3GPP TSG SA WG3 Security#31 **S3-030743**
18-21 November, 2003
Munich, Germany

Agenda Item: **6.9 (GAA)**

Source: **Siemens**

Title: **Key separation in a Generic Bootstrapping Architecture**

Document for: **Discussion and decision**

Abstract

At SA3#30, Siemens presented the contribution S3-030552 on "Key separation in a Generic Bootstrapping Architecture". Some of the proposals in contribution S3-030552 were agreed, others were agreed to be decided at meeting SA3#31. The present contribution makes a selection among the alternatives presented in S3-030552 and asks SA3 for a decision. An accompanying pseudo-CR to [33220] implements the recommended changes. It is also suggested here to leave the design of the key derivation algorithm to ETSI SAGE. An LS to ETSI SAGE should be sent if the proposals in this contribution are agreed.

1. Way forward after SA3#30

We quote from the draft meeting report on SA3#30:

1. *"Section 2* [of S3-030552] *showed threats, which become possible if only one application server is successfully attacked. SA3 is asked to endorse that the threats should be mitigated by appropriate provisions in the standard. In particular, it shall be prevented that a security breach in one application server can spread across the entire system.* Agreed."

As a consequence of this agreement, SA3 needs to define a suitable mechanism to address the threat. This contribution and the accompanying pseudo-CR to [33220] provide such a mechanism.

from the draft meeting report on SA3#30:

2. *"It was proposed in section 3* [of S3-030552] *to limit the effect of a security breach in one part of the system to a small part of the system by introducing key derivation for the keys shared between UE and NAF. SA3 is asked to endorse the use of a suitable key derivation procedure.* It was recognised that further discussion is needed on the choice of key derivation mechanisms which binds keys with adequate identity to mitigate the threat. Agreed to look for a suitable solution."

It is proposed, as in S3-030552, to use a suitable key derivation procedure to obtain such a solution. In fact, it is difficult to see how such a solution could be achieved without key derivation. This key derivation is specified in the accompanying pseudo-CR to [33220] which also shows how the key derivation is integrated in the GBA procedures specified in [33220]. Input parameters to the key derivation mechanism are also proposed, but it is suggested to leave the final choice of the input parameters as well as the choice of the key derivation algorithm to ETSI SAGE.

from the draft meeting report on SA3#30:

3. *"Section 4 [of S3-030552] proposed certain alternatives for the NAF identifier which is input to the key derivation parameters. The NAF identifier would determine the degree of assurance the UE gets about the identity of the NAF in NAF-to-UE authentication. SA3 is asked to agree to study only alternatives 1 (use DNS server name) or 2 (use defined parts of DNS server name) further and select between these alternatives at the next meeting.* It was decided to allow delegates to analyse this proposal and make a decision at the next meeting based on available scheme proposals."

It is proposed in this contribution to select alternative 2 (use defined parts of DNS server name) as it significantly enhances the flexibility without unduly increasing the complexity.

from the draft meeting report on SA3#30:

4. *"Section 5 proposed a flexible mechanism to signal that one out of possibly multiple key derivation schemes be used with a certain key. As a minimum, the mechanism could be used to signal whether no key derivation or some pre-determined key derivation is used. SA3 is asked to endorse the use of this flexible signalling mechanism.* It was decided to allow delegates to analyse this proposal and make adopt this at the next meeting if there are no alternative proposals."

It is proposed to adopt the signalling scheme presented in S3-030552 with a minor modification, as specified in the accompanying pseudo-CR.

(The modification consists in the fact, that "no key derivation" is not explicitly signalled any more, rather "no key derivation" is the default when nothing else is signalled over the Ub interface. The value "0" is then used to signal that the NAF identifier is the full DNS name of the NAF.)

In addition, we address an editor's note in 4.3.1 of [33220], which is also reflected in the main text of section 4.3.2 and which, in our opinion, tries to express that the keys at the UE and the NAF resulting from the use of the GBA may need to be adapted (e.g. shortened) in order to fulfil the specific requirements of the Ua interface. This adaptation is, however, outside the scope of the GBA specification. We propose to introduce two notes in section 4.3.2 to capture this.

2. Proposal

SA3 is asked to endorse the proposed changes, as specified in the accompanying pseudo-CR to [33220].

3. Reference

[33220] S3-030662: TS 33.220 v0.1.1 " Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (Release 6)" (Oct 2003)

CR-Form-v7

Pseudo - CHANGE REQUEST

⌘ 33.220 CR CRNum ⌘rev - ⌘ Current version: 0.1.1 ⌘

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

Title:	⌘	Key separation
Source:	⌘	Siemens
Work item code:	⌘	GBA
Date:	⌘	18 Nov 2003
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:
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:	⌘	It was agreed at SA3#30 that the threats described in S3-030552 should be mitigated by appropriate provisions in the standard. This pseudo-CR introduces these provisions.
Summary of change:	⌘	A suitable key derivation procedure is specified
Consequences if not approved:	⌘	Threat is not addressed

Clauses affected:	⌘	

	Y	N			
Other specs affected:	Y	N	Other core specifications	⌘	Related CN1 spec on Ub interface
		N	Test specifications		Related CN4 spec on Zn interface
		N	O&M Specifications		

Other comments:	⌘	

***** Begin of Change ****

4.1 Requirements and principles for bootstrapping

Editor's note: The description of AKA bootstrapping shall be added here.

- The bootstrapping function shall not depend on the particular network application function
- The server implementing the bootstrapping function needs to be trusted by the home operator to handle authentication vectors.
- The server implementing the network application function needs only to be trusted by the home operator to handle derived key material.
- It shall be possible to support network application functions in the operator's home network
- The architecture shall not preclude the support of network application function in the visited network, or possibly even 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 application server using the Generic Boostrapping Architecture can be used by an attacker to mount succesfull attacks to the other application servers using the Generic Boostrapping Architecture.

***** End of Change ****

***** Begin of Change ****

4.1.6 Requirements on Zn interface

The requirements for Zn interface are:

- ~~NAF shall be able to communicate securely with a subscriber's BSF.~~
- Mutual authentication, confidentiality and integrity shall be provided.
- The BSF shall verify that the NAF is authorised;
- The NAF shall be able to send a key material request to the BSF.
- The BSF shall be able to send the requested key material to the NAF.
- The NAF shall be able to get the subscriber profile from BSF.

Editor's note: in later phases there is an additional requirement that the NAF and the BSF may be in different operators' networks.

***** End of Change ****

***** Begin of Change ****

4.2.2 Network elements

4.2.2.1 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 key material must be generated specifically for each NAF independently. The BSF can restrict the use of the key material to a defined set of NAFs by using a suitable key derivation procedure.

Editor's note: key generation for NAF is ffs. Potential solutions may include:
- *Separate run of HTTP Digest AKA over Ub interface for each request of key material from a NAF*
- *Derivation of NAF-specific keys in BSF*

***** End of Change ****

***** Begin of Change ****

4.3.1 Bootstrapping procedures

When a UE wants to interact with an NAF, and it knows that bootstrapping procedure is needed, it shall first perform a bootstrapping authentication (see Figure 1)

Editor's notes: Zh interface related procedure will be added here in future development. It may re-use Cx interface that is specified in TS 29.228.

Otherwise, the UE shall perform a bootstrapping authentication only when it has received bootstrapping initiation required message or a key update indication from the NAF (cf. subclause 4.3.2).

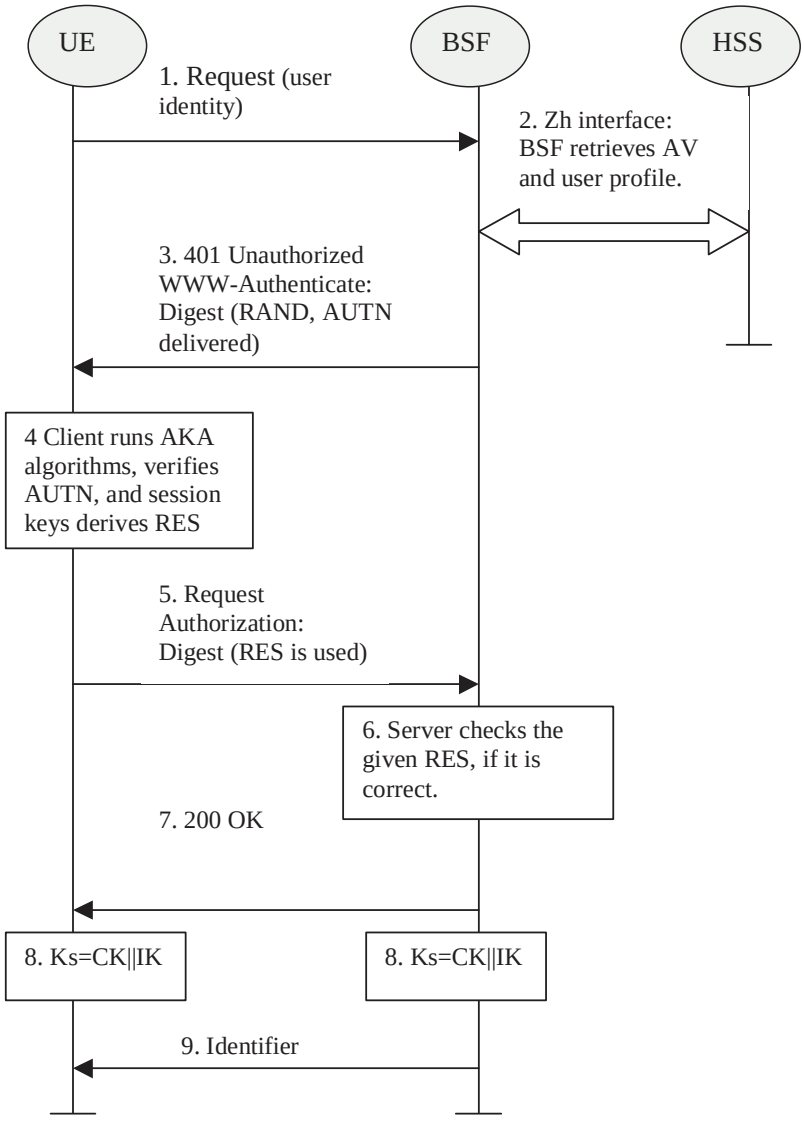

Figure 1: The bootsrapping procedure

1. The UE sends an HTTP request towards the BSF.
2. BSF retrieves the user profile and a challenge, i.e. the Authentication Vector (AV, AV = RAND||AUTN||XRES||CK||IK) over Zh interface 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 calculates the message authentication code (MAC) so as to verify the challenge from authenticated 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 request again, with the Digest AKA RES as the response to the BSF.
6. If the RES equals to the XRES that is in the AV, the UE is authenticated.
7. The BSF shall send 200 OK message to the UE to indicate the success of the authentication.
8. ~~8.~~ The key material Ks is generated in both BSF and UE by concatenating CK and IK. The Ks is used to derive the key material Ks_NAF. Ks_NAF is used for securing the Ua interface.

~~*Editor's note: The key material Ks is 256 bits long. It is up each NAF to make the usage of the key material specifically*~~

9. Ks_NAF is computed as Ks_NAF = KDF (Ks, key derivation parameters), where KDF is a suitable key derivation function, and the key derivation parameters include the user's IMSI, the NAF_Id and RAND. The NAF_Id consists of the n rightmost domain labels in the DNS name of the NAF, separated by dots (n= 1, ..., 7). For n = 0, NAF_Id equals the full DNS name of the NAF. The next bullet specifies how the UE obtains *n*.

Note: this note gives an example how to obtain the NAF_Id: if the DNS name of the NAF is "server1.presence.bootstrap.operator.com", and n = 3, then NAF_Id = " bootstrap.operator.com".

Editor's note: the definition of the KDF and the possible inclusion of further key derivation parameters is left to ETSI SAGE.

10. BSF may supply a transaction identifier to UE in the cause of Ub interface. The BSF may also supply the parameter *n* used to determine the NAF_Id (cf. previous bullet) to the UE over the Ub interface. If the parameter *n* is not supplied then no key derivation is performed, i.e. Ks = Ks_NAF.

4.3.2 Procedures using bootstrapped Security Association

After UE is authenticated with the BSF, every time the UE wants to interact with an NAF the following steps are executed as depicted in Figure 2

UE starts communication over Ua interface with the NAF

- In general, UE and NAF will not yet share the key(s) required to protect Ua interface. If they already do, there is no need for NAF to retrieve the key(s) over Zn interface.
- If the NAF shares a key with the UE, but an update of that key it sends a suitable key update request to the UE and terminates the protocol used over Ua interface. The form of this indication may depend on the particular protocol used over Ua interface and is ffs.
- It is assumed that UE supplies sufficient information to NAF, e.g. a transaction identifier, to allow the NAF to retrieve specific key material from BSF.
- The UE derives the keys required to protect the protocol used over Ua interface from the key material, as specified in clause 4.3.1.

Note: The UE may adapt the key material Ks_NAF *to the specific needs of the Ua interface. This adaptation is outside the scope of this specification.*

NAF starts communication over Zn interface with BSF

- The NAF requests key material corresponding to the information supplied by the UE to the NAF (e.g. a transaction identifier) in the start of the protocol used over Ua interface.
- The BSF derives the keys required to protect the protocol used over Ua interface from the key material and the key derivation parameters, as specified in clause 4.3.1, and supplies to NAF the requested key material. 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 key update request to the UE.
- ~~The NAF derives the keys required to protect the protocol used over Ua interface from the key material in the same way as the UE did.~~

Note: The NAF may adapt the key material Ks_NAF *to the specific needs of the Ua interface in the same way as the UE did. This adaptation is outside the scope of this specification.*

NAF continues with the protocol used over Ua interface with UE

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

Editor's note: Message sequence diagram presentation and its details will be finalized later.

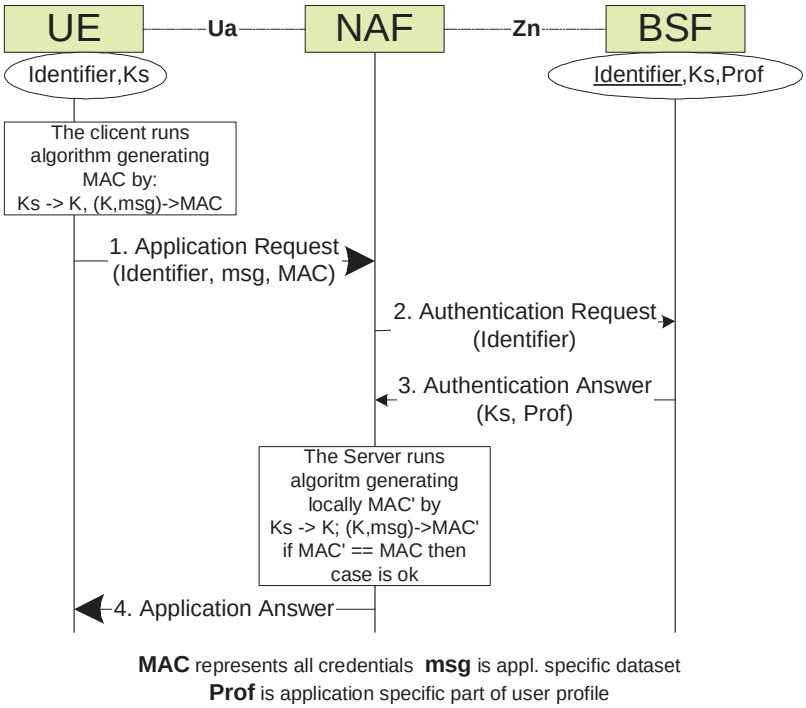

Figure 2: The bootstrapping usage procedure

***** End of Change ****