3GPP TSG SA WG3 Security — S3#36 **S3-040940**

Shenzhen, China, November 23-26, 2004

Source: **3**

Title: **Key freshness in GBA**

Document for: **Discussion and Decision**

Agenda Item: **GBA**

1 Introduction

GBA is designed to be a generic method of generating keys for use between NAF and UE. GBA uses one run of AKA to generate a key Ks(_int/ext), which can be used be used to generate several keys, Ks_(int/ext_)NAF. Each Ks_(int/ext_)NAF is used to provide security between a NAF and a UE. It may be beneficial to provide some cryptographic separation between Ks_(int/ext_)NAFs to ensure the following

1. Ks_(int/ext_)NAF is guaranteed to be different between different NAFs, i.e. two Ks_(int/ext_)NAF generated for different NAFs are unique. This is already included in GBA.

2. Ks_(int/ext_)NAF or keys derived from Ks_(int/ext_)NAF are used in Ua protocols are guaranteed to be fresh for different Ua protocols. This stops an attacker using a weak Ua protocol to obtain K_(int/ext_)NAF and subsequently masquerade as the UE. This is currently not provided by GBA. This is briefly discussed in section 2 of this contribution.

3. A NAF and a UE are guaranteed to be able to agree a fresh Ks_(int/ext_)NAF for use between themselves. This is provided by GBA with a new run of AKA, but could also be enhanced as discussed in S3-040811. A proposal for achieving this enhancement is given in section 3 of this contribution.

2 Freshness between different Ua protocols

As mentioned above, if a Ks_(int/ext_)NAF is used directly in more than one Ua protocol, then a weakness in one Ua protocol mat allow an attacker to pretend to be a particular NAF and obtain Ks_(int/ext_)NAF. The attacker can masquerade as the UE towards the real NAF. There are several ways to avoid this problem, e.g. include some information about the Ua protocol to be used when deriving the Ks_(int/ext_)NAF at the BSF and UE, generate a Ua specific version of Ks_(int/ext_)NAF at the UE and NAF, mandate not using Ks_(int/ext_)NAF directly in Ua protocols. The pros and cons of each method should be analysed by SA3 and the appropriate method be chosen (this is not intended to rule out doing nothing, if that is felt to be the best choice).

3 Enhanced key freshness

S3-040811 discussed the need for enhanced key freshness in GBA. This section provides a possible method of achieving this enhanced key freshness in a generic manner. The enhanced key freshness allows a NAF to guarantee that a Ks_(int/ext_)NAF is fresh without requesting a new run of AKA. This is useful in the following circumstances:

1. a Ua protocol that has no in-built replay protection could generate a fresh key without a re-run of AKA

2. when combined with GBA_U, Ua protocols that uses this key freshness could check the presence of a UICC without a re-run of AKA, i.e. by generating a fresh Ks_(int/ext_)NAF from a Ks(_int/ext) held on the UICC.

If it is considered not worth including enhanced key freshness, then it is proposed to add a note to the GBA specification to provide a warning about the use of GBA with Ua protocols that do have in built replay protection (see a companion contribution for a proposed CR).

It is proposed to allow both the UE and NAF to provide their own random numbers, UE_RAND and NAF_RAND respectively that are used as inputs when generating Ks_(int/ext_)NAF from Ks(_int/ext). Achieving this requires the following functionality, the UE's random number needs to be carried from the UE to the BSF via the NAF, the NAF's random number needs to be carried to the BSF and the BSF's random number also needs to be carried to the UE. Each of these issues is considered in turn.

Passing the UE's random number to the BSF: Currently the UE is required to send the B-TID to the BSF via the NAF. The B-TID is in the form RAND@BSF_address, where the RAND is from the run of AKA. It is possible to extend the B-TID for a particular Ua protocol if required to the form RAND || UE_RAND@BSF_address. This UE_RAND could then be transferred to the BSF. NAFs that do not require the use of UE_RANDs would just transparently pass B-TID onto the BSF, so there is no effect on NAFs that do not want to use this enhanced key freshness. For the BSF, this would require a small amount of additional functionality. It would be required to parse the received B-TID to extract RAND@BSF_address to find the correct Ks(_int/ext) before calculating Ks_(int/ext_)NAF.

Passing the NAF's random number to the BSF: A NAF generated number could be sent to the BSF in the same way as the UE generated one, i.e. by extending the B-TID to the following form: RAND || UE_RAND || NAF_RAND@BSF_address. This puts no more requirements on the Ub interface than is required to solve passing the UE's random number to the BSF. Which element adds the BSF_RAND to the B-TID will be further discussed below.

Passing the NAF's random number to the UE: For Ua protocols that do not require enhanced key freshness, there is no need to do anything. For Ua protocols that require enhanced key freshness, there are two ways that NAF_RAND could be added to B-TID. Both of these methods can co-exist in the sense that different Ua protocols could use the most appropriate method without applying any conditions on other Ua protocols. Firstly the NAF could pass NAF_RAND to the UE before the UE sends B-TID to the NAF. The UE would then include NAF_RAND in the B-TID, it sends to the NAF. The NAF should check that the NAF_RAND has been added correctly before passing on B-TID to the BSF. Secondly the UE could send the B-TID to the NAF, which adds the NAF_RAND to it before sending the BSF. The modified B-TID would then need to be returned to the UE. As previously stated, the choice of methods for passing the NAF_RAND to the UE would depend on the particular Ua protocol.

Overall allowing GBA to have the proposed enhanced key freshness only increases the complexity of how the BSF parses the B-TID to identify the correct Ks(_int/ext) and subsequently generate the appropriate Ks_(int/ext_)NAF. It requires no changes to the parameters passed across the Zb interface or any changes to the Ub or Zh protocols. Ua protocols that do not use this enhanced key freshness also require no changes from how they would be specified if this functionality did not exist. Ua protocols that want to use this key freshness need to have a method of passing NAF_RAND to the UE. A couple of methods of doing this have been discussed. A companion contribution is a CR to add this functionality to the specification.

4 Conclusions

This contribution discusses a couple of issues relating to key freshness in GBA. On the freshness between Ua protocols issue, it is proposed that SA3 should further study the issue to decide on an appropriate solution. For the enhanced key freshness, it is proposed that SA3 decide between

1. if the decision on freshness between Ua protocols is still open, then postpone the decision on the enhanced key freshness in case it is believed their may be overlap between possible solutions to these issues.
2. accept the CR that adds the enhanced key freshness to GBA
3. accept the CR that adds a note about Ua protocols that lack replay protection