3GPP TSG SA WG3 Security — S3#34 **S3-040500**
6 – 9 July 2004
Acapulco, Mexico

Source: **Nokia, Siemens**
Title: Solutions for transfer of User Security Settings -(User-Profile)
Work Item: 7.9.2 GBA
Document for: Discussion and approval

Abstract

This contribution explains why GBA user security settings (previously also called GAA user profiles) are needed, and how these could be transferred from HSS to NAF. The two alternatives under discussion are transfer via Zh and Zn, or transfer via an extended Sh interface. The two approaches are discussed in this paper. It is concluded that both approaches appear feasible, with one potential exception: for the extended Sh interface, inter-network communication between NAFs in visited network and the HSS, required e.g. for MBMS, is seen as problematic, both from a security and an architectural point of view.

It is therefore proposed to transfer at least the authentication part of the GBA user security settings, i.e. the set of identities associated with a user across the Zh and Zn interface. A corresponding CR to TS 33.220 is submitted for approval to SA3#34.

It is further proposed to also transfer the authorisation part of the GBA user security settings, i.e. the set of user permissions, across the Zh and Zn interface at least for certain NAFs specific to GBA, such as PKI portal (TS 33.221) and Authentication Proxy (TS 33.141). However, in case CN4 and SA2 showed strong concerns with this approach, SA3 may have to reconsider. It is further proposed to leave the decision on how to transfer the authorisation part of the GBA user security settings in general to CN4 and SA2. This is reflected by the introduction of corresponding editor's notes in the CR. It should be noted that a NAF in the home network, hosting IMS applications, can retrieve the authorisation part of the GBA user security settings, via the Sh interface according to the current specifications for Release 6, once the NAF has obtained the relevant IMPU via Zn. It also seems that, for the purposes of MBMS, no authorisation information needs to be supplied by the GBA as part of the GBA user security settings

1. Introduction

The Generic Bootstrapping Architecture (GBA), as specified in TS 33.220, provides a mechanism to supply a UE and an application server (NAF) with a shared secret, which is derived from the long-term security association between the USIM and the Authentication Centre in the HSS. In the most basic configuration option of the GBA, the NAF does not learn any permanent user identity, but only a temporary pseudonym of the user, called the transaction identifier, or B-TID. Then the only assurance the NAF obtains from the GBA is that the shared secret is associated with an authenticated subscriber of a particular mobile network operator. This may be quite sufficient for certain applications. Examples of such applications are information services hosted on a NAF, realised as a web server, which are offered by the mobile operator to his subscribers. For some applications, the use of a temporary pseudonym may be even required for privacy reasons,

But for other applications, the NAF may need to know more about the user. The NAF may need to know permanent identities of the user for **authentication purposes**, e.g. the IMSI, or, in the case of IMS, the private identity (IMPI) or (some of) the public identities (IMPUs). These identities need to be asserted to the NAF through the GBA, i.e. after completion of the GBA procedures, the NAF shall be sure that the identities belong to the user the NAF is communicating with indeed.

The NAF may also need to know certain user permissions for **authorisation purposes**, e.g. a NAF may need to know that the user is permitted to access certain applications, but not others, or, in another example, a NAF, which is a PKI portal, may need to know whether certain types of certificates can be issued to the subscriber.

Terminology: The collection of information which a NAF may need for **authentication and authorisation purposes** in the context of the Generic Authentication Architecture, is termed GAA-user-profile in TS 33.220 v610. It is suggested here to call it "GBA user security settings", to distinguish it from user profiles in general~~,~~ and to emphasise that it is required for the purposes of the GBA, not the GAA in general. The current debate in SA3 and CN4 is about how to best transfer this information from the HSS to the NAF. There is one user security setting (USS) per application, the totality of USSs is called GBA user security settings.

2. Approaches to solutions

The current text in TS 33.220v 600, and the contributions S3-040326 and S3-040331, suggest that the G~~B~~AA user security settings are transferred via Zh and Zn. Concerns have been raised against this approach in the LS from CN4 (S3-040210) and in S3-040466 by Nortel ("Transfer of application-specific user profiles in GAA ", submitted to the SA3 mailing list 7 June 2004). The alternative proposal by Nortel is to use an extended Sh interface.

2.1 Transfer of GBA user security settings via Zh and Zn

From S3-040326: user security settings (USSs) are "application specific and may contain one or more parameters, which are security-relevant in the context of GAA, cf. examples further below in this text. The complete set of all application-specific USSs is transferred from HSS to BSF via Zh interface. The BSF forwards selected application-specific USSs to the NAF over Zn, depending on the policy of the BSF and the application indicated in the request from the NAF over Zn." In this proposal, the BSF can tune its policy to send USSs according to the requirements concerning a particular NAF.

2.2 Transfer of GBA user security settings via an extended Sh interface

This is proposed in S3-040466. The key elements of this proposal are:

- The Sh interface is extended to support also non-IMS applications;
- The Sh interface is extended to support, in addition to IMPUs, also other types of identities, such as IMPIs or pseudonyms;
- The HSS generates pseudonyms for users, which are transferred to the BSF over Zh.

It remains unclear from the proposal whether these HSS-generated pseudonyms are permanent, or are freshly generated whenever a new request from the BSF to the HSS is received over Zh.

For the currently specified functionality of the Sh interface, see the quotations from TS 23.002 and TS 23.228 in the Annex to this contribution. It becomes clear in particular, that Sh is an intra-domain interface.

3. Discussion

3.1 Transfer of GBA user security settings via Zh and Zn

In S3-040466, four objections are raised against the transfer of GBA user security settings over Zh and Zn:

CON1 from S3-040466: "GBA is a 'generic' bootstrapping architecture. It should be kept simple and concern only with authenticating the UE and providing the necessary keying material to the UE's and the NAFs, in order to provide the needed keys to secure the communication between the UE and the application."

The transfer of GBA user security settings (USSs) over Zh and Zn is indeed concerned with no more than authentication and authorisation information needed for the purposes of GBA, so the objection does not seem to be valid. It is true, however, that USSs need to be transferred over Zh for all applications a user has subscribed to. But, when the USSs are limited to contain only authentication (i.e., the user's identities) and authorization (i.e., flags to indicate whether the user is permitted to use the particular service), the amount of data per user to be transferred over Zh should be fairly small. Furthermore, this approach has the advantage that the HSS is not bothered by the request of individual application servers.

CON2 from S3-040466: " If the application specific user information that were obtained using the bootstrapping procedure changes, it is not clear how the changed user profile information is propagated to the applications. After bootstrapping, even if the application knows that it needs to obtain the latest profile information, it does not seem possible without initiating a new bootstrapping run."

There seems to be not a fundamental problem with this, as the lifetime of the agreed keys is limited anyhow. If USS / profile revocation was an issue then revocation of the shared secret would have to be an issue as well. At least the authentication part of the USS (identities) will be longer-lived.

CON3 from S3-040466: "... , there may also be another kind of NAF, where the application provider does not want the BSF (or even the HSS – this can be accomplished by storing the information "opaquely" in the HSS) to know the identities used by it"

In order to accommodate this scenario, the NAF_selected pseudonym, which points to the information stored "opaquely" in the HSS, can be included in the authentication part of the USS. It cannot be the task of the G~~B~~AA USS to provided means for applications to manage user-related information in the HSS.

CON4 from S3-040466: “This approach is not flexible: For example, if an application is using other means for authentication (e.g., subscriber certificate – this is a valid case within GAA, liberty alliance identity architecture, etc), ...”

Clearly, 3GPP need not worry about solving any user profile related issues of Liberty Alliance. Rather, it should be remembered that the proposed changes are changes to TS 33.220, which is about G~~B~~AA, and not about authentication mechanisms in general. TS 33.220 should concentrate on solving the issues related to G~~B~~AA, otherwise the problem gets too big and cannot be resolved in time for Release 6.

3.2 Transfer of GBA user security settings via an extended Sh interface

CON1: The HSS is burdened with functionality which is specific to the GBA, and GBA-specific privacy policies cannot be implemented in a GBA-specific component, the BSF.

CON2: currently, the Sh interface is limited to IMS, and within IMS, limited to requests based on public user identities (IMPUs). It is not clear whether Sh can be extended for Rel 6. But on the other hand it is true that, with the alternative approach, the Zh and Zn interfaces would need to be extended.

CON3: it is unclear from S3-040466 how the generation of pseudonyms is supposed to work in the HSS. Clearly, it is not sufficient from a privacy point of view to use only pseudonyms constant over time, as these allow a NAF to link different instances of application use over time, hence make the user traceable by the NAF. On the other hand, a NAF may require a persistent mapping of the user to an identity/ pseudonym, possibly local to the NAF. Then the user identity /pseudonym must be sent from the HSS to the NAF via Zh/Zn. But if the idea in S3-040466 is that the HSS generates a pseudonym for each request by the BSF, then the HSS also needs to assign and transfer a lifetime to the BSF, which then the BSF should transfer to the NAF. But this lifetime should coincide with the key lifetime assigned by the BSF, making a co-ordination necessary. So, it may be best to use B-TID as a temporary pseudonym. If user identities and pseudonyms are transferred to the NAF via Zh/Zn there is no need to store the B-TID in the HSS.

CON4: it is unclear how the access by NAFs in visited networks over Sh shall be secured. NAFs in visited networks are needed for MBMS. Currently, there is work on a DIAMETER proxy architecture to allow NAFs in visited networks to access the BSF. Similar work would have to be performed for the extended Sh interface, but has not even started yet. The security requirements and trust implications are unclear, and seem to be more far-reaching than for a Zn interface extended across network boundaries.

CON5: it is unclear how USSs should be distributed when authentication proxies are used. Should the authentication proxy (AP) have an extended Sh interface to the HSS (yes, if the AP is to assert identities and/or perform access control on behalf of the application server AS), or should the AS have an extended Sh interface to the HSS (yes, if the AS is to verify identities and/or perform access control), or both?

Conclusion

Both approaches seem feasible in principle, with one potential exception: for the extended Sh interface, inter-network communication between NAFs in visited network and the HSS, required e.g. for MBMS, is seen as problematic, both from a security and an architectural point of view. The possibility to complete them within Release 6 needs to be discussed with CN4. Clearly the second approach needs a different mechanism to generate pseudonyms, if it is to be taken into account further. Also, the issue of authentication proxies needs to be addressed.

For a proposal how to proceed, see Abstract.

Annex on definition of Sh

From Ts 23.002, section 6a.7.16:

The Application Server (SIP Application Server and/or the OSA Service Capability Server) may communicate to the HSS. The Sh interface is used for this purpose. Details are described in 23.228 [34], sub-clause 4.2.4.

From TS 23.228, section 4.2.4:

For the Sh interface, the following shall apply:

1. The Sh interface is an intra-operator interface.
2. The Sh interface is between the HSS and the “SIP application server” and between the HSS and the “OSA service capability server”. The HSS is responsible for policing what information will be provided to each individual application server.

3. The Sh interface transports transparent data for e.g. service related data , user related information, … In this case, the term transparent implies that the exact representation of the information is not understood by the HSS or the protocol.
4. The Sh interface also supports mechanisms for transfer of user related data stored in the HSS (e.g. user service related data, MSISDN, visited network capabilities, user location (cell global ID/SAI or the address of the serving network element, etc))

Note: before providing information relating to the location of the user to a SIP Application Server, detailed privacy checks frequently need to be performed in order to meet the requirements in TS22.071 [27]. The SIP Application Server can ensure that these privacy requirements are met by using the Le interface to the GMLC (see TS 23.271) instead of using the Sh interface.

5. The Sh interface also supports mechanisms for transfer of standardised data, e.g. for group lists, which can be accessed by different application servers. Those application servers sharing the data shall understand the data format. This enables sharing of common information between application servers, e.g. data managed via the Ut reference point.