3GPP TSG SA WG3 Security — S3#34 **S3-040466**
July 6 - 9, 2004
Acapulco, Mexico

Agenda Item: **6.9.2 – GAA/GBA**

Source: **Nortel Networks**

Title: **Transfer of application-specific user profiles in GAA**

Document for: **Discussion and Decision**

1. Introduction

In SA3 meeting #33, there were some discussions on the need to transfer application specific user profiles (also referred to as “user security settings” in some proposals) as part of the Generic Bootstrapping Architecture (GAA/GBA). In the meeting, Nortel Networks indicated that we do not agree with the proposals to transfer application specific user profiles using Zh and Zn reference points.

It should be noted that application-specific profile information is neither ‘generic’ nor related to ‘authentication’ and should therefore be out of scope of Generic Authentication Architecture. In this paper, we provide more details on the disadvantages of transferring application specific (or NAF-Specific) user profile or user security settings over Zh/Zn reference points.

Finally, we note that the one aspect of the proposed ‘user security settings’ that is in scope of an Authentication architecture is communication of the authenticated user identity to the application, in a form which the application can understand and whilst also potentially supporting privacy of the IMSI. We propose a pseudonym-based approach to this problem.

2. GAA and application specific user profiles

It has been recognized that applications using Generic Bootstrapping architecture (GBA) and more generically, applications using Generic Authentication Architecture have a need to have UE identities available to them. What identity a particular application is using depends on a number of factors, including the type of application and privacy requirements, among others.

In the last SA3 meeting Siemens and Nokia have proposed that, if needed, user identities and other application specific profile information be transferred using the Zh and Zn reference points in GBA.

In our view, this approach has the following shortcomings:

1. 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. Overloading the GBA with application specific parameters will make the architecture more complex, affect the performance of the bootstrapping function (as the profile information can be potentially large depending on the NAF) and the architecture is no more generic. The performance problem becomes especially pronounced, if the GBA has to support authenticating large number of UEs (in our view, this is more likely is the case)
2. 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.
3. It has been suggested that for privacy reasons, in certain scenarios, temporary pseudonyms (such as B-TID) is used as an anonymous identity. This functionality hides the identity from the NAFs. However, 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 (NOTE: asserted identity does

not make sense in this case as the identities (and possibly other parameters) may be stored/managed in the HSS opaquely by the application itself).

4. 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), it is not clear how these applications would obtain the identities and other needed information (other than the identity provided by the certificate itself). It is preferable to have the solution that works in this scenario as well.

As a result of the above, it can in general be expected that all applications will have their own mechanisms for storing and accessing the user profile information which is applicable to that application. All that an application requires from the GAA is the authenticated user identity, which can then be used as a key into whatever database is used to store user profile information. In the case that the HSS is used, then a generic interface for applications to store profile information in the HSS is required. This could be provided using the Sh interface. This approach is architecturally cleaner and as a result more flexible than overloading the Authentication architecture with proxying of user profile operations.

3. Alternate solution

In order to overcome the shortcomings identified in section 2, we present an alternate proposal in this section.

It should be noted that user profile information can always be obtained by the application using the IMSI as key. However, it is recognised that it may not be appropriate to provide the IMSI to all applications due to privacy restrictions. Some other means to communicate the authenticated identity to the application is reuqired.

In our proposed solution, the HSS generates or assigns a ***pseudonym*** for each UE. This pseudonym is included in the data transferred (along with the authentication vectors) over the Zh reference point. After the successful authentication run, depending on the local policy (e.g., whether the NAF needs access to identities and other information from the HSS), the BSF decides whether to include the pseudonym in the response message of the NAF's key request (Ks_NAF). Depending on the local policy, the BSF may include the authenticated identity (IMSI) of UE, but this is independent of the decision on whether to include the pseudonym or not.

Once the NAF has access to the pseudonym, the NAF uses Sh interface to obtain the identities and any other application specific information needed using the pseudonym. It should be noted that Rel-5 Sh interface does not support the concept of pseudonyms. However, the Rel-6 Sh can be extended to support the retrieval of profile information using pseudonyms. If this solution is agreed, then an LS needs to be send to the CN1/CN4 groups, so that they can start their work on extending the Sh interface.

Although the Sh interface is available only for IMS based applications, in our view, there is no impediment for other non-IMS applications to use this interface from Rel-6 onwards. However, an LS needs to be sent to other relevant 3GPP groups in order to solicit their views on non-IMS applications using Sh interface.

4. Pseudonym Generation in the HSS

The proposed solution in section 3 relies on the HSS to generate pseudonyms, so that the NAFs can use them over the Sh interface in order to retrieve the profile information. The pseudonym is used by the HSS to identify the user. The format of the pseudonym is an implementation issue for the HSS. For example, the pseudonym could be generated by encrypting the IMSI with a pre-determined key, namely, pseudonym key (P_KEY).

When the HSS receives the pseudonym over the Sh interface, it would simply decrypt the pseudonym using the same key (P_KEY) and obtain the IMSI. By combining the IMSI with the NAF identity, the HSS should be able to determine the relevant profile information for that particular application.

Since the pseudonym generation is entirely internal to the HSS, any encryption/decryption algorithm available to the HSS can be utilized and need not be standardized.

5. Conclusion

In this paper, we described a flexible and scalable alternative solution to obtain the identities and other profile information needed by the NAFs from the HSS by extending the Sh interface using pseudonyms.

Proposal 1: SA3 agree that storage and retrieval of application-specific information (user profile information) is out of scope of the Generic Authentication Architecture

Proposal 2: SA3 agree to the alternative solution described in this paper for communicating the authenticated identity to the NAF in GAA.

Proposal 3: Send LS to other relevant 3GPP groups requesting their views on non-IMS applications using the Sh interface from Rel-6 onwards.

Proposal 4: Send LS to CN1/CN4, informing them of SA3's decision and requesting them to start the work on extending the Sh interface for Rel-6.

If the proposals in section are accepted by SA3, Nortel Networks is volunteering to provide the necessary change requests to implement these proposals to the GAA specifications.