3GPP TSG SA WG3 Security — S3#37 **S3-050098**
21-25 February 2005 **Comments to S3-050069**
Sophia-Antipolis, France

Source: **Gemplus, Axalto, OCS**

Title: **Comments to S3-050069 "Clarify the GBA requirements for https applications at Ua reference point"**

Document for: **Discussion and decision**

Agenda Item: **6.9.2**

1. Introduction

For "Access to NAF using HTTPS" specification, CR S3-050069 to TS 33.222 proposes to exclude the usage of GBA_U by only mentioning GBA_ME mechanism.

2. Use of GBA_U for TS 33.222

The reason for change of S3-050069 CR is:
"- By referencing the complete GBA specification it is implied that GBA_U shall be supported by both the NAF and the UE application. This has never been intended as there is no usecase for the Ks_int_NAF key within this specification.
- The BSF is not impacted by this specification."

The interest in GBA_U for TS 33.222 essentially lies in the possibility to use Ks_ext_NAF and not only in the use Ks_int_NAF as indicated in the proposed CR

The presence of GBA_U for TS 33.222 allows the use of the NAF-specific key Ks_**ext**_NAF for the authentication schemes (shared key-based UE authentication with certificate-based NAF authentication and shared key-based mutual authentication between UE and NAF).

The advantages of using Ks_ext_NAF rather than Ks_NAF are the following:

- **Security advantage**
  S3-040773 contribution, presented in the scope of GBA_U discussions, indicates that the security level associated to Ks_ext_NAF (GBA_U) is higher than the security level associated to Ks_NAF (GBA_ME). So, the authentication schemes specified in TS 33.222 should benefit from the GBA_U security features.

- **GBA_U: a generic mechanism**
  GBA specification TS 33.220 states that GBA-aware ME shall support both GBA_U and GBA_ME. So, there is no issue for a GBA-aware ME to run a GBA_U procedure when a GBA-aware UICC is present in the UE.

  Ks_ext_NAF availability
  In case of a GBA-aware UICC, the UE equipped with a HTTP capable client could use the Ks_ext_NAF for the authentication schemes:

1) Shared key-based UE authentication with certificate-based NAF authentication:
When the UE sends a response with an Authorization header field where Digest is inserted using the B-TID and the NAF-specific key as password, the Ks_ext_NAF could be used as NAF-specific key.

2) Shared key-based mutual authentication between UE and NAF
The UE and the NAF could also derive the TLS premaster secret from the NAF-specific Ks_ext_NAF.

Ks_int_NAF availability
The use of Ks_int_NAF could also be foreseen. It was clarified in S3-040774 that the use of Ks_int_NAF does not always require the definition of a new ME-UICC interface since some existing UE applications (not specified in Rel-6) may use those GBA_U NAF-specific keys without involving a new ME-UICC interface (cf (U)SIM toolkit application and JSR177 mechanisms).

Advantage of generic mechanism
TS 33.222 describes how the access over HTTP can be secured using TLS in the Generic Authentication Architecture. Since GBA_U is a generic mechanism, there is no reason to preclude the use of GBA_U for accessing in a secure manner the services that may be accessed over HTTP.

The operators issuing GBA_U-aware UICCs should have the possibility to benefit from the GBA_U security features for the different services that might be accessed. Moreover, operators could decide to have BSF and NAFs that support GBA_U only.

Consequently, the use of GBA_U for TS 33.222 authentication schemes should not be precluded. The interest lies essentially in the use of the Ks_ext_NAF.

3. Proposal

GBA_U mechanism should not be excluded to access services by means of HTTPS. Consequently, we kindly ask SA3 to reject S3-050069 CR and to accept the companion CR to TS 33.222 which completes references to TS 33.220.

3GPP TSG-SA WG3 Meeting #37
Sophia Antipolis, France, 21-25 February 2005

Tdoc ⌘att_S3-050098

CR-Form-v7.1

CHANGE REQUEST

⌘ **33.222** **CR** **018** ⌘**rev** **-** ⌘ Current version: **6.3.0** ⌘

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

Title:	⌘	Clarify the GBA requirements for https supporting applications at Ua reference point
Source:	⌘	Gemplus, Axalto, OCS
Work item code:	⌘	GBA-SSC
Date:	⌘	16/02/2005
Category:	⌘	**F**
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:
Ph2 (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)
Rel-7 (Release 7)

Reason for change:	⌘	TS 33.222 reads the following in the "Requirements on the UE" section: "*To utilise GBA as described in this document the UE shall be equipped with a HTTPS capable client (e.g. browser) implementing the particular features of GBA as specified in TS 33.220 [3]"* Additionally, TS 33.220 reads like this: *"A GBA-aware ME shall support both GBA_U, as specified in clause 5.2.1 and GBA_ME procedures, as specified in clause 4.5"* Therefore, the reference to the GBA_ME specific procedures in TS 33.222 is inconsistent with these requirements and may be misleading. GBA_U is specified as a generic mechanism, it should not be excluded to access services by means of HTTPS.
Summary of change:	⌘	- Correct the references to TS 33.220 in section 5.3. - Clarify that G Ks_ext_NAF could also be used for TS 33.222 authentication schemes.
Consequences if not approved:	⌘	Inconsistency in TS 33.222. Incomplete references to TS 33.220 .

Clauses affected:	⌘	5.3, 5.4

	Y	N		
Other specs affected: ⌘		X	Other core specifications	⌘
		X	Test specifications	

	X	O&M Specifications
Other comments:	⌘	

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

5.3 Shared key-based UE authentication with certificate-based NAF authentication

The authentication mechanism described in this section is mandatory to implement in UE and NAF.

This section explains how the procedures specified in TS 33.220 [3] have to be enhanced when HTTPS is used between a UE and a NAF. The following gives the complementary description with respect to the procedure specified in clause 4.5.3 and 5.3.3 of TS 33.220 [3]. For the purposes of this document Ks_(ext)_NAF refers to the key shared between the UE and a NAF. In the case of GBA_U, Ks_(ext)_NAF refers to Ks_ext_NAF, and in the case of GBA_ME, Ks_(ext)_NAF refers to Ks_NAF. This document specifies the logical information carried in some header fields. The exact definition of header fields is left to stage 3 specifications.

1) When the UE starts communication via Ua reference point with the NAF, it shall establish a TLS tunnel with the NAF. The NAF is authenticated to the UE by means of a public key certificate. The UE shall verify that the server certificate corresponds to the FQDN of the NAF it established the tunnel with. No client authentication is performed as part of TLS (no client certificate necessary).

2) The UE sends an HTTP request to the NAF inside the TLS tunnel (HTTPS, i.e. HTTP over TLS). The UE shall indicate to the NAF that GBA-based authentication is supported by adding a constant string "3gpp-gba" to the "User-Agent" HTTP header as a product token as specified in IETF RFC 2616 [12]. The UE shall send the hostname of the NAF in "Host" HTTP header.

NOTE 1: The ability to send the hostname of the NAF is particularly necessary if a NAF can be addressed using different hostnames, and the NAF cannot otherwise discover what is the hostname that the UE used to contact the NAF. The hostname is needed by the BSF during key derivation.

3) In response to the HTTP request received from UE over the Ua reference point, the NAF shall invoke HTTP digest as specified in RFC 2617 [10] with the UE in order to perform client authentication using the shared key as specified in section 4.5.3 and 5.3.3 of TS 33.220 [3]. The realm attribute of the WWW-Authenticate header field shall contain the constant string "3GPP-bootstrapping" and the FQDN of the NAF, to indicate the GBA as the required authentication method.

4) On receipt of the response from the NAF, the UE shall verify that the FQDN in the realm attribute corresponds to the FQDN of the NAF it established the TLS connection with. On failure the UE shall terminate the TLS connection with the NAF.

5) In the following request to NAF the UE sends a response with an Authorization header field where Digest is inserted using the B-TID as username and the session key Ks_(ext)_NAF as password.

6) On receipt of this request the NAF shall verify the value of the password attribute by means of the Ks_(ext)_NAF retrieved from BSF over Zn using the B-TID received as user name attribute in the query.

7) After the completion of step 6), UE and NAF are mutually authenticated as the TLS tunnel endpoints.

NOTE 2: RFC 2617 [10] mandates in section 3.3 that all further HTTP requests to the same realm must contain the Authorization request header field, otherwise the server has to send a new "401 Unauthorized" with a new WWW-Authenticate header. In principle it is not necessary to send an Authorization header in each new HTTP request for security reasons as long as the TLS tunnel exists, but this would not conform to RFC 2617 [10].

In addition, there may be problems with the lifetime of a TLS session, as the TLS session may time-out at unpredictable (at least for the UE) times, so any request sent by UE can be the first request inside a newly established TLS tunnel requiring the NAF to re-check user credentials.

It shall be possible for the AP/AS to request a re-authentication of an active UE, see TS 33.220 [11], clauses 4.5.3 and 5.3.3.

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

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

5.4 Shared key-based mutual authentication between UE and NAF

The authentication mechanism described in this section is optional to implement in UE and NAF.

Editor's note: If the "Pre-Shared Key Ciphersuites for TLS" Internet Draft [15] does not reach the RFC status by the time when Release 6 is frozen, this subclause shall be removed and the support for the Pre-Shared Key TLS is postponed to Release 7.

The HTTP client and server may authenticate each other based on the shared key generated during the bootstrapping procedure. The shared key shall be used as a master key to generate TLS session keys, and also be used as the proof of secret key possession as part of the authentication function. The exact procedure is specified in Pre-Shared Key Ciphersuites for Transport Layer Security (TLS) [15].

Editor's note: The exact procedure of "Pre-Shared Key Ciphersuites for TLS" is under inspection in IETF. When the procedure is ready in IETF, the description how it is used in GAA should be added to TS 24.109, and this subclause should refer to it. The following gives general guidelines for how the TLS handshake may be accomplished using a GBA-based shared secret. The exact definitions of the message fields are left to the stage 3 specifications.

This section explains how a GBA-based shared secret that is established between the UE and the BSF as specified in TS 33.220 [3] is used with Pre-Shared Key (PSK) Ciphersuites for TLS as specified in IETF Internet-Draft [15].

1. When an UE contacts a NAF, it may indicate to the NAF that it supports PSK-based TLS by adding one or more PSK-based ciphersuites to the ClientHello message. The UE shall include ciphersuites other than PSK-based ciphersuites in the ClientHello message. The UE shall send the hostname of the NAF using the server_name extension to the ClientHello message as specified in IETF RFC 3546 [8].

NOTE 1: The ability to send the hostname of the NAF is particularly necessary if a NAF can be addressed using different hostnames, and the NAF cannot otherwise discover what is the hostname that the UE used to contact the NAF. The hostname is needed by the BSF during key derivation.

NOTE 2: When the UE adds one or more PSK-based ciphersuites to the ClientHello message, this can be seen as an indication that the UE supports GBA-based authentication. If the UE supports PKS-based ciphersuites but not GBA-based authentication, the TLS handshake will fail if the NAF selected the PSK-based ciphersuite and suggested to use GBA (as described in step 2). In this case, the UE should attempt to establish the TLS tunnel with the NAF without including PSK-based ciphersuites to the CientHello message, according to the procedure specified in clause 5.3. This note does not limit the use of PSK TLS to HTTP-based services.

2. If the NAF is willing to establish a TLS tunnel using a PSK-based ciphersuite, it shall select one of the PSK-based ciphersuites offered by the UE, and send the selected ciphersuite to the UE in the ServerHello message. The NAF shall send the ServerKeyExchange message with a PSK-identity that shall contain a constant string "3GPP-bootstrapping" to indicate the GBA as the required authentication method. The NAF finishes the reply to the UE by sending a ServerHelloDone message.

NOTE 3: If the NAF does not wish to establish a TLS tunnel using a PSK-based ciphersuite, it shall select a non-PSK-based ciphersuite and continue TLS tunnel establishment based on the procedure described either in clause 5.3 or clause 5.5.

3. The UE shall use a GBA-based shared secret for PSK TLS, if the NAF has sent a ServerHello message containing a PSK-based ciphersuite, and a ServerKeyExchange message containing a constant string "3GPP-bootstrapping" as the PSK identity hint. If the UE does not have a valid GBA-based shared secret it shall obtain one by running the bootstrapping procedure with the BSF over the Ub reference point as specified in TS 33.220 [3].

   The UE derives the TLS premaster secret from the NAF specific key (Ks_(ext)_NAF) as specified in IETF Internet Draft [15].

The UE shall send a ClientKeyExchange message with the B-TID as the PSK identity. The UE concludes the TLS handshake by sending the ChangeCipherSuite and Finished messages to the NAF.

4. When the NAF receives the B-TID in the ClientKeyExchange messages it fetches the NAF specific shared secret (Ks_(ext)_NAF) from the BSF using the B-TID.

   The NAF derives the TLS premaster secret from the NAF specific key (Ks_(ext)_NAF) as specified in IETF Internet Draft [15].

   The NAF concludes the TLS handshake by sending the ChangeCipherSuite and Finished messages to the UE.

The UE and the NAF have established a TLS tunnel using GBA-based shared secret, and then may start to use the application level communication through this tunnel.

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