ETSI TR 133 919 V12.0.0 (2014-10)

TECHNICAL REPORT

Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
LTE;
3G Security;
Generic Authentication Architecture (GAA);
System description
(3GPP TR 33.919 version 12.0.0 Release 12)

Reference

RTR/TSGS-0333919vc00

Keywords

GSM,LTE,SECURITY,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**may not**", "**need**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Report 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

This section provides an introduction on the context of GAA and some clarification of why this TR was written (with some reference to related 3GPP Technical Specifications and Technical Reports).

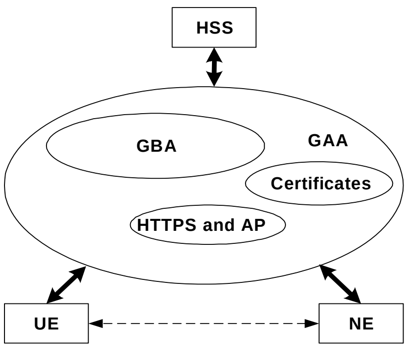


Figure 1: Schematic illustration of GAA

A number of applications share a need for mutual authentication between a client ((i.e. the UE) 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 authentication proxy), communication with a PKI portal where a client requests a digital certificate, communication with a Mobile Broadcast / Multicast Service (MBMS) content server, a BM-SC, etc.

Since a lot of applications share this common need for a peer authentication mechanism, it has been considered useful 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, HTTPS and the Authentication Proxy (AP), and Certificates are the basic building blocks of the GAA in 3GPP Release 6 and they are specified each in a separate TS. Later on, many additions were made to accommodate specific needs for various use cases.

How the different GAA and GBA related specifications and technical reports fit together in GAA is explained in this document.

1 Scope

This 3GPP Technical Report aims to give an overview of the different mechanisms that mobile applications can rely upon for authentication between server and client (i.e. the UE). 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 3GPP GAA related specifications, into perspective. It clarifies the logic for having the technical specifications and technical reports, sketches their content and explains the inter-relation between these 3GPP TSs and TRs and their relation with this TR.

The heart of GAA consists out of the Generic Bootstrapping Architecture (GBA): The GBA core specifications consist out of TS 33.220, TS 24.109 and TS 29.109. Figure 2 depicts protocols used over the GBA core interfaces and the relationships between different GBA core specifications.

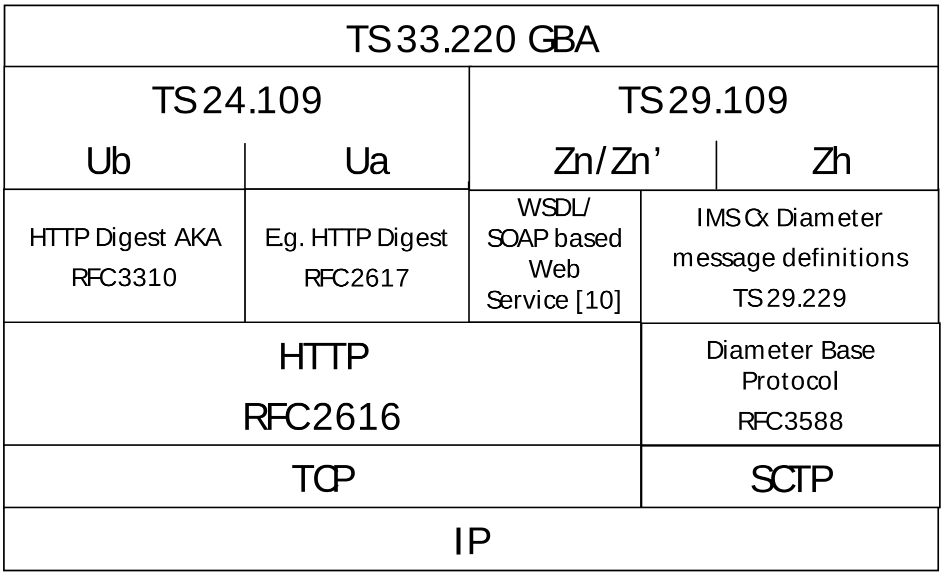


Figure 2: Relationships between GBA core specifications and the protocols used by GBA interfaces

GBA in turn is then used by many other TSs and TRs to enable specific usages e.g. HTTPS, subscriber certificates.

2 References

The following documents contain provisions which, 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*.

[1] 3GPP TS 33.102: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security architecture".

[2] 3GPP TS 33.220: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Generic Authentication Architecture (GAA); Generic bootstrapping architecture".

[3] 3GPP TS 33.221: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Generic Authentication Architecture (GAA); Support for subscriber certificates".

[4] 3GPP TS 33.222: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Generic Authentication Architecture (GAA); Access to network application functions using secure hypertext transfer protocol (HTTPS)".

[5] IETF RFC 2818: "HTTP Over TLS".

[6] 3GPP TS 29.109: "3rd Generation Partnership Project; Technical Specification Group Core Network; Generic Authentication Architecture (GAA); Zh and Zn Interfaces based on the Diameter protocol; Protocol details".

[7] 3GPP TS 24.109: "3rd Generation Partnership Project; Technical Specification Group Core Network; Bootstrapping interface (Ub) and Network application function interface (Ua); Protocol details".

[8] Void.

[9] Void.

[10] W3C: "Web Services Description Language (WSDL) Version 2.0 Part 0: Primer", http://www.w3.org/TR/2005/WD-wsdl20-primer-20050803/.

[11] IETF RFC 4279 (2005) "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)".

[12] 3GPP TS 33.246: '3rd Generation Partnership Project, Technical Specification Group Services and System Aspects; 3G Security; Security of Multimedia Broadcast / Multicast Services'.

[13] 3GPP TR 33.905: '3rd Generation Partnership Project, Technical Specification Group Services and System Aspects; Recommendations for Trusted Open Platforms'.

[14] 3GPP TR 33.920: '3rd Generation Partnership Project, Technical Specification Group Services and System Aspects; SIM card based GBA Generic Bootstrapping Architecture; Early Implementation Feature'.

[15] 3GPP TS 33.110: '3rd Generation Partnership Project, Technical Specification Group Services and System Aspects; Key Establishment between a UICC and a Terminal'.

[16] 3GPP TR 33.980: '3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Liberty Alliance and 3GPP security interworking; Interworking of Liberty Alliance Identity Federation Framework (ID-FF), Identity Web Services Framework (ID-WSF) and Generic Authentication Architecture (GAA)'.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

Subscriber certificate: a certificate issued by a mobile network operator to a subscriber based on his/her subscription. It contains the subscriber's own public key and possibly other information such as the subscriber's identity in some form.

3.2 Abbreviations

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

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

4 Generic Authentication Architecture

4.1 GAA overview

There are generally speaking two types 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 3.

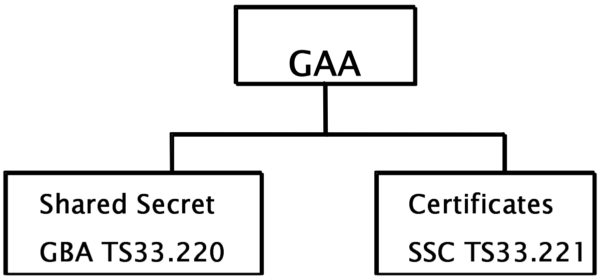


Figure 3: GAA schematic overview

4.2 Authentication using shared secret

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

The main problem with these mechanisms is how to agree on this pre-shared secret. Clause 5.2 and GBA TS 33.220 [2] describe how in a mobile context an AKA based mechanism can be used to provide both communicating entities with a pre-shared secret.

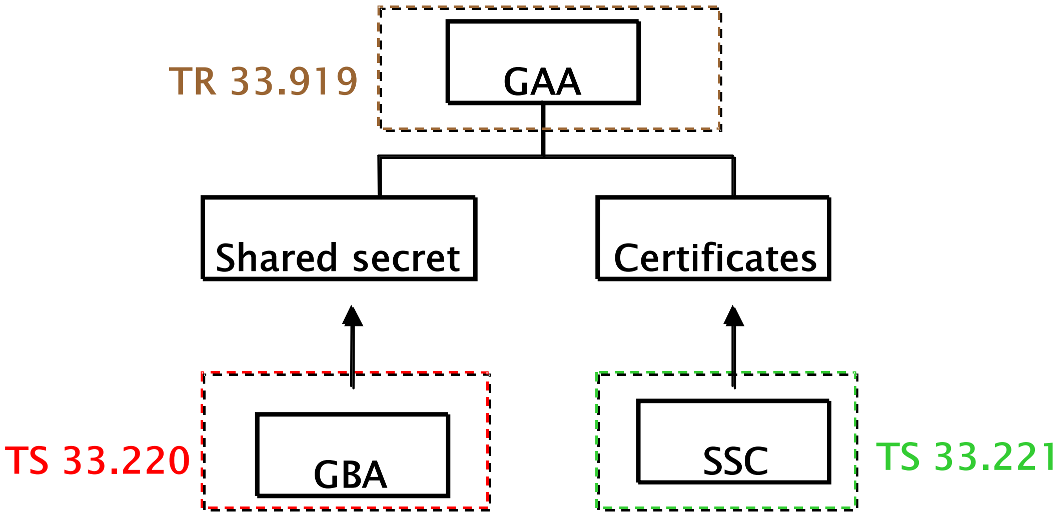
4.3 Authentication based on (public, private) key pairs and certificates

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) possesses 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 and HTTP over TLS, RFC 2818 [5] (the later is commonly called by its protocol identifier, "HTTPS").

The main disadvantage of this type of authentication is that a PKI is needed and that asymmetric key cryptographic operations often require substantially more computational effort than symmetric key operations. Clause 5.3 and Support for Subscriber Certificates (SSC) TS 33.221 [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



NOTE: Other mechanisms for issuing authentication credentials may exist but are out of scope for this TR.

Figure 4: Illustration of mechanisms to issue authentication credentials

Figure 4 illustrates the relation between this 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 TS 33.220 [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 TS 33.221 [3], specifies how to issue certificates to mobile subscribers.

5.2 GBA: Mechanism to issue shared secret

TS 33.220 [2] specifies an application independent mechanism based on the 3GPP AKA mechanism to provide a client and an application server with a common shared secret. This shared secret can subsequently be used to authenticate the communication between the client and an application server.

5.3 SSC: Mechanism to issue subscriber certificates

TS 33.221 [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 (e.g. in TLS).

6 GAA building blocks

6.1 GAA structural overview

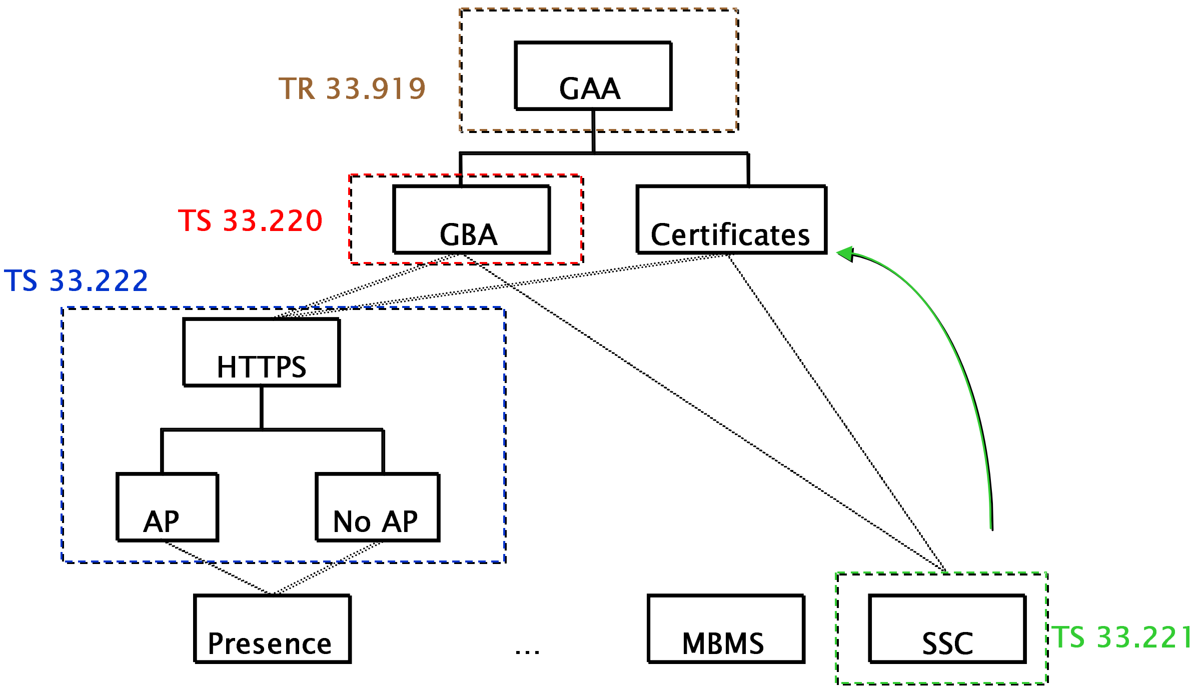


Figure 5: Detailed overview of inter-relation of GAA building blocks

This clause gives a high level overview of the content of the different GAA and GBA related documents and describes how these documents fit together.

6.2 GAA

GAA refers to this TR that describes the general framework of the Generic Authentication Architecture.

6.3 GBA

As briefly indicated in clause 5.2, GBA provides a general mechanism based on 3GPP AKA [1] to install a shared secret between a UE and a server.

AKA is a very powerful mechanism that mobile networks make use 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 derived in BSF and UE. An application server (called Network Application Function (NAF) in TS 33.220 [2]) can fetch this session key from the BSF together with subscriber profile information. In this way the application server (NAF) and the UE share a secret key that can subsequently be used for application security, in particular to authenticate UE and NAF at the start of the application session (possibly also for integrity and/or confidentiality protection although that might not be strictly in the scope of GAA). 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 described in TS 33.220 [2].

If only SIM cards or SIMs on UICC is available, and 2G_GBA is allowed, the BSF and UE mutually authenticates using the 2G AKA and TLS protocol.

The following argument leads 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 SSC

If a client wants to make use of asymmetric encryption technology, he needs a digital certificate that is created by a certification authority (CA). Such a certificate binds a public key to the identity of its legitimate owner and certifies the validity of the public key. If a mobile subscriber wants to have and make use of a (public, private) key pair, the key pair and a certificate should either be preloaded or the subscriber must have the means to either generate or obtain a key pair and dynamically obtain a corresponding digital certificate. As briefly indicated in clause 5.3, SSC specifies a mechanism to dynamically issue a digital certificate to a mobile subscriber.

To dynamically obtain a digital certificate a UE must send an appropriate certificate request to a PKI portal of his home operator, and the PKI portal must authenticate the certificate request. The certificate enrolment process i.e. the issuing of a certificate to a subscriber and the corresponding communication session between a UE and a PKI portal is in fact an example of a mobile application. As with many mobile applications 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 for any other application there are 2 options for this authentication: pre-shared secret based or based on asymmetric cryptography 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. If the shared secret is not pre-configured, GBA can be used to obtain such a shared secret.

As indicated in Figure 5, the result of the process of issuing a certificate to a mobile subscriber which is described in the SSC TS 33.221 [3] is that the UE is loaded with a certificate corresponding to its (public, private) key pair. 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 5). The (public, private) key pair and the corresponding digital certificate can also be used for integrity protection (or less likely confidentiality) but these are not part of the scope of GAA.

6.5 Access to Network Application Functions using HTTPS

It is envisaged that HTTPS (or HTTP/TLS) may be used in a number of services to secure the application session between the UE and the application server (Ua interface in TS 33.220, see TS 33.222 [4]). TS 33.222 [4] describes the details of the possible authentication options when HTTPS is used between a UE and an application server. Any existing or future application based on HTTPS or Pre-Shared Key TLS can refer to TS 33.222 [4] for details on authentication and the set up of a secure HTTP session.

6.5.1 HTTPS with Authentication Proxy

TS 33.222 [4] describes a mechanism where a reverse proxy (called authentication proxy (AP)) is used between the UE and the AS.

The AP is the TLS end point and the UE shall be able to simultaneously connect to different ASs behind one AP. The AP shall be able to authenticate the UE using the means of GAA, and shall send the authenticated UE identity to the AS. If UE authentication is based on a shared secret then the AP acts as the NAF in the GAA architecture and terminology.

Possible advantages of the use of such an AP may include reduced consumption of authentication vectors, minimization of SQN synchronization failures and reduction of number of TLS sessions that a UE needs to set up and maintain.

6.5.2 HTTPS without Authentication Proxy

HTTP based application servers can also be deployed without the use of an authentication proxy. In this case the HTTPS (or TLS) session is between the UE and the AS. In this case the AS shall be able to authenticate the UE using the means of GAA. If UE authentication is based on a shared secret then the AS acts as the NAF in the GAA architecture and terminology.

6.5.3 Pre-Shared Key TLS

The HTTP client and server can authenticate each other based on the GBA-based shared key between the UE and the BSF generated during the bootstrapping procedure created by the procedures in TS 33.220 [2]. The shared key shall be used as a master key to generate further TLS session keys, and also be used as the proof of secret key possession as part of the authentication function. The exact procedure can be found in Pre-Shared Key Ciphersuites for Transport Layer Security (TLS) [11].

7 Application guidelines to use GAA

7.0 Overview on Application Authentication

GAA provides different alternatives to an AS or an AP to perform user authentication (i.e. force the UE to run AKA with the BSF as specified in TS 33.220 [2] or use a mechanism based on subscriber certificates). Also under GAA, an AS may understand that the user request is already authenticated by an Authentication Proxy.

GAA as described in this TR has not the intention to impose any one authentication mechanism onto applications . It is rather aimed to be a tool at developers disposal which they can use to their benefit. Application developers may save development time by using GAA instead of designing and implementing application-specific authentication mechanisms. An additional advantage of the mechanisms of GAA is that they can provide global coverage, inherited from the GSM/UMTS coverage.

Depending on network configuration and policies of the operator, an AS or an AP will be able to use any of the alternatives provided by GAA or even any other user authentication mechanisms specified outside of 3GPP if such mechanisms are at their disposal. It is therefore assumed that an AS and an AP should be able to take the decision what parts of GAA shall be used if any.

This section tries to give an overview of arguments that can play a role in the choice of authentication mechanism. The authentication mechanism selected will be dependent on:

1. Requirements/policies relating to the user/server/application/device that needs authentication. This may be in both directions (mutual authentication), but the usual emphasis is user to server authentication.

2. Device and service characteristics, user capabilities and preferences as defined in the user profile.

3. Policies of the network or networks providing the transport service and the service providers of the applications.

Requirements/policies relating to authentication will depend on whether there is a need for:

a) Device authentication: The device is genuine and not a clone i.e. Authentication of a (U)SIM by challenge response.

b) Integrity protection: An example is signalling protection in UTRAN access A weakness in GSM is that it is very easy for a man in the middle to manipulate signalling message e.g. cipher mode command and a way to prevent it being compromised is to use device authentication **and** integrity protection via a keyed MAC (Message Authentication Code) on the specific signalling messages.

c) Application authentication: It will often be necessary to check the authenticity of the application software. Application authentication is however out of the scope of GAA.

d) User authentication:. This refers to authentication of the end user, the person who is using the end user device. One way of doing this is to make the USIM availability to devices/protocols/applications dependent, logically, by user PIN input or physically, by a policy of removal and insertion. The entry of a PIN may also be required before access is allowed to a specific application.

e) Transaction authentication and non-repudiation: For some business transactions that are carried out using the mobile device it is necessary to digitally sign the transaction with a users private key, specifically where there is a need for non repudiation i.e. to prevent:

- the False Denial of the: SENDING of the Message, e.g. "I never sent it!"

- the CONTENT of the Message, e.g. "I said you should sell, not buy!"

- the TIME of the Message, e.g. "I sent it a different time!"

NOTE: Many authentication techniques such as 3GPP AKA are based on a single key which is shared between the network and the user - this is OK for authentication between sender and recipient, but non repudiation provable to a third party may require the use of public key technique where the private key is only held by the sender.

Figure 6 shows how device and service characteristics can impact the choice of a particular technique from the Spectrum of Authentication Mechanisms.

	authentication type		
client (device) charateristics	device (client) auth	server auth	transaction auth
PIN/password	stored PIN/password	signature or password	x
GAA: subscriber certificate	client private key, signature	signature	private key, signature
GAA: GBA at UE	shared secret (GBA), keyed MAC	shared secret (GBA), keyed MAC OR server private key, signature	x

x = client characteristics do not allow authentication requirement to be met.

Figure 6: Authentication characteristics comparison

7.1 Use of shared secrets and GBA

Some examples of where shared secrets from the innovation of GBA can be used are:

- distribution of symmetric ciphering and integrity keys for securing applications running between the UE and a server in the network. Example protocols that can be used to secure an application and that require a shared secret include HTTP Digest, shared secret TLS and IPsec;
- distribution of passwords and PIN for third party applications;
- for protecting the distribution of certificates between the UE and the certificate authority.

7.2 Use of certificates

Some examples of where certificates can be used for authentication are:

- when it is necessary to check the identity of the end user;
- when the application security protocol works smoothly with (public, private) key pair authentication and subscriber certificates are available (e.g. normal TLS);
- where there is a need for non-repudiation and where the user is required to digitally sign the transaction with a user's private key as many authentication techniques such as 3GPP AKA are based on a single key, which is shared between the network and the user. Non-repudiation provable to a third party may require the use of public key technique where the private key is only held by the sender.

7.3 NAF Recommendations

7.3.1 Overview

GBA can be used by various types of applications, which utilize different types of Ua protocols. In this section, some recommendations and motivations are given to help the design of the Ua protocol. The security and the service delivery would benefit from the following recommendations in some cases, but not all recommendations are applicable to all application types and hence, each NAF Ua application has to validate, if these recommendations are applicable or not.

7.3.2 Key Lifetime Management

If the NAF has a NAF defined lifetime to be used with the NAF specific key material Ks_(ext/int)_NAF key (or further adapted key material), which is shorter compared to the default Ks lifetime received from the BSF, (due to NAF having it''s own policy), then to avoid different lifetimes in UE and NAF and to ensure continuous service delivery to the UE at least two approaches could be taken. One approach is that the NAF could send the key lifetime to the UE on the Ua reference point. This way the UE is able to make a new bootstrapping before the NAF defined key lifetime expires. Another approach, is that the NAF instructs the UE on the Ua reference point to make a new bootstrapping before the NAF defined key lifetime expires.

7.3.3 User Identity Validation

If the UE sends one or more identities as part of the application protocol that is run over reference point Ua, then it would avoid some fraud scenarios, if the NAF verifies that the identities indeed belongs to the subscriber by checking that the identity or identities are present in the service specific USS fetched from the BSF. This is only possible if the BSF and NAF support usage of USS and the NAF can extract the send identities or identity from the message send by the UE.

7.4 Event Monitoring Principles for GAA

Monitoring of GAA usage requires the logging of certain events. The following list of events, grouped by interfaces on which they occur, may serve as a baseline for implementing monitoring in the network nodes.

1 Ub interface: Bootstrapping between UE and BSF over Ub interface,

2 Ua interface:

a) Authentication of UE and NAF based on bootstrapped NAF specific key.

b) Enrollment of subscriber"s public key.

c) Delivery of operator"s root CA certificate.

3 Zn interface:

a) Delivery of NAF specific key material to the NAF,

b) Delivery of USS to NAF

4 Zh interface: Delivery of AV and GUSS.

Note: This list may not be exhaustive.

8 Usage of GBA

This chapter contain some examples of usage of GBA within 3GPP specification and GBA extension.

8.1 GAA and Trusted Open Platforms

TR 33.905 [13] outlines the GAA related functionalities and their interworking in the terminal. It describes the relationship of the GAA server in the terminal that communicates with the BSF server over Ub reference point, and with the UICC through the relevant device drivers, the GAA client that is part of a terminal application and communicates over the network with a NAF server and the GAA server in the terminal to obtain the NAF specific GAA credentials.

8.2 2G GBA

TR 33.920 [14] outlines needed changes to provision an ME and a NAF with a shared secret based on the SIM card. TS 33.220 focuses on the usage of 3G USIM / ISIM for GBA and the SIM card specific details can be found in the Annex I. 2G GBA provides allows building services where authentication is performed and managed in an analogous way as using USIM. The protocol wherein the SIM card is used, decides the strength of the security of the whole system. Therefore, the solution described for an early implementation feature in this specifications targets to enhance GSM security to address the known GSM vulnerabilities when using 2G GBA. 2G GBA does not require any change to the existing SIM specifications, in particular GBA_U as in 3G GBA is not part of 2G GBA.

8.3 Key Establishment between UICC and a Terminal

TS 33.110 [15] describes a GBA based key establishment between a UICC and a terminal. The UICC is rarely a stand-alone device; it usually interacts with a terminal. For sensitive applications that are split between a smart card and a terminal with sensitive data exchanged between those two. TS 33.110 provide means to establish keys, that then can be used to establish a secure channel between a UICC and a terminal.

8.4 Liberty Alliance and GBA

TR 33.980 [16] defines the Interworking between GBA and Liberty Alliance protocols. GBA and the Liberty Alliance Identity Federation and Web Service Framework were developed and can be deployed indepently of each other. The Liberty Alliance Identity Federation and Web Service Framework offers simplified single sign-on and session management for complex web service business interaction protocols. The GBA offers a mechanism to provide a shared secret and certificates to two communicating entities for mobile applications, based on GSM and UMTS authentication and key agreement protocols. TR 33.980 provides guidelines on the interworking of the Generic Authentication Architecture (GAA) and the Liberty Alliance architecture. It studies the details of possible interworking methods between the Liberty Alliance Identity Federation Framework (ID-FF), the Identity Web Services Framework (ID-WSF) and GBA for various Single Sign On scenarios.

8.5 MBMS Security

TS 33.246 [12] specifies the usage of GBA TS 33.220 [2]. GBA is used to provision the keys that are needed to run an MBMS User Service. If protection for the MBMS User Service is required, then the UE needs to share GBA-keys with the BM-SC that is acting as a Network Application Function (NAF) according ot TS 33.220 [2]. The MMBS Serivice Keys for an MBMS User Service shall be stored on either the UICC if the UICC is capable of MBMS key management or the ME if the UICC is not capable of MBMS key management.

Annex A: 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
2003-11	SA3#31				Text has been added to the introduction and to sections 1 to 6	0.1.0	0.2.0
2003-12	SP-22	SP-030582	-	-	Presentation to TSG SA#22 for Information	0.2.0	1.0.0
2004-01	SA3#32	-	-	-	Editorial changes plus an additional reference	1.0.0	1.0.1
2004-02	SA3#32	-	-	-	References have been added. An editor's note in section 6.5 has been removed, Figure with explanation has been added and text has been added to sections 6.3, 6.4, 6.5 and 7.	1.0.1	1.2.0
2004-06	SP-24	SP-040365	-	-	Editorial changes for Presentation to TSG SA#24 for information	1.2.0	1.2.1
2004-07	SA3#34	-	-	-	Text has been added to section 7	1.2.1	1.3.0
2004-09	SP-25	SP-040625	-	-	Edited by MCC for presentation to TSG SA #25 for approval	1.3.0	2.0.0
2004-09	-	-	-	-	Updated to v6.0.0 after TSG SA approval	2.0.0	6.0.0
2004-12	SP-26	SP-040861	0002	-	Removal of unnecessary editor's notes	6.0.0	6.1.0
2005-03	SP-27	SP-050140	0003	-	Correct the 'Application guidelines to use GAA'	6.1.0	6.2.0
2006-03	SP-31	SP-060218	0004	1	Add a text to include the 2G_GBA scenario	6.2.0	7.0.0
2006-03	SP-31	SP-060069	0005	-	Correction of some editorial errors	6.2.0	7.0.0
2006-09	SP-33	SP-060662	0006	1	Verification of UE information and key management	7.0.0	7.1.0
2006-09	SP-33	SP-060502	0007	-	Event monitoring principles for GAA/GBA	7.0.0	7.1.0
2007-03	SP-35	SP-070159	0008	1	Update of TR 33.919	7.1.0	7.2.0
2008-12	SP-42	--	--	--	Upgrade to Release 8	7.2.0	8.0.0
2009-12	-	-	-	-	Update to Rel-9 version (MCC)	8.0.0	9.0.0
2010-06	SP-48	SP-100251	0009	-	removal of Rel-9 OSA outdated references	9.0.0	9.1.0
2011-03	-	-	-	-	Update to Rel-10 version (MCC)	9.1.0	10.0.0
2012-09	-	-	-	-	Update to Rel-11 version (MCC)	10.0.0	**11.0.0**
2014-09	-	-	-	-	Update to Rel-12 version (MCC)	11.0.0	**12.0.0**

History

Document history		
V12.0.0	October 2014	Publication