3GPP TSG SA WG3 Security — S3#37 **S3-050067**

21 - 25 February 2005

Sophia Antipolis, France

Source:	Nokia, Siemens, Vodafone
Title:	Bootstrapping timestamp
Agenda item:	6.9.2 (GBA)
Document for:	Discussion/Decision

1 Introduction

During SA3#36, Nokia proposed in S3-040982 that the BSF would be able to indicate the exact bootstrapping timestamp to the NAF over Zn reference point. In the meeting, it was noted that the bootstrapping time could be determined by the NAF by discovering the operator specific key lifetime which would be an operator specified predefined parameter out of a business agreement in the NAF, and the expiration time that is delivered from the BSF to the NAF over the Zn reference point. This contribution discusses why this approach causes unnecessary complexity in the NAF, and argues why the bootstrapping timestamp is needed in the Zn reference point.

2 Discussion

Currently, the NAF can only implictly discover the actual bootstrapping time. The NAF may need the actual bootstrapping time to discover the freshness of the original GBA session key Ks. Upon discovering the bootstrapping time the NAF can determine whether the original bootstrapping procedure is too old according to its policies and whether it requires the UE re-run the bootstrapping procedure. Also, the Identity Provider (IdP) of Liberty may require the actual time of initial authentication of the end user. In this case, the initial authentication means the bootstrapping time - not the time when the bootstrapped shared secret was used with the IdP (over Ua reference point).

The current bootstrapping time procedure is not good due to the fact that the NAF can be in a visited network. The mere possibility that operators may have different key lifetimes from bootstrapped key Ks warrants the NAF to keep a table that contains all the operator specific key lifetimes in order for the NAF to discover the actual bootstrapping time. This causes unnecessary complexity in the NAF as each NAF must keep up-to-date information about the key lifetimes of all the possible operators. For example, each time an operator changes the default key lifetime this information has to be updated in the NAF or if the NAF makes an agreement with a new operator this new key lifetime entry needs to be added and maintained. Also, if the operator has subscriber specific lifetimes for some users the exact bootstrapping time cannot be determined in the NAF without even more complex table structure (i.e., subscriber specific key lifetimes in the NAF). The management of such information in the NAFs is not necessary if the NAF receives the actual bootstrapping time from the BSF.

An operator might also with to use a flexible key lifetime to enable load balancing in the BSF. The BSF could, for example, set the key lifetime so that the amount of keys that expire is distributed uniformly in time.

Also, the operator may have subscriber specific bootstrapping lifetimes (e.g., for prepaid subscribers). This can be set in subscriber's GBA User Security Settings (cf. clause 4.2.3 of TS 33.220). In this case the implicit discovery of the actual bootstrapping time is not possible as the bootstrapping lifetime time may vary per subscriber.

3 Conclusion & Proposal

To simplify the NAF procedures and management, and avoid unnecessary complexity in the NAF, we propose that the bootstrapping timestamp shall be transferred over the Zn reference point from the BSF to the NAF.

3GPP TSG-SA WG3 Meeting #37
Sophia Antipolis, France, February 21-25, 2005

Tdoc ⌘Att_S3-050067

CR-Form-v7.1

CHANGE REQUEST

⌘	**33.220**	**CR**	**047**	⌘ **rev**	**-**	⌘	Current version:	**6.3.0** ⌘

*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:	⌘	Bootstrapping timestamp		
Source:	⌘	Nokia, Siemens, Vodafone		
Work item code:	⌘	SEC1-SC	***Date:*** ⌘	14/02/2005
Category:	⌘	**C**	***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:

Ph2	*(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)*
Rel-7	*(Release 7)*

Reason for change:	⌘	Currently, the NAF can only implictly discover the actual bootstrapping time. The NAF may need the actual bootstrapping time to discover the freshness of the original GBA session key Ks. Upon discovering the bootstrapping time the NAF can determine whether the original bootstrapping procedure is too old according to its policies and whether it requires the UE re-run the bootstrapping procedure. Also, the operator may have subscriber specific bootstrapping lifetimes (e.g., for prepaid subscribers). This can be set in subscriber's GBA User Security Settings (cf. clause 4.2.3 of TS 33.220). In this case the implicit discovery of the actual bootstrapping time is not possible as the bootstrapping lifetime time may vary per subscriber.
Summary of change:	⌘	The bootstrapping time is sent from the BSF to the NAF (in addition to the bootstrapping lifetime time).
Consequences if not approved:	⌘	The NAF cannot reliably determine the actual bootstrapping time.

Clauses affected:	⌘	4.4.6, 4.5.3, 5.3.2				
		Y	N			
Other specs affected:	⌘	X		Other core specifications	⌘	TS 29.109
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘					

===== BEGIN CHANGE =====

4.4.6 Requirements on reference point Zn

The requirements for reference point Zn are:

- mutual authentication, confidentiality and integrity shall be provided;
- If the BSF and the NAF are located within the same operator's network, the Zn reference point shall be secured according to NDS/IP [13];
- If the BSF and the NAF are located in different operators' networks, the Zn' reference point between the D-Proxy and the BSF shall be secured using TLS as specified in RFC 2246 [6];

NOTE 1: Annex E specifies the TLS profile that is used for securing the Zn' reference point.

- The BSF shall verify that the requesting NAF is authorised;
- The NAF shall be able to send a key material request to the BSF, containing NAF's public hostname used by the UE's corresponding request. The BSF shall be able to verify that a NAF is authorized to use this hostname, i.e. the FQDN used by UE when it contacts the NAF;
- The BSF shall be able to send the requested key material to the NAF;
- The NAF shall be able to get a selected set of application-specific USSs from the BSF, depending on the policy of the BSF and the application indicated in the request from the NAF over Zn;
- The NAF shall be able to indicate to the BSF the single application or several applications it requires USSs for;

NOTE 2: If some application needs only a subset of an application-specific USS, e.g. only one IMPU, the NAF selects this subset from the complete set of USS sent from BSF.

- If a NAF requests USSs from the BSF and they are not present in subscriber's GUSS, it shall not cause an error, provided the conditions of the local policy of the BSF are fulfilled. The BSF shall then send only the requested and found USSs to the NAF;
- The BSF shall be able to be configured on a per NAF or per application basis if private subscriber identity and which application-specific USSs may be sent to a NAF;
- The BSF shall be able to be configured locally by the MNO in such a way that the BSF is able to decide on a per NAF basis if one or more application-specific USSs shall be present in subscriber's GUSS, and to reject the request from the NAF in case the conditions are not fulfilled;
- The BSF shall be able to indicate to the NAF the bootstrapping time and the lifetime of the key material. The key lifetime sent by the BSF over Zn shall indicate the expiry time of the key, and shall be identical to the key lifetime sent by the BSF to the UE over Ub.

NOTE 3: This does not preclude a NAF to refresh the key before the expiry time according to the NAF's local policy.

NOTE 4: If one or more of the USSs that have been delivered to the NAF has been updated in subscriber's GUSS in the HSS, this change is propagated to the NAF the next time it fetches the USS from the BSF over Zn reference point (provided that the BSF has updated subscriber's GUSS from the HSS over Zh reference point).

===== BEGIN NEXT CHANGE =====

4.5.3 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 4.5.1.

Once the UE and the NAF have established that they want to use GBA then every time the UE wants to interact with an NAF the following steps are executed as depicted in figure 4.4.

1. UE starts communication over reference point Ua with the NAF:

    - in general, UE and NAF will not yet share the key(s) required to protect the reference point Ua. If they already do (i.e. if a key Ks_NAF for the corresponding key derivation parameter NAF_Id is already available),, the UE and the NAF can start to securely communicate right away. If the UE and the NAF do not yet share a key, the UE proceeds as follows:

        - if a key Ks for the selected UICC application is available in the UE, the UE derives the key Ks_NAF from Ks, as specified in clause 4.5.2;

        - if no key Ks for the selected UICC application is available in the UE, the UE first agrees on a new key Ks with the BSF over the reference point Ub, and then proceeds to derive Ks_NAF;

    NOTE 1: If it is not desired by the UE to use the same Ks for the selected UICC application to derive more than one Ks_NAF then the UE should agree on a new key Ks with the BSF over the reference point Ub, and then proceed to derive Ks_NAF;

    - if the NAF shares a key with the UE, but the NAF requires an update of that key, e.g. because the key’s lifetime has expired or will expire soon, or the key can not meet the NAF local validity condition, it shall send a suitable bootstrapping renegotiation request to the UE, see figure 4.5. If the key's lifetime has expired the protocol used over reference point Ua shall be terminated. The form of this indication depends on the particular protocol used over reference point Ua. If the UE receives a bootstrapping renegotiation request, it starts a run of the protocol over reference point Ub, as specified in clause 4.5.2, in order to obtain a new key Ks.

    NOTE 2: To allow for consistent key derivation in BSF and UE, both have to use the same FQDN for derivation (see 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 3: If the shared key between UE and NAF is invalid , the NAF can set deletion conditions to the corresponding security association for subsequent removal.

    - the UE supplies the B-TID to the NAF, in the form as specified in clause 4.3.2, to allow the NAF to retrieve the corresponding keys from the BSF;

    NOTE 4: The UE may adapt the key material Ks_NAF to the specific needs of the reference point Ua. This adaptation is outside the scope of this specification.

    - key management for GBA related keys in the ME (i.e. Ks and Ks_NAF keys):

        - all GBA related keys shall be deleted from the ME when a different UICC is inserted. Therefore the ME needs to store in non-volatile memory the last inserted UICC-identity to be able to compare that with the used UICC-identity at UICC insertion and power on;

        - the Key Ks shall be deleted from the ME when the ME is powered down;

        - all other GBA related keys may be deleted from the ME when the ME is powered down. If the ME does not delete the GBA keys at power down then the GBA keys need to be stored in non-volatile memory.

    - when a new Ks is agreed over the reference point Ub and a key Ks_NAF, derived from one NAF_Id, is updated, the other keys Ks_NAF, derived from different values NAF_Id, stored on the UE shall not be affected;

    NOTE 5: According to the procedures defined in clauses 4.5.2 and 4.5.3, in the UE there is at most one Ks_NAF key stored per NAF-Id.

2. NAF starts communication over reference point Zn with BSF

    - The NAF requests key material corresponding to the B-TID supplied by the UE to the NAF over reference point Ua. 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 2 on key derivation in this clause);

    - The NAF may also request one or more application-specific USSs for the applications, which the request received over Ua from UE may access;

- With the key material request, the NAF shall supply NAF's public hostname that UE has used to access NAF to BSF, and BSF shall be able verify that NAF is authorized to use that hostname;

3. The BSF derives the keys required to protect the protocol used over reference point Ua from the key Ks and the key derivation parameters, as specified in clause 4.5.2, and supplies to NAF the requested key Ks_NAF, as well as the bootstrapping time and the lifetime of that key, and the requested application-specific and potentially NAF group specific USSs if they are available in subscriber's GUSS and if the NAF is authorized to receive the requested USSs. If the key identified by the B-TID 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 to the UE.

NOTE 6: The NAF can further set the local validity condition of the Ks_NAF according to the local policy,for example a limitation of reuse times of a Ks_NAF.

NOTE 7: The NAF shall adapt the key material Ks_NAF to the specific needs of the reference point Ua in the same way as the UE did. This adaptation is outside the scope of this specification.

- The BSF may require that one or more application-specific and potentially NAF group specific USSs shall be present in subscriber's GUSS for the NAF (see clause 4.4.6). If one or more of these required settings are missing from the GUSS, the BSF shall indicate this in the reply to the NAF.

- The BSF may also send the private user identity (IMPI) and requested USSs to NAF according to the BSF's policy;

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

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

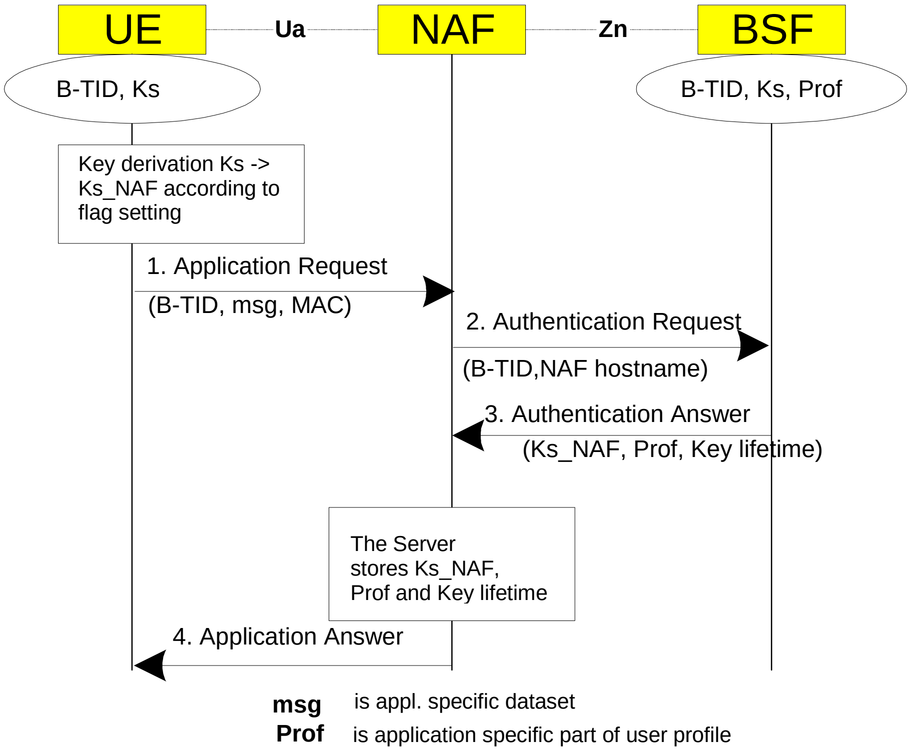
UE
Ua
NAF
Zn
BSF
B-TID, Ks
B-TID, Ks, Prof
Key derivation Ks -> Ks_NAF according to flag setting
1. Application Request
(B-TID, msg, MAC)
2. Authentication Request
(B-TID,NAF hostname)
3. Authentication Answer
(Ks_NAF, Prof, Key lifetime)
The Server stores Ks_NAF, Prof and Key lifetime
4. Application Answer
msg is appl. specific dataset
Prof is application specific part of user profile

Figure 4.4: The bootstrapping usage procedure

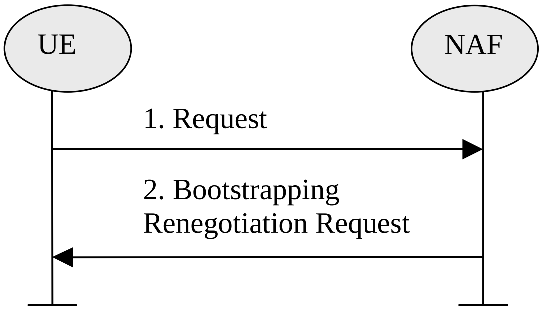

Figure 4.5: Bootstrapping renegotiation request

===== BEGIN NEXT CHANGE =====

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 Ks_ext_NAF and Ks_int_NAF 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. Any such agreement overrules the default use of the keys. How this agreement is reached is application-specific and is not within the scope of this document.

NOTE 1: This agreement may be mandated by the specification, which defines the Ua reference point between UE and NAF, e.g. TS 33.246 for the use of GBA in MBMS, or negotiated by the NAF and the UE over the Ua reference point, or reached by configuration.

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 for the selected UICC application is available in the UICC, the ME requests the UICC to derive the key Ks_ext_NAF from Ks, as specified in clause 5.3.2;

- if Ks_int_NAF is required and a key Ks for the selected UICC application is available in the UICC, the ME requests the UICC to derive the key Ks_int_NAF from Ks, as specified in clause 5.3.2;

NOTE 2: If it is not desired by the UE to use the same Ks for the selected UICC application to derive more than one Ks_ext/int_NAF then the UE should first agree on new key Ks 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 for the selected UICC application is not available in the UE, the UE first agrees on a new key Ks 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. If the key's lifetime has expired the protocol used over reference point Ua shall be terminated. 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.

1. 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 B-TID 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.

   - key management for GBA related keys in the ME (i.e. Ks_ext_NAF keys):

     - all GBA related keys shall be deleted from the ME when a different UICC is inserted. Therefore the ME needs to store in non-volatile memory the last inserted UICC-identity to be able to compare that with the used UICC-identity at UICC insertion and power on.

     - all GBA related keys may be deleted from the ME when the ME is powered down. If the ME does not delete the GBA keys at power down then the GBA keys need to be stored in non-volatile memory.

     - all GBA related keys in the UICC do not need to be deleted when the ME is powered down.

   NOTE 7: After each run of the protocol over the Ub reference point, a new key Ks, associated with a new B-TID, are derived in the UE according to clause 5.3.2, so that it can never happen, that key Ks with different B-TIDs simultaneously exist in the UE.

   - When new key Ks is 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: According to the procedures defined in clauses 5.3.2 and 5.3.3, in the UE there is at most one Ks_int_NAF/Ks_ext_NAF key pair stored per NAF_Id.

   NOTE 9: This rule ensures that the keys Ks_ext_NAF and Ks_int_NAF are always in synch at the UE and the NAF.

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

   - The NAF requests from the BSF the keys corresponding to the B-TID, 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).

   - The NAF may also request one or more application-specific USSs for the applications, which the request received over Ua from UE may access;

   - 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.

   3. 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 bootstrapping time and the lifetime time of these keys, and the requested application-specific and potentially NAF group specific USSs if they are available in subscriber's GUSS and if the NAF is authorized to receive the requsted USSs. If the key identified by the B-TID 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 10: The NAF can further set the local validity condition of the Ks_NAF according to the local policy, for example a limitation of reuse times of a Ks_NAF.

NOTE 11: 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 BSF may require that one or more application-specific and potentially NAF group specific USSs shall be present in subscriber's GUSS for the NAF (see clause 4.4.6). If one or more of these required settings are missing from the GUSS, the BSF shall indicate this in the reply to the NAF.

- The BSF may also send the private user identity (IMPI) and requested USSs to NAF according to the BSF's policy.

4. 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.

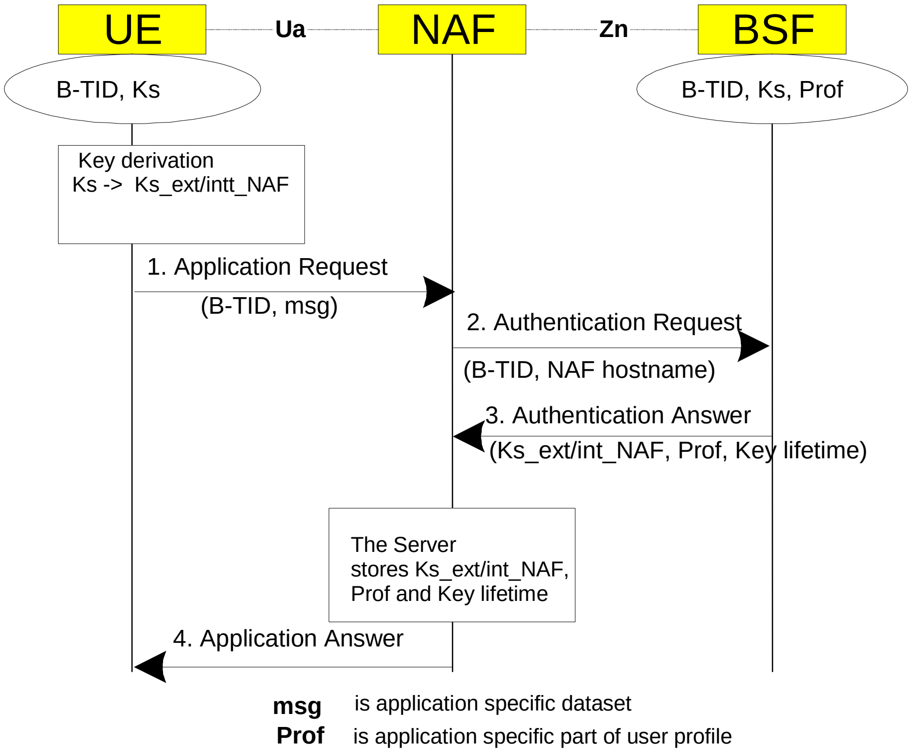

Figure 5.3: The bootstrapping usage procedure with UICC-based enhancements

===== END CHANGE =====