3GPP TSG SA WG3#31 **S3-030716**
18-21 Nov 2003
Munich, Germany

Agenda Item: **6.9 GAA and support for subscriber certificates**

Source: **Alcatel**

Title: **Pseudo CR to GAA TR 33.919**

Document for: **Discussion and decision**

Contents

Contents 1
Foreword 1
Introduction 2
1 Scope 2
2 References 3
3 Definitions, symbols and abbreviations 3
3.1 Definitions 3
3.2 Symbols 3
3.3 Abbreviations 4
4 Generic Authentication Architecture 5
4.1 GAA overview 5
4.2 Authentication using shared secret 5
4.3 Authentication based on (public, private) key pair and certificates 5
5 Issuing authentication credentials 5
5.1 Schematic overview 5
5.2 GBA: Mechanism to issue shared secret 6
5.3 SSC: Mechanism to issue subscriber certificates 6
6 GAA building blocks 6
6.1 Overview ..... **Error! Bookmark not defined.**
6.2 GAA 7
6.3 GBA 7
6.4 SSC 7
6.5 HTTPS 8
6.5.1 HTTPS with AP 8
6.5.2 HTTPS without AP 8
7 Application guidelines to use GAA 8
7.1 Use of shared secrets and GBA 8
7.2 Use of certificatesAnnex <X> (informative): Change history 8
7.2 Annex <X> (informative): Change history 9

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

Context of GAA and clarification of how we end up writing this TR (with some reference to 3 TS documents).

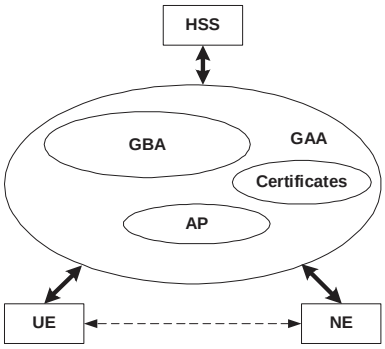

Figure 1. Schematic illustration of GAA

A number of applications share a need for mutual authentication between a client (human user and/or device) and an application server before further communication can take place. Examples include (but are not limited to) communication between a client and a presence server (possibly via an application proxy), communication with a PKI portal where a client requests a digital certificate, communication with a content server, a BM-SC etc.

Since a lot of applications share this common need for a peer authentication mechanism, it has been considereduseful to specify a Generic Authentication Architecture (GAA). This GAA describes a generic architecture for peer authentication that can a priori serve for any (present and future) application.

This TR can be considered as a framework document for the generic authentication architecture as is illustrated in *Figure 1*. GBA, AP and Certificates are building blocks of the GAA and they are specified each in a separate TS. How they fit together in GAA is explained in this document.

1 Scope

This Technical Report aims to give an overview of the different mechanisms that mobile applications can rely upon for authentication between server and user (person and/or device). Additionally it provides guidelines related to the use of GAA and to the choice of authentication mechanism in a given situation and for a given application.

To this end the TR Puts the different specifications under the work item Support for Subscriber Certificates which are releated to peer authentication, into perspective. It cClarifyiesng the logic for having three technical specifications, sketchesing their content and explanesing the inter-relation among these three TSs and their relation with this TR.

~~Give an overview of the different mechanisms that applications can rely upon for authentication between server and user (person and/or device). Give guidelines for applications related to the use of GAA and the choice of authentication mechanism.~~

2 References

The following documents contain provisions that, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[<seq>] <doctype> <#>[([up to and including]{yyyy[-mm]|V<a[.b[.c]]>}[onwards])]: "<Title>".

[1] 3GPP TS 33.102: "Security Architecture".

[2] 3GPP TS 33.220: “Generic Authentication Architecture; Generic Bootstrapping Architecture”

[3] 3GPP TS 33.221: “Generic Authentication Architecture; Support for Subscriber certificates”

[43] 3GPP TS ab.cde: “Generic Authentication Architecture; Access to Network Application Functions using HTTPS”

3 Definitions, symbols and abbreviations

Delete from the above heading those words which are not applicable.
Subclause numbering depends on applicability and should be renumbered accordingly.

3.1 Definitions

For the purposes of the present document, the [following] terms and definitions [given in ... and the following] apply.

Subscriber certificate: a certificate issued to a subscriber. It contains subscriber's own public key and possibly other information such as subscriber's identity in some form.

3.2 Symbols

For the purposes of the present document, the following symbols apply:

Symbol format

<symbol> <Explanation>

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AKA	Authentication and Key Agreement
AP	Authentication Proxy
BSF	Bootstrapping Server Function
GAA	Generic Authentication Architecture
GBA	Generic Bootstrapping Architecture
HSS	Home Subscriber System
NE	Network Element
SSC	Support for Subscriber Certificates
UE	User Equipment

4 Generic Authentication Architecture

4.1 GAA overview

There are generally speaking two tpyes of authentication mechanisms. One is based on a secret shared between the communicating entities, the other one is based on (public, private) key pairs and digital certificates. Also in GAA these are the two options that are a priori available for mobile applications as is illustrated in Figure 2.

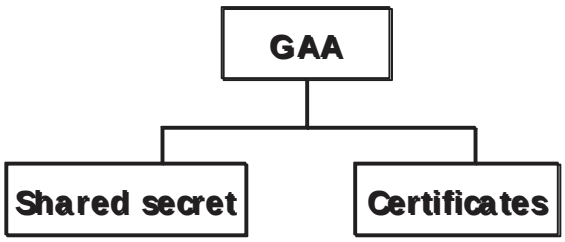

Figure 2: GAA schematic overview

4.2 Authentication using shared secret

There are several authentication mechanisms that rely on a pre-shared secret between the two communicating entities. Popular examples include HTTP Digest, IKE with pre-shared secret and a priori any mechanism based on username and password.

The main problem with these mechansims is how to agree on this pre-shared secret. Section 5.2 and GBA [2] describe how in a mobile context an AKA based mechanism can be used to provide both communicating entities with a pre-shred secret.

4.3 Authentication based on (public, private) key pair and certificates

An alternative to using shared secrets for authentication is to rely on asymmetric cryptography, This assumes that the entity that needs to be authenticated (one or both partners in the communication) possess a (public, private) key pair and a corresponding digital certificate. The latter validates the key pair and binds the key pair to its legitimate owner. Well-known protocols whose authentication is based on (public, private) key pairs include PGP, TLS and HTTPS.

The main disadvatage of this type of authentication is that a PKI structure is needed and that asymmetric key cryptographic operations often require substantially more computational effort than symmetric key operations. Section 5.3 and SSC [3] describe how a mobile operator can issue digital certificates to its subscribers (hence providing a basic PKI).

5 Issuing authentication credentials

5.1 Schematic overview

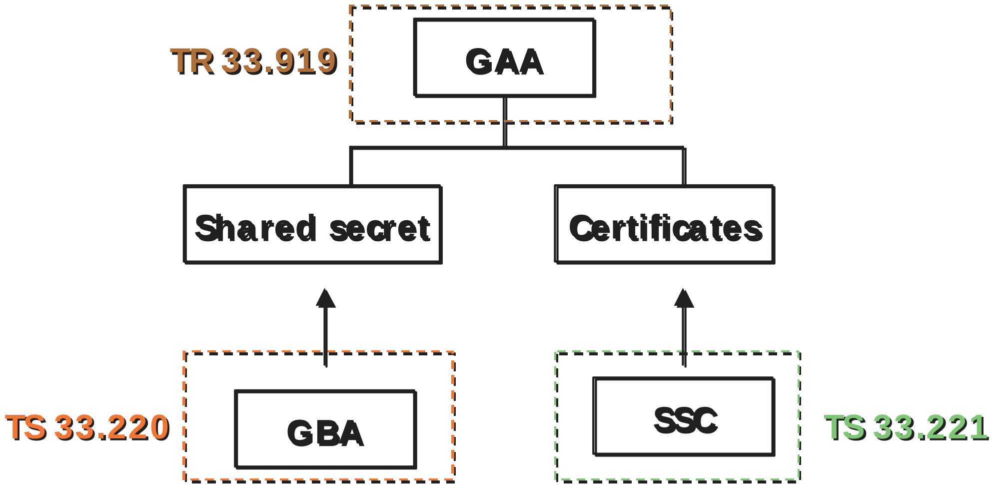

Figure 3 Illustration of mechanisms to issue authentication credentials

Note: other mechanisms for issuing authentication credentials may exist but are out of scope for this TR and the TSs under the referenced WI and will not be discussed here.

Figure 3 illustrates the relation between this GAA TR and TS 33.220 [2] and TS 33.221 [3]. There are on the one hand authentication methods that are based on shared secrets and GBA described in [2] specifies a mechanism to provide communicating parties with such a shared secret. On the other hand there are authentication methods that rely on (public, private) key pairs and digital certificates and SSC described in [3] specifies how to issue certificates to mobile subscribers.

5.2 GBA: Mechanism to issue shared secret

~~*Editor's note: To be completed with a very short explanation and reference to GBA TS.*~~

TS 33.220 Generic Boostrapping Architecture (GBA) [2] specifies an application independent mechanism based on AKA to provide a user and an application server with a common shared secret. This shared secret can subsequently be used to authenticate the communication between the user and an application server.

5.3 SSC: Mechanism to issue subscriber certificates

Editor's note: To be completed with a very *short explanation and reference to SSC TS.*

TS 33.221 Support for Subscriber Certificates [3] specifies a mechanism to issue a digital certificate to a mobile subscriber.

Once a mobile subscriber has a (public, private) key pair and has obtained a certificate for it, he can use the certificate together with the corresponding key pair to produce digital signatures in e.g. m-commerce applications but also to authenticate to a server (as e.g. in TLS).

6 GAA building blocks

6.1 GAA structural overview

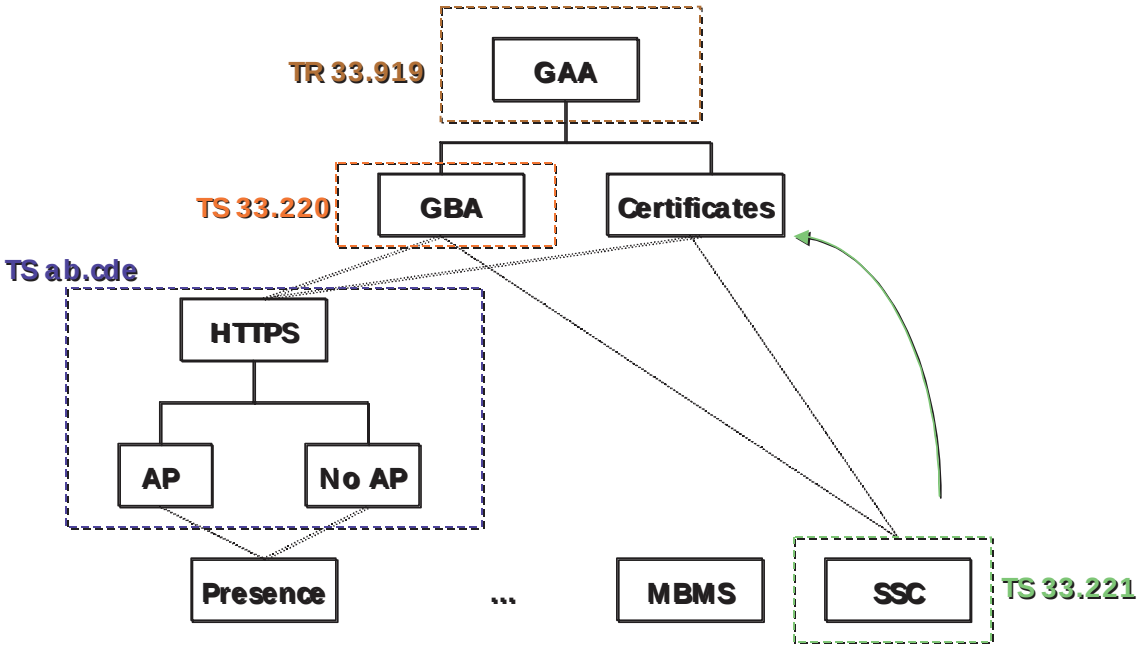

Figure 4 Detailed overview of inter-relation of GAA building blocks

This section gives a high level overview of the content of the different documents under the WI Support for Subscriber Certificates and describes how these documents fit together in the Generic Authentication Architecture framework.

6.2 Generic Authentication Architecture

GAA refers to this TR which descibes the general framework of the Generic Authentication Architecture.

6.3 Generic Bootstrapping Architecture

As briefly indicated in section 5.2, GBA provides a general mechanism based on AKA to install a shared secret between a UE and a server.

AKA is a very powerful mechanism that mobile networks dispose of. GBA takes benefit of this mechanism and re-uses AKA to bootstrap application security. GBA introduces a new network element (NE) called the Bootstrapping Server Function (BSF). This BSF has an interface with the HSS. The UE runs AKA with the HSS via the BSF. From the resulting (CK, IK), a session key is deduced in BSF and UE. An application server (called Network Application Function (NAF) in [2]) can fetch this session key from the BSF together with user profile information. In this way the application server (NAF) and the UE share a secret key that can subsequently be used for application security. The communication between the UE and the BSF as well as that between NAF and BSF and between BSF and HSS are application independent and are specified in [2].

The following arguments lead to the introduction of this new NE (BSF)

- Keep the number of different types of NEs as well as the total number of NEs that retrieve AVs from the HSS to a minimum.
- One generic mechanism for different applications avoids a large diversity of mechanisms and allows to address security issues once and in a consistent way.

6.4 Support for Subscriber Certificates

To obtain a digital certificate a UE must send an appropriate certificate request to a PKI portal of his home operator. The process of issuing subscriber certificates and the corresponding communication session between a UE and a PKI portal is in fact an example of a mobile application. As any mobile application it requires authentication of the communicating entities, in this case the UE and the PKI portal (the latter plays the role of the application server). As any other application there are

2 options for this authentication: pre-sared secret based or based on asymmetric crypto and certificates. The latter is only an option when a new certificate is requested from the PKI portal while another still valid certificate is already loaded in the UE. The former method requires a shared secret between the PKI portal and the UE. And if that is not pre-configured, GBA can be used to obtain such a shared secret.

As indicated in Figure 4, the result of process of issuing a certificate to a mobile subscirber which is descibed in the SSC TS [3] is that the UE is loaded with a (public, private) key pair and a correspondingcertificate. This is indicated by the green upward arrow.

Once the certificate is in place it can be used (together with the corresponding (public, private) key pair) to authenticate the UE. This is indicated by the black dotted lines that connect “certificates” to the underlying applications (HTTPS and SSC in Figure 4).

6.5 Access to Network Application Functions using HTTPS

6.5.1 HTTPS with Authentication Proxy

6.5.2 HTTPS without Authentication Proxy

7 Application guidelines to use GAA

7.1 Use of shared secrets and GBA

7.2 Use of certificates

Annex <X> (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
2003-10	*SA3#30*				*New Draft TR: Generic Authentication Architecture (GAA).*		*0.1.0*