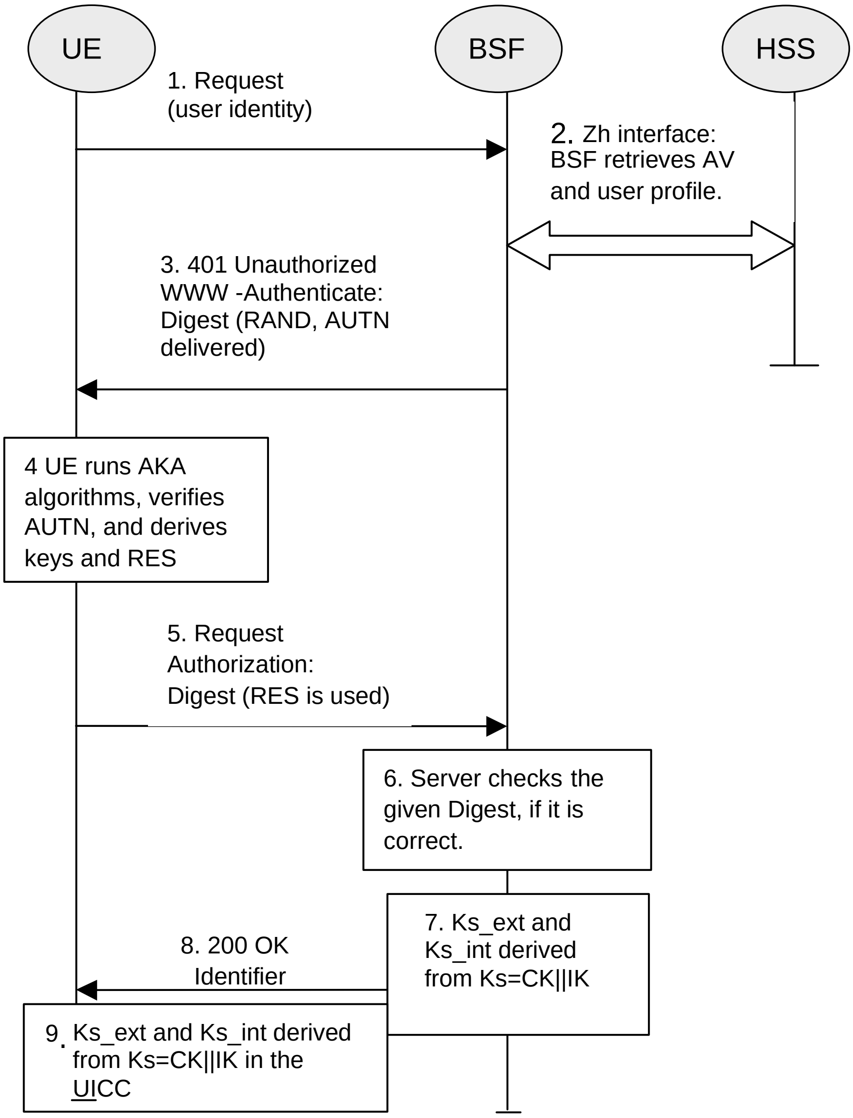
UE
BSF
HSS
1. Request (user identity)
2. Zh interface: BSF retrieves AV and user profile.
3. 401 Unauthorized WWW -Authenticate: Digest (RAND, AUTN delivered)
4 UE runs AKA algorithms, verifies AUTN, and derives keys and RES
5. Request Authorization: Digest (RES is used)
6. Server checks the given Digest, if it is correct.
7. Ks_ext and Ks_int derived from Ks=CK||IK
8. 200 OK Identifier
9. Ks_ext and Ks_int derived from Ks=CK||IK in the UICC

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: 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 key derivation parameters) = Ks_ext || Ks_int (see also figure 5.2). The UICC then transfers RES (after flipping the least significant bit) and optional Ks_ext to the ME and stores Ks_int/ks_ext on the UICC. UICC determine send Ks_ext to ME or remain it based on the capability of ME (GBA_ME-aware or GBA_U-aware ME), if it is the GBA_ME-aware ME only, UICC will send Ks_ext to ME, otherwise UICC will remain Ks_ext.

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.

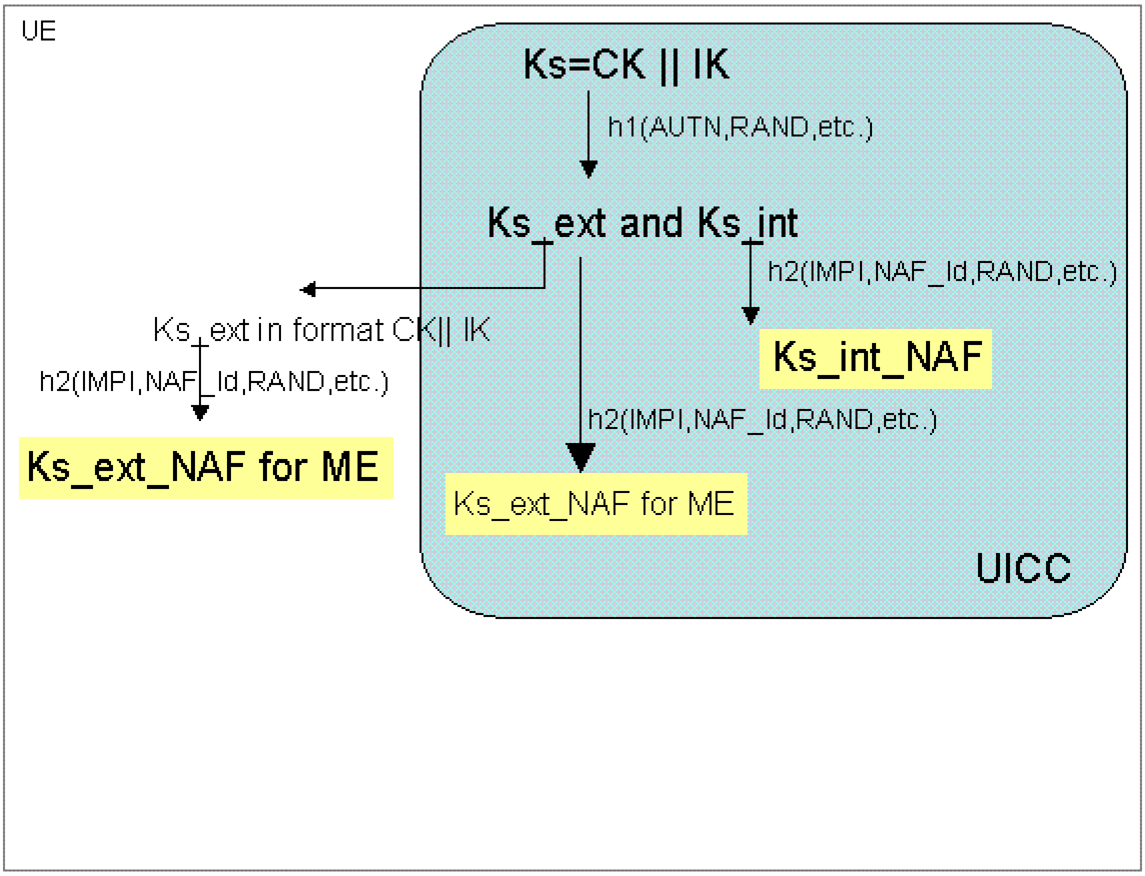

Figure 5.2: Key derivation for GBA-aware UICC when GBA-run was triggered

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 clause 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 1: Such an agreement could e.g. be reached by manual configuration, or by an application-specific protocol step.

Editors' 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 Ua 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; This used to be compatible with GBA_U unaware ME.

- if Ks_ext_NAF is required and a key Ks_ext is available in the UICC, the ME requests the UICC explicitly to derive the key Ks_ext_NAF from Ks_ext, as specified in clause 5.3.2; This used to the GBA_U aware ME.

- if Ks_int_NAF is required and a key Ks_int is available in the UICC, the ME requests the UICC explicitly to derive the key Ks_int_NAF from Ks_int, as specified in clause 5.3.2;

NOTE 2: 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 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 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 clause 5.3.2, in order to obtain new keys.

NOTE 3: 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 4: 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 5: To allow for consistent key derivation in BSF and UE, both have to use the same FQDN for derivation (cf. NOTE 2 of clause 4.5.2). For each protocol used over Ua it shall be specified if only cases (1) and (2) of NOTE 2 of clause 4.5.2 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 6: 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/Ks_ext and Ks_int_NAF from storage in the UICC;

NOTE 7: 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 8: 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 (see note above on key derivation in this clause).

- 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.

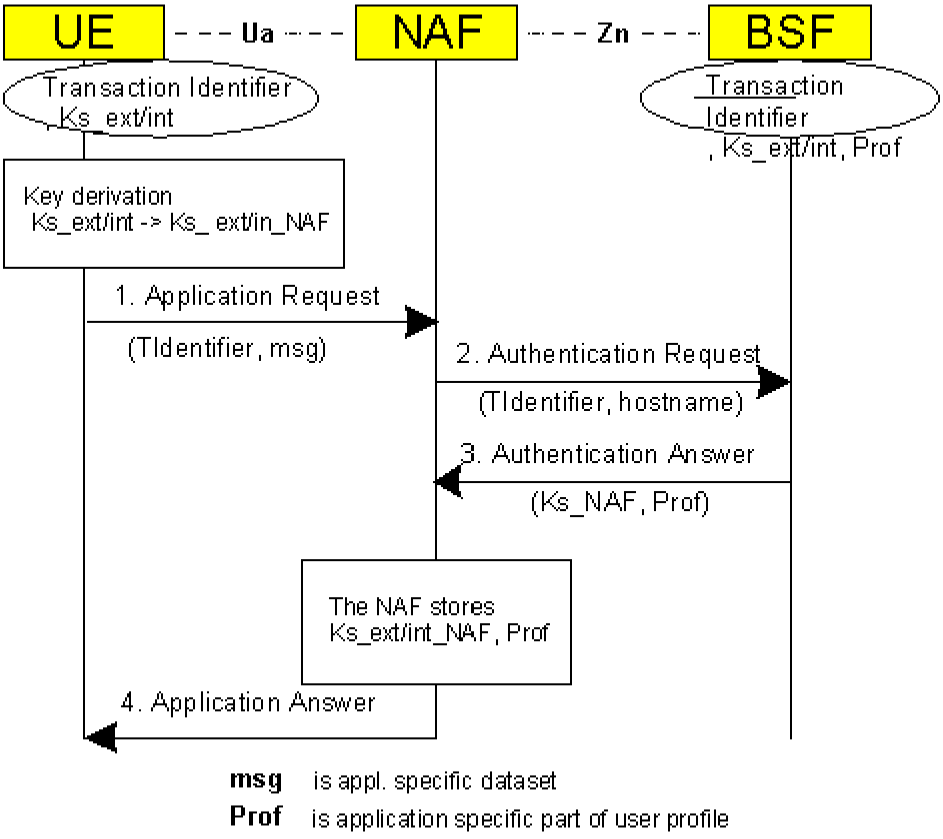

Figure 5.3: The bootstrapping usage procedure with UICC-based enhancements

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