

CR-Form-v7

## CHANGE REQUEST

• **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 •

ME

Radio Access Network

Core Network

Title:	• ISIM support		
Source:	• Nokia, Gemplus, Alcatel, <a href="#">Motorola</a>		
Work item code:	• GBA and SSC	Date:	• <a href="#">0629/076/2004</a>
Category:	• <b>C</b>	Release:	• Rel-6
Use <u>one</u> of the following categories:		Use <u>one</u> of the following releases:	
F (correction)		2 (GSM Phase 2)	
A (corresponds to a correction in an earlier release)		R96 (Release 1996)	
B (addition of feature),		R97 (Release 1997)	
C (functional modification of feature)		R98 (Release 1998)	
D (editorial modification)		R99 (Release 1999)	
Detailed explanations of the above categories can be found in 3GPP <a href="#">TR 21.900</a> .		Rel-4 (Release 4)	
		Rel-5 (Release 5)	
		Rel-6 (Release 6)	

Reason for change:	• For a service that utilizes the GBA, e.g., Presence, it should be possible to access the server with an ISIM, since the presence account based on an ISIM may be different than that in the USIM which are received from a BSF, e.g., an IMPI value, and IMPUs added to enable new services.
Summary of change:	• ISIM support in GAA is added, <del>and</del> Default selection logic is added to make the selection on the UE whether to use <a href="#">an</a> ISIM or <a href="#">a</a> USIM in GBA.
Consequences if not approved:	• Service may have conflicts when handling the UE's identities.

Clauses affected:	• 2, 4, 4.2.3, 4.2.4, 4.3.1, 4.5.2						
Other specs affected:	<table border="1" style="display: inline-table; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Y</td> <td style="padding: 2px;">N</td> </tr> <tr> <td style="text-align: center; padding: 2px;"><input checked="" type="checkbox"/></td> <td style="text-align: center; padding: 2px;"><input checked="" type="checkbox"/></td> </tr> </table>	Y	N	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Other core specifications	•
	Y	N					
	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>					
<input checked="" type="checkbox"/>	Test specifications	•					
<input checked="" type="checkbox"/>	O&M Specifications	•					
Other comments:	•						

---

## 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

- [1] 3GPP TS 31.102: "3rd Generation Partnership Project; Technical Specification Group Terminals; Characteristics of the USIM application".
- [2] 3GPP TS 33.102: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security architecture".
- [3] Franks J., et al.: "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.
- [4] A. Niemi, et al.: "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)", RFC 3310, September 2002.
- [5] 3GPP TS 33.221: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Generic Authentication Architecture (GAA); Support for Subscriber Certificates".
- [6] T. Dierks, et al.: "The TLS Protocol Version 1.0", RFC 2246, January 1999.
- [7] OMA: "Provisioning Content Version 1.1", Version 13-Aug-2003. Open Mobile Alliance.
- [8] 3GPP TS 23.228: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS); Stage 2 (Release 6)".
- [9] IETF RFC 3546 (2003): "Transport Layer Security (TLS) Extensions".
- [10] [3GPP TS 31.103: "3rd Generation Partnership Project; Technical Specification Group Terminals; Characteristics of the IP Multimedia Services Identity Module \(ISIM\) application"](#).
- [11] [3GPP TS 23.003: "3rd Generation Partnership Project; Technical Specification Group Core Network; Numbering, addressing and identification"](#).

---

## 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.

**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, see clause 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, see clause 5 of this specification.

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

*Editor's note: Definition to be completed.*

**Bootstrapping Transaction Identifier:**

*Editor's note: Definition to be completed.*

## 3.2 Abbreviations

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

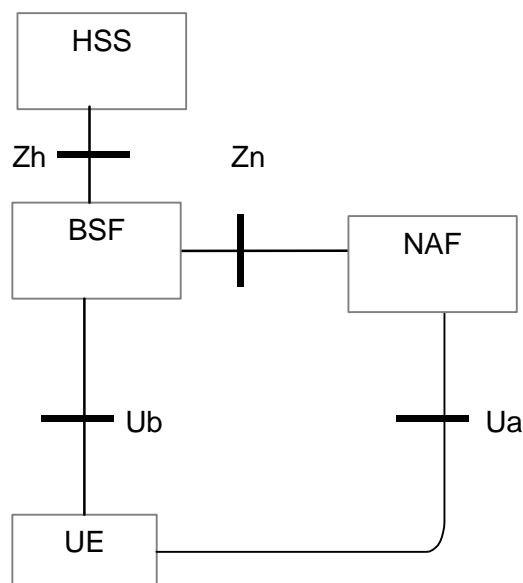
AK	Anonymity Key
AKA	Authentication and Key Agreement
B-TID	Bootstrapping Transaction Identifier
BSF	Bootstrapping Server Function
CA	Certificate Authority
FQDN	Fully Qualified Domain Name
GAA	Generic Authentication Architecture
GBA	Generic Bootstrapping Architecture
GBA_ME	ME-based GBA
GBA_U	GBA with UICC-based enhancements
HSS	Home Subscriber System
IK	Integrity Key
KDF	Key Derivation Function
Ks_int	Derived key in GBA_U which remains on UICC
Ks_ext	Derived key in GBA_U
MNO	Mobile Network Operator
NAF	Network Application Function
PKI	Public Key Infrastructure

## 4 Generic Bootstrapping Architecture

The 3GPP authentication infrastructure, including the 3GPP Authentication Centre (AuC), the USIM/[ISIM](#), and the 3GPP AKA protocol run between them, is a very valuable asset of 3GPP operators. It has been recognised that this infrastructure could be leveraged to enable application functions in the network and on the user side to establish shared keys. Therefore, 3GPP can provide the "bootstrapping of application security" to authenticate the subscriber by defining a Generic Bootstrapping Architecture (GBA) based on AKA protocol.

### 4.1 Reference model

Figure 4.1 shows a simple network model of the entities involved in the bootstrapping approach, and the reference points used between them.



**Figure 4.1: Simple network model for bootstrapping**

Figure 4.1a shows a simple network model of the entities involved when the network application function is located in the visited network.

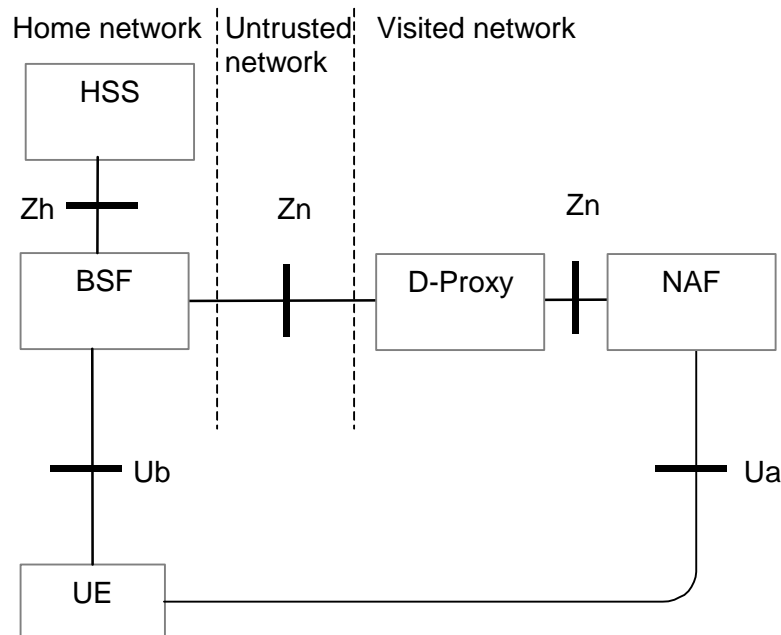


Figure 4.1a: Simple network model for bootstrapping in visited network

## 4.2 Network elements

### 4.2.1 Bootstrapping server function (BSF)

A generic Bootstrapping Server Function (BSF) and the UE shall mutually authenticate using the AKA protocol, and agree on session keys that are afterwards applied between UE and an operator-controlled Network Application Function (NAF). The BSF shall restrict the applicability of the key material to a specific NAF by using a suitable key derivation procedure. The key derivation procedure may be used with multiple NAFs during the lifetime of the key material. The lifetime of the key material is set according to the local policy of the BSF. The generation of key material is specified in section 4.5.2.

### 4.2.2 Network application function (NAF)

After the bootstrapping has been completed, the UE and an operator-controlled NAF can run some application specific protocol where the authentication of messages will be based on those session keys generated during the mutual authentication between UE and BSF.

General assumptions for the functionality of an operator-controlled NAF are:

- there is no previous security association between the UE and the NAF;
- NAF shall be able to locate and communicate securely with the subscriber's BSF;
- NAF shall be able to acquire a shared key material established between UE and the BSF during the run of the application-specific protocol;
- NAF shall be able to check lifetime of the shared key material.

#### 4.2.2a Diameter proxy (D-Proxy)

In the case where UE has contacted a NAF that is operated in another network than home network, this visited NAF shall use a diameter proxy (D-Proxy) of the NAFs network to communicate with subscriber's BSF (i.e. home BSF).

NOTE: D-Proxy functionality may be implemented as a separate network element, or be part of any NE in the visited network that implements Diameter proxy functionality (examples of such NE's are the BSF of the network that the visited NAF belongs to, or an AAA-server).

General requirements for the functionality of D-Proxy are:

- D-Proxy shall be able to function as a proxy between the visited NAF, and the subscriber's home BSF;
- D-Proxy shall be able to locate subscriber's home BSF and communicate with it over secure channel;
- D-Proxy shall be able to validate that the visited NAF is authorized to participate in GBA and shall be able to assert to subscriber's home BSF the visited NAFs DNS name. The D-Proxy shall also be able to assert to the BSF that the visited NAF is authorized to request the GBA specific user profiles contained in the NAF request;
- the physical security level of the D-proxy shall not be lower than the highest level of the NAFs which it interfaces with.

### 4.2.3 HSS

HSS shall store new parameters in the subscriber profile related to the use of the bootstrapping function. Possibly also parameters related to the usage of some NAFs are stored in the HSS. [In the case where the subscriber has multiple identities \(e.g., both a USIM and an ISIM\), a subscriber profile related to the bootstrapping function shall be referable using all these identities.](#)

*Editor's note: Needed new subscriber profile parameters are FFS.*

### 4.2.4 UE

The required functionalities from the UE are:

- the support of HTTP Digest AKA protocol;
- [the capability to use both a USIM and an ISIM in bootstrapping;](#)
- [the capability to select either a USIM or an ISIM to be used in bootstrapping, when both of them are present;](#)
- the capability to derive new key material to be used with the protocol over Ua interface from CK and IK;
- support of NAF-specific application protocol (For an example see TS 33.221 [5]).

## 4.3 Bootstrapping architecture and reference points

### 4.3.1 Reference point Ub

The reference point Ub is between the UE and the BSF. Reference point Ub provides mutual authentication between the UE and the BSF. It allows the UE to bootstrap the session keys based on 3GPP AKA infrastructure.

The HTTP Digest AKA protocol, which is specified in RFC 3310 [4], is used on the reference point Ub. It is based on the 3GPP AKA TS 33.102 [2] protocol. The interface to the USIM is as specified in TS 31.102 [1] [and to the ISIM is as specified in TS 31.103 \[10\]](#).

### 4.3.2 Reference point Ua

The reference point Ua carries the application protocol, which is secured using the keys material agreed between UE and BSF as a result of the run of HTTP Digest AKA over reference point Ub. For instance, in the case of support for subscriber certificates TS 33.221 [5], it is a protocol, which allows the user to request certificates from the NAF. In this case the NAF would be the PKI portal.

### 4.3.3 Reference point Zh

The reference point Zh used between the BSF and the HSS allows the BSF to fetch the required authentication information and subscriber profile information from the HSS. The interface to the 3G Authentication Centre is HSS-internal, and it need not be standardised as part of this architecture.

### 4.3.4 Reference point Zn

The reference point Zn is used by the NAF to fetch the key material agreed during a previous HTTP Digest AKA protocol run over the reference point Ub from the UE to the BSF. It may also be used to fetch subscriber profile information from the BSF.

## 4.4 Requirements and principles for bootstrapping

The following requirements and principles are applicable to bootstrapping procedure:

- the bootstrapping function shall not depend on the particular NAF;
- the server implementing the bootstrapping function needs to be trusted by the home operator to handle authentication vectors;
- the server implementing the NAF needs only to be trusted by the home operator to handle derived key material;
- it shall be possible to support NAF in the operator's home network and in the visited network;
- the architecture shall not preclude the support of network application function in a third network;
- to the extent possible, existing protocols and infrastructure should be reused;
- in order to ensure wide applicability, all involved protocols are preferred to run over IP;
- it shall be prevented that a security breach in one NAF who is using the GBA, can be used by an attacker to mount successful attacks to the other NAFs using the GBA.

### 4.4.1 Access Independence

Bootstrapping procedure is access independent. Bootstrapping procedure requires IP connectivity from UE.

### 4.4.2 Authentication methods

Authentication between the UE and the BSF shall not be possible without a valid cellular subscription. Authentication shall be based on the 3GPP AKA protocol.

### 4.4.3 Roaming

The roaming subscriber shall be able to utilize the bootstrapping function in the home network. The subscriber shall be able to utilize network application function that is in a visited network.

### 4.4.4 Requirements on reference point Ub

The requirements for reference point Ub are:

- the BSF shall be able to identify the UE;
- the BSF and the UE shall be able to authenticate each other based on AKA;
- the BSF shall be able to send a Transaction Identifier to the UE.

### 4.4.5 Requirements on reference point Zh

The requirements for reference point Zh are:

- mutual authentication, confidentiality and integrity shall be provided;

NOTE: This requirement may be fulfilled by physical or proprietary security measures if BSF and HSS are located within the same operator's network.

- the BSF shall be able to send bootstrapping information request concerning a subscriber;
- the HSS shall be able to send 3GPP AKA vectors to the BSF in batches;
- the HSS shall be able to send the subscriber's GAA profile information needed for security purposes to the BSF;

**Editor's note: It's ffs how to proceed in the case where profile is updated in HSS after profile is forwarded. The question is whether this profile change should be propagated to BSF.**

- no state information concerning bootstrapping shall be required in the HSS;
- all procedures over reference point Zh shall be initiated by the BSF;

**Editor's note: This requirement may need to be modified depending on what happens in the case where the profile in the HSS is updated.**

- the number of different interfaces to HSS should be minimized.

#### 4.4.6 Requirements on reference point Zn

The requirements for reference point Zn are:

- mutual authentication, confidentiality and integrity shall be provided;

**NOTE:** This requirement may be fulfilled by physical or proprietary security measures if BSF and NAF are located within the same operator's network.

**Editors' Note: In the visited NAF scenario, it should be decided how the communication between a D-Proxy and a BSF is secured. The possible solutions for securing this link include TLS and IPsec.**

- 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 the subscriber profile information needed for security purposes from BSF;
- The BSF shall be able to indicate to the NAF the lifetime of the key material.

**Editor's note: Relationship between Transaction Identifier and subscriber identity is ffs. In the case of Presence reference point Ut, there are several potential identities that are related to Transaction Identifier, i.e. IMPI and IMPUs. The subscriber may have several Presence accounts related to same IMPI. Transaction Identifier does not carry enough information on which IMPU the end-user is trying to use.**

#### 4.4.7 Requirements on Transaction Identifier

Transaction identifier shall be used to bind the subscriber identity to the keying material in reference points Ua, Ub and Zn.

Requirements for Transaction Identifier are:

- Transaction Identifier shall be globally unique;
- Transaction Identifier shall be usable as a key identifier in protocols used in the reference point Ua;
- NAF shall be able to detect the home network and the BSF of the UE from the Transaction Identifier.

**NOTE 1:** NAF can remove the security association based on deletion conditions after the key has become invalid.



NOTE 2: Care has to be taken that the parallel use of GBA and non-GBA authentication between UE and NAF does not lead to conflicts, e.g. in the name space. This potential conflict cannot be resolved in a generic way as it is dependent on specific protocol and authentication mechanism used between UE and application server. It is therefore out of scope of this specification.

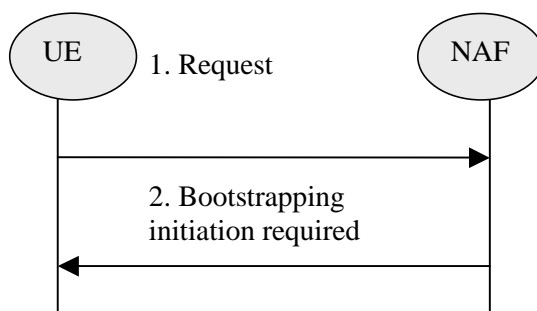
For the example of HTTP Digest authentication used between UE and NAF, parallel use is possible as the following applies: <username,password>-pairs must be unique to one realm only. As the NAF controls the realm names, it has to ensure that only the GBA based realm is named with the reserved 3GPP realm name. In the special case that the NAF wants to allow non GBA based authentication in the GBA realm also, it has to ensure that no usernames in the format of a Transaction Identifier are used outside GBA based authentication.

## 4.5 Procedures

This chapter specifies in detail the format of the bootstrapping procedure that is further utilized by various applications. It contains the AKA authentication procedure with BSF, and the key material generation procedure.

### 4.5.1 Initiation of bootstrapping

Before communication between the UE and the NAF can start, the UE and the NAF first have to agree whether to use the GBA. When a UE wants to interact with a NAF, but it does not know if the NAF requires the use of shared keys obtained by means of the GBA, the UE shall contact the NAF for further instructions (see figure 4.2).



**Figure 4.2: Initiation of bootstrapping**

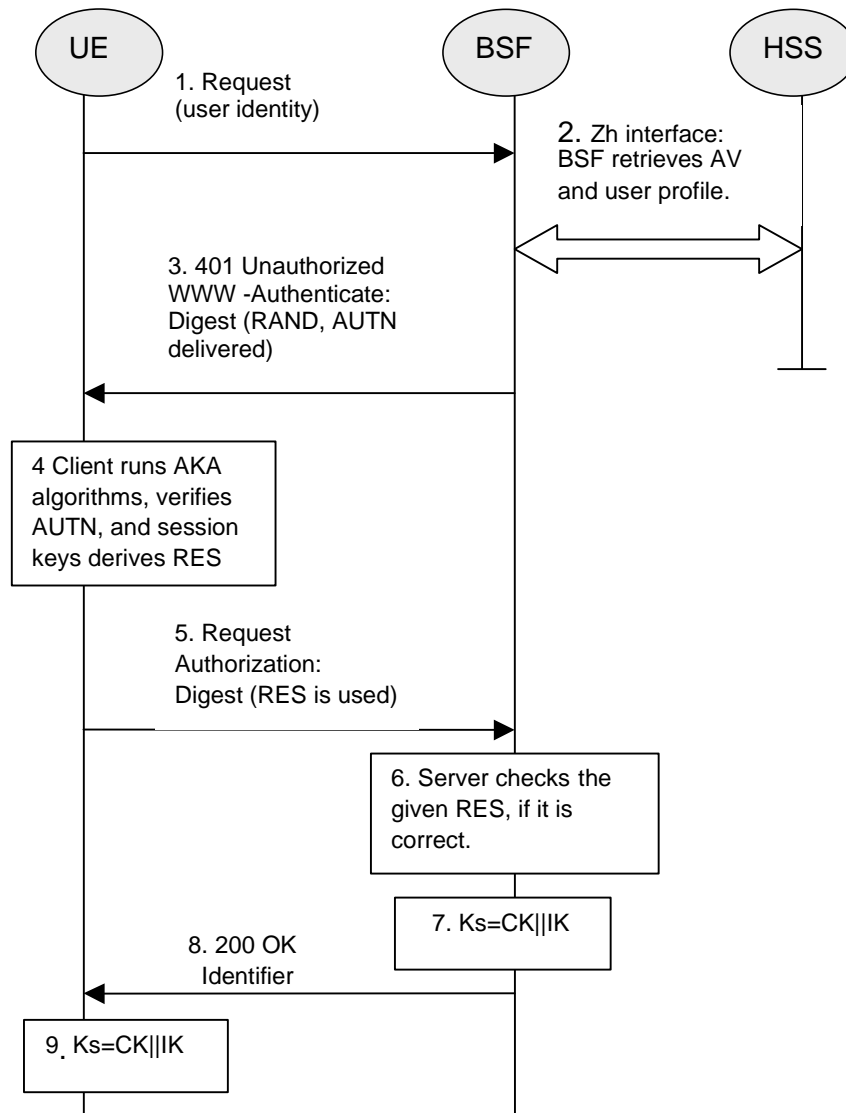
1. UE starts communication over reference point Ua with the NAF without any GBA-related parameters.
2. If the NAF requires the use of shared keys obtained by means of the GBA, but the request from UE does not include GBA-related parameters, the NAF replies with a bootstrapping initiation message. The form of this indication may depend on the particular reference point Ua and is specified in the relevant stage 3-specifications.

### 4.5.2 Bootstrapping 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 4.3). Otherwise, the UE shall perform a bootstrapping authentication only when it has received bootstrapping initiation required message or a bootstrapping negotiation indication from the NAF, or when the lifetime of the key in UE has expired (cf. subclause 4.5.3).

The User's IMPI is used as the user identity in the bootstrapping procedure. By default, if an ISIM is present in the UE, the UE shall use the ISIM for the bootstrapping procedure. ~~and~~ The IMPI is obtained from the ISIM as specified in TS 33.103 [10]. If an ISIM is not present, the UE shall use the USIM; and derive the IMPI from the IMSI as specified in TS 23.003 [11].

NOTE 1: 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 3 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 4.3: The bootstrapping procedure**

1. The UE sends an HTTP request [with the subscriber's IMPI as the user identity](#) towards the BSF.
2. BSF retrieves the user profile and one or a whole batch of Authentication Vectors (AV, AV = RAND||AUTN||XRES||CK||IK) over the reference point Zh from the HSS.
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 UE checks AUTN to verify that the challenge is from an authorised network; the UE also calculates CK, IK and RES. This will result in session keys IK and CK in both BSF and UE.
5. The UE sends another HTTP request, containing the Digest AKA response (calculated using RES), to the BSF.
6. The BSF authenticates the UE by verifying the Digest AKA response.
7. The BSF generates key material  $K_s$  by concatenating CK and IK. 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.
8. The BSF shall send a 200 OK message, including a 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 key  $K_s$ . The key material  $K_s$  is generated in UE by concatenating CK and IK.

9. Both the UE and the BSF shall use the Ks to derive the key material Ks\_NAF during the procedures as specified in clause 4.5.3. Ks\_NAF shall be used for securing the reference point Ua.

Ks\_NAF is computed as  $Ks\_NAF = KDF(Ks, \text{key derivation parameters})$ , where KDF is a suitable key derivation function, and the key derivation parameters consist of the user's IMPI, the NAF\_Id and RAND. The NAF\_Id consists of the full DNS name of the NAF. KDF shall be implemented in the ME.

NOTE 2: To allow consistent key derivation based on NAF name in UE and BSF, at least one of the three following prerequisites shall be fulfilled:

- (1) The NAF is known in DNS under one domain name (FQDN) only, i.e. no two different domain names point to the IP address of the NAF. This has to be achieved by administrative means. This prerequisite is not specific to 3GPP, as it is necessary also under other circumstances, e.g. for TLS V1.0 without use of wildcard or multiple-name certificates.
- (2) Each DNS entry of the NAF points to a different IP address. The NAF responds to all these IP addresses. Each IP address is tied to the corresponding FQDN by NAF configuration. The NAF can see from the IP address, which FQDN to use for key derivation.
- (3) Ua uses a protocol which transfers the host name (FQDN of NAF as used by UE) to NAF (e.g. HTTP/1.1 with mandatory Host request header field). This requires the NAF to check the validity of the host name, to use this name in all communication with UE where appropriate, and to transfer this name to BSF to allow for correct derivation of Ks\_NAF. In case of a TLS tunnel this requires either multiple-identities certificates or the deployment of RFC 3546 [9] or other protocol means with similar purpose.

**Editor's note: The definition of the KDF is left to ETSI SAGE and is to be included in the Annex B of the present specification.**

The UE and the BSF shall store the key Ks with the associated Transaction Identifier for further use, until the lifetime of Ks has expired, or until the key Ks is updated.

### 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.

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 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 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 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, it shall send a suitable bootstrapping renegotiation request to the UE and terminates the protocol used over reference point Ua, see figure 4.5. 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 section 4.5.2). For each protocol used over Ua it shall be specified if only cases (1) and (2) of NOTE 2 of section 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 Transaction Identifier 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.

- when the UE is powered down, or when the UICC is removed, any keys Ks and Ks\_NAF shall be deleted from storage;
- 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;

NAF starts communication over reference point Zn with BSF

- The NAF requests key material corresponding to the Transaction Identifier 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);
- 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 to verify that NAF is authorized to use that hostname;
- 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 lifetime of that key. 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 to the UE.

NOTE 5: 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.

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.

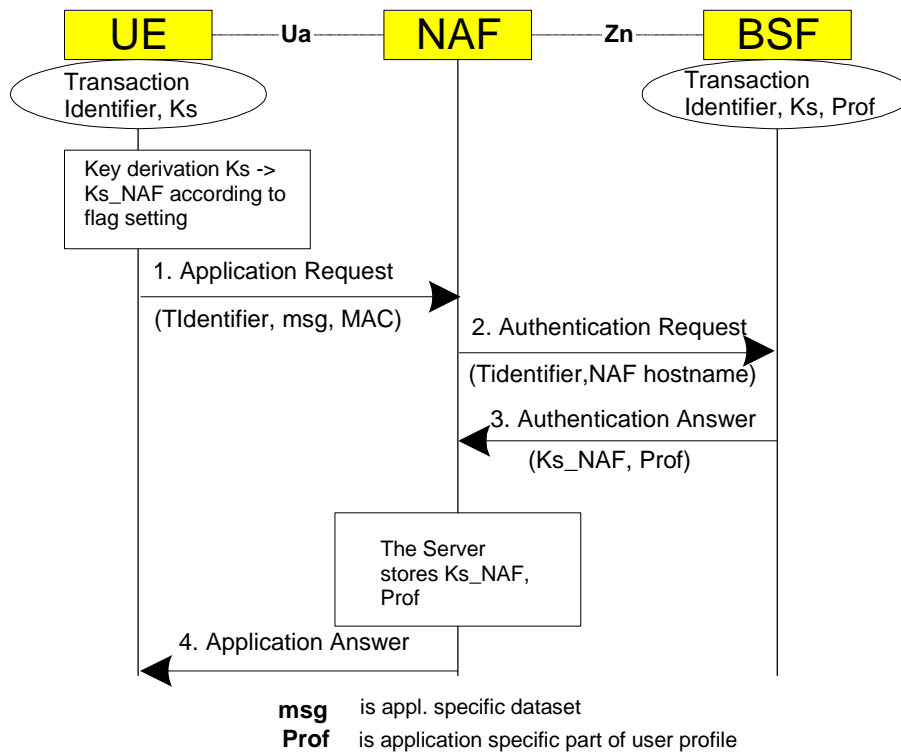


Figure 4.4: The bootstrapping usage procedure

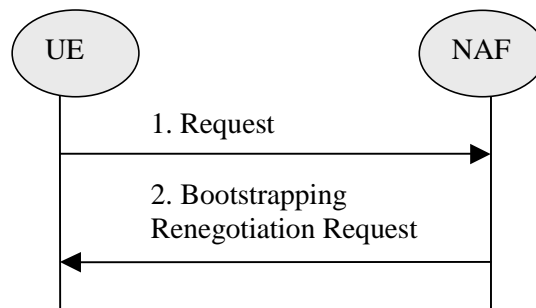


Figure 4.5: Bootstrapping renegotiation request

#### 4.5.4 Procedure related to service discovery

To enable the bootstrapping procedure, a procedure needs to be described on how to discover the location of BSF. It shall be possible to enable the terminal to be configured either manually or automatically via one of the following approaches:

- The address information shall be published via reliable channel. Subscribers shall store all the parameters as part of the initial establishment of IP connectivity. The addresses need to be input only once;
- The address information shall be pushed automatically to the UE over the air interface when the subscription to bootstrapping service is accepted. All the parameters shall be saved in the UE and used the same manner as above. The procedure is specified in [7];
- The location information shall be discovered automatically based on DHCP, after the IP connectivity has been established. The DHCP server shall provide the UE with the domain name of a BSF and the address of a Domain Name Server (DNS) that is capable of resolving the Fully Qualified Domain Name (FQDN) of the BSF. The procedure is specified in TS 23.228 [8].

NOTE: The location of DHCP server may be pushed to UE through the procedure specified in [7].

---

## 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 clause 4.4 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 clause 5.2.1, details on the procedures are in clause 5.3.

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

The requirements and principles from clause 4.3 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.

**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

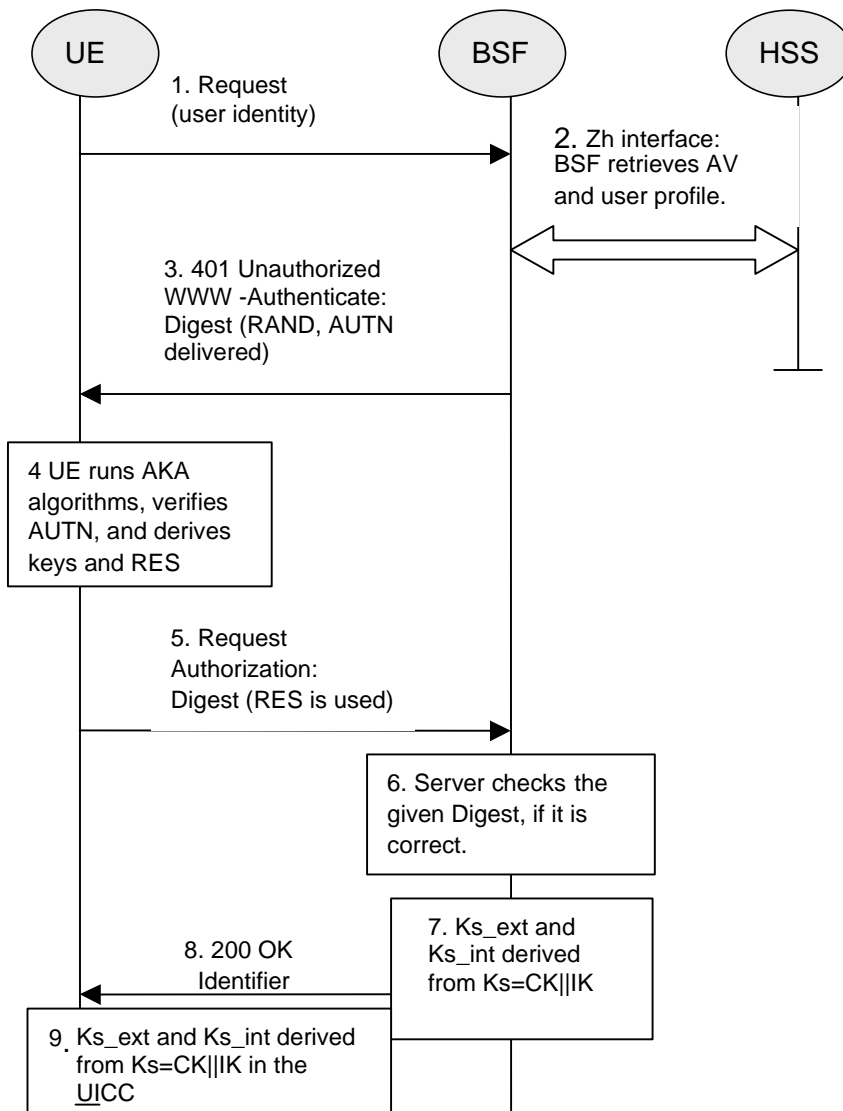
The text from clause 4.5.1 of this document applies also here.

#### 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.



**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.  $h1(Ks, h1 \text{ key derivation parameters}) = Ks\_ext \parallel Ks\_int$  (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  $Ks\_ext\_NAF = h2(Ks\_ext, h2\text{-key derivation parameters})$ , 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.

**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.



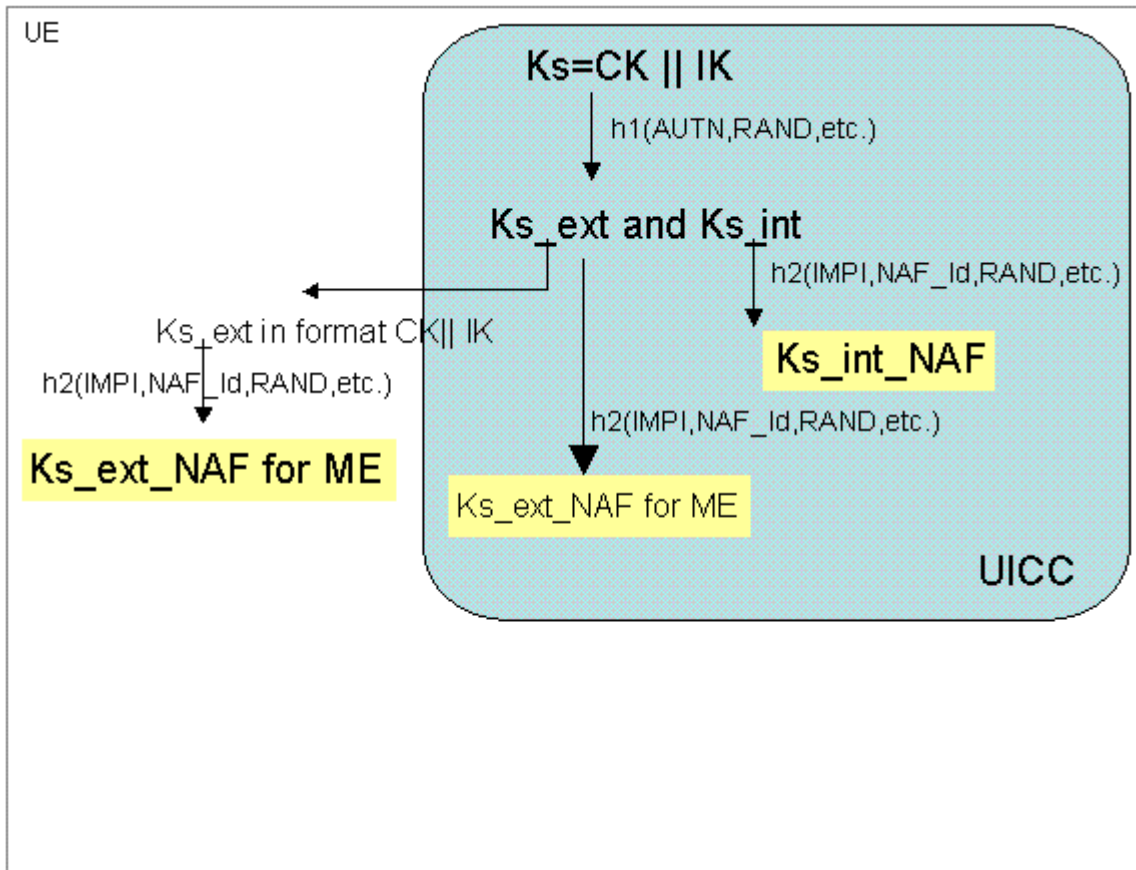


Figure 5.2: Key derivation for GBA-aware UICC when GBA-run was triggered

### 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  $U_a$  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 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 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.

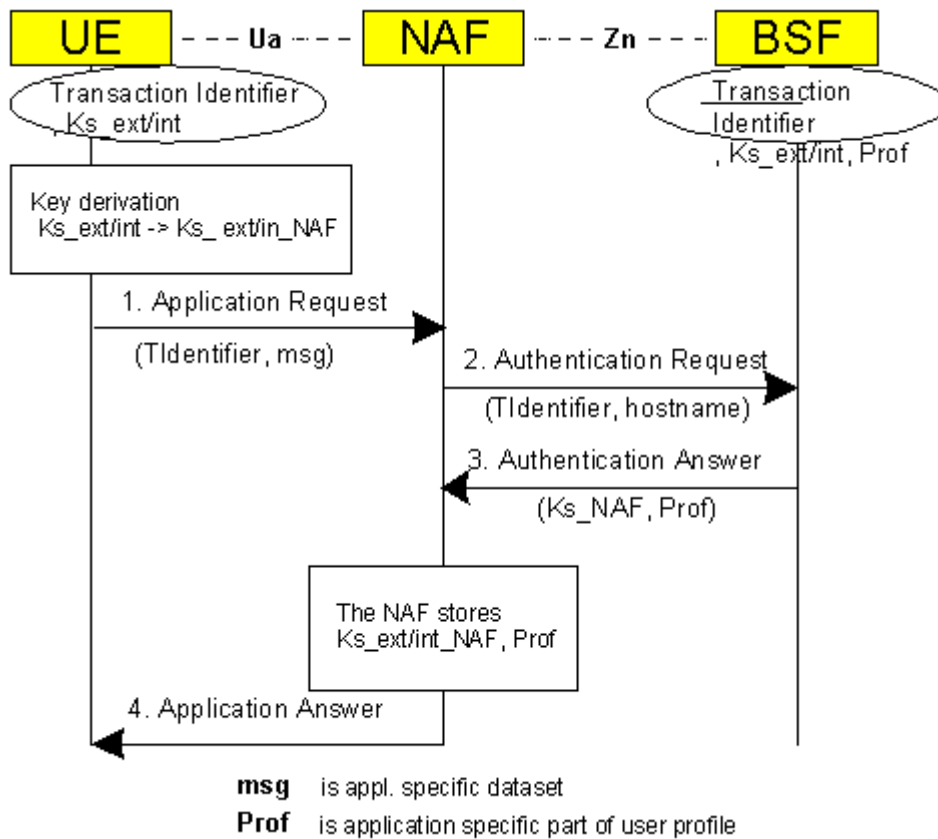


Figure 5.3: The bootstrapping usage procedure with UICC-based enhancements

### 5.3.4 Procedure related to service discovery

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