3GPP TSG SA WG3 Security — S3#32 **S3-040089**
09 - 13 February 2004
Edinburgh, Scotland, UK

Agenda Item: **6.9.2 (GBA)**

Source: **Siemens**

Title: **Introducing a UICC-based Generic Bootstrapping Architecture**

Document for: **Discussion and decision**

Abstract

In the current version of the Generic Bootstrapping Architecture specification (TS 33.220 v100), the keys Ks shared between UE and NAFs are available to the ME. But in the future, it may be desirable for some applications (e.g. MBMS) that Ks does not leave the UICC. In this contribution, we therefore propose to introduce also a UICC-based version of GBA, called GBA_U. It shall be backwards-compatible with the currently discussed version, which we propose to call GBA_ME, in the sense that no changes to GBA_ME are required by the introduction of GBA_U, but a maximum re-use by GBA_U of functionality existing from GBA_ME is possible. We also propose a new structure to TS 33.220 in this document. More technical detail on how GBA_U would differ from GBA_ME is contained in a companion contribution.

1. Rationale for introduction of a UICC-based version of the Generic Bootstrapping Architecture

The GBA establishes a shared key between a UE and a NAF, as specified in TS 33.220 v100. The shared key is computed on the ME. This is quite sufficient for many applications. An example is http digest being used for access to http-based services. The ME obtains CK and IK from the USIM by invoking the normal UMTS AKA authentication procedure, sending a challenge RAND, AUTN to the card. A major advantage with this approach is that any USIM from Rel99 onwards can be used. In order to distinguish the GBA concepts from the UICC-based ones, we propose to call the former GBA_ME and the latter GBA_U.

But there may be applications with a different threat model, for which it would be, at least for high value applications, more desirable to establish a shared key on the UICC, which never leaves the UICC. The different threat model would involve a user who has an interest in divulging his own key to many others. This assumption is not normally made in security protocols, for good reasons, but for certain applications they may be valid. Such applications include MBMS where one of the major threats consists in a rogue user with a valid subscription recovering a key from his own UE. This key would be used to encrypt and decrypt the subscribed multicast stream. The rogue user would then divulge the key to others. In this way, those others could also decrypt the multicast stream without having to subscribe to it. The rogue user could be paid by the others for this “service” and, and this way, benefit from the attack. (A sort of MBMS analogue to call selling.) The consequences of such an attack could be reduced if a shared key used in the process resided on the UICC rather than in the ME. This may be of particular relevance for high-value content. As an example, for MBMS with low- or medium-value content it may be quite appropriate to use GBA_ME. But for MBMS with high-value content GBA_U may give significant security benefits. When using GBA_U for MBMS, the key Ks or a NAF-specific key derived from Ks would remain on the UICC and would be used as the point-to-point key to transfer the multicast (master) keys in a secure way to the UICC.

Other advantages of GBA_U include that the keys Ks could be assigned a longer lifetime. In addition, plastic roaming would become possible without a refresh of Ks.

2. How to proceed with the standardisation of GBA?

It is proposed to progress the work on the current version of GBA as planned before, but under the name of GBA_ME. Pseudo-CRs to TS 33.220 v100 should be handled as before. The work on the GBA_ME is fairly advanced and stage 2 may be perhaps even stabilised as early as this meeting.

It is also proposed to start work on a UICC-based enhancement of GBA, the GBA_U, based on this contribution and the companion contribution. The ongoing work on GBA_ME shall not be affected in any negative way by the new work on GBA_U.

GBA_U shall be backwards-compatible with the currently discussed version of GBA (i.e. GBA_ME) in the sense that no changes to GBA_ME are required by the introduction of GBA_U, but a maximum re-use by GBA_U of functionality existing from GBA_ME is possible.

3. Proposed new table of contents for TS 33.220

In order to accommodate any agreed new specification text on GBA_U in a Technical Specification we propose to introduce a new section 5 in TS 33.220. The new section shall be introduced in such a way that

- TS 33.220 constitutes a self-contained, complete specification of GBA_ME without the proposed new section 5;

- Section 5 makes reference to section 4 as much as possible, so as to minimise repetitions.

The proposed ToC is based on that of TS 33.220 v100. The newly introduced parts are marked.

4. Conclusion

3GPP SA3 is asked to endorse the proposal made in this contribution after studying the companion contribution giving more technical detail on GBA_U.