USECA

Project Number	AC336
Project Title	**USECA: UMTS Security Architecture**
Deliverable Type	Report
Security Class	Public
Deliverable Number	D08
Title of Deliverable	**Intermediate report on UMTS security architecture**
Nature of the Deliverable	Intermediate deliverable
Document reference	AC336/ATEA/WP23/DS/P/08/1
Contributing WPs	WP 2.3
Contractual Date of Delivery	May 1999 (Y02M03)
Actual Date of Delivery	12 August, 1999
Editor	Bart Vinck, Siemens Atea

Abstract	This report describes the preliminary security architecture contained in April release of the 3rd generation mobile telecommunications system standardised by 3GPP, identifies the open issues and provides further proposals on several of these. At the end also a start is made to indicate how the security mechanisms can be integrated in the network architecture.

Keywords	ACTS, USECA, UMTS, security, security mechanisms, security architecture, authentication, key agreement, identity confidentiality, data confidentiality, data integrity, data origin authentication, functional security architecture, integration in the network architecture

Executive Summary

This report provides a snapshot of the current state of the 3rd generation mobile telecommunications system security architecture as it is developing in the Third Generation Partnership Project (3GPP). It describes the preliminary April release; it identifies the open issues which have been identified subsequently, and provides further proposals on a number of them. A initial view of the integration of the security mechanisms into the network architecture is given.

The security architecture consists of four major security feature classes:

network access security:	protecting the (radio) access link between user and provider domain;
provider domain security:	protecting fixed links in the provider domain;
user domain security:	protecting access to the equipment in the user domain (user equipment and access module);
application layer security:	protecting communication between applications in the user and the provider domain.

UMTS security features and mechanisms are carried over from GSM features and mechanisms where possible and sufficient. Modifications and enhancements are included only in the feature classes *network access security* and *provider domain security*. In release '99 however new mechanisms are only introduced in *network access security*.

The network access security mechanisms are:
conventional user identity confidentiality mechanism: a mechanism using temporary identities known to the MS and the SN/VLR (carried over from GSM);
mechanism for enhanced user identity confidentiality: a mechanism between using encryption of the permanent user identities between the user access module and a home environment network entity (a small part of which - the transport mechanism - is standardised);
authentication and key agreement: in addition to the protocol goals achieved by the GSM mechanism, an integrity key is derived alongside a cipher key, both keys are considerably longer than in GSM; assurance of the freshness of the derived cipher and integrity keys is provided by means of an authentication token that is sent alongside the random challenge containing a signed sequence number;
encryption mechanism: a symmetric-key encryption mechanism between user equipment and radio network controller applied to user and signalling data;
data integrity mechanism: a symmetric-key message authentication mechanism between user equipment and radio network controller applied to signalling data;
network-wide encryption mechanism: a mechanism using symmetric-key encryption re-using the algorithms available for access link encryption.
These mechanisms are described in detail and open issues and scope for improvement are identified.

The core of this deliverable is formed by a number of further proposals on the open issues. For the mechanism for enhanced user identity confidentiality we propose a solution that protects the identity not only for mobile-originated but also for mobile-terminated connection establishment. For authentication and key agreement we propose a) modifications to the integration of the re-synchronisation mechanism, b) a sequence number mechanism that assures freshness and at the same protects against long-time lock-up of USIMs and c) a modification to the re-synchronisation procedure that ensures the accuracy of the information received by the home environment on the counter in the USIM. Then follow two contributions on interoperation between UMTS and two 2G mobile communications systems. For GSM both inter-system registration and handover are discussed, for IS-41 based networks only registration is discussed. Further, for data confidentiality and data integrity an overview is provided of the discussions within the RAN-2 group and between SA-3 and RAN-2. Finally some alternative key management schemes for network-wide encryption are proposed and their properties discussed.

The April release of the security architecture together with the modifications and additions proposed here are taken as a basis for a functional network architecture: for each functional network entity, i.e., for each network entity and for each security mechanism, the requirements concerning data storage and implementation of cryptographic functions are given.

A start is made of the description of the modifications to the different network protocols required to support the new and enhanced network access security mechanisms.

1. Table of contents

2. Document management

2.1 Contributors

Else Backx	Siemens Atea Atealaan 34, B-2200, Herentals, Belgium Phone: +32 14 252972 Fax: +32 14 253339 E-mail: Else.Backx@siemens.atea.be
Günther Horn	Siemens AG ZT IK 3, D-81730, München, Germany Phone: +49 89 636 41494 Fax: +49 89 636 48000 E-mail: Guenther.Horn@mchp.siemens.de
Klaus Müller	Siemens AG ZT IK 3, D-81730, München, Germany Phone: +49 89 636 41126 Fax: +49 89 636 48000 E-mail: Klaus.Mueller@mchp.siemens.de
Bart Vinck (editor)	Siemens Atea Atealaan 34, B-2200, Herentals, Belgium Phone: +32 14 252592 Fax: +32 14 253339 E-mail: Bart.Vinck@siemens.atea.be

2.2 Document history

Version	Date	Comment
A	23/05/1999	First draft
B	09/07/1999	Second draft
C	20/07/1999	Third draft
D	02/08/1999	Final version

2.3 References

[1] 3G TS 33.120. 3G Security: Security principles and objectives. Version 3.0.0. April 1999.

[2] 3G TS 21.120. 3G Security: Security threats and mechanisms. Version 3.0.0. April 1999.

[3] 3G TS 33.102. 3G Security: Security architecture. Version 3.0.0. April 1999.

[4] 3G TS 33.103. 3G Security: Integration guidelines. Version 1.0.0.. June 1999.

[5] 3G TS 33.106: 3G Security: Cryptographic algorithm requirements. Version 3.0.0. June 1999.

[6] 3G TR 33.901: 3G Security: Criteria for the algorithm design process. Version 3.0.0. June 1999.

[7] USECA Deliverable 6: Intermediate report on security mechanisms. Final Version. May 1999.

[8] USECA Deliverable 5: Intermediate report on UMTS terminal security. Final Version. July 1999.

[9] S3-99186. CR to TS 33.102 on MAP Security. June 1999.

[10] GSM 02.31: Fraud Information Gathering System (FIGS): Service description; Stage 1. Version 7.1.0. December 1998.

[11] GSM 33.20: Security-related network functions. Version 7.1.0.

[12] N2-99679.Proposed Technical Report. Super-charger. Source: Nortel Networks. May 1999.

[13] ISO/IEC 9798-4: Security techniques - Part 1: Mechanisms using a cryptographic check function ; 1995.

[14] GSM 11.14: Specification of the SIM-terminal interface. Version 7.1.0.

[15] GSM 02.22: Personalisation of GSM UE (ME); Mobile functionality specification. Version 6.0.0.

[16] GSM 03.48: Security Mechanisms for the SIM application toolkit; Stage 2. Version 6.1.0.

[17] S3-99180: CR to TS 33.102 on modified re-synchronisation procedure for AKA protocol. Source: Siemens (USECA). June 1999.

[18] S3-99181: CR to TS 33.102 on sequence number management scheme protecting against USIM lockout. Source: Siemens (USECA). June 1999.

[19] S3-99170: Results of formal analysis of the 3G authentication protocol with modified sequence number management. Source: Siemens AG (USECA). June 1999.

[20] S3-99171: Modified sequence number management. Source: Siemens AG (USECA). June 1999.

[21] S3-99113: Proposal for use of a common RAND in GSM and UMTS authentication. Source: Ericsson. May 1999.

[22] S3-99158: Interoperation between GSM and UMTS. Source: Siemens Atea (USECA). June 1999.

[23] S3-99177: Conversion functions for interoperation between GSM and UMTS. Source: Siemens Atea (USECA). June 1999.

[24] S3-99152, rp-99333: CR to TS 253.01-003. Description of the ciphering model. June 1999.

[25] S3-99123: Integrity Protection Mechanism. Source: Nokia. May 1999.

[26] S3-99174: Integrity protection of RRC messages. Source: Nokia. June 1999.

[27] R2-99390: Further clarifications of the MAC based ciphering solution. Source: Nokia. May 1999.

[28] TIA/EIA PN 2991: Cellular radio telecommunications intersystem operations IS-41rev. D; May 4, 1995.

[29] S3-99164: Key management for network-wide encryption. Source: Vodafone Ltd. (USECA). June 1999.

[30] M.P. Hoyle and C.J. Mitchell, ‘On solutions to the key escrow problem’. In B. Preneel and V. Rijmen (eds.), COSIC ‘’97 Course, Springer-Verlag LNCS **1528** (1998) 307—331.

[31] GSM 04.08. Mobile radio interface layer 3 specification. Version 6.2.0. December 1998.

[32] 3G TS 25.331. RRC Protocol Specification. Version 1.0.0. April 1999.

[33] 3G TS 29.002. Mobile Application Part (MAP) specification. Version 3.0.0. April 1999.

[34] 3G TS 25.413. RANAP Specification. Version 1.0.0. April 1999.

2.4 Abbreviations

3GMS	Third Generation Mobile Communication System
3GPP	Third Generation Partnership Project.
AK	Anonymity Key
AKA	Authentication and Key Agreement
AUTN	Authentication Token for Network-to-user authentication
AUTS	Authentication Token for Re-synchronisation
AV	Authentication Vector
BSS	Base Station System
CK	Cipher Key
CN	Core Network

CS	Circuit Switched
DC	Data Confidentiality
DI	Data Integrity
CUIC	Conventional User Identity Confidentiality
EUIC	Enhanced User Identity Confidentiality
EMUI	Encrypted Mobile User Identity
GI	Group Identity
GK	Group Key
GPRS	General Packet Radio System
GSM	Global System for Mobile Communication
HE	Home Environment
HLR	Home Location Register
IK	Integrity Key
IMSI	International Mobile Subscriber Identity
IMUI	International Mobile User Identity
KSI	Key Set Identifier
KSS	Key Stream Segment
LAI	Location Area Identity
MAC	Medium Access Control
MAC	Message Authentication Code
MAC-A	MAC for user authentication
MAC-I	MAC for data integrity
MAC-S	MAC for resynchronisation
MAP	Mobile Application Part
MM	Mobility Management
MS	Mobile Station
MSC	Mobile Services Switching Centre
PS	Packet Switched
P-TMSI	Packet-TMSI
RAI	Routing Area Identity
RAN	Radio Access Network
RANAP	Radio Access Network Application Part
RAND	Random challenge
RES	Response
RLC	Radio Link Control
RRC	Radio Resource Control
SA	Services and Systems Aspects (3GPP)
SA3	3GPP TSG SA WG3 - Security
SGSN	Serving GPRS Support Node
SQN	Sequence number
SN	Serving Network
SSD	Shared Secret Data
TMSI	Temporary Mobile Subscriber Identity
TMUI	Temporary Mobile User Identity
UE	UE
UEA	UMTS Encryption Algorithm
UIA	UMTS Integrity Algorithm
UIC	User Identification Centre
UIC	User Identity Confidentiality
USIM	User Services Identity Module
UTRAN	UMTS Terrestrial Radio Access Network
VLR	Visited Location Register
XMAC-A	Expected MAC for user authentication
XMAC-I	Expected MAC for data integrity
XMAC-S	Expected MAC for resynchronisation
XRES	Expected Response

3. Introduction

In early May we started writing this intermediate report on the UMTS security architecture. At that time the April release of the specifications of the 3G mobile system to be specified by the Third Generation Partnership Project (3GPP) was recently issued. Among those specifications three security-related specifications [1]-[3] that were considered sufficiently stable to be raised to version 3.0.0 and put under rigorous change control. And two more specifications [4], [5] and one technical report [6] were planned to follow in June. For the specification of the security principles and objectives [1] and of the security threats and requirements [2] this status was appropriate, but for the security architecture specification [3] containing all security features and mechanisms, the many change requests proposed by SA-3 at the SA plenary meeting in June showed that the work on the security mechanism and architecture was still far from completed and the document itself not yet stable. At various points [3] contains several options which are equivalent—or at least acceptable—from a security point of view, and SA-3, the group within 3GPP responsible for security issues, has been waiting for feedback of experts outside the security area to provide some guidance on the best way forward. The security architecture has developed in a well structured way, starting from principles (among which is to keep closely to the existing GSM architecture) and objectives, via requirements and threat analysis, all the way through to definition of features and mechanisms, little known outside the security group. Perhaps because of this rigidity, hardly any feedback from outside has reached SA-3. Further open issues were raised by taking into account relatively new features and concepts originating from outside SA-3. This has led to some uncertainty, which was probably most significant for the security features for data integrity on signalling messages. At the time of writing, there is still discussion on the allocation of the functionality, at the user and the network side, and hence on the protocol layer in which it has to be integrated.

This lack of stability and detail made it necessary that a large part also of this deliverable is devoted to further proposals on the open issues concerning the mechanisms themselves, rather than on the later stage of integration into the network architecture. This means that the original goal for this document, as complementary to [7], concerned with the integration of the 3G security mechanisms, has had to be changed.

Chapter 4 and chapter 5 summarise the April release of the security architecture [3]. Chapter 4 provides an overview of the security architecture and of the different security features or services that are part of the different security feature classes. Chapter 5 contains a description of the security mechanisms that support the security features for network access, predominantly those that protect the radio access link, as the only new mechanisms in Release '99 will be to support features from that class.

The proposals in Chapter 6 on open issues, all on network access security mechanisms, form the core of this deliverable. Some sub-sections there describe modifications already proposed by USECA members to SA-3 (and usually agreed), some contains proposals not yet presented and some describe the discussions partly going on in RAN-2, the group within 3GPP responsible for the definition of the layer 2 functions within the radio access network, on cipher and data integrity, the security features with mechanisms entirely integrated in the radio access network.

Chapters 7 and 8 the outline a security architecture that by the unsettled state of the security mechanisms still has to be provisional. Chapter 7 provides a functional security architecture, and describes for each functional entity, i.e., for each network entity and security mechanism, the requirements as regards data storage and implementation of cryptographic algorithms. Chapter 8 describes the modifications that are required to support the different security mechanisms in several network protocols.

Finally in chapter 9 we present the conclusion we attach to this work as well as an outlook to the further work that needs to be undertaken by USECA WP 2.3 and—we believe—by 3GPP SA-3.

4. 3G Security architecture

4.1 Overview

The April release of the 3G security architecture specification [3] defines five security feature classes:

(I) **Network access security:** the set of security features that provide users with secure access to 3G services, and which in particular protect against attacks on the (radio) access link.

(II) **Network domain security:** the set of security features that enable nodes in the provider domain to securely exchange signalling data, and protect against attacks on the wireline (fixed) network.

(III) **User domain security:** the set of security features that secure access to mobile stations

(IV) **Application domain security:** the set of security features that enable applications in the user and in the provider domain to securely exchange messages.

(V) **Visibility and configurability of security:** the set of features that enables the user to inform himself whether a security features is in operation or not and whether the use and provision of services should depend on the security feature.

Figure 4-1 shows the entities and links that are involved in the above security feature groups.

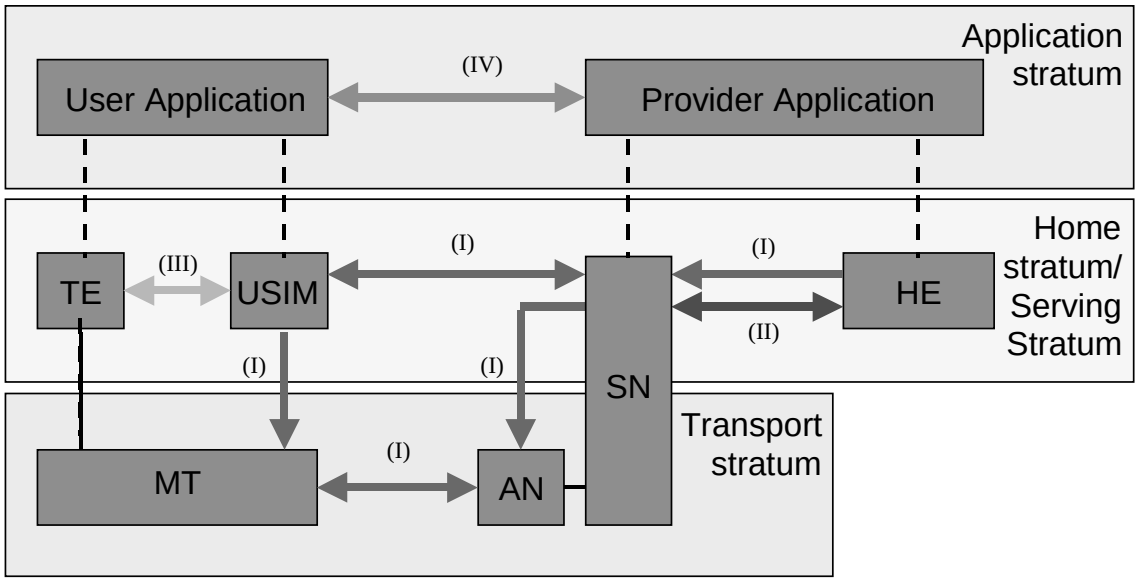

Figure 4-1: Overview of the 3G security architecture [3]

4.2 Network access security

4.2.1 Overview

This security feature class contains the security features that provide users with secure access to 3G services, and that in particular protect against attacks on the (radio) access link. The following security features are listed in that section:

a) User identity confidentiality

b) Entity authentication (access link)

c) Data confidentiality (access link)

d) Data integrity (access link)

e) User equipment identification

The first three security features are carried over from GSM, but the security mechanisms that provide them are enhanced. The fourth, data integrity is a new security feature to protect against hijacking of services, to authenticate the user and the network during and before service provision and to enable bot h parties to securely set-up connections without executing an authentication and key agreement (AKA) protocol.

The need for the fifth security, user equipment identification, is questioned and especially the mechanism which is not secure. It should be replaced by a security feature that, in 5[3], is placed under application layer security:

f) Network-wide encryption.

4.2.2 User identity confidentiality

For **user identity confidentiality** the GSM mechanism using temporary identities agreed between the serving network (SN) and the user/subscriber is retained (see 5.1.2). However, the GSM mechanism allows the SN to request the user to send his user identity in cleartext over the radio access link. The fact that this procedure cannot be removed allows an active attack, using a so-called identity catcher, that reveals the user's identity [7]. A UMTS HE has the option to protect its users against such attack by implementing a mechanism for enhanced user identity confidentiality (EUIC) between its user's USIMs and a HE network entity: the User Identification Centre (UIC). To provide that, the SN will support a transport mechanism for enhanced user identity confidentiality (see 5.1.3). The implementation of mechanism for EUIC between USIM and UIC is optional and the mechanism itself may be HE proprietary. An example mechanism is included in an annex of [1] (see 5.1.4).

4.2.3 Entity authentication

The security feature **entity authentication** for user and network is provided by the UMTS authentication and key agreement (AKA) mechanism (see 5.2). The authenticating parties are the USIM issued by the HE and the Authentication Centre (AuC) in the HE domain.. In addition to the security features provided by the GSM mechanism, the UMTS AKA mechanism assures that the user only accepts fresh authentication data (random challenge and consequently the derived keys). Furthermore, the mechanism derives a cipher key for data confidentiality and an integrity key for data integrity of data transmitted over the radio access link. The mechanism described in the main body of [3] uses sequence numbers and counters both in the AuC and the USIM to assure the freshness of the authentication data. According to the definitions used in ISO/IEC 9798-4 [13] the mechanism provides **mutual authentication** between the user and the network.

Note: In annex D of [3] an alternative mechanism for user authentication is described. The mechanism is kept there to have a backup in case the integration of the mechanism in the main body in the overall network architecture would proof difficult. The mechanism in annex D however is a much larger deviation from the GSM security architecture, which is the primary reason why the "sequence number" mechanism is currently retained as the working assumption. We will not discuss the alternative mechanism any further in this deliverable.

4.2.4 Data confidentiality

As in GSM, in the 3G mobile communication system **data confidentiality** will be applied to user data and signalling data transmitted over the radio access link. To achieve this, a stream cipher will be implemented at either side of the access link. Compared with GSM, UMTS ciphering will be terminated further into the network and be applied at a higher layer (see 5.3). Also, the effective key length is defined as longer than the key length used in GSM 33.20 [10].

4.2.5 Data integrity

The first new network access security feature in the 3G mobile communication system is **data integrity** that will be applied to selected signalling messages transmitted over the radio access link. To provide this a message authentication function will be implemented at either side of the access link (see 5.4).

4.2.6 Network-wide encryption

The second new network access security feature the 3G mobile communication system is **network-wide encryption**. It is an extension of this security feature that provides a protected mode of transmission on user traffic channels across the entire network. This gives users assurance that their user data is protected against eavesdropping on every link within the network, i.e., not just the particularly vulnerable radio links in the access network, but also on the fixed links within the core network. The mechanism is described in 5.5.

Note: This security feature is included in [3] in the category of application layer security features. However, the mechanism re-uses the encryption for network access security, and at least the mechanism should therefore be considered a network access security feature, although its objective—with respect to (access link) data confidentiality—primarily is to protect the user data when transmitted over the core network connections.

4.2.7 User equipment identification

GSM implements a mechanism for **user equipment identification** but that mechanism is not secure. Several mechanisms have been proposed in 3GPP SA-3. However, many delegates have expressed the view that the effort required for a secure solution is cost effective. In order to deter theft, a personalisation feature between the USIM and the UE might provide an alternative mechanism, without involvement of network entities. In [3] no mechanism for identification is included, and for the time being it is assumed that the GSM procedure is retained. We will not discuss mechanisms for UE identification in this deliverable. They are the subject of a separate deliverable [8].

4.3 Provider domain security

4.3.1 Overview

This security feature class contains the security features that provide serving network operators and home environments with the ability to securely communicate over the core network links and to monitor the usage of their system and to detect and to counter fraudulent behaviour . The following security features are included:

a) Entity authentication (between core network entities);

b) Key agreement (between core network entities);

c) Key distribution (between core network entities);

d) Data confidentiality (of signalling over core network links);

e) Data integrity (of signalling over core network links);

f) Fraud detection.

4.3.2 MAP security

The first five security features—**Entity authentication**, **key agreement**, **key distribution, data confidentiality** and **data integrity** enable nodes in the provider domain to securely authenticate and agree on session keys, and subsequently securely exchange signalling data. To provide this, a mechanism has been proposed that may be used to secure MAP messages on the SS7 signalling network. For its most up-to-date version we refer to [9].

These security mechanisms provide an essential security feature that is certainly required before UMTS (as well as GSM) providers can securely migrate from dedicated to shared signalling links, for instance, over an internet. The implementation of the mechanisms for MAP over SS7 security are however not part of the release '99 of UMTS. For that reason it will not be discussed further in this deliverable.

4.3.3 Fraud detection

The security feature **fraud information gathering** enables providers to detect and combat fraudulent behaviour. SA-3 agreed that the mechanisms for release '99 are to be taken over entirely from the relevant GSM specifications [10].

4.4 User domain security

4.4.1 Overview

This security feature class contains those security features that control the access to the USIM or to the terminal and that are completely implemented in the user domain (UE + USIM).

The following security features are described:

– User-to-USIM authentication

– USIM-to-UE authentication

4.4.2 User-to-USIM authentication

User-to-USIM authentication restricts access to the USIM to an authorised user or to a number of authorised users. To achieve this, the users and the USIM must share a secret (e.g. a PIN) that is stored securely in the USIM. The mechanism is to be carried over from GSM 11.14 [14].

4.4.3 USIM-to-UE authentication

USIM-to-UE authentication restricts access to UE to a particular USIM or a set of USIMs. To this end, the USIM and the UE must share a secret that is stored securely in the USIM and in the UE. The mechanism is carried over from GSM 02.22 [15].

4.5 Application layer security

This security feature class provides security interfaces and sublayers that allow applications to securely communicate across wireline and radio links, i.e., to provide secure application security between applications in the user domain and in the provider domain.

Special attention is devoted to the **SIM Application Toolkit.** This is a standardised interface that supports several security services, e.g., entity authentication, message authentication, replay detection, sequence integrity assurance, confidentiality and proof of receipt. The interface is carried over from GSM 03.48 [16].

4.6 Visibility and configurability of security features

Although in general the security features should be transparent to the user, for certain events and according to the user's concern, greater user **visibility** of the operation of security features should be provided. This may include: a) an indication of access network encryption; b) an indication of network-wide encryption; c) an indication of 2G/3G "level of security" (e.g., when the user moves from 3G → 2G).

Configurability is the property that the user and the user's HE can configure whether the use or the provision of a service should depend on whether a security feature is in operation. This may include: a) enabling/disabling user-USIM authentication for certain services, b) accepting/rejecting incoming non-

ciphered calls; c) setting up or not setting-up non-ciphered calls; d) accepting/rejecting the use of certain ciphering algorithms.

Currently only the indication of ciphering is the only feature that is part of UMTS release ’99.

Note: The value of this security feature class is questionable. Often the “security features” are either implemented in a single network entity or they provide small boundary conditions on existing security features and mechanisms and could be included in the descriptions of these.

5. Access link security mechanisms

5.1 User identity confidentiality

5.1.1 Outline

This section describes the mechanisms for user identity confidentiality as they are part of the April release of the 3G security architecture [3].

Essentially, the GSM mechanism using a layer of temporary user identities known to the SN/VLR[1] and the user/subscriber is retained (see 5.1.2). However, the GSM mechanism allows the SN/VLR to request the user to send his user identity in cleartext over the radio access link. The fact that this procedure cannot be removed allows an active attack, using a so-called identity catcher, for further details see USECA D06 [7]. In UMTS a HE will be given the option to protect its users against such attacks by implementing a mechanism for enhanced user identity confidentiality between its user's USIMs and a HE network entity. To achieve this the SN will support a transport mechanism for enhanced user identity confidentiality (see 5.1.3). The implementation of the security feature itself is optional for the HE and the HE may choose its own mechanism (as long as it fits in the transport mechanism). An example mechanism is included in annex B of [3] (see 5.1.4).

5.1.2 Mechanism using temporary identities

This mechanism ([3], 6.1; [11]) allows the identification of a user on the radio access link by means of a temporary mobile user/subscriber identity (TMUI/TMSI). A TMUI has significance only in the location area in which the user is registered. Outside that area it should be a accompanied by an appropriate location area identity (LAI) in order to avoid ambiguities. The association between the permanent and temporary user identities is maintained in the SN/VLR in which the user is registered.

In order to prevent user traceability, i.e., the property that an intruder may detect whether several services are delivered to the same person, the visited SN should frequently update the temporary user identity of a user. This is performed using the TMUI allocation procedure shown in Figure 5-1 and after the initiation of ciphering.

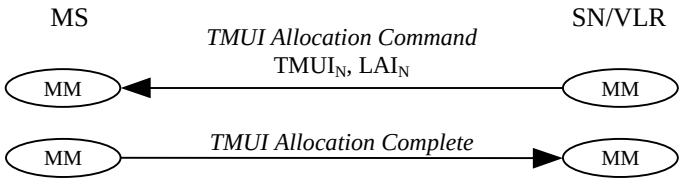

Figure 5-1: TMUI Allocation (see [3])

Note that to achieve user identity confidentiality and avoid user traceability in addition it is required that any signalling or user data that might reveal the user's identity is ciphered on the radio access link.

After a successful TMUI allocation the user stores TMUI/LAI and the SN/VLR stores the association between TMUI/LAI and the IMUI. The current rule is that at location updates in a newly visited VLR_N, the VLR_N first attempts to retrieve the IMUI (and other user data such as unused authentication vectors)

[1] Throughout the text, the mechanisms for user identity confidentiality are described using the terminology for the circuit-switched part of the GSM/UMTS architecture. An identical mechanism is in place for the packet-switched part of the architecture, however, with different names for similar aspects. The following terms should be replaced: SN/VLR → SN/SGSN, TMSI → P-TMSI, LAI → RAI.

from the SN/VLR_O that controls the location area LAI_O. However, an alternative mechanism is required for identification of the user by a SN/VLR that does not have the valid TMUI.

1) In the case of ***mobile-originated*** connection establishments the SN/VLR has no valid TMUI when:

   a) The VLR_N sees that LAI_O is controlled by a VLR_O that is part of a different SN, or a VLR_O with which the VLR_N does not exchange user data (it might be the SN policy not to exchange user data among VLRs, e.g. in a super-charged SN [12]).

   b) The VLR_N cannot associate an IMUI with the $TMUI_O$ because the VLR_N (or the VLR_O) database is corrupted or has been in recovery, or the $TMUI_O$ has been corrupted when sent over the radio access link.

   In each of the above cases, [3] specifies that the SN/VLR initiates the identity request procedure. The GSM-like UMTS identity request procedure is shown in Figure 5-2.

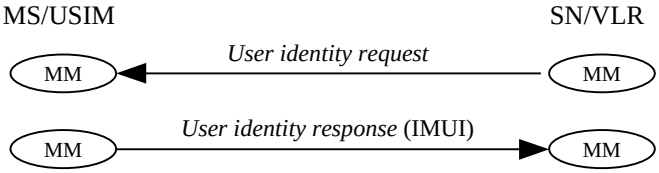

Figure 5-2: User identification procedure with GSM-like response

   This procedure is still allowed in UMTS, but it represents a flaw in the provision of user identity confidentiality. To allow the HE to mend this hole for its user's, a transport mechanism is described in 5.1.3.

2) In the case of ***mobile-terminated*** connection establishment the SN/VLR has no valid TMUI when:

   a) The SN/VLR database has been in recovery. In this case the SN/VLR has no TMUI. The TMUI in the SN/VLR_N database is corrupted or the TMUI sent to the user is corrupted. In this case the SN/VLR has an invalid TMUI.

   b) The SN/VLR_N has not received a TMUI allocation complete message from the user. In this case the SN/VLR maintains an association between the IMUI and two TMUIs ($TMUI_N$ and $TMUI_O$).

   In each of the above cases, [3] specifies that the SN/VLR pages the user with the IMUI. For this procedure no alternative is provided as yet. Nevertheless, the use of the IMUI in cleartext on the radio access link in paging requests equally leads to an attack against the user identity confidentiality.

→ Section 6.1 describes the necessary countermeasures to always have an alternative for cleartext identity paging.

5.1.3 Transport mechanism for enhanced user identity confidentiality

This transport mechanism ([3], 6.2) allows the HE to protect its users against unauthorised disclosure of their identity, by implementing a mechanism for enhanced user identity confidentiality between the USIM and a HE network entity, referred to as the User Identification Centre (HE/UIC). The mechanism is shown in Figure 5-3.

The mechanism is initiated by the visited SN/VLR that requests the user to send its permanent identity. If the user has enhanced user identity confidentiality, he will not send his IMUI in cleartext, but include in his response a HE-message from which the HE/UIC should be able to retrieve the IMUI, and a HE/UIC-identity to allow the SN/VLR to route the HE-message to the user's HE/UIC. The SN/VLR forwards the HE-message to the HE/UIC which sends back the IMUI to the SN/VLR.

Therefore, the dialogue between the SN/VLR and the HE/UIC shall be included in the MAP authentication data request/response procedure as an option that is mandatory for implementation. Also the dialogue between the MS/USIM and the SN/VLR shall be included in the MM identity request/response procedure.

A concern was raised as whether this procedures met the requirements on lawful interception, as the SN will now have to rely on the willingness of possibly a foreign HE to provide the IMUI. It will be impossi-

ble to derive the IMUI locally from the MS. However, although it is required that the HE provides the IMUI, it is not required to leak to the HE that the user data or its identity is being intercepted, as the interrogation of the HE/UIC is part of the normal procedures, e.g., when a user first registers in a SN or after database failures in the SN/VLR.

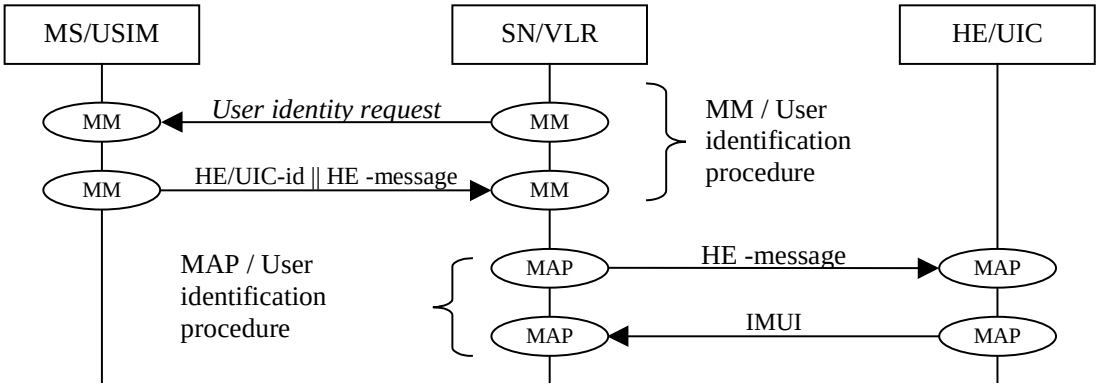

Figure 5-3: Transport mechanism for enhanced user identity mechanism

→ Section 6.1 describes the necessary modification to the network protocols to always have an alternative for cleartext identity paging.

5.1.4 Example mechanism using group keys

This mechanism ([3], annex B) allows the identification of a user on the radio access by means of the permanent user identity encrypted by means of a group key. The mechanism makes use of the transport mechanism described in 5.1.3.

The mechanism assumes that the user belongs to a user group with group identity GI. Associated to the user group is a group key GK which is shared between all members of the user group and the user's HE/UIC. The mechanism is shown in Figure 5-4.

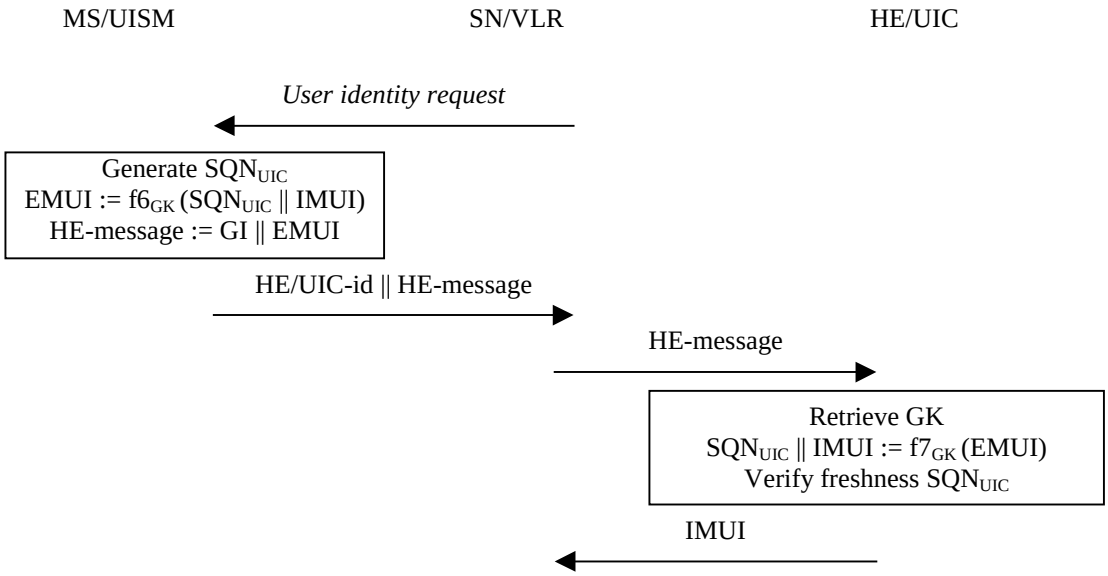

Figure 5-4: Mechanism for enhanced user identification using group keys

Upon receipt of the user identity request from the SN/VLR, the MS/USIM generates a fresh sequence number SQN_{UIC}, encrypts the sequence number SQN_{UIC} and the user's identity IMUI with the enciphering

algorithm f6 and the user's group key GK and constructs the HE-message that consists of the group identity GI and the encrypted user identity EMUI.

Upon receipt of the HE-message, the HE/UIC retrieves the group key GK associated with the group identity GI from its database and decrypts the EMUI with the deciphering algorithm f7 ($f7 = f6^{-1}$) and the group key GK so that it retrieves SQN_{UIC} and IMUI. If the SQN_{UIC} is verified to be fresh, the HE/UIC sends the IMUI to the visited SN/VLR.

→ Section 6.1 describes the necessary addition to the group key mechanism to always have an alternative for cleartext identity paging.

5.2 User authentication and access link key agreement

5.2.1 Outline

The mechanism described here achieves mutual authentication by the user and the network showing knowledge of a permanent secret key K which is shared between and available only to the USIM and the HE/AuC in the user's HE. In addition the MS/USIM and the HE/AuC keep track of one or several counters SQN_{MS} and SQN_{HE} respectively to support network-to-user authentication.

The method was chosen in such a way as to achieve maximum compatibility with the current GSM security architecture and facilitate migration from GSM to UMTS. The method is composed of a challenge/response protocol identical to the GSM subscriber authentication and key establishment protocol combined with a sequence number-based one-pass protocol for network authentication derived from the ISO standard ISO/IEC 9798-4 (section 5.1.1) [13].

The mechanism consists of the following procedures and functionality:

- A procedure to distribute authentication vectors from the HE/AuC to the SN/VLR and the functionality to generate authentication vectors in the HE/AuC. This is discussed in 5.2.3.
- The generation of authentication vectors itself is discussed in 5.2.4.
- A procedure to distribute authentication vectors among SN/VLR and rules for authentication vector management in the SN/VLR. This is discussed in 5.2.5.
- A procedure to authenticated the user and agree on new access link keys between the SN/VLR and the MS/USIM and the functionality in the MS/USIM and the SN/VLR to verify the authentication parameters. This is discussed in 5.2.6.
- The verification of authentication vectors in the USIM is discussed in 5.2.7.
- A procedure to synchronise the counters in the HE/AuC and in the MS/USIM, i.e., a mechanism to prevent a semi-permanent lock-out of the MS/USIM in the case of a corruption of the AuC database. This is discussed in 5.2.8.

5.2.2 Overview

An overview of the mechanism is shown in Figure 5-5. Upon receipt of a request from the SN/VLR, the HE/AuC sends an ordered array of *n* authentication vectors (AV) (the equivalent of a GSM "triplet") to the SN/VLR. See 5.2.4 for the generation of authentication vectors by the HE/AuC. Each authentication vector consists of the following components: a random number RAND, an expected response XRES, a cipher key CK, an integrity key IK and an authentication token AUTN. Each authentication vector is good for one authentication and key agreement between the SN/VLR and the USIM.

When the SN/VLR initiates an authentication and key agreement, it selects the next authentication vector from the array and sends the parameters RAND and AUTN to the user. The USIM checks whether AUTN can be accepted and, if so, produces a response RES which is sent back to the SN/VLR. See 5.2.7 for the verification of AUTN in the MS/USIM. The MS/USIM then also computes CK and IK. The SN/VLR compares the received RES with XRES. If they match the SN/VLR considers the authentication and key agreement exchange to be successfully completed. The established access link keys CK and IK will then

be transferred by the USIM and the SN/VLR to the entities which perform ciphering and integrity functions (for ciphering that are respectively the UE and the RNC).

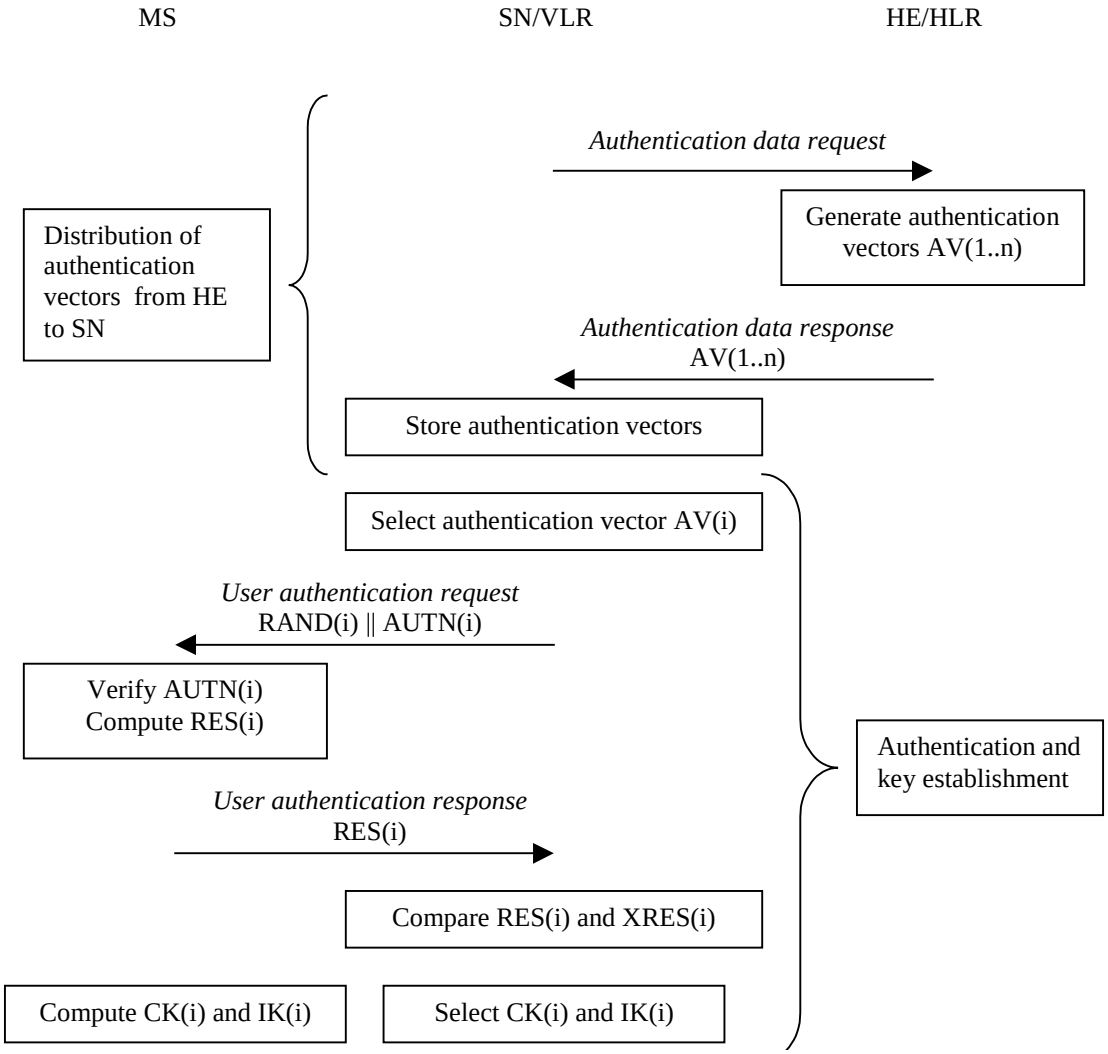

Figure 5-5: User authentication and access link key agreement

5.2.3 Distribution of authentication vectors from HE/AuC to SN/VLR

The purpose of this procedure is to provide the SN/VLR with an array of fresh authentication vectors from the user's HE to perform a number of user authentications and access link key agreements.

Figure 5-6 shows the protocol used to distribute authentication vectors from the HLR to the VLR. When the user is known in the SN/VLR by means of an IMUI, the IMUI is used as the "User-id" in the authentication data request, and no "User-id" is required in the response. If the user is known in the SN/VLR by means of a HE-message, the HE-message is used as the "User-id" in the authentication data request, and the IMUI and a $TMUI_{HE}$ are sent as "User-id" in the authentication data response (see also 5.1.3).

The parameter MODE is used to distinguish between several SN nodes that need to use arrays of authentication vectors in parallel. In the April release MODE can take two values and can distinguish between CS and PS core network nodes. The authentication vectors AV[MODE;1..n] returned by the HE/HLR are valid only for the mode indicated in the request.

Note: To support roaming users as well as certain types of super-charged serving networks [12] this mechanism should be extended such that several serving networks or VLRs can store and use arrays of authentication vectors (see also 5.2.7).

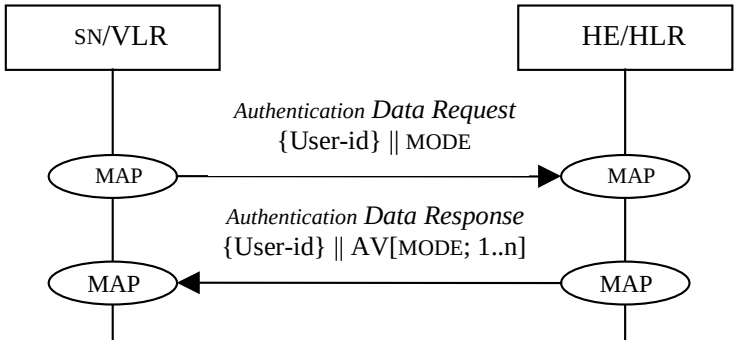

Figure 5-6: Distribution of authentication vectors from HE/HLR to SN/VLR

5.2.4 Generation of authentication vectors in the HE/AuC

Figure 5-7 shows the generation by the HE/AuC of an authentication vector (AV) characterised by the following parameters:

- it is intended for the user with permanent secret key K;
- it is intended for use in the SN/VLR identified by MODE;
- it contains a sequence number SQN, derived from a counter in the HE/HLR;
- it contains a random challenge RAND, generated by the HE/AuC.

After the random challenge RAND is generated, the following elements of the authentication vector can be derived:

- an expected response XRES = $f2_K$ (RAND), where f2 is a (possibly truncated) data integrity function;
- a cipher key CK = $f3_K$ (RAND), where f3 is a key generating function ;
- an integrity key IK = $f4_K$ (RAND), where f4 is a key generating function ;
- an anonymity key AK = $f5_K$ (RAND), where f5 is a key generating function .

After the sequence number SQN is derived from the counter and the parameter MODE is known, the fifth element, the authentication token AUTN = Conc(SQN) || MODE || MAC-A, can be derived. It consists of:

- the concealed sequence number: Conc(SQN) = SQN ⊕ AK;
- the mode MODE;
- a data integrity code MAC-A = $f1_K$(SQN || RAND || MODE), where f1 is a data integrity function .

The anonymity key AK is used to conceal the sequence number sent in the authentication token. If the sequence number would be sent in cleartext, it might enable an intruder to link several authentication and services to the same user. Whether this is the case depends on how sequence numbers are generated.

The April release contains the following mechanisms for sequence number generation:

- **Individual and per mode counters.** The HE/AuC maintains a counter $SQN_{HE/MODE}$ per user and per mode. Normally this sequence number is equal to the highest SQN included in an authentication token for that user and mode. Each time a sequence number is generated, the counter is incremented by one and the sequence number is set to the new counter value. This mechanism is currently in the main body of [3].

– **Global counter.** The HE/AuC maintains a global counter SQN_{HE} for all users and modes. This counter may be derived from a clock value. This mechanism is currently in an annex of [3].

Note: The range of the sequence number should be large enough, such that it is very unlikely that a counter would roll-over during the lifetime of a USIM. The June release of the cryptographic algorithm requirements [5] lets the HE choose a size between 32 and 64 bits. If individual counters are used, a range of 2^{32} is certainly several orders of magnitude larger than the expected number of authentications during the lifetime of a USIM. For a time-derived unique counter that starts in 2000 and that increments every second, a 32 bit counter would run until the year 2136, which is also far enough in the future.

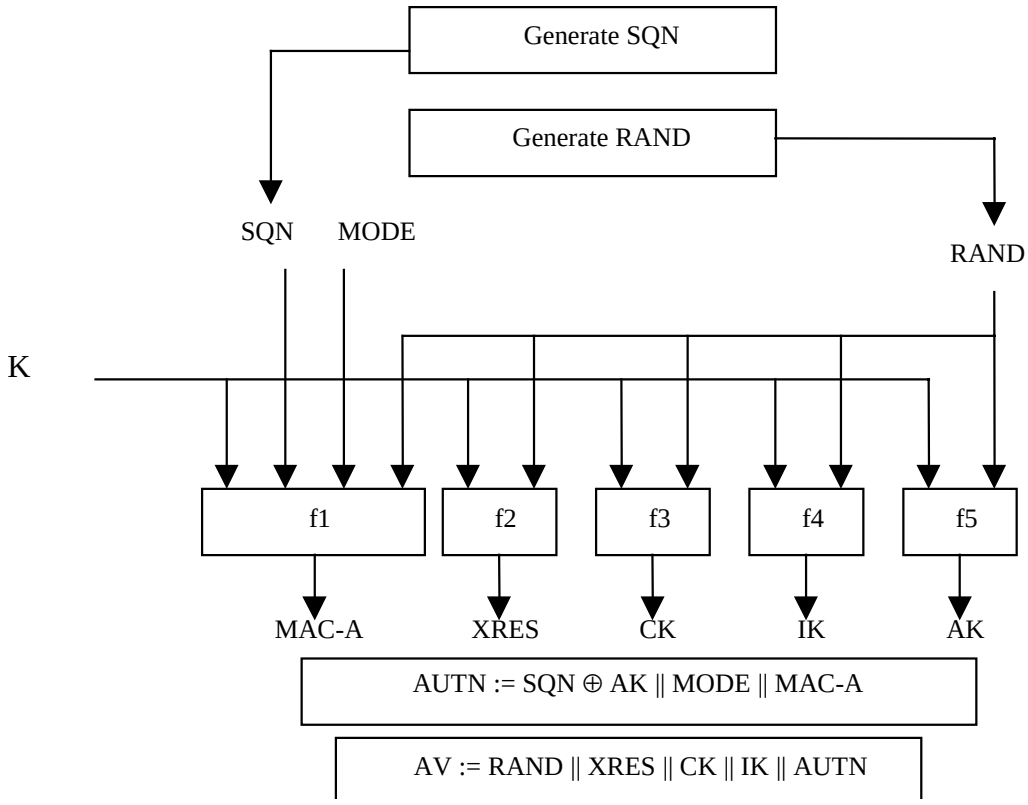

Figure 5-7: Generation of an authentication vector in the HE/AuC

The generation of the sequence numbers in the HE/AuC in combination with the verification of the sequence number in the MS/USIM assure the user the freshness of the authentication vectors used. Generally, however, these mechanisms may lead to a long-time lock-out of an MS/USIM if the counter in the HE/AuC (for reasons such as database failures, etc.) is set to a smaller value than the counter value in the MS/USIM (note that normal operation assures that $SQN_{HE} \geq SQN_{MS}$). Therefore, a re-synchronisation mechanism is included, described in 5.2.8.

5.2.5 Management and distribution of authentication vectors in the SN

The purpose of this procedure is to provide a newly visited VLR_N with unused authentication vectors from a previously visited VLR_O.

The procedure is invoked by the SN/VLR_N after the receipt of a location update request from the user. The parameter "User-id" is thus the $TMUI_O/LAI_O$ pair that was used by the user. In the authentication data response, the SN/VLR_O includes the IMUI as "User-id" and sends the unused authentication vectors to the SN/VLR_N.

When a user changes from one VLR to another one and the new VLR requests remaining authentication vectors from the old VLR then the old VLR shall not retain any copies of these authentication vectors.

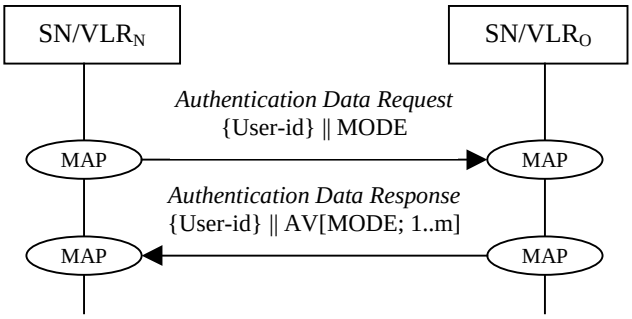

Figure 5-8: Distribution of authentication vectors from SN/VLR$_O$ to SN/VLR$_N$

The current rule is that when a VLR receives a "cancel location" request for a certain user it shall delete all authentication vectors relating to that user.

Equivalently, when a SN/VLR receives a location update request from a user and the SN/VLR notices that authentication vectors relating to that user are still stored in the SN/VLR, it will delete this information and request fresh authentication vectors from the AuC.

5.2.6 User authentication between SN/VLR and MS/USIM

The purpose of this procedure is to authenticate the user and establish a new pair of cipher and integrity keys between the SN/VLR and the MS/USIM. During the authentication, the user verifies the freshness of the authentication vector that is used.

The procedure is by the SN/VLR that select the next unused authentication vector from the SN/VLR data-base. The user sends back a response that is either of the following:

- ACCEPT. The user has successfully verified the data origin, data integrity of the freshness of the authentication token. The parameter RES is included for user-to-network authentication.
- REJECT. The user has not successfully verified the data origin and data integrity of the authentication token. No parameter RES is sent.
- FAILURE. The user has successfully verified the data origin and data integrity of the authentication token, but not the freshness. No parameter RES is sent.

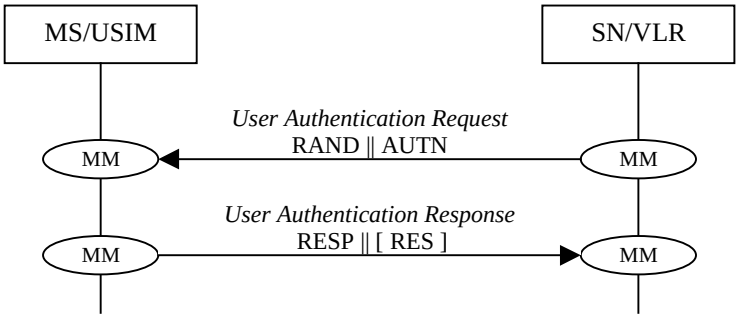

Figure 5-9: User authentication between SN/VLR and MS/USIM

The procedure is terminated successfully only if the user has accepted the authentication token and the SN/VLR has successfully verified that RES = XRES. In case the MS/USIM has not accepted the authentication vectors the HE/HLR is informed. If the procedure is not successfully terminated, the SN/VLR shall inform the HE/HLR.

Unlike in GSM authentication vectors cannot be re-used by the SN/VLR. A concern was raised whether this might lead to a temporary lockout when communication between the SN/VLR and the HE/HLR would be impossible. However, SN/VLRs can offer secure service even when HE/AuC links are unavailable by allowing them to use previously derived access link keys for a user so that a secure connection can still be

set up without the need for an authentication and key agreement. Authentication is in that case based on a shared integrity key, by means of data integrity protection of signalling messages.

Furthermore, the SN/VLR should use the authentication vectors in the order in which they are generated by the HE/AuC and sent to the SN/VLR. To achieve this a set of authentication vectors that is received in one MAP message should be treated as an ordered array. If the SN/VLR does not comply to this, some of the mechanisms for verification of the freshness will cause the USIM to reject the authentication vector, and even if this is not the case, it decreases the efficiency of other mechanisms to correctly assert the freshness of other authentication vectors[2].

5.2.7 Verification of the authentication token in the MS/USIM

Figure 5-10 shows the MS/USIM actions for user authentication and key agreement.

The MS/USIM first computes the anonymity key AK = $f5_K$ (RAND) and retrieves the sequence number in clear text SQN = (SQN ⊕ AK) ⊕ AK.

Next the user computes XMAC-A = $f1_K$ (SQN || RAND || MODE) and compares this with MAC-A which is included in AUTN. If they are different, the MS/USIM terminates unsuccessful, and the MS/USIM sends a response back to the SN/VLR with the indication "REJECT".

Next the user verifies whether SQN is acceptable. The April release already points to different mechanisms available to the MS/USIM. All of these tests are sufficient but not necessary conditions for freshness of the sequence number. Those mechanisms are:

- **Mode.** For each mode the MS/USIM keeps track of one counter: $SQN_{MS/CS}$ for authentications with CS core network nodes, and $SQN_{MS/PS}$ for authentications with PS core network nodes. To verify the freshness of the sequence number SQN, the MS/USIM compares the received SQN with $SQN_{MS/MODE}$. If SQN > $SQN_{MS/MODE}$ the MS accepts SQN and sets $SQN_{MS/MODE}$ to SQN such that $SQN_{MS/MODE}$ represents the highest SQN ever seen (and accepted) by the MS/USIM. This mechanism is included in the main body of [3].
- **Window.** The USIM keeps track of a counter SQN_{MS} which equals the highest sequence number the USIM has seen and accepted, and an array of Boolean values that indicates for each sequence number in the interval [SQN_{MS} − w, SQN_{MS}) whether the USIM has already seen it. The USIM then accepts all sequence numbers SQN > SQN_{MS} and also all SQN ∈ [SQN_{MS} − w, SQN_{MS}) that it has not seen before. After accepting a sequence number, the counter value and the window are updated accordingly. This mechanism is included in annex C of [3].
- **List.** The USIM keeps track of an ordered list {$SQN_{MS,i}$ > $SQN_{MS,j}$ | 1 ≤ i < j ≤ k} with the k highest sequence numbers the USIM has seen and accepted. The USIM accepts a sequence number if it ranks among the k highest and has not been seen before. This mechanism is included in annex C of [3].

The latter two mechanisms are very similar. The window mechanism is appropriate when individual counters per user are used. When a global counter is used, the list mechanism should be used instead.

All three methods are a loosening of the initial test to let the USIM maintain a single counter value SQN_{MS} and check whether SQN > SQN_{MS}. Such a test works fine as long as no CN nodes simultaneously use arrays of authentication vectors. However, this is definitely the case for CS and PS core network nodes.

Only the first mechanism, a separate counter per mode, allows simultaneous usage in CS and PS nodes of authentication vectors without the risk of a fresh authentication vector being rejected by the user. The one counter per mode mechanism and the test SQN > $SQN_{MS/MODE}$ has therefore become the working assumption and is included in the main body. However, the "window" or the "list" mechanism may be used in addition to the mode mechanism, to increase the efficient use of authentication vectors when users are roaming or when they are registered in super-charged serving networks [12].

[2] In the window mechanism (in [3], annex C.3) the out-of-order usage of authentication vectors would make the top value (an the entire window) increment earlier. The list mechanism (Ibid., annex C.4) is not affected by out-of-order use of authentication vectors.

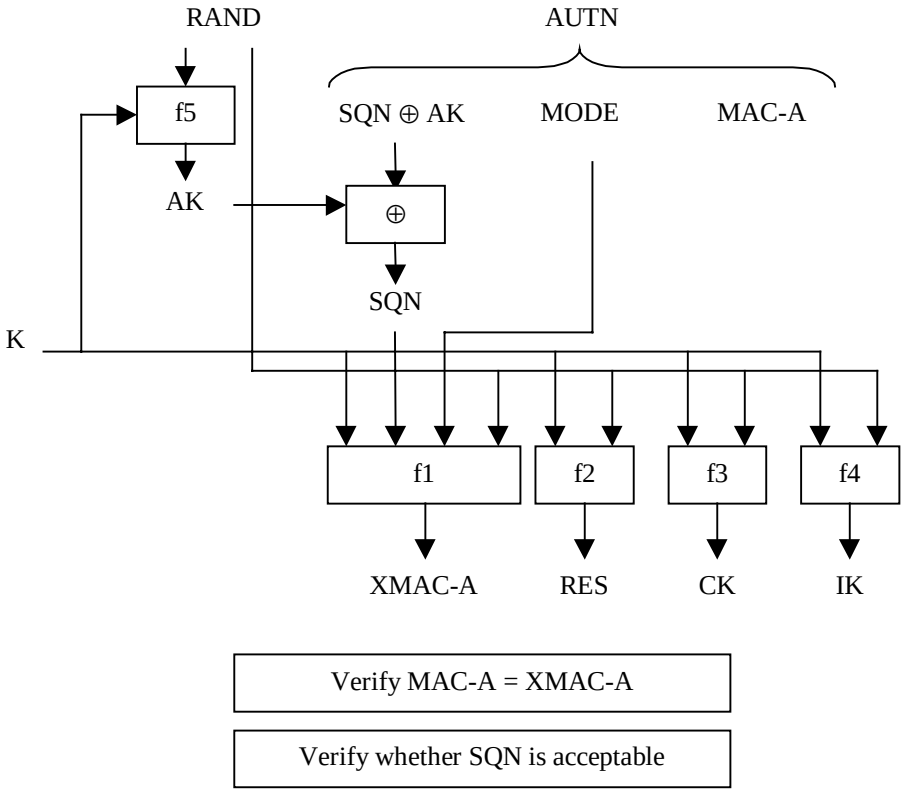

Figure 5-10: User authentication in the USIM

Note that the mode mechanism (the current working assumption) always rejects authentication vectors when the SN/VLR does not use them in order of receipt. But even in the window mechanism, the use of authentication vectors would advance the window too soon. Only the performance of the list mechanism is not influenced by the out-of-order use of authentication vectors (at least if the list size is larger than the batch size).

All three methods may however lead to USIM lock-up, in the event of a serious malfunction of the HE/AuC or a malicious attack on the HE/AuC. Such an event is expected to be very unlikely if proper precautions are taken. Nevertheless, in view of the consequences that this event may have (USIM lock-up or unlimited use of stolen authentication vectors) it may be desirable to have countermeasures available also for this event.

→ Section 6.3 presents an alternative optional scheme for the management of sequence numbers that gives protection against lock-up of a MS/USIM.

5.2.8 Synchronisation of the counters in the MS/USIM and the HE/AuC

The purpose of this procedure over MM and MAP is to synchronise a individual counter SQN_{HE} in the HE/AuC whose value has been lost or has been corrupted.

The procedure is initiated by the HE/HLR (or HE/AuC) that sends a synchronisation request to the USIM via the SN/VLR, with an indication of the required MODE. Upon receipt of that request the user sends the authentication parameters with he highest sequence numbers that it has accepted: $RAND_{MS}$, $AUTN_{MS}$ for that MODE.

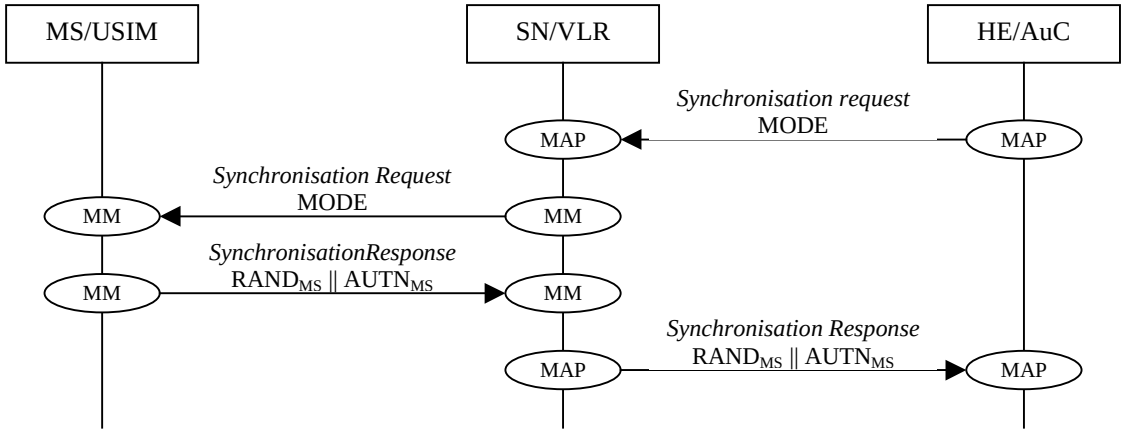

Figure 5-11: Procedures for synchronisation of the counter in the HE/AuC

Upon receipt of the response, the HE/AuC retrieves the sequence number in cleartext and verifies its data integrity, in a similar way as the USIM did (see 5.2.7). It then sets its counter value SQN_{HE} to the received value SQN_{MS} only if $SQN_{MS} > SQN_{HE}$, i.e., the HE/AuC allows its counter value to be set to a higher value, never to a smaller value.

The mechanism described here requires additional messages to be exchanged on the interface between SN/VLR and HE/HLR.

→ Section 6.2 contains an alternative implementation of the re-synchronisation procedure using piggy backing on existing messages that avoids additional messages and doing so provides a more efficient implementation.

The mechanism described here does not provide assurance to the HE/AuC of the accurateness of the SQN_{MS}, i.e., the HE has no assurance that the value it receives is the actual value stored in the MS/USIM now, and not a replay of a previous value.

→ Section 6.4 presents a modification of the mechanism that provides assurance of accurateness and freshness of the USIM counter value.

The current text here and in [3] may suggest that the re-synchronisation procedure should be used to re-synchronise after recovery of the transient HE/AuC database. A perfectly valid concern was raised that this would lead a sudden and large load on the system.

→ Section 6.5 presents an alternative to the re-syncrhonisation procedure.

5.3 Data confidentiality

The April release of the security architecture ([3], 6.6) only contains the high-level mechanism for data confidentiality on the network access link. It specifies that the encryption will be based on a symmetric cipher, that uses a secret cipher key CK that is established using the mechanism for authentication and key agreement (see 5.2). It also specifies that the encryption functions for both the signalling and user data are allocated to the UE at the user side and at the RNC at the network side. Access link encryption will thus be extended further into the network, to protect also the links between the base stations and the RNCs which may be radio links. However, the April release does not go into detail on the integration of the ciphering in the radio access network architecture (except that it will be applied on a higher layer than the physical layer because the physical layer does not extend to the RNC) and many other open issues such as the synchronisation of the cipher stream generation, the negotiation of the ciphering algorithm and the cipher key selection (selection between the cipher keys provided by two core networks connected to one UTRAN, in case of separate mobility management), which messages start ciphering and how the cipher key lifetime is monitored and controlled by both parties is not specified.

5.4 Data integrity

Equally, the April release of the security architecture ([3], 6.4) only contains the high-level description of the mechanism for data integrity of signalling messages on the network access link. It specifies that the encryption will be based on a data integrity code, that uses a secret integrity key IK that is established using the mechanism for authentication and key agreement (see 5.2).

Almost all other aspects were still open for discussion, as there are the allocation of the integrity functions, the mechanism to prevent replay of old signalling messages and data integrity codes, the integrity mechanism negotiation, the integration in the (UTRAN) architecture and even which messages that require protection.

5.5 Network-wide encryption

5.5.1 Introduction

If network-wide confidentiality of user traffic is provided we assume that access link confidentiality of user traffic between UE and RNC will be replaced with the network-wide service. However, we note that access link confidentiality of signalling information and user identity between UE and RNC will be applied regardless of whether the network-wide user traffic confidentiality service is applied or not.

The provision of an network-wide confidentiality service in UMTS has an obvious impact on lawful interception. We assume that the same lawful interception interface is required in UMTS as in second generation systems regardless of whether network-wide confidentiality is applied by the network or not. Thus, we assume that it must be possible to remove any network-wide confidentiality protection within the core network to provide access to plaintext user traffic at the lawful interception interface.

We assume that network-wide confidentiality will be provided by protecting transmissions on user traffic channels using a synchronous stream cipher. This will involve the specification of a standard method for ciphering user traffic on an end-to-end basis and a standard method for managing the ciphering key required at the end points of the protected channel.

5.5.2 Ciphering method

It is assumed that the network-wide encryption algorithm shall be a synchronous stream cipher similar to the access link encryption algorithm. Indeed, it would be desirable to use the same algorithm for access link encryption and for network-wide encryption.

The network-wide synchronous stream cipher shall contain a key stream generator which shall have (at least) two inputs: the end-to-end cipher key (Ks) and an initialisation value (IV). The plaintext shall be encrypted using the key stream by applying an exclusive-or operation to the plaintext on a bit per bit basis to generate the ciphertext. The decryption operation shall involve applying the same key stream to the ciphertext to recover the plaintext.

Synchronisation of the key stream shall be achieved using the initialisation value. Synchronisation information shall be available at both end points of the communication and shall be used to maintain alignment of the key stream. For example, it might be necessary to transmit explicit end-to-end synchronisation frames with the user traffic at certain intervals. Alternatively, it might be possible to use some existing frame structure for network-wide encryption synchronisation purposes. The frequency at which synchronisation information must be made available at each end to ensure reliable transmission will depend on the exact nature of the end-to-end user traffic channel.

Protection against replay of user traffic shall be achieved through the use of a time variable initialisation vector combined with a time variable cipher key. If the same cipher key is used in more than one call then it may be necessary to include a third input to the key stream generator such as a call-id or a time-stamp to protect against replay of the whole call. Note that the stream cipher does not protect against bit toggling so other mechanisms must be used if this type of integrity protection is required on user traffic.

For encryption of voice traffic we assume that Transcoder Free Operation (TFO) is used between the two end points such that the structure and ordering of the transmitted data is maintained with the same boundary conditions at each end of the link. Note that in the initial phases of UMTS, transcoder free operation may only be possible for user traffic channels which terminate within the same SN. Furthermore, TFO may only be possible if the entire communication path is within the same SN. Thus, in non optimal routing cases where the tromboning effect occurs, TFO may not be available, even if the traffic channel terminates within the same SN.

For encryption of data traffic we assume that a transparent data service is used between the two end points such that the structure and ordering of transmitted data is maintained with the same boundary conditions at each end of the link.

To satisfy lawful interception requirements it must be possible to decrypt end-to-end encrypted traffic within the core network to provide access to plaintext user traffic. Thus decryption facilities (and the end-to-end encryption key) must be available in the core network for lawful interception reasons. Note also that if transcoder free operation is used on voice traffic channels, transcoders must be available in the core network for lawful interception reasons whether network-wide encryption is provided or not.

Issues for further study:

- Specification of encryption synchronisation mechanism;
- Adaptation of TFO voice traffic channels for network-wide confidentiality;
- Adaptation of data traffic channels for network-wide confidentiality;
- The ability to terminate network-wide encryption at network gateways for inter-network user traffic channels;
- The ability to handle multiparty calls, explicit call transfer and other supplementary services;
- Network-wide encryption control – algorithm selection, mode selection, user control

5.5.3 Key management

5.5.3.1 General case

We assume that signalling links within the network are confidentially protected on a link-by-link basis. In particular, we assume that the UE to RNC signalling links are protected using access link security domain keys (see clause 6). We also assume that VLR to RNC signalling links and core network signalling links are protected using network security domain keys (see clause 7). Note that if network-wide encryption can be provided across SN boundaries (e.g. because inter-network TFO is available) then the signalling links requiring protection will cross network boundaries. In this situation it is important to note that the two SNs may not be roaming partners yet they still must be able to confidentially protect inter-network signalling by establishing appropriate keys.

The key management scheme for network-wide encryption involves establishing an end-to-end session key between the end points of the traffic channel. It should not be possible to obtain this key by eavesdropping on any transmission links within the network. However, it may be possible to obtain the end-to-end key by compromising certain nodes within the network (e.g. nodes where link encryption terminates).

To satisfy lawful interception requirements it must be possible to decrypt end-to-end encrypted traffic within the core network to provide access to plaintext user traffic. Thus, the end-to-end encryption key (and decryption facilities) must be available in the core network for lawful interception reasons.

Issues for further study:

- Specification of key management scheme for the general case;
- The ability to terminate network-wide encryption key management at network gateways for inter-network user traffic channels.

5.5.3.2 Outline scheme for intra-SN case

In this case we make the following assumptions:

- Two UEs registered on the same SN wish to set up an network-wide confidentiality protected call
- The appropriate user traffic channel for encryption can be established between the two UEs
- During connection establishment, the appropriate control information is transmitted to the called party indicating that the incoming connection is end-to-end encrypted.
- During connection establishment, the appropriate control information is transmitted to the relevant VLRs (or other core network entities) indicating that the connection being established is end-to-end encrypted.
- The keys Ka and Kb used to derive the end-to-end session key shall not be used for access link encryption of other data, nor for the derivation of end-to-end session keys with other parties.

The key management scheme is illustrated in the diagram below.

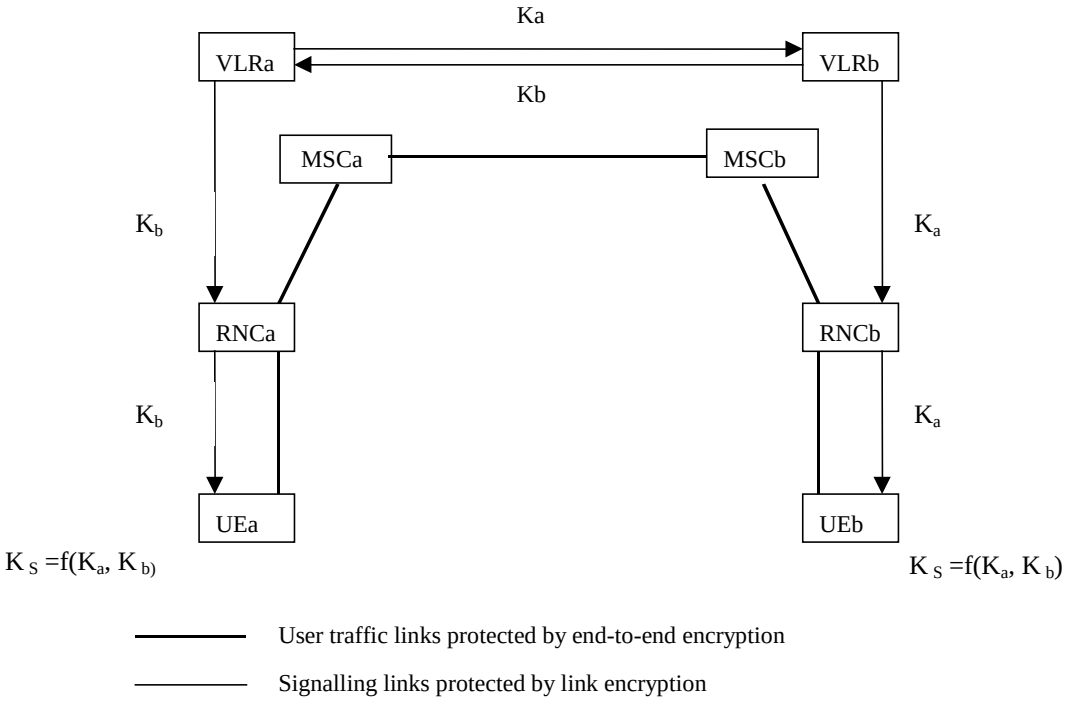

Figure 5-12: Key management scheme for network-wide encryption

In this scheme VLRa and VLRb exchange access link cipher keys for UEa and UEb. VLRa then passes Kb to UEa, while VLRb passes Ka to UEb. At each end the access link key is transmitted to the UE over protected signalling channels (which may be protected using different access link keys Ka' and Kb'). When each UE has received the other party's access link key, the end-to-end session key Ks is calculated as a function of Ka and Kb.

This key management scheme satisfies the lawful interception requirement since Ks can be generated by VLRa or VLRb and then used by decryption facilities in the core network to provide plaintext user traffic at the lawful interception interface.

Issues for further study:

- The exact mechanism by which the VLRs exchange access link keys during connection set up.

5.5.3.3 Variant on the outline scheme

VLRa and VLRb mutually agree Ks over a secure signalling link using an appropriate key agreement protocol. VLRa then passes Ks to UEa and VLRb passes Ks to UEb.

Note: As opposed to the scheme in 5.5.3.2, the access link keys Ka and Kb could be used for access link encryption of other data.

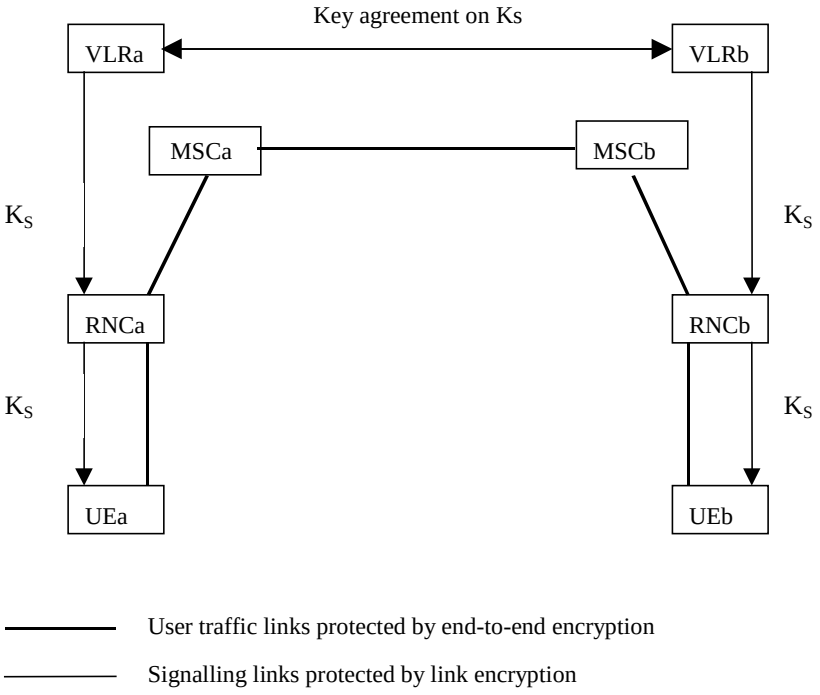

Figure 5-13: Key management scheme for network-wide encryption

6. Further proposals on open issues

6.1 Alternative for clear-text permanent identity paging

6.1.1 Introduction

In this section we discuss the countermeasures that are required to prevent attacks against the confidentiality of the user identity exploiting the clear-text permanent identity paging procedure for those users that have enhanced user identity confidentiality. To prevent these attack it is required that an alternative identity is available to the SN/VLR in each of the situations where IMUI paging is used (see 5.1.2).

A first situation is when the SN/VLR has not received a TMUI allocation complete message. Then, the SN/VLR has two TMUI stored for the user and normally one of them is valid. IMUI paging can thus easily be avoided by paging the user twice, once with each TMUI. If no TMUI is valid, the user may not be attached, or both TMUI may be corrupted. The SN/VLR should then use the mechanism described below.

After a VLR database recovery, or a corruption of the TMUI in the VLR database, the SN/VLR requires a second temporary identity by which it can page the user. This temporary identity has to be provided by the user's HE, otherwise it cannot be assured that it is available at the SN/VLR after a VLR recovery. For similar reasons, it cannot be provided to the SN/VR in advance. $TMUI_{HE}$ can be provided to the SN/VLR by either one of the following mechanisms:

1. a separate MAP dialogue that allows the SN/VLR to interrogate the HE/UIC, or
2. an optional parameter ($TMUI_{HE}$) that has to be included in every MAP service request message (for users with enhanced user identity confidentiality).

The $TMUI_{HE}$ that is provided by the HE/UIC should have a limited lifetime. If a $TMUI_{HE}$ would remain valid for a long time, an intruder who has linked a $TMUI_{HE}$ to the user's identity, would able to detect the presence of that user in a certain area. Therefore, it is proposed that a new $TMUI_{HE}$ is agreed between MS/USIM and HE/UIC each time the dialogue for enhanced user identity confidentiality is executed and that each time the user is paged by a $TMUI_{HE}$, a new $TMUI_{HE}$ is agreed between user and HE/UIC.

This mechanism has the disadvantage that it may lock out a user for incoming calls after a simultaneous VLR and HLR recovery (when also the data in the HE/UIC database is lost). Note however that the lock-out only applies until the next user-initiated connection establishment. Note further that after a simultaneous VLR and HLR recovery, the HLR possibly also has lost track of the user's location, such that it is not (only) the absence of a valid temporary identity for paging that prevents the network from establishing mobile-terminated services.

6.1.2 Implementation via a separate dialogue

A first implementation option requires the addition to MAP of the dialogue shown in Figure 6-1.

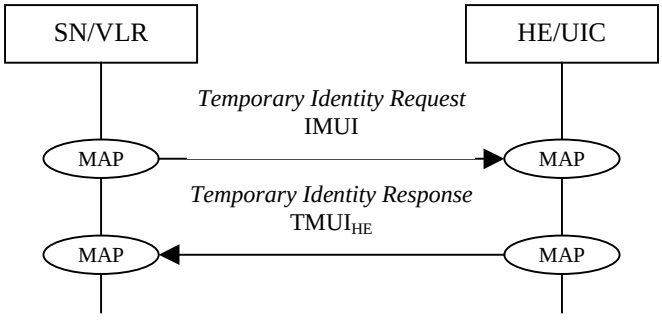

Figure 6-1: MAP procedure "Temporary identity request"

In addition, the SN/VLR should send an IMUI request to the user after he has paged him using $TMUI_{HE}$. The complete message flow for a mobile-terminated service request after a VLR recovery is shown in Figure 6-2.

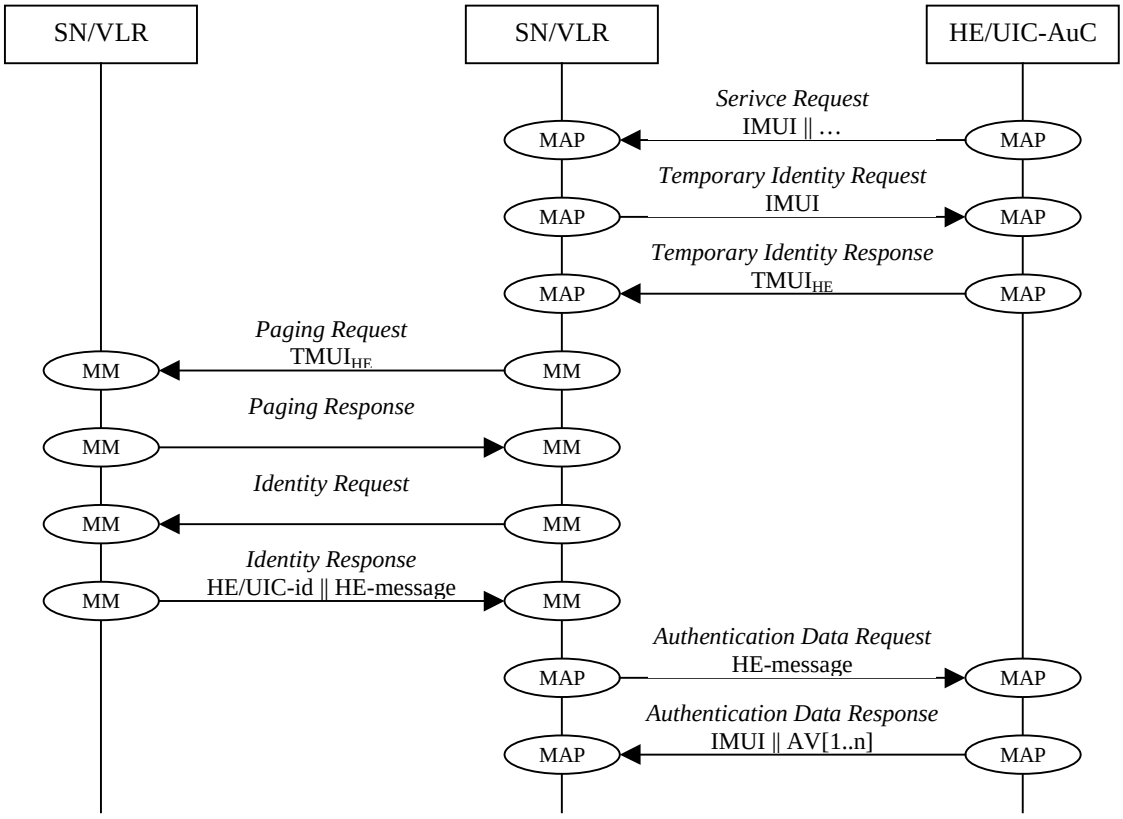

Figure 6-2: Mobile-terminated service establishment after VLR recovery

6.1.3 Implementation via piggy-backing

A slightly more efficient implementation might be arrived at as follows:

- Instead of creating a new MAP dialogue, $TMUI_{HE}$ is an optional parameter in the MAP/service request message that is included by the HE/UIC when the user has enhanced user identity confidentiality.
- Instead of initiating a new identity request by the SN/VLR, the HE-message is an optional parameter in the paging response message from the user that is included whenever a user is paged with a $TMUI_{HE}$.

In this way, the message flow for a mobile-terminated service request after a VLR recovery is reduced to the one shown in Figure 6-3.

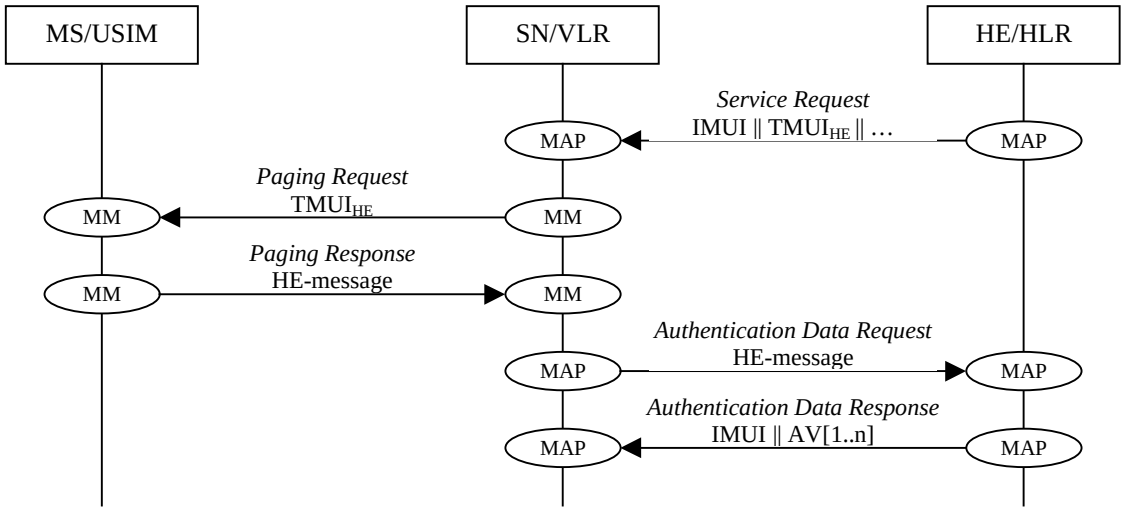

Figure 6-3: Optimised mobile-terminated service establishment after VLR recovery

6.2 Re-synchronisation of the SQN_{HE} counter by means of piggy-backing

6.2.1 Current situation – implementation via separate procedures

The April release of the security architecture [3] (see also 5.2.8) contains a re-synchronisation procedure that allows the HE/AuC to request the MS/USIM to send the current value of SQN_{MS} and to re-synchronise its counter SQN_{HE} to that value.

That procedure is typically triggered by a synchronisation failure detected by the USIM.

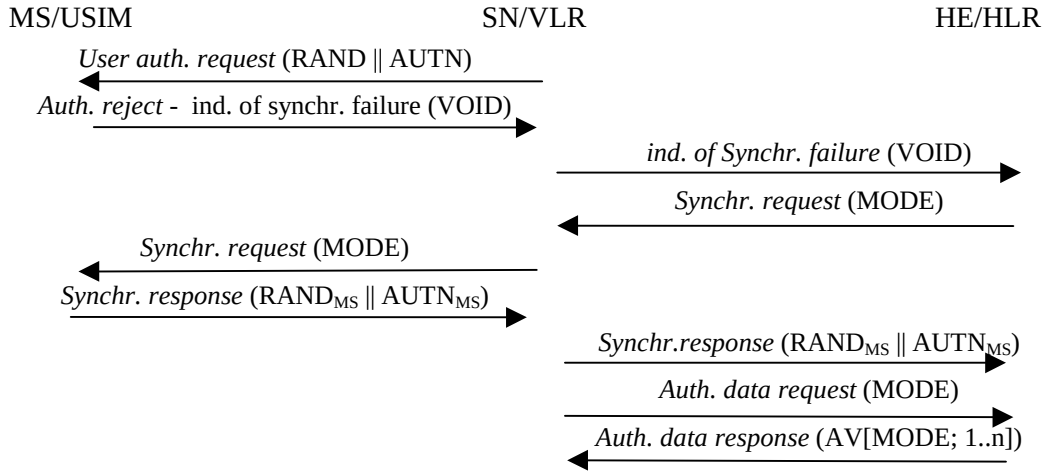

Figure 6-4: Complete message flow for re-synchronisation of the SQN_{HE} counter

This procedure can be optimised, as the information provided in the sixth, seventh and eighth message can be provided already in the second and third.

6.2.2 Proposal – implementation via piggy backing

Instead of the above mechanism it was proposed [17] and agreed by SA-3 to implement the re-synchronisation mechanism via piggy-backing of the required parameters on existing messages.

This is achieved via the following measures:

– If the MS/USIM detects a synchronisation failure, it sends the parameters for re-synchronisation along with the indication of a synchronisation failure in the user authentication response to the SN/VLR;

– The SN/VLR then forwards the indication of synchronisation failure and the parameters for re-synchronisation to the HE/HLR in an authentication data request.

In this way there is no need anymore for separate MM and MAP dialogues for re-synchronisation. If these changes are implemented, the message flow after a synchronisation failure is as shown in Figure 6-5.

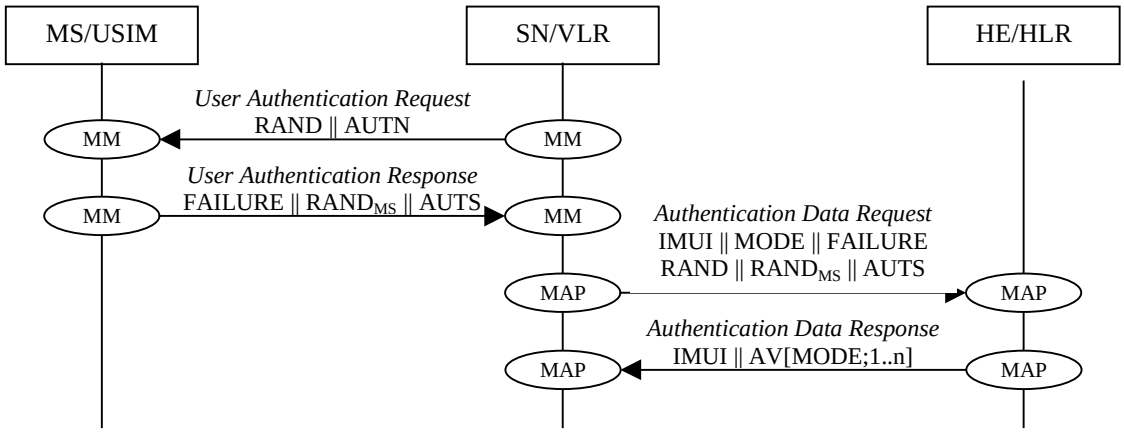

Figure 6-5: Re-synchronisation procedure using piggy backing

The parameters needed for re-synchronisation included in Figure 6-5 are those that are needed to assure the freshness of the value of SQN_{MS} that is sent to the HE/HLR according to the mechanism described in 6.4.2.

6.3 Management of sequence numbers

6.3.1 Current situation

In the April release of the security architecture ([3], 6.3.6), sequence numbers shall be sufficiently long so that the counter SQN_{MS} cannot reach its maximum value SQN_{max} during the lifetime of the system (in normal operations). We assume that failures in the USIM need not be taken into account. So, only a serious malfunction of the HE/AuC or a malicious attack on the HE/AuC can cause counters in the MS to reach their maximum value. Such an event is expected to be very unlikely if proper precautions are taken. Nevertheless, in view of the consequences that this event may have (USIM lock-up or unlimited use of stolen authentication vectors) it may be desirable to have countermeasures available also for this event. To this end, an alternative optional scheme for the handling of sequence numbers is proposed.

6.3.2 Proposed mechanism

The basic idea of the alternative sequence number handling is that the MS will not accept arbitrary jumps in sequence numbers.

The sequence number SQN is now accepted by the MS if and only if the following holds for some Δ: $SQN_{MS} < SQN \leq SQN_{MS} + \Delta$. This means that SQN_{MS} can reach its maximum value only after a minimum of SQN_{max}/Δ successful authentications have taken place.

Conditions on Δ:

(1) Δ shall be sufficiently large so that the MS will not receive any $SQN > SQN_{MS} + \Delta$ if the HE/AuC functions correctly.

(2) SQNmax /Δ shall be sufficiently large to prevent that SQN_{MS} ever reaches SQN_{max} during the lifetime of the USIM.

The choice of parameters, of course, depends on assumptions on the maximum number of authentications a USIM is to accept during its lifetime. $SQN_{max} = 2^{48}$ and $\Delta = 2^{24}$ ($\approx 16 \times 10^6$) appears to fulfil conditions (1) and (2). (Note that the range for SQN currently agreed is 32 to 64 bits.)

The choice of these values seems to be on the cautious side. Note that if $SQN > SQN_{MS} + 2^{24}$ would mean (in the absence of AuC failures) that 16 million authentication vectors were sent by the HE/AuC without being successfully received by the MS which is practically impossible.

For the counter SQN_{MS} to reach SQN_{max} it would again then take at least $SQN_{max}/\Delta = 2^{24} \approx 16$ million successful authentications. A denial of service attack requiring this amount of effort over the air may be deemed tolerable. Note also that a successful authentication requires write-operations on the EEPROM of a USIM. Current smart cards guarantee only about 100 000 read-write-cycles per EEPROM cell.

This mechanism for sequence number management was proposed to SA-3 in [18] and included in the security architecture specification.

6.4 Freshness assurance in the re-synchronisation procedure

6.4.1 Current situation

In the April release of the 3G security architecture ([3], 6.3.5), the MS includes a $RAND_{MS} \parallel AUTN_{MS}$ in the re-synchronisation response (in view of the agreed changes described in 6.2.2, now the authentication response) with $RAND_{MS} \parallel AUTN_{MS}$ whereby

$$AUTN_{MS} = SQN_{MS} \oplus AK_{MS} \parallel MODE \parallel MAC\text{-}A_{MS},$$

with $RAND_{MS}$ a random parameter, $AK_{MS} = f5_K(RAND_{MS})$ an anonymity key and

$$MAC\text{-}A_{MS} = f1_K(SQN_{MS/MODE} \parallel \mathbf{RAND_{MS}} \parallel MODE).$$

$AUTN_{MS}$ thus contains the value of SQN_{MS} for the appropriate mode (The reference to MODE is omitted further on to improve readability). The USIM has several ways to produce $RAND_{MS} \parallel AUTN_{MS}$. Either it stores and returns the latest received RAND ‖ AUTN pair, or it only stores the received RAND and recomputes AUTN, or it generates a RAND and computes the corresponding AUTN.

However, neither solution provides assurance of the freshness of SQN and/or MAC-A to the HE/AuC receiving the re-synchronisation response. Concern was raised that this could lead to a denial of service attack by a replay of a re-synchronisation response. Although the scope of such a denial of service attack was quite limited (forcing just an additional round of authentication) it is proposed here to modify the message sent by the MS because the required change is quite small and prevents the limited denial of service attack.

6.4.2 Proposal

The proposed change consists in using a different random number in the computation of the data integrity code in the parameters for synchronisation, namely the random number RAND received by the MS in the current authentication request.

In order to avoid interference with the MAC-A in the user authentication request it is proposed to use a message authentication function f1* to produce the data integrity code MAC-S. (It is expected that f1* may derived in a uniform fashion from a single MAC function , along with f1, ..., f5.)

Thus, AUTN is replaced by a new parameter

$$AUTS = SQN_{MS} \oplus AK_{MS} \parallel MODE \parallel MAC\text{-}S$$

whereby

$$MAC\text{-}S = f1^*_K(SQN_{MS/MODE} \parallel \mathbf{RAND} \parallel MODE).$$

Figure 6-6 shows the construction of the authentication token for re-synchronisation AUTS:

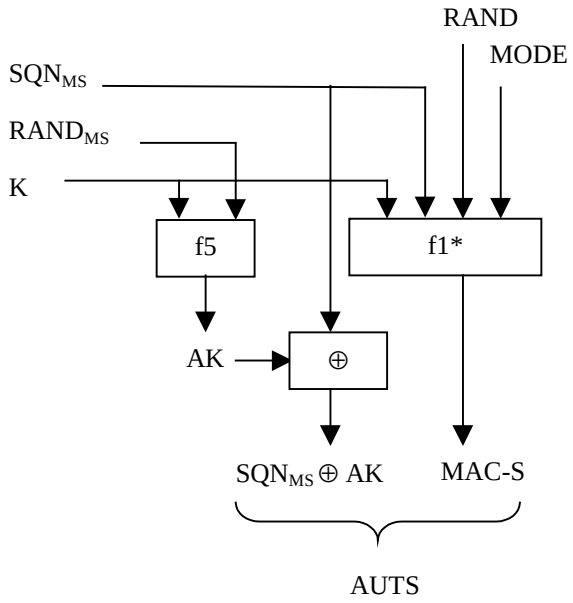

Figure 6-6: Construction of the re-synchronisation parameter AUTS

The idea behind the proposed change is that if the SN/VLR plays along the rules (i.e., does not replay RAND || AUTN pairs), RAND guarantees to the HE/AuC that MAC-S is accurate, and hence that also SQN_{MS} is accurate.

This yields then to the message flow shown in Figure 6-7:

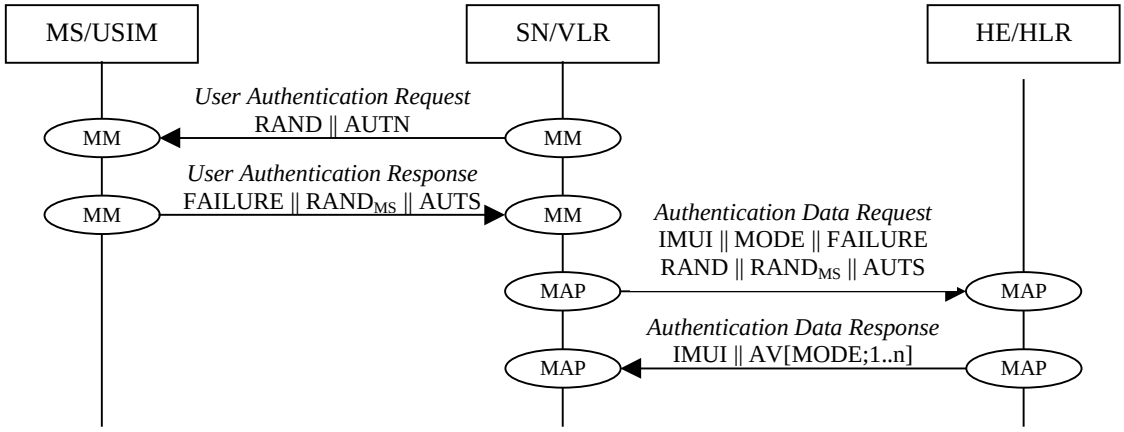

Figure 6-7: Transfer of SQN_{MS} to the HE/HLR with assurance of accurateness

When a user detects that the received SQN in a user authentication request is not in the correct range, the user computes AUTS from K, SQN_{MS}, RAND and $RAND_{MS}$ as shown in Figure 6-6 whereby RAND is the random challenge that was included in the last (and unsuccessful) authentication request, whereas $RAND_{MS}$ is the RAND which was included in the last successful authentication request. The MS then sends an authentication response message back to the network with an indication of a synchronisation failure and the parameters $RAND_{MS}$ and AUTS.

Upon receiving an authentication response with an indication of a synchronisation failure from the user, the SN/VLR sends an authentication data request with a “synchronisation failure indication” to the HE/AuC, together with the parameters RAND sent to the MS in the preceding user authentication request and $RAND_{MS}$ || AUTS received by the SN/VLR in the response to that request.

An SN/VLR shall not react to an unsolicited authentication response with a "synchronisation failure indication". The SN/VLR does not send new user authentication requests to the user before having received the response to its authentication data request from the HE/AuC (or before it is timed out).

When the HE/AuC receives an authentication data request with a "synchronisation failure indication" it acts as follows:

1) The HE/AuC computing $AK_{MS} = f5_K(RAND_{MS})$ and retrieves SQN_{MS} as $(SQN_{MS} \oplus AK_{MS}) \oplus AK_{MS}$.
2) The HE/AuC computes XMAC-S from K, RAND, $RAND_{MS}$ and SQN_{MS}.
3) The HE/AuC now verifies that XMAC-S matches MAC-S otherwise SQN_{HE} is left unchanged
4) The HE/AuC now verifies whether SQN_{MS} is in the correct range and if that is not the case, resets the counter SQN_{HE} to the value of SQN_{MS}.

In case sequence numbers are managed according to the mechanism described in Annex C.1 of [1], SQN_{HE} is in the correct range if $SQN_{MS} \leq SQN_{HE}$.

In case sequence numbers are managed according to the mechanism described in 6.3.2 of this document, SQN_{HE} is in the correct range if $SQN_{MS} \leq SQN_{HE} < SQN_{MS} + \Delta$. This implies that using the mechanism described there SQN_{HE} may be reset to a lower value, provided that $SQN_{MS} + \Delta < SQN_{HE}$. This is required to prevent a lock-out in the unlikely case that SQN_{HE} has increased very much ahead of SQN_{MS} (without any of the generated authentication vectors reaching the USIM) such that the USIM does not accept them anymore. Note however that the freshness assurance provided by the parameter RAND in computation of the parameters AUTS assures the HE/AuC that SQN_{HE} is never set to a lower value than the current value of SQN_{MS}.

The above change has been proposed to SA-3 [17] and the meeting agreed that it is included in future releases of the security architecture specification.

6.4.3 Notes

6.4.3.1 How does the re-synchronisation mechanism presented above address the possible causes for synchronisation failures?

We discuss each of the possible causes of synchronisation failures:

1) **An SN/VLR uses an authentication vector out of sequence.** The resulting MM user authentication response and MAP authentication data request with indication of synchronisation failure will cause the HE/AuC to send new authentication vectors from storage corresponding to correct sequence numbers values. The counter in the HE/AuC is not reset.
2) **An attacker replays RAND || AUTN, injecting it in a MM user authentication request sent by the SN/VLR.** The resulting MM authentication response and MAP authentication data request with indication of synchronisation failure will cause the HE/AuC to send new authentication vectors from storage corresponding to correct sequence numbers values. The counter in the HE/AuC is not reset. Optionally the HE/AuC may tell the SN/VLR to use the authentication parameters it already has.
3) **There was a failure in the HE/AuC and consequently the counter in the HE/AuC had a wrong value.** Then the resulting MM authentication response and MAP authentication data request with indication of synchronisation failure cause the HE/AuC to generate new authentication vectors corresponding to the correct sequence numbers values. The counter in the HE/AuC is reset.

6.4.3.2 How is replay of an authentication response with indication of synchronisation failure towards the SN/VLR prevented?

According to the conditions on the handling of authentication vectors by the SN/VLR, an SN/VLR never sends the same user authentication request twice. Consequently, the MS cannot have seen RAND and cannot have produced the data integrity code $MACS = f1^*_K(SQN_{MS} \| RAND \| MODE)$ in the authentication response with indication of synchronisation failure before the current user authentication request was

sent by the SN/VLR. The HE/AuC trusts that the SN/VLRs are playing by the rules and hence believes that the parameter MACS is not a replay, i.e., it believes that SQN_{MS} was sent by the MS recently. This gives the HE/AuC confidence that it can securely update its counter $SEQ_{HE.}$

6.4.3.3 What could happen if the SN/VLRs were not handling authentication vectors according to the rules or the security in the SN/VLRs was compromised?

The result could be a temporary denial of service necessitating just one extra round of authentication data request/response before the service was restored to normal. But note that incorrect function ing of the SN/VLRs may lead to (even long-lasting) denial of service to a user in many different ways. So, no additional risk is introduced.

6.4.3.4 Setting back counters in the HE/AuC

In case there was a failure in the HE/AuC and the counter in the HE/AuC is reset to a lower value – as could happen in the alternative proposed in 6.3 above – then the HE/AuC may produce more than one parameter AUTN with the same SQN. (Note, however, that the corresponding RAND always will be different.) But still only one of these AUTN will ever be accepted by the user. (The counter in the MS is always monotonously increasing.) In particular, if an authentication vector was compromised the attacker may still use it only in one session. Note also that an attacker has an additional problem using the compromised authentication vector when the sequence number is concealed because he then does not know when to use it. Furthermore, if the failure in the HE/AuC is an accidental one and the counter is equally likely to assume any value in the range $0 \ldots SQN_{max}$ then authentication vectors based on the incorrectly high counter value are unlikely to be ever usable in a successful authentication with the user. (This is true if Δ is small compared to SQN_{max}.)

6.4.3.5 Cryptographic requirements on f1*

For computing the parameter MACS sent in an authentication response with indication of synchronisation failure the MS uses a MAC-function f1*. Let $(RAND_1, SQN_1)$, $(RAND_2, SQN_2)$, …, $(RAND_n, SQN_n)$ be the random values and sequence numbers in the authentication vectors produced by the HE/AuC over time. Then an attacker can make the MS produce $f1^*_K (SQN_j \| RAND_k \| MODE)$ for all $k < j \leq n$ by replaying old user authentication requests towards the user. Hence, f1* must be resistant against this attack. Note however, that the messages for which an attacker can possibly see the images under f1* are completely determined by choices made by the HE/AuC. Note also that we require f1* to be a MAC function which, by definition, is resistant against adaptively chosen plaintext attacks (a considerably stronger requirement than the one described above).

Note also that the number of images of messages under f1* an attacker can see over time by conducting active attacks is not larger than the number of images of messages under f1 (the MAC-A in a MM authentication request) solicited by spoof service requests. (This is assuming that these active attacks take a similar amount of time.)

If felt necessary the risk could be further limited by limiting the number of synchronisation failure messages which an MS is allowed to produce. (A cryptographically significant number is expected to be so high that denial of service attacks trying to exploit this limit by conducting a corresponding number of replay attacks will be unattractive.)

6.5 Restoring the counters in the HE/AuC database

6.5.1 Current situation

The April release of the security architecture ([3], section 6.3 and Annex C) seems to suggest that for restoration of counters after crash in the AuC database (which may be an extremely rare event, but cannot be ruled out completely), either global time-based counters should be used or a potentially large number of re-synchronisation procedures would have to be run. This section is to show that there is an alternative.

6.5.2 Proposed alternative

We assume that a back-up of the counter values in the AuC is made at regular intervals which is securely stored. We also assume that the operator of the database has an estimate of the maximum increase of counter values during the back-up interval for his most active users. The operator may obtain a robust upper bound from this by applying a safety margin. We further assume that the operator of the AuC notices a crash before the time of the next back-up.

When the database crashes and the counter values are lost then the counter values are restored to the value from the back-up plus the upper bound for increases in counter values during the back-up interval. (In case the sequence number handling described in 6.3 above is used the upper bound is, of course, to be smaller than the Δ introduced there.) The restored counter values are then likely to be larger than what they would have been in normal operations, but they would still be in the correct range.

6.6 Interoperation between UMTS and GSM

6.6.1 Outline

The April release of the security architecture [3] does not specify how interoperation between UMTS and GSM can be achieved. Here we present several mechanisms. In 6.6.2 the different scenarios are identified and the requirement for each scenario is defined. Finally priorities are assigned. Sections 6.6.3 –6.6.5 describe the currently proposed mechanisms. In 6.6.6 they are compared and. In the subsequent sections the preferred proposals is then further developed.

The results of this analysis where present to SA-3 meeting #4 in June ([22], [23]).

6.6.2 Scenarios, requirements and priorities

Inter-operation between UMTS and GSM comprises the following scenarios:

1) **Registration** of a user of the one type in a network of the other type, typically including authentication and agreement on access link keys for the visited network. This includes:

   a) **Registration of a UMTS user in a GSM SN.** [Highest priority.] In countries with existing GSM networks, UMTS networks are expected to be introduced in islands; for nation-wide coverage for GSM-like services the UMTS user will have to rely on the existing GSM network coverage. ***This is called USIM roaming.***

   b) **Registration of a GSM user in a UMTS SN.** [Low priority.] Whether there is an important need for GSM users to access the UMTS network is under dispute. This scenario might be interesting for GSM operators who want to offer their customers roaming opportunities in those countries that are covered by a UMTS network but not with a GSM network. ***This is called GSIM roaming.***

2) **Inter-system handover** of a user from a network of the one type to a network of the other type. This includes:

   a) **Inter-system handover from a UTRAN to a GSM BSS**

      i) **Of a UMTS user.** [High priority.] In countries with existing GSM networks, UMTS networks are expected to be introduced in islands; for nation-wide coverage for GSM-like services the UMTS user will have to rely on the existing GSM network coverage. Inter-system handover will provide service continuation when the UMTS user leaves an area with UMTS coverage. ***This is part of USIM inter-system handover.***

      ii) **Of a GSM user.** [Lowest priority.] In UMTS-only countries there obviously is no need for inter-system handover from UTRAN to GSM BSS, and vice versa. In countries with both GSM and UMTS networks there is hardly any need for allowing GSM users on the UMTS network, as the GSM coverage is likely to be larger than the UMTS coverage. The only rea-

son to connect a GSM user to a UMTS network might be congestion of the GSM network in some area. ***This is part of GSIM inter-system handover.***

b) **Inter-system handover from a GSM BSS to a UTRAN**

i) **Of a UMTS user.** [Medium priority.] In countries with existing GSM networks, UMTS networks are expected to be introduced in islands; for nation-wide coverage for GSM-like services the UMTS user will have to rely on the existing GSM network coverage. This type of handover would allow a UMTS user who initiated a service through a GSM BSS in an area without UMTS coverage, to be handed over to the UTRAN, as soon as possible, and receive the better quality of service. As soon as he is handed over, he may also initiate extra (UMTS) service capabilities. ***This is part of USIM inter-system handover.***

ii) **Of a GSM user.** [Lowest priority.] The same arguments apply as for inter-system handover in the opposite direction (see 2)a)ii)). ***This is part of GSIM inter-system handover.***

While designing a mechanism to meet these objectives, we assumed that there will be UMTS MSC/VLRs that control both UTRAN and GSM BSS. In addition to that there will be GSM MSC/VLRs controlling GSM BSS. We seek for a mechanism that allows inter-system handover between GSM BSS and UTRAN controlled by the same UMTS MSC/VLR, as well as inter-system handover between access networks controlled by GSM MSC/VLRs and UMTS MSC/VLRs.

6.6.3 Mechanism 1

Ericsson [21] outlined a mechanism for USIM roaming and inter-system handover that provides a mechanism for USIM roaming and intersystem handover. It consists of the UMTS HLR/AuC pre-computing the GSM cipher key Kc* along with the UMTS access link keys.

The mechanism can be summarised as follows:

1) **Generation of an authentication vector.** The UMTS HLR/AuC generates a RAND and derives the UMTS authentication parameters XRES, CK, IK and AUTN, and in addition, it derives (from the same RAND) a GSM cipher key Kc*.

2) **Distribution of an authentication vector.** The UMTS/GSM authentication vector is distributed to UMTS VLRs or GSM VLRs that support inter-system handover for UMTS users.

Figure 6-8 shows the state-transition diagram for mechanism 1. It shows the authentication parameters and the access link keys derived at authentication (at registration, location update and re-authentication initialised by the network) and at intersystem handover.

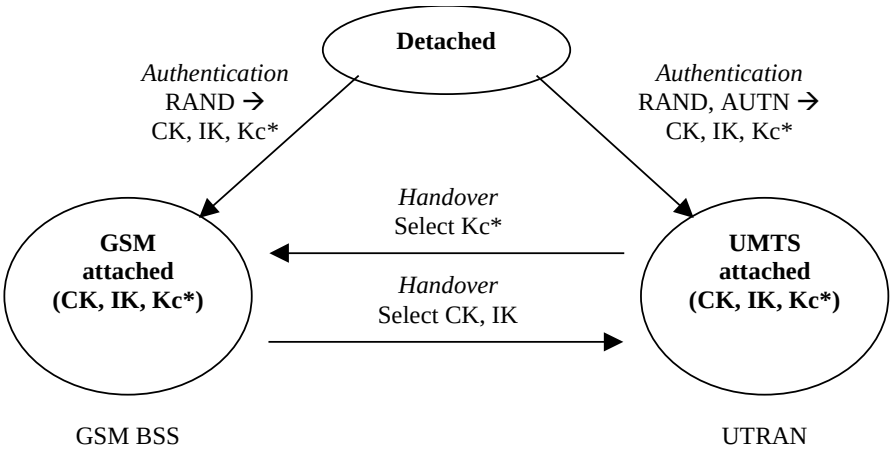

Figure 6-8: State-transition diagram for UMTS interoperation with mechanism 1

The registration and intersystem handover for UMTS users are supported as follows:

3) **UMTS user authentication in UTRAN.** When a user is attached through a UTRAN, the controlling UMTS VLR initiates UMTS authentication and key agreement, i.e., the authentication request contains RAND and AUTN. Upon receipt, the USIM computes the UMTS response RES, the UMTS access link keys CK and IK as well as the GSM cipher key Kc*. It sends back RES. After successful authentication network and user select the UMTS access link keys CK and IK.

4) **UMTS user authentication in GSM BSS.** When a user is attached through a GSM BSS, the controlling UMTS or GSM VLR initiates GSM authentication (i.e., the authentication request contains only RAND). Upon receipt, the MS computes the UMTS response XRES, the UMTS access link keys CK and IK as well as the GSM cipher key Kc*. It computes RES* from RES (conversion function c1) and sends RES* back[3]. The network converts XRES into SRES (conversion function c1) and compares RES with SRES. After successful authentication network and user select the GSM cipher key Kc*.

5) **Inter-system handover of UMTS user from GSM BSS to UTRAN.** When a UMTS user is handed over from GSM BSS to UTRAN, the network and the user select the UMTS access link keys already available at both ends for the new connection.

6) **Inter-system handover of UMTS user from UTRAN to GSM BSS.** When a UMTS user is handed over from UTRAN to GSM BSS, the network and the user select the GSM cipher key Kc* already available at both ends for the new connection.

No mechanism for GSIM roaming and handover was provided. A similar mechanism is not possible, as the GSIM does not compute CK and IK.

6.6.4 Mechanism 2

Another approach at intersystem handover is to derive the access link keys from the target access network from the access link keys of the source access network. This is the approach of the mechanism presented here. It is also characterised by the fact that the network and the user delete the access link keys from the source SN after they are handed over to the target network.

The alternative mechanism can be summarised as follows:

1) **Generation of an authentication vector.** The UMTS HLR/AuC generates a RAND and derives from that the UMTS authentication parameters XRES, CK, IK and AUTN. No GSM cipher key Kc* is generated.

2) **Distribution of an authentication vector.** The distribution depends on the type of VLR that requests authentication vectors:

   a) UMTS VLR (controlling UTRAN or both UTRAN and GSM BSS) receive UMTS authentication vectors.

   b) GSM VLR (controlling only GSM BSS) receive GSM authentication vectors (RAND*, SRES*, Kc*). The HLR/AuC constructs them from the UMTS authentication vectors in the following way: RAND* = RAND, SRES* = c1(XRES) and Kc* = c2(CK).

Figure 6-9 shows the state-transition diagram for UMTS interoperation with mechanism 2. It shows the authentication parameters and the access link keys derived at authentication (at registration, location update and re-authentication initialised by the network) and at intersystem handover. Note that the UMTS user has two state when attached to a UTRAN. In both states he has CK and IK available, but in the state "Handed over from GSM" these keys have the effective key length of the GSM cipher key (64 bits). Therefore, he should perform a re-authentication as soon as he sees fit and go to state "UMTS attached".

[3] It is for further study whether RES* should be different from RES.

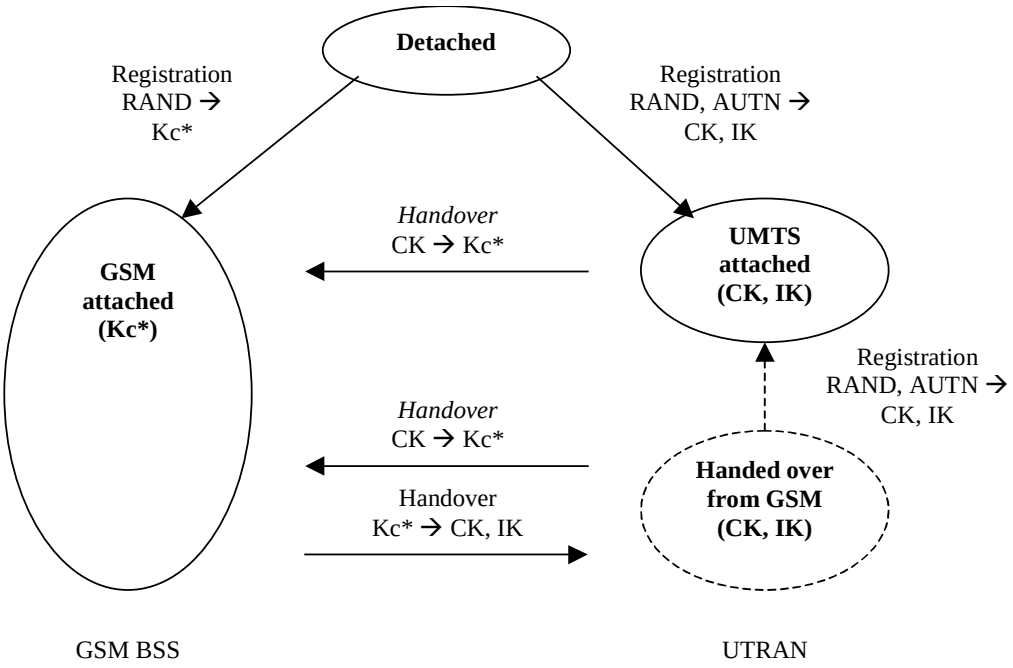

Figure 6-9: State-transition diagram for UMTS interoperation with mechanism 2

The registration and intersystem handover for UMTS users are supported as follows:

3) **UMTS user authentication in UTRAN.** When a UMTS user is attached through a UTRAN, the controlling VLR initiates UMTS authentication and key agreement. No GSM cipher key Kc* is derived.

4) **UMTS user authentication in GSM BSS.** When a UMTS user is attached through a GSM BSS, the controlling VLR initiates GSM authentication and key agreement. This is done using a UMTS authentication vector or a GSM authentication vector, depending on the type of VLR controlling the GSM BSS:

   a) A user attached to a GSM BSS controlled by a UMTS VLR (controlling UTRAN or both UTRAN and GSM BSS). The UMTS VLR converts the UMTS authentication vector into a GSM authentication vector in the same way as the HLR/AuC did before it distributed authentication data to a GSM VLR, see 2) b). The UMTS VLR sends RAND* to the user. The USIM derives RES and CK and converts these parameters to their GSM counterparts in the same way as the VLR did: RES* = c1(RES) and Kc* = c2(CK).

   b) A user attached to a GSM BSS controlled by a GSM VLR (controlling only GSM BSS). The GSM VLR sends the UMTS user RAND*. The USIM derives RES* and Kc* is the same way as in a GSM BSS controlled by a UMTS VLR (see 4)a)).

5) **Inter-system handover of UMTS user from GSM BSS to UTRAN.** At the network side the old VLR sends Kc* to the new VLR. The new VLR then derives CK and IK from Kc*: CK = c3(Kc*) and IK = c4(Kc*). At the user end, the dual-mode UE derives CK and IK in the same way.

6) **Inter-system handover of UMTS user from UTRAN to GSM BSS.** At the network side the old VLR derives Kc* from CK: Kc* = c2(CK) and sends Kc* to the new VLR. At the user end, the dual-mode UE derives Kc* in the same way.

Surprisingly, all functionality is now in place for GSIM roaming too.

1) **Generation of an authentication vector.** The GSM HLR/AuC generates a GSM authentication vector that consists of (RAND*, SRES*, Kc*).

2) **Distribution of an authentication vector.** The GSM HLR/AuC distributes GSM authentication vectors to all VLR, regardless its type.

Figure 6-10 shows the state-transition diagram for mechanism 1. It shows the authentication parameters and the access link keys derived at authentication (at registration, location update and re-authentication initialised by the network) and at intersystem handover.

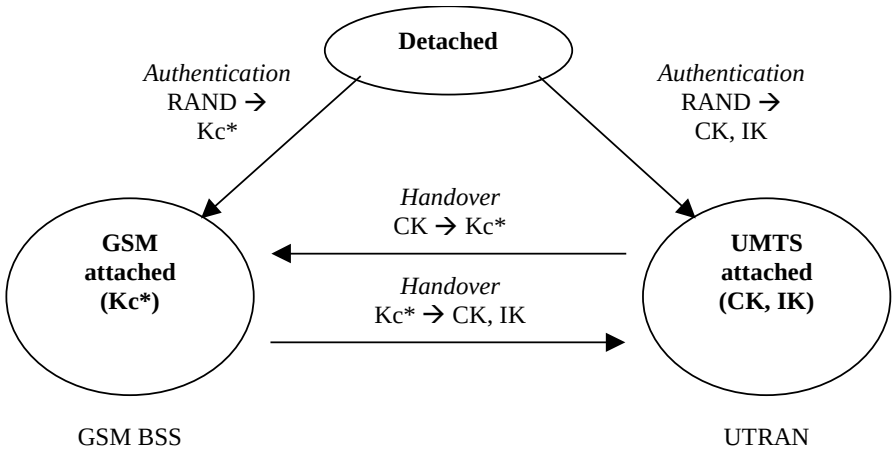

Figure 6-10: State-transition diagram for GSIM interoperation with mechanism 2

The registration and intersystem handover for GSM subscribers are supported as follows:

3) **GSM user authentication in UTRAN.** When a GSM user is attached through a UTRAN, the controlling VLR initiates GSM authentication and key agreement. It sends RAND* to the user. The user derives RES* and Kc*. The dual-mode UE sends RES* back to the network. The VLR compares RES* with SRES*. After successful authentication and agreement of a GSM cipher key Kc*, the VLR as well as the UE derive UMTS access link keys as already explained under inter-system handover of UMTS users from GSM BSS to UTRAN: CK = c3(Kc*) and IK = c4(Kc*).

4) **GSM user authentication in GSM BSS.** When a GSM user is attached through a GSM BSS, the controlling VLR initiates GSM authentication and key agreement.

5) **Inter-system handover of GSM user from GSM BSS to UTRAN.** The procedure is identical to the one explained under inter-system handover of a UMTS user from GSM BSS to UTRAN. At the network side the old VLR sends Kc* to the new VLR. The new VLR then derives CK and IK from Kc*: CK = c3(Kc*) and IK = c4(Kc*). At the user end, the dual-mode UE derives CK and IK in the same way.

6) **Inter-system handover of GSM user from UTRAN to GSM BSS.** The procedure is identical to the one explained under inter-system handover of a GSM user from UTRAN to GSM BSS. At the network side the old VLR derives Kc* from CK: Kc* = c2(CK) and sends Kc* to the new VLR. At the user end, the dual-mode UE derives Kc* in the same way.

6.6.5 Mechanism 3

in this section a third approach is proposed. In this approach, the GSM cipher key is derived from the UMTS access link keys when the user is handed-over from the UTRAN to the GSM BSS for the first time, but the UMTS access link keys are not forgotten. They are sent to the GSM VLR and may be used again, for instance when the user returns to the UTRAN.

The alternative mechanism can be summarised as follows:

1) **Generation of an authentication vector.** The UMTS HLR/AuC generates a RAND and derives from that the UMTS authentication parameters XRES, CK, IK and AUTN.

2) **Distribution of an authentication vector.** The UMTS HLR/AuC distributes the UMTS authentication vectors to the UMTS VLR and the GSM VLR in which the user is registered.

Figure 6-11 shows the state-transition diagram for USIM interoperation with mechanism 3. It shows the authentication parameters and the access link keys derived at authentication (at registration, location update and re-authentication initialised by the network) and at intersystem handover. Note that the UMTS user has two states when attached to a UTRAN. This time the state "handed over from GSM" does not have weaker access link keys, instead it has in addition to the other state, a GSM cipher key. There is no need to re-authenticate as soon as the VLR sees fit.

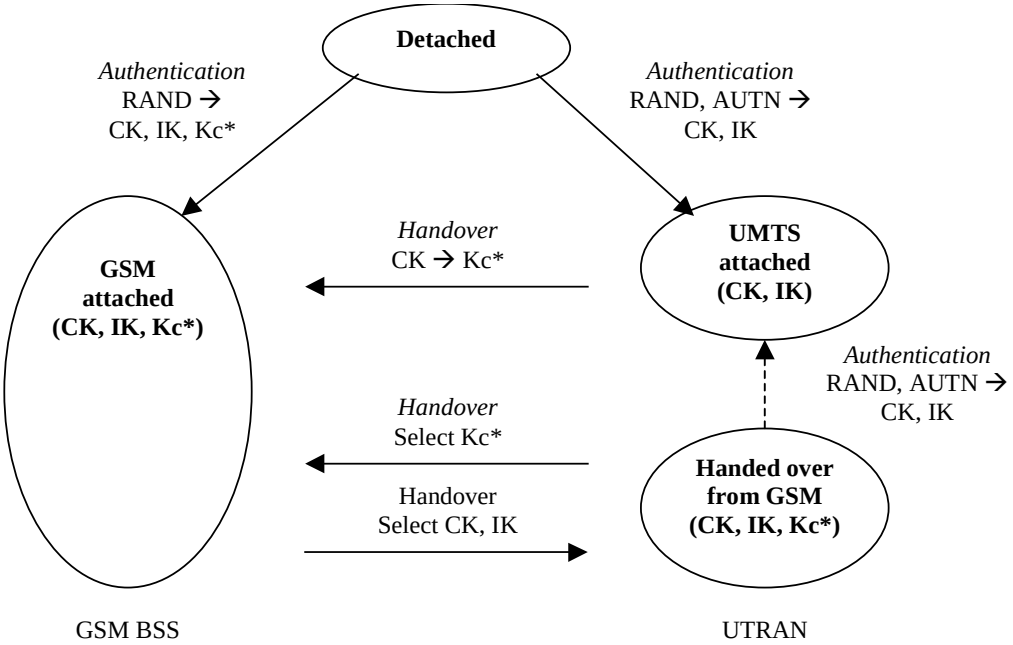

Figure 6-11: State-tansition diagram for USIM interoperation with mechanism 3

The registration and intersystem handover for UMTS users are supported as follows:

3) **UMTS user authentication in UTRAN.** The controlling UMTS VLR initiates UMTS authentication and key agreement.

4) **UMTS user authentication in GSM BSS.** The controlling VLR initiates GSM authentication and key agreement. The VLR sends RAND to the user, who computes RES and derives RES, CK and IK and then derives RES* = c1(RES) and Kc* = c2(CK). Kc*. The user then returns RES*. Upon receipt the VLR derives SRES* = c1(XRES) and verifies the received RES* and derives Kc* = c2(CK). The user and the network then use Kc*, but also CK and IK are stored in the MS and in the VLR.

5) **Inter-system handover of UMTS user from GSM BSS to UTRAN.** At the network side the old VLR selects CK and IK and sends them to the new VLR, at the user side, the MS selects CK and IK.

6) **Inter-system handover of UMTS user from UTRAN to GSM BSS.** The procedure depends on whether the user has already been in the GSM BSS before or not:

   a) If the user has already been in a GSM BSS since the last authentication and key agreement, then user and network already have the corresponding CK and IK stored. At the network side the old VLR sends CK, IK and Kc* to the new VLR. The network and the user subsequently select Kc* for protection on the radio access link.

   b) If it is the first time the user enters a GSM BSS since the last authentication and key agreement, the user and the network derive Kc* from CK: Kc* = c2(CK). At the network side the old VLR sends CK, IK and Kc* to the new VLR. The network and the user subsequently select Kc* for protection on the radio access link.

6.6.6 Evaluation of the proposals

6.6.6.1 Provide the required level of security

All mechanisms provide the UMTS users in the UMTS network the same level of security after a UMTS authentication and key agreement has been performed. All mechanisms also provide the UMTS users from the start of entering the network, mutual authentication between network and user through the use of the data integrity mechanism and therefore, protection against those attacks that rely on the suppression or absence of encryption.

However, there is one noticeable difference between mechanism 2 and the other two: after an inter-system handover from GSM BSS to UTRAN mechanism 2 provides access link keys with the strength and effective key length of the GSM cipher key (64 bits), whereas mechanism 1 and mechanism 3 provide the full UMTS effective key length. Nevertheless, mechanism 2 can still be found acceptable, as one can argue that the user has accepted the GSM level of security (still very much lower, offering no data integrity protection) for this service when he initiated the service in the GSM network or was handed over to the GSM network. Furthermore, the UMTS user still has assurance of key freshness when he initiated his service in the UTRAN. Nevertheless, using mechanism 2 the user should require a full UMTS authentication and key agreement, before he initiates new services. This can be accomplished by setting the key lifetime of the UMTS access link keys CK and IK derived at intersystem handover such that they are only valid for the ongoing service.

6.6.6.2 Have a minimal impact

A further difference between mechanism 2 and the mechanisms 1 and 3 is that mechanism 2 does not require any changes to the GSM VLR. Instead, mechanism 1 and 3 require that the GSM VLR be upgraded

Mechanism 3 requires the implementation of a standardised conversion function in the UMTS VLR. Mechanism 2 requires the implementation of two additional conversion functions, but two that do not require any computation, and which would be required for mechanism 3 too, if mechanism 3 were to support GSIM interoperation. However, all conversion functions are unkeyed. Mechanism 1 has the advantage that no conversion functions are required in the UMTS VLR.

6.6.6.3 Be computationally efficient

Mechanism 1 computes the GSM cipher key Kc* efficiently but computes and distributes this key for each authentication vector regardless of whether the user is likely to roam into or be handed over to a GSM network.

Mechanisms 2 and 3 computes the GSM cipher key Kc* rather inefficiently, but only when needed. Further, mechanism 3 only converts keys once, whereas mechanism 2 converts keys at each handover.

The relative efficiency of the competing schemes are dependent on the frequency of inter-system roaming and handover. The mechanisms 2 and 3 have the advantage when inter-system roaming and handover are relatively rare events.

6.6.6.4 Conclusion

We have a slight preference for mechanism 2 as it provides a sufficient level of security and is the only mechanism that is able to operate without upgrading the GSM/VLRs.

6.6.7 Procedures

6.6.7.1 Authentication and key agreement of UMTS users attached to a GSM BSS

Figure 6-12 shows how authentication and key agreement is run between the UMTS/GSM network and the UMTS user, re-using the cryptographic function for UMTS authentication and key agreement.

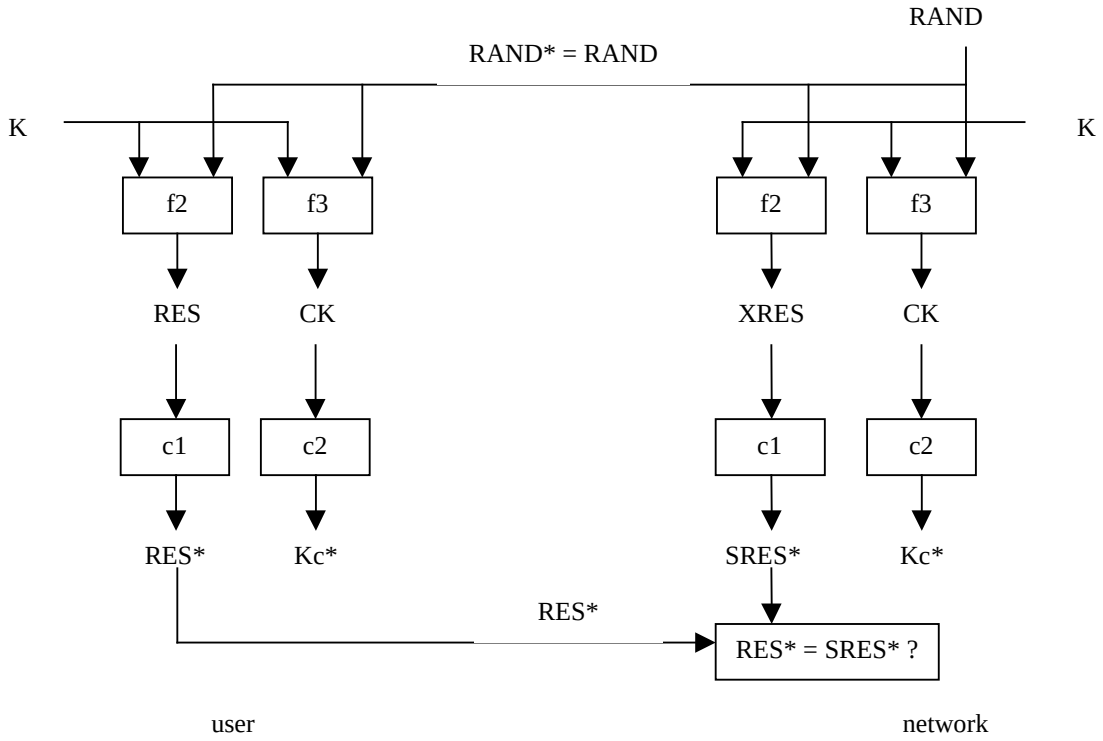

Figure 6-12: Authentication and key agreement of UMTS users attached to a GSM BSS

There are two possibilities:

1) A UMTS user in a GSM BSS controlled by a GSM VLR.

⇒ c1 and c2 in the UMTS HLR/AuC [Highest priority] .

2) A UMTS user in a GSM BSS controlled by a UMTS VLR.

⇒ c1 and c2 in the UMTS VLR [Highest priority].

For both scenarios the allocation of c1 and c2 provides the advantage that the USIM can be inserted in a GSM-only terminal.

⇒ c1 and c2 in the USIM [Highest priority].

6.6.7.2 Authentication and key agreement of GSM subscribers attached to a UTRAN

Figure 6-12 shows how authentication and key agreement is run between the UMTS network and the GSM user, deriving the UMTS access link keys CK and IK from the GSM cipher key Kc*.

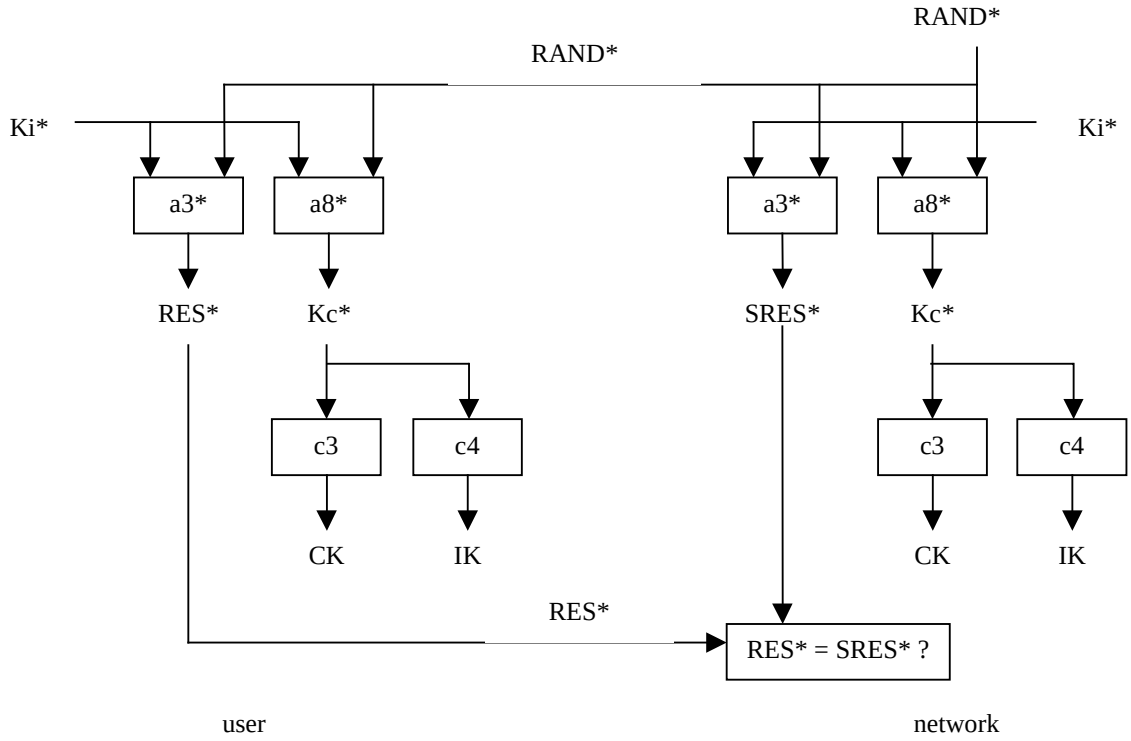

Figure 6-13: GSM AKA between GSM users and UMTS VLR

At the network side the UTRAN can only be attached to a UMTS VLR:

⇒ c3 and c4 in the UMTS VLR [Low priority] .

At the user side it is impossible to introduce c3 and c4 in the existing GSIM:

⇒ c3 and c4 in the UMTS UE [Low priority].

6.6.7.3 Intersystem handover from GSM BSS to UTRAN

Figure 6-14 shows how UMTS access link keys are derived from a GSM cipher key.

Figure 6-14: Intersystem handover from GSM BSS to UTRAN

At the network side the UTRAN can only be attached to a UMTS VLR:

⇒ c3 and c4 in the UMTS VLR [Medium priority] .

At the user side it is impossible to introduce c3 and c4 in the existing GSIM:

⇒ c3 and c4 in the UMTS UE [Medium priority].

6.6.7.4 Intersystem handover from UTRAN to GSM BSS

Figure 6-15 shows how a GSM cipher key Kc* is derived from a UMTS cipher key CK.

Figure 6-15: Intersystem handover from UTRAN to GSM BSS

In order not to impact the VLR that controls the GSM BSS, which might be a GSM VLR, the source VLR converts the UMTS cipher key into the GSM cipher key.

⇒ c2 in the UMTS VLR [High priority] .

At the user side having c2 in the UE allows interoperation with GSIMs:

⇒ c2 in the UMTS UE [High priority].

6.6.8 Conversion functions

6.6.8.1 The conversion function c1

The conversion function c1 converts XRES into SRES* (resp. RES into RES*). XRES has a length between 32 and 64 bits, whereas SRES* has a length of 32 bits.

At the SA-3 meeting #3 in Bonn the question was raised whether XRES is allowed to be equal to SRES*. We are convinced this is acceptable indeed. The only attack we see when XRES equals to SRES* is shown in Figure 6-16; a user is camping on a false GSM BSS connected to a UTRAN. The network initiates UMTS authentication and key agreement and is made to believe, by receiving the correct response RES that the user is authenticated through the UMTS authentication and key establishment mechanism. However, the user has only seen RAND. The attack is useless however, and this for the following reasons:

1) The network will select the UMTS access link keys CK and IK, whereas the user will select Kc*. The communication will not work. Even if the intruder suppresses encryption on the downlink channel, and the UMTS does the same on the channel between intruder and network. Even then, the intruder will not have the correct integrity key IK.
2) The intruder must choose RAND* = RAND to receive the proper RES* = RES, i.e., the RAND is fresh after all, unless the network tries to re-use an old authentication vector, which it cannot expect to succeed.

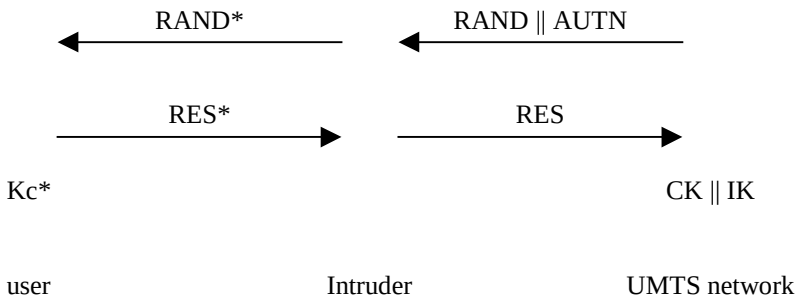

Figure 6-16: Useless false base station attack when (X)RES = (S)RES*

We conclude that there is no danger in XRES being of equal length as SRES, in which case XRES and SRES may be identical, and c1 is nothing but the identity function . If XRES is longer than SRES, a truncation function may be chosen for f1.

6.6.8.2 The conversion function c2

The conversion function c2 converts CK into Kc*. CK has a length of 128 bits, whereas Kc* has a length of 64 bits.

A simple truncation to 64 bits is not sufficient, as a possible compromise of the (too short) GSM cipher key Kc* would in that case give too much information on the UMTS cipher key CK. The remaining part of CK might then be guessed by the intruder (i.e., derived by cryptanalysis of encrypted data). A one-way function may be appropriate. This is a cryptographic function that has the following property:

> A one-way function (OWF) is a function *f* such that for each *x* in the domain of *f*, it is easy to compute *f*(*x*); but for essentially all *y* in the range of *f*, it is computationally infeasible to find any *x* such that $y = f(x)$.

Using a one-way function would thus provide a GSM cipher key Kc* that would not convey information on CK.

Note however, that once Kc* is derived from CK, CK is never going to be used again. If after being registered or handed over to the GSM BSS he subsequently re-enters the UTRAN, a new UMTS authentication and key agreement is performed and new keys are established before any new services is started. If he is again handed over to the UMTS network, during the same service, new UMTS access link keys CK' and IK' will be derived from the GSM cipher key Kc*, which convey as much information about the original UMTS access link keys CK and IK as the GSM cipher key Kc* does. Therefore by choosing a one-way function c2 rather than a truncation one only protects the confidentiality of user data encrypted before the inter-system handover from being compromised. A possible alternative to the choice for a one way function , might be a truncation function and the additional measure that also a user after handover to the GSM network, performs a new GSM authentication and key agreement before any new service is started, in order to minimise the risk of a compromise of the GSM cipher key Kc* and hence of a part of the UMTS access link keys.

6.6.8.3 The conversion function c3

The conversion function c3 converts Kc* into CK. Kc* has a length of 64 bits, whereas CK has a length of 128 bits, which however, are not all used.

It appears sufficient to define a function that pads Kc* with a fixed sequence of ones and zeros.

6.6.8.4 The conversion function c4

The conversion function c4 converts Kc* into IK. Kc* has a length of 64 bits, whereas IK has a length of 128 bits, which however, are not all used.

It appears sufficient to define a function that pads Kc* with a fixed sequence of ones and zeros, which should however, be different from the one used to derive the cipher key CK..

6.7 Interoperation between UMTS and IS-41

6.7.1 Provided services and priorities

Interoperation between UMTS users and networks and IS-41 users and networks comprises the following services:

Registration of a roaming user of the one type in a network of the other type, typically including authentication and key agreement. This includes:

(1) **Registration of a UMTS user in an IS-41 SN.** [High priority.] In countries with existing IS-41 networks, UMTS networks are expected to be introduced in islands; for nation-wide coverage for IS-41-like services the UMTS user will have to rely on the existing IS-41 network coverage. ***This is called USIM roaming.***

(2) **Registration of an IS-41 user in a UMTS SN.** [Low priority.] Whether there is an important need for IS-41 users to access the UMTS network is under dispute. This scenario might be interesting for IS-41 operators who want to offer their customers roaming opportunities in those countries that are covered by a UMTS network but not by a IS-41 network. ***This is called IS-41ME roaming.***

Inter-system handover of a user from a network of the one type to a network of the other type is expected to be hardly feasible with justifiable effort, because of the big differences between both systems, of which security only plays a small part. Therefore security mechanisms to provide inter-system handover are not investigated.

Different scenarios are conceivable for an interoperation between UMTS and IS-41 systems. Only scenarios are taken into account where permanent key material is never disclosed to a network component apart from the HE/AuC. In this section these scenarios are discussed.

A pre-condition for all such interoperation scenarios is that the MS/USIM can run both protocols, the UMTS AKA as well as the IS-41 AKA.

6.7.2 Interoperation based on location data base interworking

6.7.2.1 General

In the scenarios discussed in this section users have only one subscription, but their MS/USIM has two full sets of security related data. When a user is roaming in a UMTS or an IS-41 system the appropriate MS/USIM data is utilised. For a more detailed description of the scenarios two cases have to be distinguished.

6.7.2.2 Registration of a UMTS user in an IS-41 serving network

In this scenario the UMTS HE which provides to his UMTS users capabilities for roaming in IS-41 networks has a contractual agreement with a specific IS-41 network. This IS-41 network acts as a IS-41 HE for the user when he is roaming in arbitrary IS-41 networks. We denote this network by “contractual IS-41 HE”.

The UMTS user gets two full sets of security related data including user identity, keying material, one set for UMTS and one for IS-41 systems. But the user has only one subscription, the one with his UMTS HE. The MS utilised by this user has the capabilities of a normal IS-41 MS but also of a UMTS terminal including a UICC containing the USIM application. Depending on the kind of network in which the user is roaming the MS runs the appropriate protocols.

As long as the user is roaming in UMTS networks the contractual IS-41 HE is not involved. In case of roaming in an IS-41 network the contractual IS-41 HE acts as the HE of this user. The UMTS HE at least needs to have the information about the fact that the user is roaming in an IS-41 network (e.g. for incoming calls). The exact location information may be held by the contractual IS-41 HE. The necessary inter-

operation between the databases of both networks has to be facilitated by their database management functions.

6.7.2.3 Registration of an IS-41 user in a UMTS SN

In case of an IS-41 user who wants to roam in UMTS SNs, the interoperation is carried out analogously to that described in section (1) above, with the difference that the roles of the involved networks are exchanged.

The scenarios described in this section may be a valuable option especially

– when HE and contractual HE coincide or

– when (more generally) HE and contractual HE have a common database and common access to the service profile of the UMTS user.

6.7.3 Interoperation based on IS-41/GSM interoperation

6.7.3.1 General

Because of market demands in the US it seems to be likely that interoperation between IS-41 and GSM systems will be in operation before UMTS systems are going to be introduced. A consortium of companies called GAIT (GSM/ANSI136 Interoperability Team) is already working on this topic with the aim to specify interoperation features by the end of this year. The results of GAIT are introduced to the standardisation body T1P1 working group 5, which has established an ad-hoc group on the interoperation topic.

The scenario proposed in this section is based on the pre-requisite that interoperation between IS-41 and GSM systems is already in operation. Then interoperation between UMTS and IS-41 may be provided by a two step approach, where one of the steps makes use of the interoperation between IS-41 and GSM. Two cases have to be considered.

6.7.3.2 Registration of a UMTS user in a IS-41 SN

If a UMTS user wants to register in an IS-41 SN, the user's UMTS HE at first carries out a mapping of the UMTS AKA parameters to the GSM AKA parameters as described in 6.6 for a UMTS user roaming in a GSM network attached to a GSM BSS controlled by a GSM VLR. Then as a result GSM triplets are available at the network side.

The interoperation can now be facilitated using the already existing features for GSM users roaming in IS-41 networks.

Apart from the UMTS capabilities the user needs to have the capabilities to carry out the mapping mentioned above but also the ones needed for a GSM user roaming in an IS-41 network.

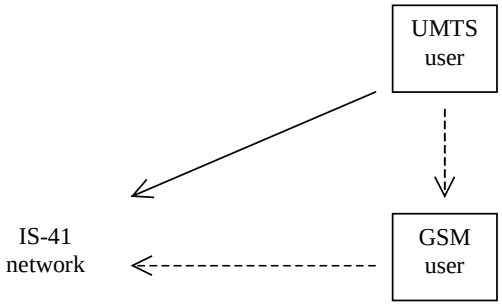

Figure 6-17: UMTS users roaming in IS-41 networks based on IS-41/GSM interoperation

6.7.3.3 Registration of an IS-41 user in a UMTS SN

If an IS-41 user wants to register in a UMTS SN, the user's IS-41 HE at first acts according to the already existing features which provide roaming for IS-41 users in GSM networks.

The further steps depend on the way roaming of IS-41 users in GSM networks is really facilitated.

If after the run of the interoperation features on the user as well as on the network side GSM triplets are available, then interoperation can be facilitated as described in 6.6 for a GSM user in a UMTS network attached to a UTRAN.

Apart from the IS-41 capabilities the user needs to have the capabilities to carry the mapping features needed for a IS-41 user roaming in a GSM network but also the ones needed for a GSM user in a UMTS network as mentioned above.

If no GSM triplets are available, then additional features may be needed to provide the IS-41 user with GSM triplets. This however depends on the exact way in which IS-41/GSM interworking facilitated and can therefore not further be examined here.

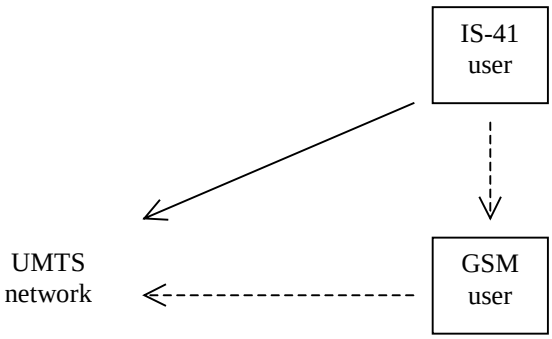

Figure 6-18: IS-41 users roaming in UMTS networks based on IS-41/GSM interoperation

The AKA protocol for UMTS specified in 5.2 is closely related to the existing GSM protocol but quite different from the AKA used in IS-41. Therefore AKA mapping is felt to be easier between UMTS and GSM systems compared to UMTS and IS-41 systems and hence the approach described in this section may be easier than the interworking examined in section 2.3 below. This however depends on the exact way in which interworking of GSM and IS-41 is going to work. This scenario therefore has to be investigated in depth when IS-41/GSM interoperation is standardised and its actual value can only be assessed thereafter.

6.7.4 Interoperation based on UMTS/IS-41 gateway components

6.7.4.1 General

It is investigated how security can interoperate for UMTS systems based on the authentication and key-agreement mechanism specified in [3] (see also 5.2) and IS-41 systems standardised in [19].

The aim in this section is that interoperation between UMTS and IS-41 networks should be provided without (or at least with as little as possible adaptation in the SN and in the HE of the roaming user. All adaptation needed has to facilitated by features on the user side and in a newly introduced UMTS/IS-41 gateway component GW, which provides interfaces between UMTS and IS-41 networks.

Figure 6-19 shows the entities involved in the roaming scenarios. The user (MS = USIM + UE) and the HE/AuC of the appropriate HE belong to system B. This B-user is roaming in a SN A-SN of system A. The A-SN only has capabilities to run protocols according to the standard for systems of type A and does not know anything about the protocols of type B systems.

The gateway component GW has two different faces. To the A-SN it acts like a HE which also belongs to system A and to the actual HE B-HE/AuC of the B-user it acts like a SN which is also of type B. There-

fore it has to have implemented the security functionality for both systems as well as additional features to provide the conversion from the protocols and parameters of the one type to that of the other type.

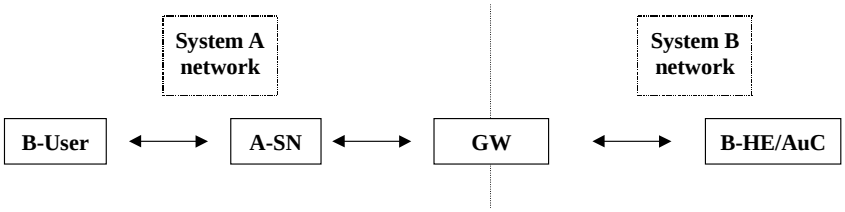

Figure 6-19: System interoperation with intermediate gateway component

The operator of the gateway component may coincide with the operator of the HE, he may also coincide with an operator of the type of the SN, the gateway may even be operated by a third party. Depending on the actual gateway operator different relationships of trust may be needed for interoperation.

The general pre-requisite in section 6.7.1 that only scenarios are taken into account where long term key material is never disclosed to a network component apart from the HE/AuC, is in the scenarios in this section also applied to the gateway, i.e., gateways shall never have permanent key material available.

6.7.4.2 Registration of a UMTS user in an IS-41 serving network

In order to initially register in an IS-41 SN both UMTS user and IS-41 SN need to agree on a common temporary authentication key *SSD*. Then, as long as the UMTS user is roaming in IS-41 networks, *SSD* can be used in security protocols which may have to be run to provide IS-41 services to that UMTS user. These security protocols are carried out between UMTS-user, IS-41 SN and GW which acts like an IS-41 HE of the UMTS user. For *SSD Update* the so-called "*SSD* shared" variant of [19] is used, i.e., the *SSD* is shared between GW and IS-41 SN. A description of the *Initial Registration* but also of the *SSD Update* procedure with *SSD* shared can also be found in [6] (6.2.2.4).

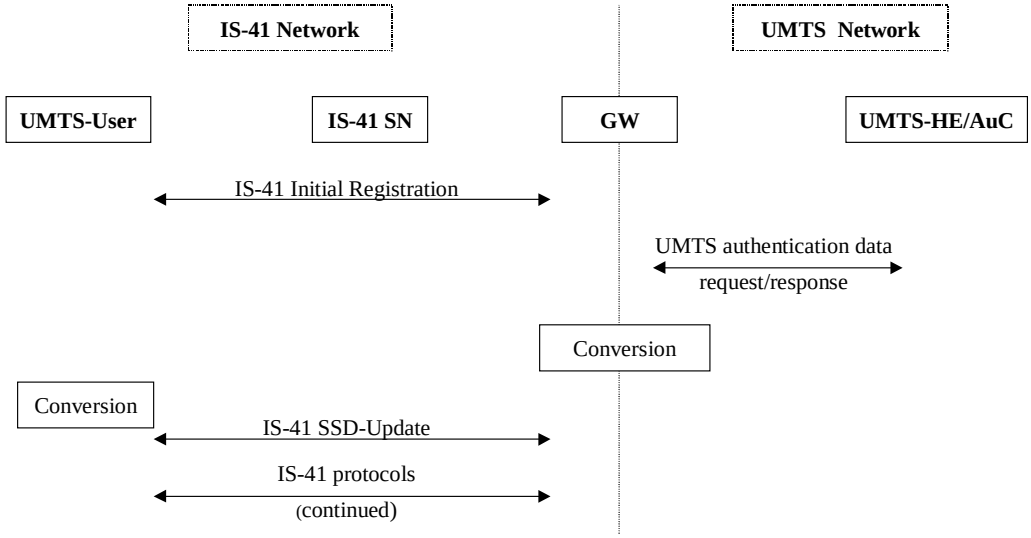

Figure 6-20: Overview - UMTS users roaming in IS-41 networks based on intermediate gateway components

Figure 6-20 provides an overview of the initial registration procedure which can be roughly described by the following steps:

(1) The UMTS user determines that a serving system of IS-41 type has been entered and initiates an IS-41 *Initial Registration* procedure with the IS-41 SN.

(2) The IS-41 SN contacts the gateway (as the presumed HE of this user). The GW detects that the user is a UMTS user roaming in an IS-41 SN and demands a set of UMTS AKA parameters by initiating a

UMTS *Authentication Data Request/Response* procedure according to [1] with the UMTS HE/AuC of that user.

(3) The GW in a predefined way converts the received UMTS AKA parameters and initiates a run of the IS-41 *SSD Update* protocol with the UMTS user via the IS-41 SN using the converted parameters. The main idea of the conversion is that the integrity key IK of UMTS which is also to provide "local" authentication, i.e., serves as a temporary authentication key in UMTS, may be used as a temporary authentication key *SSD* in IS-41.

(4) The UMTS user in the same predefined way as the gateway converts its UMTS AKA parameters to the parameters needed for the IS-41 *SSD Update* protocol.

(5) Now UMTS user and IS-41 SN continue IS-41 registration as it is usually foreseen in [19] when in course of the registration procedure *SSD Update* was demanded.

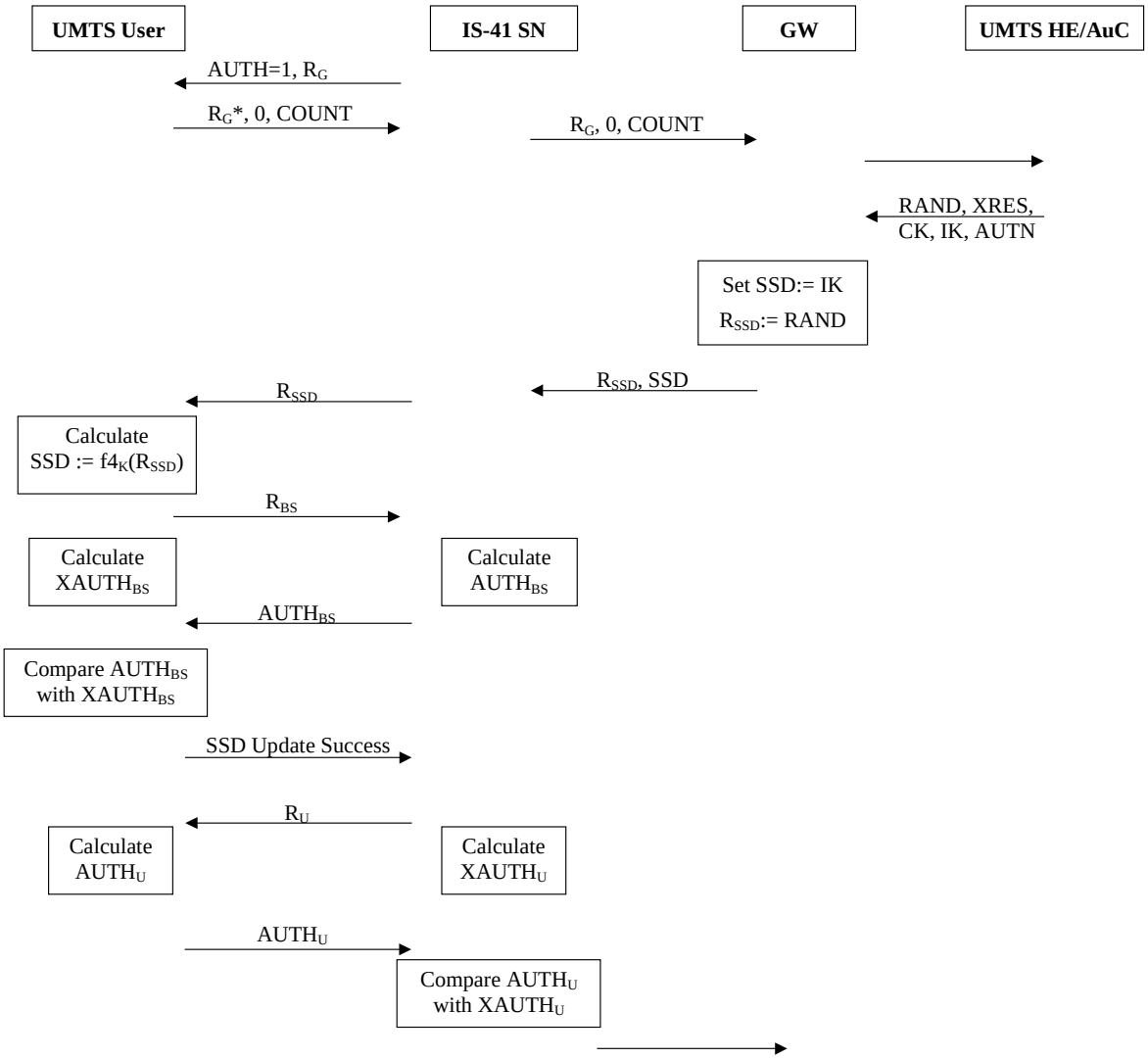

Figure 6-21: Details - UMTS users roaming in IS-41 networks based on intermediate gateway components

The precise description of the protocol and the parameters exchanged in course of the information flows is given below and is depicted in **Figure 6-21**. Differences to the original IS-41 procedures are specifically mentioned in the text. A description of these original IS-41 procedures, the *Initial Registration* and the *SSD Update* procedure with *SSD* shared, can also be found in [6] (6.2.2.4).

The UMTS user determines that a SN of IS-41 type has been entered and by the parameter *AUTH=1* that authentication is required on all system accesses. The UMTS user extracts the random number R_G from this broad-casted message sent by the IS-41 SN and initiates an IS-41 *Initial Registration* procedure with the IS-41 SN. As no *SSD* is available on the user side, the user is not able to authenticate himself. Therefore instead of the usual calculation of the IS-41 authentication parameter *AUTR*, the user sets *AUTR* to a predefined value, e.g. "0" and sends *(R_G*, 0, COUNT)* to the IS-41 SN, where R_G* denotes the first 8 significant bits of R_G. The SN forwards these parameters to GW, replacing R_G* by R_G. The IS-41 SN contacts the gateway (as the presumed HE of this user).

The next steps are not part of the usual IS-41 protocols. The GW detects that the user is a UMTS user roaming in an IS-41 SN and demands a set of UMTS AKA parameters by sending a UMTS *Authentication Data Request* message according to the UMTS HE/AuC of this user. The HE/AuC produces a quintuple *(RAND, XRES, CK, IK, AUTN)* according to [3] and sends it to the GW.

The GW converts the received UMTS quintuple in order to use it for an IS-41 *SSD Update* protocol with the UMTS user. The *SSD Update* protocol according to [19] makes use of a random number R_{SSD} and a temporary key *SSD*. The GW interprets the received *RAND* as R_{SSD} and the temporary key *IK* as *SSD*. All other parameters of the quintuple are ignored.

Note: *RAND* is usually a 128 bit value whereas R_{SSD} is only 56 bits long. As the full length of *RAND* is needed on the user side in order to calculate the correct *SSD* value, the IS-41 SN would have to allow for a longer random number parameter to be transmitted.

Another possibility would be that GW somehow (e.g. by (mis-)using the parameter MODE) indicates to the UMTS HE/AuC that only a 56 bit random number *RAND* should be used to calculate the UMTS quintuple.

Now the IS-41protocol is continued. The GW transmits the random number R_{SSD} and *SSD* to the IS-41 SN, which stores *SSD* and forwards R_{SSD} to the UMTS user.

On the user side (in contrast to the IS-41 protocol) *SSD* is computed analogously to the way in which *IK (=: SSD)* was calculated in the UMTS HE/AuC. I.e., *SSD* is computed from R_{SSD} using the UMTS key generating function $f4_K$ under control of the users permanent UMTS key *K*.

The remainder of the procedure is identical to the SSD-update procedure for IS-41: In order to authenticate the IS-41 SN the UMTS user generates a random number $RAND_{BS}$ and sends it to the SN. The user then computes the expected authentication result $XAUTH_{BS}$ from $RAND_{BS}$ using *CAVE* under control of the first 64 bits SSD_A of *SSD*.

The IS-41 SN analogously computes the authentication result $AUTH_{BS}$, and sends it to the user. If the $AUTH_{BS}$ result provided by the IS-41 SN matches the value $XAUTH_{BS}$ computed by the UMTS user then the user stores the new *SSD* value for use in future executions of *CAVE* and sends a *SSD Update Confirmation* message to the IS-41 SN.

The IS-41 SN generates a unique random number R_U and calculates the expected authentication response $XAUTH_U$ from R_U using *CAVE* under control of SSD_A. The SN now sends the generated unique challenge R_U to the user. The UMTS user computes the authentication response $AUTH_U$ analogously to the calculations for $XAUTH_U$ in the SN and sends it to the SN.

The IS-41 SN compares the value of $AUTH_U$ with the expected and previously calculated authentication response $XAUTH_U$. In case of a matching the *SSD* updating has been successfully completed and an appropriate note is sent to the GW.

The GW stores *SSD* for use in future executions of *CAVE* for the user. The GW sends an indication to the SN that service is to be provided to the user.

Now UMTS user and IS-41 SN continue IS-41 registration as it is usually foreseen in [28] when in course of the registration procedure an *SSD Update* is demanded.

When a new SSD_update is required the GW retrieves a new authentication vector from the UMTS – HE/AuC as above.

Requirements for the entities involved

In order to support the protocols described above, the following additional capabilities are required at the different entities involved:

The UMTS user

- has to be able to run the IS-41 registration protocol without carrying out the usual calculation of *AUTR*;
- has to be able to run an *SSD Update* protocol using his permanent UMTS key *K* and his UMTS key generating function $f4_K$ for the calculation of the *SSD*;

The IS-41 SN

- may have to allow for a transmission of a longer random number value R_{SSD} in course of an *SSD Update* protocol.

The UMTS/ IS-41 gateway

- has to be able to run UMTS authentication data request/response procedures.
- has to be able to convert the received UMTS quintuple into parameters needed for an IS-41 *SSD Update* protocol.
- after establishment of an *SSD* has to be able to run IS-41 security protocols with the UMTS user and the IS-41 SN.

6.7.4.3 Registration of an IS-41 user in a UMTS serving network

After initial registration in a UMTS SN, the UMTS SN needs to have UMTS quintuples and the IS-41 user needs to have the key material available on the basis of which the quintuples were calculated. In the initial registration protocol described below also the gateway after initial registration posses the same key material as the user. Therefore, as long as the IS-41 user is roaming in UMTS networks the gateway is able to provide further quintuples to the UMTS SN and the according security protocols can be carried out between IS-41user, UMTS SN and GW which acts like a UMTS HE of the IS-41 user.

Figure 6-22 provides an overview of the procedure for initial registration which can be roughly described by the following steps:

(1) The IS-41 user determines that a UMTS SN has been entered and sends a UMTS *Registration Request* to the UMTS SN which forwards it as an UMTS *Authentication Data Request* (request for authentication vectors) to the GW (as the presumed UMTS HE of this user).

(2) The GW, acting as an IS-41-SN, demands *SSD Update* by the IS-41 HE/AuC by "abuse" of a message indicating a security violation in the SN. (Specific parameter values may have to be reserved in [19] for this case. It is unclear whether there are other possibilities for the IS-41 SN to trigger an *SSD Update*.) The IS-41 HE/AuC subsequently initiates an IS-41 *SSD Update* protocol. For *SSD Update* the so-called "*SSD* shared" variant of [19] is used, i.e., the *SSD* is shared between IS-41 HE/AuC and GW.

(3) The GW in a predefined way converts the IS-41 parameters received from the IS-41 HE/AuC to a UMTS quintuple and sends it to the UMTS SN using the UMTS *Authentication Data Response* message.

(4) Now IS-41 user and UMTS SN run the UMTS authentication protocol for registration, where the UMTS user in the same predefined way as the gateway converts its IS-41 parameters to the parameters needed for UMTS authentication. The IS-41 user instead of sending a usual authentication response, “improperly” uses the UMTS *Synchronisation Failure* message to the SN and the *Authentication Data Request* with *Synchronisation Failure* indication sent by the UMTS-SN to the GW to facilitate authentication to the gateway. (Specific parameter values may have to be reserved in [1] for this case.) This “abuse” of the *Synchronisation Failure* message is suggested as this is the only means in UMTS for the user to send a message to the HE/AuC (even if indirectly via the SN).

(6) After successful authentication of the IS-41 user by the gateway, GW in an IS-41 *SSD Update Report* message sends an indication to the IS-41 HE/AuC that *SSD* update was successfully completed.

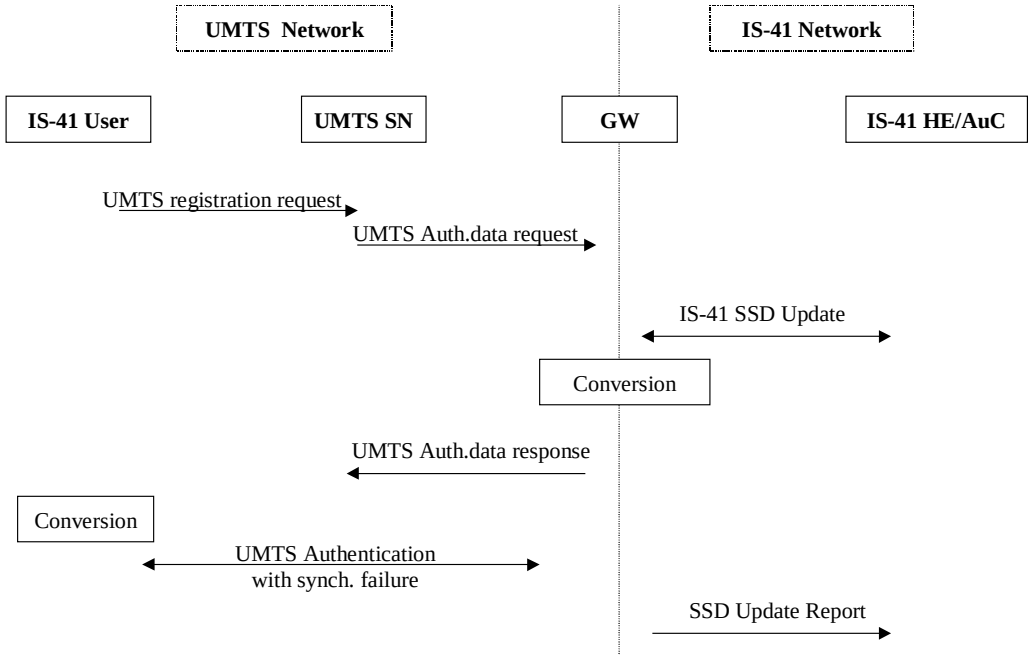

Figure 6-22: Overview - IS-41 users roaming in UMTS networks based on intermediate gateway components

The detailed description of the protocol and the parameters exchanged in course of the information flows is given below. It is depicted in Figure 6-23.

The IS-41 user determines that a UMTS SN has been entered and sends a UMTS *Registration Request* to the UMTS SN which forwards it as a UMTS *Authentication Data Request* for UMTS quintuples to the GW (as the presumed HE of this IS-41 user).

The GW, acting as an IS-41-SN, detects that the user is an IS-41 user roaming in a UMTS SN and demands an IS-41 *SSD Update* from the IS-41 HE/AuC of this user. This is facilitated by "abuse" of a message which is usually used to report a security violation in the SN. (A specific parameter value may be reserved in [19] to indicate this specific case to the IS-41 HE/AuC.)

Now the IS-41 HE/AuC initiates the *SSD Update* and therefore generates a random number R_{SSD} and calculates a pending value of the *SSD* from the random number R_{SSD} using *CAVE* under control of the user specific master key *A-key*. Note that the HE/AuC must retain both the current and pending values of the *SSD* until informed by the GW of the outcome of the updating procedure. The HE/AuC transmits the random number R_{SSD} and the new (pending) value of *SSD* to GW.

The GW stores the pending *SSD*, generates a random number R_D of 72 bits (remember that R_{SSD} has only 56 bits) and uses it together with the received data to form a UMTS quintuple (*RAND*, *XRES*, *CK*, *IK*, *AUTN)* by setting *RAND:*= R_{SSD} || *RD* and the other four parameters to a predefined value, e.g. "0". GW sends the quintuple *(*R_{SSD} || *RD, 0, 0, 0, 0)* to the UMTS SN in an UMTS *Authentication Data Response* message.

Then the UMTS SN initiates the UMTS AKA and therefore as usually forwards *(RAND, AUTN)*, i.e., *(*R_{SSD} || *RD, 0)* to the IS-41 user. The received *AUTN* = 0 is ignored by the IS-41 user.

The IS-41 user computes the pending value of the *SSD* from the random number R_{SSD} using *CAVE* under control of the user specific master key *A-key* in the same way as it is done in the IS-41 HE/AuC. In order to authenticate himself to the GW the IS-41 user abuses the protocol used in UMTS in case of synchroni-

sation failures. The IS-41 user calculates the UMTS parameter *AUTS:= SQN ⊕ AK || f1*$_K$(SQN, RAND, MODE)* using the previously received value RAND and the following settings: *SQN:=0, AK:=0, K:= SSD, MODE:= special value.* I.e., *AUTS:= 0 || f1*$_{SSD}$(0, R_{SSD} || RD, MODE).*

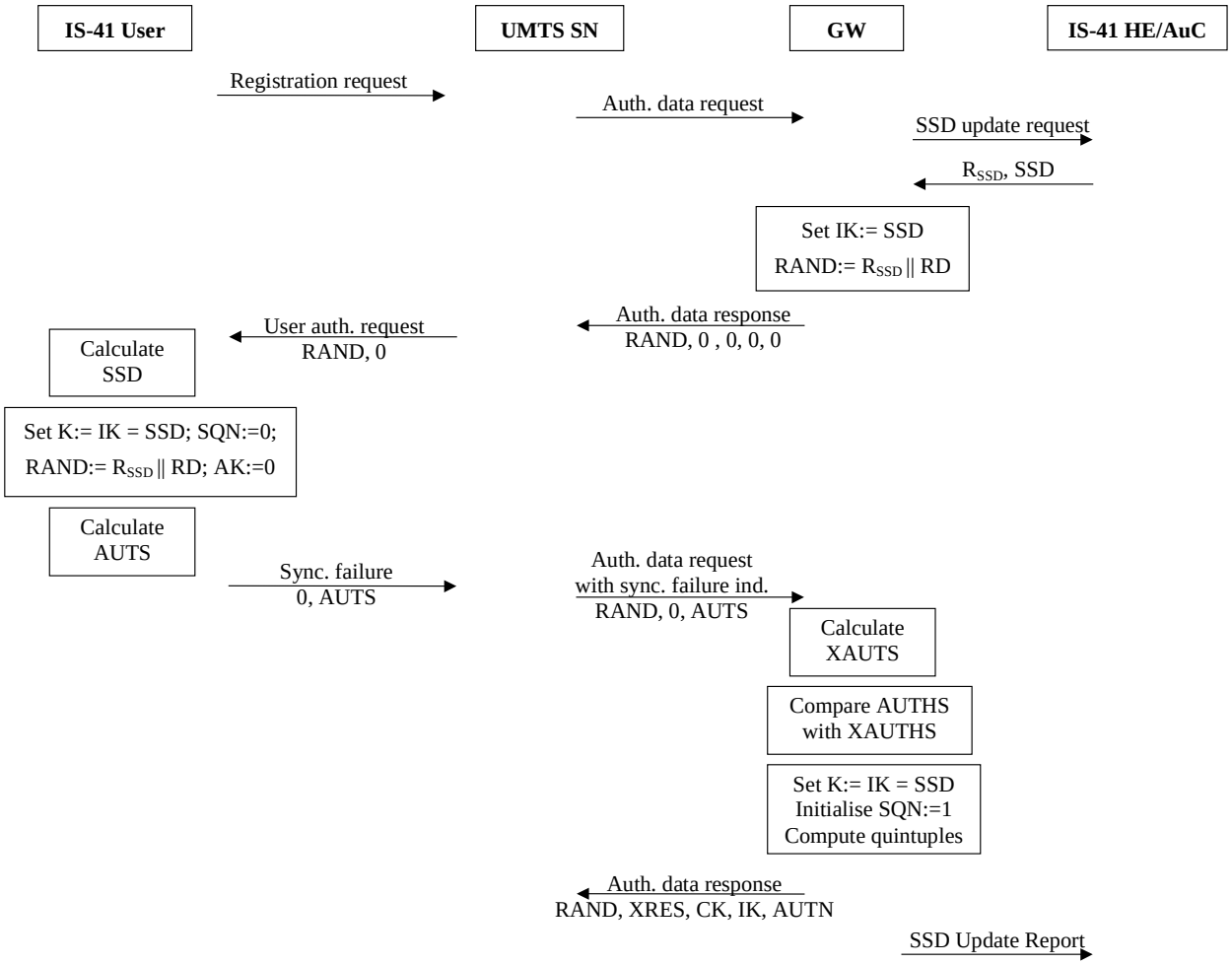

Figure 6-23: Details - IS-41 users roaming in UMTS networks based on intermediate gateway components

The user sends a UMTS *Synchronisation Failure* command with the parameters *(0, AUTS)* to the UMTS SN. The SN adds R_{SSD} || *RD* to the parameters and sends an *Authentication Data Request with Synchronisation Failure Indication* command with parameters *(R_{SSD} || RD, 0, AUTS)* to the gateway. GW determines that it is not really the case of a synchronisation failure, calculates an expected value *XAUTHS* analogously to the calculations of the IS-41 user and checks if *XAUTHS* is equal to the received value *AUTS.*

GW now computes a set of UMTS quintuples (*RAND*, *XRES*, *CK*, *IK*, *AUTN)* using the previously established temporary key *SSD* as substitute for the permanent UMTS key *K*. In the subsequent UMTS *Authentication Data Response* message the GW sends these quintuples to the UMTS SN. These quintuples can further on be used to run UMTS AKA protocols between the UMTS SN and the IS-41 user.

After successful authentication of the IS-41 user by the gateway, GW in an IS-41 *SSD Update Report* message sends an indication to the IS-41 HE/AuC that update of the *SSD* was successfully completed.

Notes

1. In the UMTS AKA protocol for the calculation of the quintuples for a UMTS user a sequence number *SQN* in the correct range is required. A security requirement for a such a sequence number is that it shall never be used twice for this specific user (with the same key).

   For usual IS-41 users such kind of sequence numbers are not available. The gateway may provide sequence numbers for roaming IS-41 users in UMTS networks in the following way:

   Each time when the IS-41 user initially registers in a SN (even if he has already roamed in this SN before) the sequence number of this user in the involved gateway is set to the value one and on the user side the sequence number is reset to the same value. The application in the IS-41 user's terminal has to allow for that. For each calculated quintuple (i.e., for each authentication attempt) the sequence number is incremented by one.

   Obviously sequence numbers for IS-41 users defined in the way described above will usually be used more than once. But in difference to the UMTS user case where always the same user specific permanent key *K* is used for the calculation of quintuples, in the scenario described here, each time the user demands initial registration an unpredictable temporary IS-41 key *SSD* is used. This reduces the requirement to the sequence number for an IS-41 user to the following: For the life-time of the key *SSD*, the sequence number of this user shall never be used twice.

2. It may seem contradictory that no freshness guarantee for the key *SSD* is given to the user while this is the case for the UMTS cipher and integrity keys derived from *SSD*. This, however, merely reflects the properties of the IS-41 security protocols: The establishment of the temporary authentication key *SSD* under the control of the permanent *A-key* is not replay protected (no network authentication here), but the subsequent IS-41 protocols under the control of *SSD* provide mutual authentication. (This situation is similar to the original TETRA protocol.)

   The IS-41 user roaming in UMTS only gets IS-41 grade security.

3. It may be wise to establish a means to replace *SSD* from time to time. A suitable trigger is ffs.

Requirements for the entities involved

In order to support the protocols described above, the following additional capabilities are required at the different entities involved:

The IS-41 user

- has to be able to run a modified UMTS AKA using the previously calculated temporary IS-41 key *SSD* instead of a usually used UMTS permanent key *K*;
- has to be able to abuse the UMTS *Synchronisation Failure* procedure to authenticate himself to the GW and to use the previously calculated *SSD* as a key for authentication to the GW;
- has to be able to handle a UMTS sequence number which is allowed to be reset in each initial authentication procedure when roaming in an UMTS SN.

The IS-41/UMTS gateway

- has to be able to calculate UMTS quintuples for UMTS SNs using the temporary IS-41 key *SSD* previously received from the IS-41 HE/AuC instead of a usually used long-term key *K*;
- has to be able to allow the abuse of the UMTS *Synchronisation Failure* procedure to facilitate authentication of the user to the GW.
- has to be able to handle a user specific UMTS sequence number which is allowed to be reset in each initial authentication procedure when roaming in an UMTS SN.

6.7.5 Security considerations for interoperation based on gateway components

6.7.5.1 General

In the interoperation scenarios based on gateway components described in section 6.7.4.2 and 6.7.4.3 key material from one mobile system is used in the other mobile system. In one scenario the temporary UMTS integrity key *IK* is used as a temporary IS-41 authentication key *SSD* and in the other scenario the temporary IS-41 authentication key *SSD* is used as permanent UMTS authentication key *K*. In this section the security implications posed to the involved systems are discussed.

6.7.5.2 Registration of a UMTS user in an IS-41 SN

This scenario is described in section 6.7.4.2 above. The temporary UMTS integrity key is used as temporary IS-41 authentication key *SSD*.

(a) **Implications on security parameters of UMTS and IS-41 systems.** On the one hand IS-41 systems taking *SSD* as an input use algorithms different from the ones used in UMTS systems and on the other hand UMTS algorithms taking *IK* as input are different from IS-41 algorithms. Therefore security parameters generated under control of *SSD* in IS-41 cannot be used in UMTS and vice versa.

(b) **Implications of compromised IK* in UMTS.** Suppose a specific temporary integrity key *IK** once used when a UMTS user was roaming in a UMTS network was compromised and an attacker also got hold the random number *RAND** of this old protocol run. Assume that the attacker wants to impersonate an IS-41 network. The attacker starts the protocol as in **Figure 6-21** but without contacting the gateway or the HE/AuC, and just replays the old value *RAND** as random challenge in the current protocol run. He thereby forces the UMTS user to reuse the temporary key *IK** (*=: SSD**).

Note that in IS-41 systems such a forced reuse is always possible, as these systems anyhow do not provide freshness of the temporary authentication key to the user. But in order to pass the subsequent network authentication within the *SSD Update* procedure an attacker could only successfully carry out the whole protocol run if he knows this key, which is assumed to be the case in the attack described here.

Therefore the compromise of an integrity key in UMTS affects the security of IS-41. However, even in case of a successful attack the roaming UMTS user only has to pay for services taken in IS-41 networks; he is not affected as long as he only roams in UMTS networks.

(c) **Implications of compromised SSD* in IS-41 systems.** Suppose a specific temporary authentication key *SSD** once used by a UMTS user when roaming in an IS-41 network (i.e., *SSD* : = IK**) is compromised in the IS-41 system. As long as the user roams in the IS-41 system he can no longer be registered in UMTS and therefore *IK** could not be (mis-) used at the same time in UMTS. A later misuse of *IK** is also not possible, as the UMTS AKA guarantees mutual key freshness, and an attacker would therefore not be able to force the usage of the same key *IK** for integrity protection in UMTS. Hence the compromise does not pose problems to the UMTS system.

6.7.5.3 Registration of an IS-41user in a UMTS serving network

This scenario is described in section 6.7.4.3 above. The temporary IS-41 authentication key *SSD* is used as permanent UMTS authentication key *K*.

(a) **Implications on security parameters of UMTS and IS-41 systems.** On the one hand IS-41 systems taking *SSD* as an input use algorithms different from the ones used in UMTS systems and on the other hand UMTS algorithms taking *IK* or *K* as input are different from IS-41 algorithms. Therefore security parameters generated under control of *SSD* in IS-41 cannot be used in UMTS and vice versa.

(b) **Implications of compromised K* in UMTS.** Suppose a specific temporary authentication key *K** once used when an IS-41 user was roaming in a UMTS network was compromised and an attacker also has the random number *RAND** of this old protocol run available. Then the attacker is able to impersonate an IS-41 network by sending *RAND** within an *SSD Update* procedure and thereby forcing this user to reuse *SSD** (*=: K**). Note, however, that K* is on the UMTS network side only

available in the clear in the GW (which has to provide the same level of security against disclosure of key material as the HE/AuC) but not in any UMTS SN.

Hence for an IS-41 user who has subscribed to roaming capabilities in UMTS networks, the compromise of an integrity key in UMTS affects the security of the IS-41 user even when he is roaming in IS-41 networks.

(c) **Implications of compromised SSD* in IS-41 systems.** Suppose a specific temporary authentication key *SSD** once used by an IS-41 user when roaming in an IS-41 network was compromised and an attacker also got hold of the random number *RAND** of this old protocol run. Assume that the attacker wants to impersonate a UMTS network. By sending *RAND** to the IS-41 user the attacker can force the IS-41 user to use the same key *K*:=SSD** again in UMTS.

Note that in IS-41 systems such a forced reuse is always possible, as these systems anyhow do not provide freshness of the temporary authentication key to the user. This means that an IS-41 user roaming in a UMTS network only gets IS-41 grade security. However, the compromise of the temporary authentication key *SSD* in IS-41 affects the security of the IS-41 user when roaming in UMTS. But even in case of a successful attack the roaming IS-41 user only has to pay for services taken in UMTS networks; he is not affected as long as he only roams in IS-41 networks.

As a conclusion of the discussion in this section it should be noted that the solution proposed in section 6.7.4 has the property that the compromise of keys in one system may affect the other system. It has to be further discussed whether this property is acceptable to operators. Note, however, that UMTS users are not affected by key compromises in IS-41 as long as these UMTS users roam inside UMTS.

6.7.5.4 Remarks on key lifetimes

The IS-41 key *SSD* and the UMTS keys *K* and *IK* have different lifetimes. The key with the longest lifetime is the UMTS key *K*, as it is a permanent key. Temporary IS-41 authentication keys *SSD* should usually live longer than temporary UMTS integrity keys *IK*, because *IK* changes each time the UMTS AKA is carried out but *SSD* in IS-41 may remain unchanged even when a user roams into a new network. These different lifetimes may have implications on the security of the authentication schemes for interoperation and may therefore pose requirements on the frequency in which these keys may have to be updated when used in the other system.

In case of a UMTS user roaming in an IS-41 network, *SSD* is derived from IK (cf. 6.7.4.2). According to the shorter lifetime of *IK* in UMTS a more frequent *SSD Update* than for usual IS-41 users in IS-41 systems should be initiated by the gateway.

In case of an IS-41 user roaming in a UMTS network, *K* is derived from SSD (cf. 6.7.4.3). According to the shorter lifetime of *SSD* in IS-41 systems a frequent update of *K* initiated by the gateway is recommended.

6.8 Data confidentiality

6.8.1 General

In the following section we discuss the progress that was made on the open issues as regards ciphering and the issues that were proposed in the period between April and June 1999.

6.8.2 Inputs to the ciphering algorithm

The June release of the cryptographic algorithms specification [5] describes the input parameters to the ciphering algorithm (see Figure 6-24).

Compared to the external specification of the ciphering algorithm A5 [11] (annex C), three new parameters are added: BEARER, DIRECTION and LENGTH.

The input parameters to the algorithm are the following:

- **A cipher key (CK).** (128 bits) This is the key agreed between the MS/USIM and the SN/VLR during or after the last authentication and key agreement mechanism. The cipher key is subsequently transported from these entities to the entities at each side of the radio link, that allocate ciphering, i.e., the UE and the RNC. Since the MS/USIM may have different cipher keys agreed with different SN/VLR that are connected to the RNC, the RNC and the UE have to select the cipher key that will use. This is discussed in 6.8.5.
- **A time dependent input (COUNT).** (32 bits) This is a value that is incremented at both sides of the radio link to assure the differentiation over time of the keystream blocks. The range should be large enough to avoid repetition. How COUNT is synchronised is detailed in 6.8.4.
- **A bearer identity (BEARER).** (8 bits) The inclusions of this identity assures that different keystream block are used to cipher plaintext blocks that are simultaneously transmitted over different logical channels.
- **A direction of transmission (DIRECTION).** (1 bit) Similarly, the inclusion of this parameter this assures that different keystream blocks are used to cipher upstream and downstream plaintext blocks.
- **The length of the keystream block (LENGTH).** This input parameter shall affect only the length of the keystream block, not the actual bits in it.

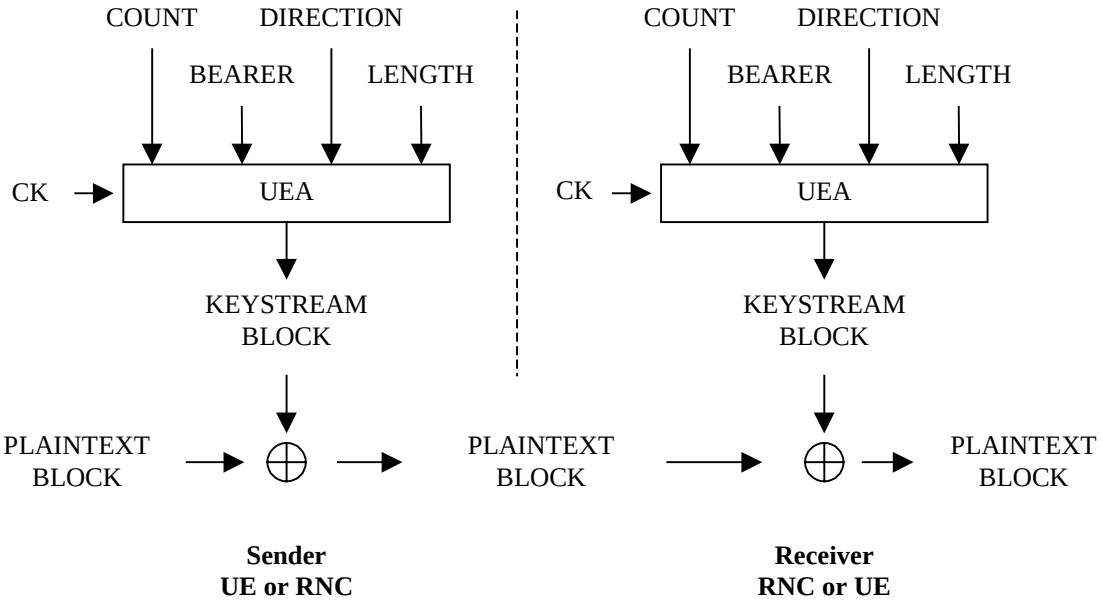

Figure 6-24: Ciphering user and signalling data transmitted over the radio access link

Based on these input parameters the algorithm generates the output keystream block (KEYSTREAM) which is used to encrypt the input plaintext block (PLAINTEXT) to produce the output ciphertext block (CIPHERTEXT).

6.8.3 Integration of ciphering in the radio access network protocol architecture

Encryption will be applied in the Medium Access Control (MAC) sublayer or in the Radio Link Control (RLC) sublayer of the data link layer (Layer 2) according to the rules specified in [21]. According to the model there, ciphering is always performed in the SRNC, and the context needed for ciphering (CK, HFN, etc.) is only known in SRNC.

When ciphering is performed in the RLC sub-layer, it performs the encryption/decryption of the data part of an RLC PDU. When ciphering is performed in the MAC sub-layer, it performs the encryption/decryption of a MAC SDU (RLC PDU).

6.8.4 Synchronisation

To produce different cipherstream blocks at each side of the radio link, the time variant parameter COUNT (a.k.a. the Ciphering Sequence Number (CSN)) is input to the ciphering algorithm (see XXX). To recover the plaintext it is essential that the receiver and the sender use the same value of COUNT. Also, in order not to repeat, the range of COUNT should be sufficiently large, and in order to not be smaller than the range for 2G GPRS, it should have a length of at least 32 bits. How ciphering is synchronised is again detailed in [21].

COUNT consists of a short part (SHORT) and a long part (LONG, a.k.a. Hyperframe Number (HFN)). The length of SHORT (and as a consequence of that, the length of LONG) and what it is derived from depends on the mode of transmission:

- **RLC TM (Transparant mode):** SHORT = CFN (7 bits). CFN is not included in the MAC-header, and transmitted over the radio access link, instead it is independently maintained at the UE MAC sublayer and the RNC MAC-d sublayer. The CFN is incremented every 10 ms physical layer time slot. When CFN reaches 72 it is reset to 0 and the value for LONG (25 bits) for that logical channel is incremented by 1.
- **RLC UM (Unacknowledged mode):** SHORT = RLC SN (7 bits). The RLC SN is part of the RLC header and sent over the radio access link. The RLC SN is incremented for each RLC PDU. When RLC SN reaches 128 it is reset to 0 and the value for LONG (25 bits) for that logical channel is incremented by 1.
- **RLC AM (Acknowledged mode):** SHORT = RLC SN (12 bits). The RLC SN is again part of the RLC header. The RLC SN is incremented for each RLC PDU. When RLC SN reaches 4096 it is reset to 0 and the value for LONG (20 bits) for that logical channel is incremented by 1.

The proposed lengths would cause the COUNT to repeat after $2^{25} \times 72 \times 10$ ms = 280 days. This is much longer than the expected lifetime of any access link key pair which is expected to be at most one or several days. For RLC UM and RLC AM the period depends on the number of RLC PDUs sent over the radio access link. More than one RLC PDU may be send in one Layer 1 frame of 10 ms. However, the number of RLC PDUs that will actually be sent in any conceivable scenario is such that over any practical time period, the COUNT values for RLC UM and RLC AM bearers are expected to fall behind the COUNT for the RLC TM bearers.

In [21] is also specified how the initial values of COUNT are synchronised. At the RRC connection establishment a parameter START is sent by the MS to the RNC before ciphering is started. It is used as the initial value for the parameter LONG of the bearers to be set-up. (In case of RLC AM, only the 20 most significant bits are used).

A different COUNT parameter is subsequently maintained for the different logical channels in the way described above; i.e., a separate COUNT is maintained for each RLC UM and RLC AM logical channel, and a single RLC TM for all RLC TM logical channels. When a new RLC UM or RLC AM logical channel is added to the connection, the highest value of LONG currently in use (this will typically be $LONG_{TM}$) is used as the initial value for LONG of the new logical channel. (In case of RLC AM, only the 20 most significant bits are used).

6.8.5 Cipher key (and integrity key) selection

6.8.5.1 General

The April release of the security architecture [3] describes two rules for access link key pair (cipher key and integrity key) selection, commonly referred to as the “one key option” and the “two key option.” The issue is due to the possibility that the RNC is connected to two (or more) core network nodes that each conduct separate mobility management, and agree on different access link keys with the mobile user. The issue occurs when users simultaneously use CS and PS services. The question is which key pair to use to protect the traffic?

- In the "one key option" at any given time instant a single key pair is used to protect all data exchanged between UE and RNC. In this case a second question immediately follows: which key pair should be used?
- In the "two key option" two key pairs will be active when a user accesses simultaneously a service delivered via the 3G SGSN and a service delivered by the 3G MSC/VLR.

The "two key option" was preferred by the security group both for signalling and user traffic. In the one key option both core networks have to rely on the security of the other core network, which is felt to be a disadvantage although according to the existing role model, both core networks belong to the same SN.

6.8.5.2 User data

As far as the ciphering of the user data channels is concerned, the following arguments can be brought forward:

- the "two key option" has the security advantage that the core network is assured that its own cipher key is used; this allows the network operator to independently introduce security measures in both core networks;
- the "one key option" has the security advantage that the most recent cipher key is always used (it may be desirable to update a long packet session, each time a CS call is set-up);
- the "two key option" has the advantage that the cipher keys of on-going user data channels are not updated; an update of a cipher key might lead to complications with re-transmission schemes and might lead to data-loss;
- the "one key option" has the advantage that the UE and the RNC have to manage fewer keys and that the cipher key selection mechanism is identical to the one applied for signalling channels;
- a further advantage of the "two key option" for user data channels in combination with separate common logical channels for signalling data is that it has as a built-in property the ability to cipher signalling channels and user data channels with different cipher keys, which is a requirement for the mechanisms proposed for network-wide encryption (see 5.5 and 6.9).

The "two key option" for user data appears to have become the working assumption in the RAN-2 as well as in the SA-3 group.

6.8.5.3 Signalling data

The radio access network architecture contains logical channels used for signalling data that support both CS and PS services. To use two key pairs on those channels would introduce a major complication and as a result, for signalling data, the "one key option" appears the only viable option.

For signalling data we thus must answer which key pair to use when both the CS and the PS domain provide an access link key pair. The answer to this question is closely linked to that of the network control over key lifetime and what the network considers to be the "most secure" key. The April release [3] here suggests (or merely specifies) that the RNC and the UE select "the most recently established key pair." The text is confusing and apparently assumes that the start of a service always involves an authentication and key agreement:

- It reads: "*This requires that the cipher key of an (already ciphered) ongoing signalling connection is changed.*" → This should only be the case when the key pair that comes with the service that is started later has a more recently established access link key pair (the time of agreement should come from the CN/VLR and the USIM, together with the key pair;
- It reads: "*This change should be completed within five seconds after an authentication and key establishment protocol has been executed.*" → It should be within a reasonable time after the second service is started.

6.8.6 Cipher key (and integrity key) lifetime

6.8.6.1 General

The April release of the security architecture [3] (6.6.6) states

> "A mechanism is needed to ensure that a particular cipher key is not used for an unlimited period of time to avoid attacks using compromised keys. (…) The USIM shall (…) contain a mechanism to limit the number of calls that can be made with a specific cipher key. Operators shall decide on the value of this number of calls, and write this parameter on the USIM. (…)"

Note that it is the unlimited usage of a cipher key that needs to be avoided, rather than "the number calls" that has to be counted and restricted. An extensive note in [3] (6.6.6) indeed says that the decision on when a key needs to be updated may depend on a number of factors, such as the time since the last key update, the amount of data protected using the key and the cost/value of the services protected using that key. The note concludes saying that the number of calls may not be a good measure.

In addition we can add that not only the user but also the network should control the access link key lifetime.

6.8.6.2 User control

The current working assumption was proposed in [25] and is already described to some extent in 6.8.4. For the lifetime of a connection, the increase of the time-varying parameter COUNT, i.e., the ciphering sequence number (CSN) is for RLC TM logical channels a measure of the time and for RLC AM and RLC UM logical channels a measure of the number of RLC PDU, i.e., the number of data, that is transmitted. Both are thus good measures of the usage of the cipher key. The mechanism that avoids repetition of COUNT, i.e., the storage in the USIM of the last value reached during a certain connection and the usage of that value to initialise COUNT at the start of the following connection thus provides a measure for the usage of the key pair.

It is then proposed that the USIM contains a THRESHOLD value and that the USIM at connection set-up decides whether a new authentication and key agreement is required.

Note that the mechanism can be easily enhanced to maintain a different measure that reflects more the usage, i.e., the amount of data protected under a certain key pair:

- currently the value START stored in the USIM is incremented at the end of a connection by the maximum of the increments of the different COUNT values:

$$START_N - START_O := Max\left\{COUNT_{N,i} - START_O \middle| i \in C\right\}$$

- the value START stored in the USIM could be incremented by the sum of the increments, rather than with the maximum of all increments:

$$START_N - START_O := \sum_{i \in C}\left(COUNT_{N,i} - START_O\right)$$

- in addition one might take into account the different bandwidth and cost associated to the different logical channels:

$$START_N - START_O := \sum_{i \in C} a_i\left(COUNT_{N,i} - START_O\right)$$

The values of a_i should be characteristic for the type of connection and may be specified for each connection type or sent over the radio link at the set-up of a logical channel.

6.8.6.3 Network control

Also the network should limit the lifetime of the active access link key pair. This, to prevent that an intruder who has obtained an active access link key pair might use it for an indefinite amount of time and an indefinite number of services.

The network has much more resources to monitor the use of the access link keys and may at each connection set-up decide to execute an authentication and key agreement. However, note that also the SN already needs to store information on the key lifetime of the cipher key: namely the time that the key pair was agreed. The core network needs to store that information in order to allow the RNC to select the "most secure" key pair to protect signalling. As in the USIM the SN should use that time and compare it with current time and ensure in that way that an active key pair is no longer than a threshold set by the SN operator (somewhere in the order of a day).

6.8.7 Ciphering (and integrity) algorithm negotiation

The negotiation of the security mechanisms should occur in a safe way. To achieve this, the network entity that is to decide on the encryption and integrity mode, should have assurance about the data integrity and the data origin of the security capabilities received from the user.

The negotiated algorithms should, off course, be chosen from the intersection of the capability sets of both the RNC and the UE. In addition, according to the text part of the RANAP specification [34], the CN node sends the RNC a set of ciphering and integrity capabilities acceptable to the CN. This yields to a rather complicated negotiation.

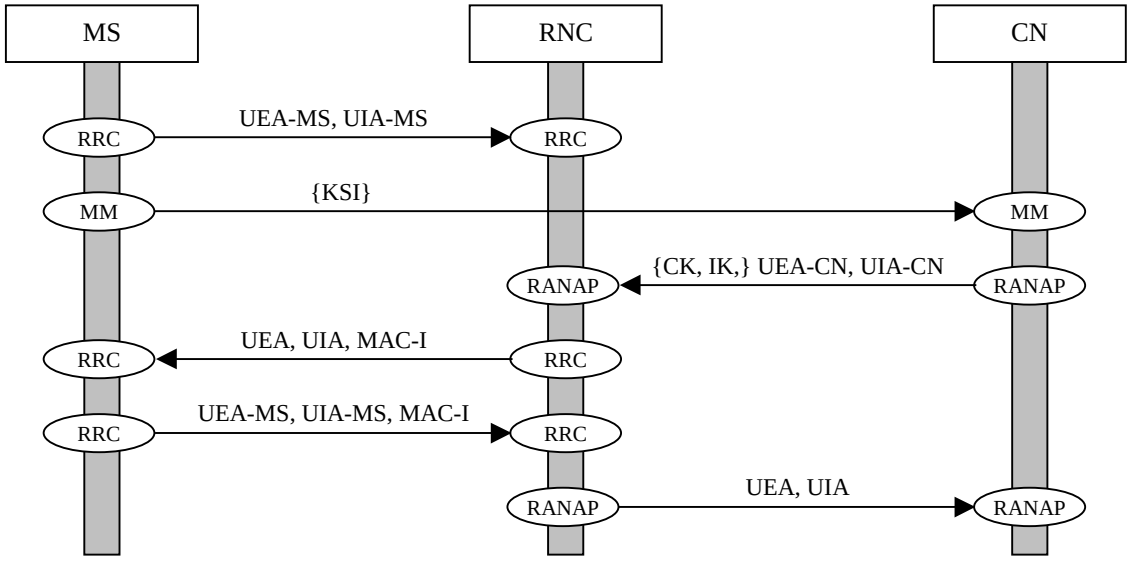

Figure 6-25: Negotiation of the encryption and data integrity capabilities

The complete negotiation is shown in Figure 6-24. The UE starts by sending the RNC the user capabilities UEA-MS and UIA-MS. The UE subsequently sends the identifier KSI to the CN. The CN then decides whether he initiates an authentication and key agreement. If that is the case, a new KSI is assigned.

The CN then sends a RANAP/security mode command to the RNC. In that message the CN indicates which cipher and integrity key should be used, as well as the set of encryption and integrity capabilities UEA-CN and UIA-CN are acceptable to the CN.

Note: The need for the CN to indicate the set of encryption and integrity capabilities acceptable to the CN is questionable, unless the different algorithms really constitute different levels of security that are to be applied to different services. Note that in the tables in [34] no such parameters are included.

The RNC then sends the RRC/security mode command message to the UE. It contains the encryption and data integrity mode. The message is mandatory and is the first one for each connection that is signed with a message authentication code for integrity. The RNC starts decryption of the receiver channel immediately after it has sent the cipher mode command.

The UE verifies the data integrity and data origin of UEA and UIA and enables encryption and data integrity. The UE now mandatorily sends the RRC/ciphering confirm message to the RNC that includes the initially sent UE encryption and integrity capabilities, but now data integrity protected. Only after successful verification of the integrity of the UEA-MS and UIA-MS the negotiation is successfully terminated. The RNC also enables encryption on the down-link and informs the CN of the selected encryption and integrity mode.

6.9 Data integrity

6.9.1 General

In the following section we discuss the progress that was made on the open issues as regards data integrity and the issues that were proposed in the period between April and June 1999.

6.9.2 Signalling messages that require integrity protection

The April release of the security architecture specification [1] contains a list of signalling messages that should be integrity protected, to counteract certain false base station attacks [6]. That list consists of:

- **MS capabilities** (GSM: MS Classmark) including a list of the authentication mechanisms, the ciphering algorithms and the data integrity functions that are available in the MS;
- **Security mode command/confirm message.** (GSM: cipher mode command/confirm) whereby the command includes the ciphering and integrity algorithm selected by the network, and the confirm includes the capabilities protected with data integrity.
- **Called party number in MOC.** This is to limit the value of channel hijacking in case no in-call re-authentication messages are sent.
- **In-call re-authentication messages.** These are signalling messages that are send during a connection and by means of the use of the integrity key on some fresh data re-authenticate the network and the user. They are sent at the initiative of the network, and may be send by the network periodically. Hence they are called “periodic data integrity messages.”

This list was to be considered as a minimum. In a further contribution [26] also a less involved attack was considered: one that only requires a false mobile station which spoofs a critical signalling message with the objective to hijack a channel. That contribution concluded that in addition the following RRC signalling messages needed protection:

- **Cell update (and URA update):** messages for in-call mobility management;
- **Inter system cell reselection from GSM/GPRS to UTRAN:** messages for intersystem handover;
- **Transmission of UE capability information:** used to transmit the MS capabilities (see above);
- **Direct transfer messages**: used to transmit the MM messages (see above).

It should be noted however that the stealing of a connection by means of a cell update, i.e., by means of a false mobile station, only works until the next in-call re-authentication message, just as is the case with the false base station attacks. Nevertheless, it has some value to protect the cell update separately, since the required investment to engage in the attack is much smaller.

6.9.3 Allocation of the data integrity function

At the **network side** the working assumption is that the integrity functionality is allocated to the **RNC**. This allows the protection of signalling messages that terminate at the RNC (such as the cell update mes-

sages), and the implementation may profit from the integration with the existing functionality for ciphering.

A recent contribution to the RAN-2 group however advocates an allocation to the **VLR**. This would allow integrity protection to apply also to GSM BSS attached to UMTS VLRs. Further it is argued that a limited set of messages that originate from the SN/VLR is sufficient to protect against hijacking attacks.

At the user side, the working assumption is that the integrity functionality is allocated to the **UE**. The main concern here is the limited resource available on the USIM despite the answer of the USIM group (TSG T WG 3) saying that the computation of message authentication function on the USIM would not cause capacity problems on the smart card nor on the UE-USIM interface. Allocation of the message authentication function on the USIM would have as an advantage that the integrity key would never leave the USIM. It would, for instance, prevent the misuse of an integrity key in a UE after the USIM has been removed from the UE.

In the remaining of this document we apply the working assumption that the integrity functionality is applied to the UE and the RNC.

6.9.4 Inputs to the data integrity algorithm

Figure 6-24 shows the symmetric key mechanism for data integrity and the input parameters to the UMTS integrity algorithm (UIA).

The input parameters to the algorithm are

- **An integrity key (IK).** (128 bits) This is a key agreed between the MS/USIM and the SN/VLR during or after the last authentication and key agreement mechanism. The integrity key is subsequently transported from these entities to the entities at each side of the radio link, that allocate the data integrity algorithms, i.e., the UE and the RNC. Since the MS/USIM may have different integrity keys agreed with different SN/VLR that are connected to the same RNC, the RNC and the UE have to select the cipher key that will use on the signalling bearers.
- **A time dependent input (COUNT).** (32 bits) The input parameter COUNT protects against replay during a connection. The range should be large enough to avoid repetition. How COUNT is synchronised is detailed in 6.9.5.
- **A network challenge (FRESH).** (32 bits) The input parameter FRESH protects network against replay of signalling messages by the user. At connection set-up the network generates the unpredictable value FRESH and sends it to the user. The value FRESH is subsequently used by both the network and the user throughout the duration of a single connection. This mechanism assures the network that the user is not replaying any old MAC-Is.
- **The signalling message (MESSAGE).**

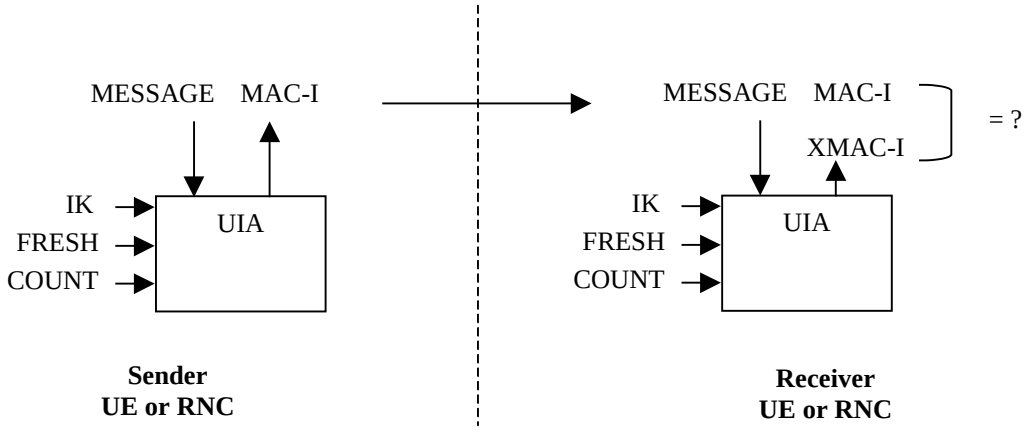

Figure 6-26: Data integrity mechanism for signalling messages

Based on these input parameters the user computes with the function the data integrity code for data integrity (MAC-I) which is appended to the signalling message when sent over the radio access link. The receiver computes XMAC-I on the messages received in the same way as the sender computed MAC-I on the message sent and verifies the data integrity by comparison of XMAC-I and MAC-I.

6.9.5 Integration in the UTRAN architecture

If also messages originating and terminating in the RNC require data integrity protection, the RRC layer appears to be the proper place to allocate the data integrity function . Synchronisation would be achieved using a separate counter that has to be introduced in the RRC header uniquely for this purpose [25].

The reason why data integrity cannot be an RLC or MAC function are that unlike for ciphering, also signalling messages that may be sent on a common control channel (CCCH) require data integrity protection (such as the 'Cell Update Request'). For these CCCH, the transparent RLC mode is used, thus no "time varying parameter" to prevent replay is available in the RLC layer. The MAC layer "time varying parameter" is the CFN and the reasons why it cannot be used in RACH transmission were discussed when the decision of ciphering layers was done [27].

It is further proposed that the sequence number COUNT that is input to the message authentication function is thus derived from counters at each side of the radio access link (one counter pair for up-link and one for down-link traffic.) To keep peer RRC entities synchronised even in case of a message is lost, the *N* least significant bits (LSB) of the counter may need to be added into the RRC PDU. Before setting an exact value for *N*, it needs to be clarified which RRC messages will be integrity protected and how many RRC signalling procedures can be active in parallel.

6.10 Alternative key management for network-wide encryption

6.10.1 Introduction

Key management for network-wide encryption enables two UEs (UEs) on the same SN to establish a shared secret key for use in end-to-end encryption. It is assumed that the UEs share an access link cipher key with their VLR (and also with their RNC). The establishment of this access link key is assumed to be an automatic by-product of the UE/VLR authentication process, and hence is available 'for free'.

In the mechanism described in [3] and also in 5.5.3.2, to establish a shared secret, the VLRs of the two UEs simply exchange the two UEs' access link cipher keys K_a and K_b. The two VLRs then send the newly acquired keys to the respective UEs and finally, both UEs compute the shared secret key as $K_s = f(K_a, K_b)$.

As mentioned above, the main motivation for using the access link cipher keys to establish the shared secret key is that these keys are present in any case (they are assumed to be an automatic by-product of the authentication process) and hence they are available for use 'for free'.

One other motivation for the scheme is that both VLRs retain the capability to compute the session key K_s, which is desirable for key recovery purposes.

We observe however the following:

- **Efficiency.** Despite the fact that the keys K_a and K_b already exist, they have to be exchanged by the two VLRs. This means that at least two messages have to be exchanged between VLRs to establish each session key, and so it hardly fair to claim that the process is 'free'. Indeed, as we discuss below, there are plenty of key establishment schemes with the same or even smaller overhead, which accomplish the same purpose.
- **Security.** The design of the scheme means that VLR_b, RNC_b and UE_b all have access to UE_a's access link cipher key K_a. This may be highly undesirable if this key is used to protect sensitive data transferred to and from UE_a.

In the following paragraphs we therefore consider some alternative schemes for key agreement. The first one using asymmetric key techniques (6.10.2), the second one using symmetric key techniques (6.10.3).

6.10.2 Diffie-Hellman key agreement

Preliminaries

We assume that there is a globally agreed modulus *p*, and also a globally agreed 'base' *g* (*g* will typically be an element of large prime multiplicative order modulo *p*). All exponentiations are to be computed modulo *p*.

Protocol description

As in the previous solution, UE_a and UE_b establish a shared secret key using their respective access link secret keys K_a and K_b. The scheme is very similar to the schemes in 5.5.3.2 and 5.5.3.3.

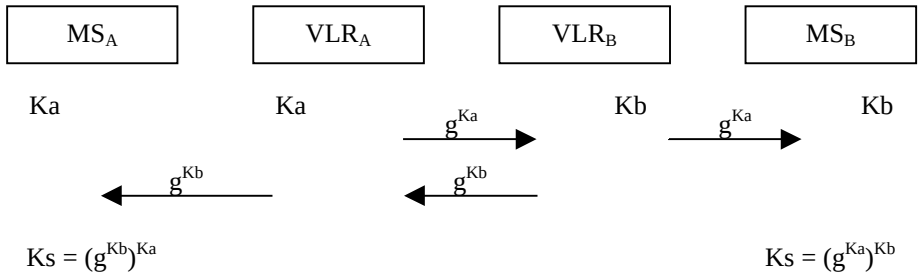

Figure 6-27: Diffie-Hellman key agreement

The main differences with the mechanism in 5.5.3.2 are as follows. Instead of exchanging K_a and K_b, VLR_a and VLR_b exchange g^{Ka} and g^{Kb}. This exchange does not need to preserve secrecy (as was the case in the previous scheme), although it does need to preserve data integrity. Thus VLR_b does not learn the secret key K_a, and VLR_a does not learn the secret key K_b. VLR_a then forwards g^{Kb} to UE_a, and VLR_b forwards g^{Ka} to UE_b.

UE_a can now compute

$$K_s = (g^{Kb})^{Ka} = g^{KaKb}.$$

Similarly, UE_b can compute

$$K_s = (g^{Ka})^{Kb} = g^{KaKb}.$$

Comments on the scheme

As already remarked, this scheme avoids the need for UE_a and UE_b to exchange keys K_a and K_b. However, the possibilities for key recovery remain undiminished. Thus it preserves the 'best' properties of the previous scheme, and avoids the main disadvantages.

6.10.3 A solution using conventional cryptography

Preliminaries

The second alternative solution we consider is taken from a recent paper by Hoyle and Mitchell [30]. This scheme has the advantage that it avoids any need for on-line communications between VLR_a and VLR_b. As previously, we assume that UE_a and VLR_a share a secret key K_a, and UE_b and VLR_b share the secret key K_b. Note that, as in [30], the described protocol does not provide entity authentication; this can be added by incorporating appropriate time variant parameters into the protocol. To use this scheme it is necessary that VLR_a and VLR_b share a long term secret key $K(VLR_a, VLR_b)$.

Protocol description

The description is an adapted version of the text appearing in Section 4.1 of [30].

1. UE_a sends VLR_a a request for a key for communicating with UE_b (and an indication of the identity of the VLR of UE_b):

   $UE_a \rightarrow VLR_a$: UE_b, VLR_b

2. VLR_a responds by sending UE_a a pair of tokens. Each contains a copy of the required session key K_s, one encrypted so that only UE_a can read it and the other so that only UE_b can read it. The encryptions are performed using keys K^*_a and K^*_b. VLR_a also sends K^*_a to UE_a, encrypted under K_a.

   $VLR_a \rightarrow UE_a$: $e_{Ka}(K^*_a)$, $e_{K^*a}(K_s, ID_{UEa}, ID_{UEb})$, $e_{K^*b}(K_s, ID_{UEa}, ID_{UEb})$

   where $e_K(X)$ denotes encryption of data X using key K, $K^*_a = f(K(VLR_a, VLR_b), ID_{UEa})$, and $K^*_b = f(K(VLR_a, VLR_b), ID_{UEb})$.

   UE_a first obtains the key K^*_a, and then decrypts the first token to obtain the session key K_s.

3. When UE_a wishes to commence secure communications with UE_b, UE_a protects the transmitted data using the session key K_s, and at the first opportunity sends both tokens to UE_b:

   $UE_a \rightarrow UE_b$: $e_{K^*a}(K_s, ID_{UEa}, ID_{UEb})$, $e_{K^*b}(K_s, ID_{UEa}, ID_{UEb})$

4. When UE_b receives the tokens, there are two possible cases to consider (we assume that UE_b knows which VLR UE_a is using). If UE_b already knows K^*_b (which is a function only of the identity of VLR_a), then the protocol is complete. If UE_b does not know K^*_b, then UE_b sends a message to VLR_b requesting a copy:

   $UE_b \rightarrow VLR_b$: VLR_a

5. VLR_a now computes K^*_b, using the function f and the shared secret key $K(VLR_a, VLR_b)$, and sends it back to UE_b:

   $VLR_b \rightarrow UE_b$: $e_{Kb}(K^*_b)$

   UE_b can now recover K_s.

Both

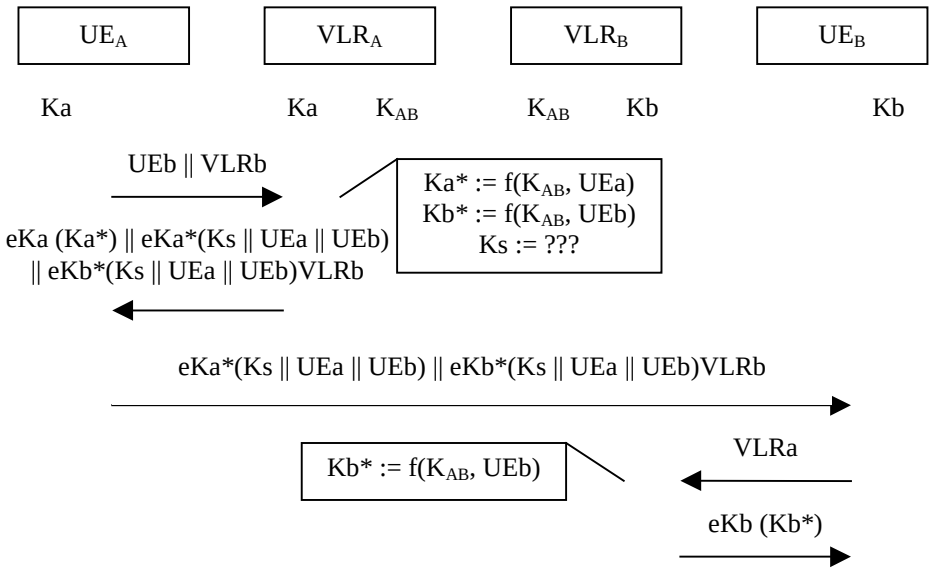

Figure 6-28: Key agreement using conventional cryptography

VLRs now possess all they need to perform key recovery, should they be required to do so (for further details see [30]).

Comments on the scheme

As already remarked, this scheme avoids any need for communication between VLRs. It also avoids the need for UE_b to obtain the key from VLR_b (except on the first occasion that a key is received from a client of VLR_a). Perhaps most importantly, it avoids the need for UE_a and UE_b to exchange keys K_a and K_b, and at the same time also avoids the use of any computationally expensive public key cryptography.

7. Integration in network entities

7.1 Functional network architecture

Figure 7-1 shows the functional security architecture of UMTS. The vertical bars represent the five network elements: the USIM issued by the HE, and the UE both in the user domain, the RNC and the VLR both in the serving network domain and the HLR/AuC in the HE domain. The horizontal lines represent the five security mechanisms: the mechanism for enhanced user identity confidentiality (which is optional), the conventional mechanism for user identity confidentiality (between user and serving network), the mechanism for authentication and key agreement, including the functionality to control the access key life time, i.e., to trigger re-authentication, the mechanism for user identity confidentiality and the mechanism for data integrity.

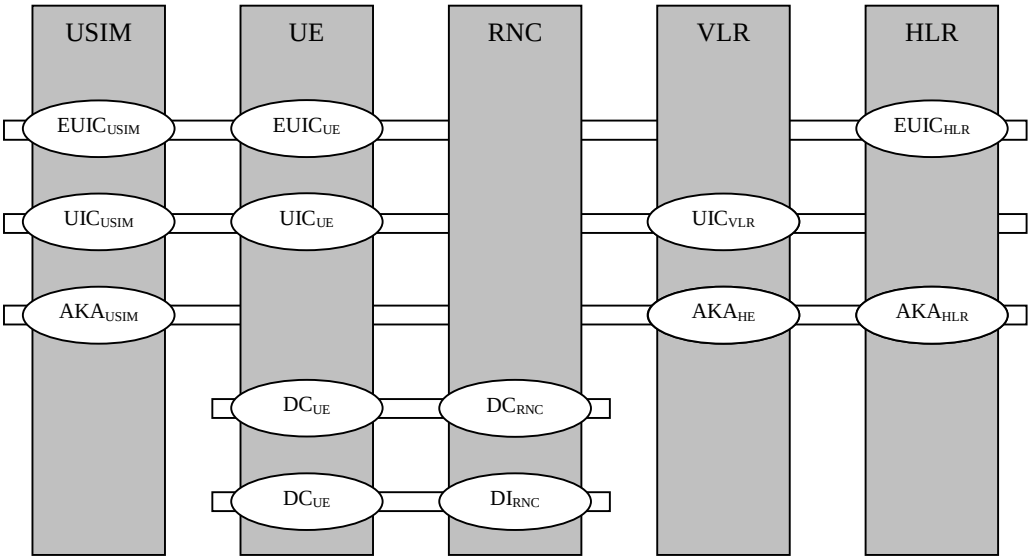

Figure 7-1: UMTS functional security architecture

In the remaining section of this chapter we describe what data elements and functions need to be implemented in each of the above network elements for each of the above mechanisms and functions.

7.2 User services identity module

7.2.1 Enhanced User Identity Confidentiality ($EUIF_{USIM}$)

UMTS users with enhanced user identity confidentiality, have to store additional data to encrypt the permanent user identity and derive a global temporary identity as an alternative for SN generated temporary identities in paging messages.

We describe the data elements that the USIM when the example mechanism described in 0 is implemented:

a) a counter $SQN_{UIC/MS}$ that is equal to the highest SQN_{UIC} generated and sent to the HE/AuC;

b) per user group the user belongs to: the group key GK (128 bits) to encrypt the IMUI and the SQN_{UIC};

c) a temporary user identity $TMUI_{HE}$ that is valid for user identification.

Also, the following cryptographic functions need then to be stored:

– f6: the user identity encryption function .

– f6*: the temporary identity derivation function .

For a summary of the data elements and cryptographic function of the $EUIC_{USIM}$ functionsee Table 7-1.

Table 7-1: USIM – Enhanced User Identity Confidentiality (optionally)

Name	Comment	Multiplicity	M/O
GI	Group identity	1 per user group	O
GK	Group key	1 per user group	O
$SQN_{UIC/MS}$	Sequence number	1 per user	O
$TMUI_{HE}$	Temporary user identity	1 per user	O
f6	Permanent identity encryption function	1	O
f6*	Global temporary identity derivation function	1	O

7.2.2 User identity confidentiality

Optionally the USIM may store the TMUI/LAI pairs for the serving networks in which the user is registered such that the UE can store these values after the UE has been switched off.

Table 7-2: USIM – Conventional User Identity Confidentiality

Name	Comment	Multiplicity	M/O
TMUI	Temporary Mobile User Identity	1 per user per mode	O
LAI	Location Area Identity	1 per user per mode	O

7.2.3 Authentication and key agreement

To support user authentication and key agreement the USIM stores:

a) the permanent secret key K (128 bits).

b) per mode: the value of the highest accepted sequence number SQN (32—64 bits).

   In addition, the “window” or “list” mechanism may be implemented to allow simultaneous use of various sets of authentication vectors in different core network nodes. In that case additional data need to be stored:

   – [WINDOW:] per mode: a Boolean array with elements that indicate whether the MS/USIM has accepted already the authentication tokens with sequence number in the range [SQN-w, SQN);

   – [LIST:] per mode: an (ordered) array of k-1 sequence numbers of the 2^{nd}, 3^{rd}, … k^{th} largest sequence number, in addition to SQN.

c) the value of the random challenge $RAND_{MS}$ (128 bits) received together with the highest accepted sequence number to support re-synchronisation. (Assuming the HE/AuC generates sequence numbers from counters per user.)

   Alternatively, the HE/AuC may generate sequence numbers from a counter per user per mode. In that case a RAND per mode needs to be stored in the USIM.

   A third alternative is that the HE/AuC derives sequence numbers from a global counter, possibly derived from time. In that case, no re-synchronisation procedure is feasible and no RAND needs to be stored.

To support connection establishment without authentication and key agreement it may be useful that the MS/USIM stores:

d) [optionally] per mode: the key set identifier (4 bits).

e) [optionally] per mode: the value of the last received random challenge RAND (128 bits) received to support re-derivation of the active access link key pair after the UE has been switched off. (Unless a "window" or "mode" is implemented, this random challenge is identical to the random challenge stored for re-synchronisation and thus requires no additional storage).

   Alternatively, the MS/USIM may store the access link key pair (256 bits) itself.

The USIM stores the following data to support control over the cipher key lifetime and thus to trigger re-authentication:

f) a threshold value $THRESH_C$ set by the home environment to limit the usage of a cipher key;

g) per mode: the initialisation value of the $COUNT_{CIPHER}$ variables for ciphering.

In addition, the USIM may store data to support control over the integrity key lifetime and thus to trigger re-authentication:

h) [optionally] a threshold value $THRESH_I$ set by the home environment to limit the usage of an integrity key;

i) per mode: the initialisation value of the $COUNT_{UP}$ and $COUNT_{DOWN}$ variables for data integrity.

   Note: Currently there is no threshold to limit the usage of an integrity key in the specifications, however, there is nothing in the specifications that can prevent the HE from implementation of such a $THRESH_I$.

Table 7-3: USIM – Authentication and key agreement

Name	Comment	Multiplicity	M/O
K	Permanent secret key	1 per user	M
SQN_{MS}	Sequence number counter	1 per user per mode	M
$RAND_{MS}$	Random challenge	1 per user or 1 per user per mode	O
KSI	Key set identifier	1 per user per mode	O
$START_C$	Initialisation value for ciphering	1 per user per mode	M
$START_{IU}$	Initialisation value for data integrity / up	1 per user per mode	M
$START_{ID}$	Initialisation value for data integrity / down	1 per user per mode	M
$THRES_C$	Threshold value for ciphering	1 per user	M
$THRES_I$	Threshold value for data integrity	1 per user	O
f1	Network authentication function	1 (proprietary)	M
f1*	Message authentication function for re-synchronisation	1 (proprietary)	O
f2	User authentication function	1 (proprietary)	M
f3	Cipher key derivation function	1 (proprietary)	M
f4	Integrity key derivation function	1 (proprietary)	M
f5	Anonymity key derivation function	1 (proprietary)	O

7.3 User equipment

7.3.1 Enhanced user identity confidentiality

To support users with enhanced user identity confidentiality the UE stores the global temporary identity $TMUI_{HE}$.

Optionally the USIM may store the TMUI/LAI pairs for the serving networks in which the user is registered such that the UE can store these values after the UE has been switched off.

Table 7-4: UE – Enhanced user identity confidentiality

Name	Comment	Multiplicity	M/O
$TMUI_{HE}$	Global Temporary Mobile User Identity	1 per user	M

7.3.2 User identity confidentiality

To support identification towards the network, the UE stores the local temporary mobile user identity TMUI. In addition, for users without enhanced user identity confidentiality, it stores the IMUI.

Table 7-5: USIM – conventional user identity confidentiality

Name	Comment	Multiplicity	M/O
TMUI	Local temporary mobile user identity	1 per user per mode	M
$TMUI_{HE}$	Global temporary mobile user identity	1 per user	M

7.3.3 Data confidentiality

To support data confidentiality the UE stores:

a) the ciphering capabilities (16 bits) of the UE;

b) per user and per mode: the cipher key CK (128bits).

c) per user and per SN: the selected ciphering function UEA (4 bits);

d) [in addition, in dedicated mode] per logical channel: the time varying parameter $COUNT_C$ (32 bits), the logical channel identifier BEARER (8 bits) and the direction bit (1 bit);

Furthermore, the following functions are implemented in the UE:

– f8: the access link encryption/decryption function .

Table 7-6: UE – Data confidentiality

Name	Comment	Multiplicity	M/O
UEA-MS	Ciphering capabilities UE	1	M
CK	Cipher key	1 per user per mode	M
UEA	Selected ciphering capability	1 per SN	M
$COUNT_C$		1 per logical channel	M
BEARER		1 per logical channel	M
f8	Ciphering function	n / standardised	M

There may be several ciphering functions implemented on the UE.

7.3.4 Data integrity

To support data integrity the UE stores:

a) the integrity capabilities (16 bits) of the UE;

b) per user and per mode: the integrity key IK (128bits);

c) per user and per SN: the selected ciphering function UEA (4 bits);

d) [in addition, in dedicated mode] per logical channel user for signalling data: the time varying parameters $COUNT_{I/UP}$ (32 bits) and $COUNT_{I/DOWN}$ (32 bits), the network challenge FRESH (32 bits) and the selected integrity function (4 bits);

Furthermore, the following functions are implemented in the UE:

– f9: the access link data integrity function .

Table 7-7: UE – Data integrity

Name	Comment	Multiplicity	M/O
UIA-MS	Integrity capabilities UE	1	M
IK	Integrity key	1 per user per mode	M
UIA	Selected integrity capability	1 per SN	M
$COUNT_{I/UP}$	Counter for upstream messages	1 per SN	M
$COUNT_{I/DOWN}$	Counter for downstream messages	1 per SN	M
FRESH	Network challenge	1 per SN	M
f9	Data integrity function	n / standardised	M

There may be several data integrity functions implemented on the UE.

7.4 Radio network controller

7.4.1 Data confidentiality

To support data confidentiality the RNC stores:

a) the ciphering capabilities (16 bits) of the RNC;

b) per user and per mode: the cipher key CK (128bits);

c) per user: the selected ciphering capability UEA;

d) [in addition, in dedicated mode] per logical channel: the time varying parameter $COUNT_C$ (32 bits), the logical channel identifier BEARER (8 bits).

Furthermore, the following functions are implemented in the RNC:

– f8: the access link encryption/decryption function .

Table 7-8: RNC – Data confidentiality

Name	Comment	Multiplicity	M/O
UEA-RNC	Ciphering capabilities RNC	1	M
CK	Cipher key	1 per user per mode	M
UEA	Selected ciphering capability	1 per SN	M
$COUNT_C$		1 per logical channel	M
BEARER		1 per logical channel	M
f8	Ciphering function	n / standardised	M

There may be several ciphering functions implemented on the RNC.

7.4.2 Data integrity

To support data integrity the RNC stores:

a) the integrity capabilities (16 bits) of the RNC;

b) per user and per mode: the integrity key IK (128bits);

c) per user: the selected integrity function (4 bit);

d) [in addition, in dedicated mode] per logical channel user for signalling data: the time varying parameters $COUNT_{I/UP}$ (32 bits) and $COUNT_{I/DOWN}$ (32 bits) and the network challenge FRESH (32 bits).

Furthermore, the following functions are implemented in the UE:

– f9: the access link data integrity function .

Table 7-9: RNC – Data integrity

Name	Comment	Multiplicity	M/O
UIA-RNC	Integrity capabilities RNC	1	M
IK	Integrity key	1 per user per mode	M
UIA	Selected integrity capability	1 per user	M
$COUNT_{I/UP}$	Counter for upstream messages	1 per user	M
$COUNT_{I/DOWN}$	Counter for downstream messages	1 per user	M
FRESH	Network challenge	1 per user	M
f9	Data integrity function	n / standardised	M

There may be several data integrity functions implemented on the RNC.

7.5 Visited location register

7.5.1 User identity confidentiality

The VLR (equivalently the SGSN) has to store the association between at most two TMUI/LAI pair and a IMUI for each user.

Table 7-10: VLR – Conventional User Identity Confidentiality

Name	Comment	Multiplicity	M/O
TMUI	Local temporary mobile user identity	2 per user	M
LAI	Location Area Identity	2 per user	M
IMUI	Permanent user identity	1 per user	M

7.5.2 Authentication and key agreement

To support authentication and key agreement, the VLR stores authentication vectors AV[1…n]. In addition, the VLR stores the CK and the IK and the KSI for the access link keys that are in use. To control the lifetime of an access link key pair, the VLR shall also store the time at which the access link key pair was derived (KSAT).

Table 7-11: VLR – Authentication and key agreement

Name	Comment	Multiplicity	M/O
AV	Authentication vectors	n per user	M
KSI	Key set identifier	1 per user	M

KSAT	Key set agreement time	1 per user	M
CK	Cipher key	1 per user	M
IK	Integrity key	1 per user	M

7.6 Home location register / Authentication centre

7.6.1 Enhanced user identity confidentiality

Optionally, the HLR/AuC may implement a mechanism for enhanced user identity confidentiality. The mechanism is proprietary. We present here the data elements and functions required to support the example mechanism using group keys.

- per user group: the group key GK (128 bits) to decrypt the IMUI and the SQN_{UIC};
- per user: a counter $SQN_{UIC/HE}$ that is equal to the last SQN received by the MS/AuC in an HE-message from the user;
- a temporary user identity $TMUI_{HE}$ that is valid for user identification.

In addition the HLR/AuC should implement the functions

- f7: decryption of the permanent identity;
- f6*: derivation of a global temporary identity.

Table 7-12: HLR/AuC – Enhanced user identity confidentiality (optional)

Name	Comment	Multiplicity	M/O
$TMUI_{HE}$	Global temporary mobile user identity	1 per user	M
SQN_{UIC}	Counter for verification of freshness	1 per user	M
GK	Group key	1 per user group	M
f7	Decryption function	1 (proprietary)	M
f6*	Global temporary identity derivation function	1 (proprietary)	M

7.6.2 Authentication and key agreement

To support authentication and key agreement, the HLR/AuC has to store:

a) per user: the permanent secret key K (128 bits).

b) per user: the value of the highest generated sequence number SQN_{HE} (32—64 bits).

   Alternatively, the HE/AUC may derive sequence numbers from a counter $SQN_{HE/MODE}$ per user and per mode or a single global counter SQN_{HE} for all users.

c) per user and per mode: an array of authentication vectors that the HE/AUC has computed in advance.

   Alternatively, the HE/AUC may compute the authentication vectors on demand in which case no authentication vectors need to be stored.

To generate authentication vectors, the following functions are implemented in the HE/AuC:

- f0: the random number generation function ;
- f1: the network authentication function ;
- f2: the user authentication function ;
- f3: the cipher key derivation function ;

– f4: the integrity key derivation function ;

– f5: the anonymity key derivation function

(all required for authentication and key agreement) and

– f1*: the authentication function for re-synchronisation

required for re-synchronisation of the counter in the HE/AuC.

Table 7-13: HLR/AuC – Authentication and key agreement

Name	Comment	Multiplicity	M/O
K	Permanent secret key	1 per user	M
SQN_{HE}	Sequence number counter	1 per user *	M
AV	Authentication vector	n per user per mode	O
f1	Network authentication function	1 (proprietary)	M
f1*	Message authentication function for re-synchronisation	1 (proprietary)	O
f2	User authentication function	1 (proprietary)	M
f3	Cipher key derivation function	1 (proprietary)	M
f4	Integrity key derivation function	1 (proprietary)	M
f5	Anonymity key derivation function	1 (proprietary)	O

8. Network protocols

8.1 Outline

In this chapter, the modifications to the network protocols are described. The modifications are necessary to support the security features as they are described in the chapter 5 and 6. As far as possible the protocols specified for UMTS in the 3GPP 'April release 1999' are taken as a base to describe the modifications.

Following protocols are discussed:

– Mobility Management (MM),

– Radio Resource (RR),

– Mobile Application Protocol (MAP),

– MSC-UTRAN interface (RANAP, comparable to GSM 08.08 MSC-BSS layer 3 specification).

For the MM protocol, the specification 3G TS 24.008 is not available in the April release of 3GPP. The specification 3G TS 24.008 will be created by taken the CC and MM parts out of GSM 04.08 [31]. Therefor [31] is used as a base to describe the modifications that are required for the security features.

For the RRC protocol, 3G TS 25.331[32] is available in the April release.

For the MAP protocol, 3G TS 29.002 [33] is available in the April release.

For the RANAP protocol on the Iu interface, 3G TS 25.413 is available in the April release.

Following security features are considered:

- Authentication and key agreement (5.2 and modifications in 6.2 and 6.4),
- Data confidentiality Ciphering (5.3 and 6.8),
- Data integrity (described in chapter 5.4 and 6.9),
- Enhanced user identity confidentiality (5.1.3 and 6.1).

Access to UTRAN with a GSM SIM (described in chapter 5), Following security features are not considered:

- Handling of cipher key and integrity key during handover,
- Handling of cipher key and integrity key during UMTS-GSM handover.

8.2 Mobile Application Part (MAP) protocol

8.2.1 Overview

Table 8-1 provides an overview of the MAP messages that need to be modified to implement the new security mechanisms.

The notation for mobility service description of [33] (7.3) is used and can be summarised as follows:

In defining the service-primitives the following convention is used for categorising parameters:

M **Mandatory.** The parameter must be present in the indicated primitive type;

O **Optionally.** The inclusion of the parameter is a home environment option;

U **User option.** The inclusion of the parameter is a user option;

C **Conditional.** Whether the inclusion of the parameter is required depends on the value of another parameter.

(=) **Equal.** When appended to one of the above (M, O, U, C), this symbol means that the parameter takes the same value as the parameter appearing immediately to its left (in the columns of the tables);

blank **Absent.** The parameter is not included in this message.

Table 8-1: MAP messages that need to be modified

	AKA	DC	DI	EUIC
MAP Authenticate	ρ	ρ	ρ	
MAP Send Authentication Info	ρ	ρ	ρ	ρ
MAP Update Authentication Area		ρ	ρ	
MAP Process Access Request		ρ	ρ	
MAP Set Ciphering Mode		ρ	ρ	
MAP Send Identification	ρ	ρ	ρ	

8.2.2 Overall modifications to MAP

There is a single overall modification to all MAP messages: the replacement of the information element CKSN (Ciphering Key Sequence Number) to the information element KSI (Key Set Identifier). The KSI is not only an identifier for the ciphering key but also for the integrity key. It remains to be assigned in a circular fashion, but it is no longer called sequence number to avoid confusion with the sequence number in the authentication mechanism.

The following MAP mobility services include the parameter ~~CKSN~~ KSI:

– MAP Update Location Area,

– MAP Process Access Request,

– MAP Authenticate.

This modification is described in detail for the MAP Authenticate service because this MAP service is also modified for other purposes. The other mentioned MAP services are only modified for the Ciphering key sequence number.

8.2.3 MAP Authenticate service

This service is used between the VLR and the MSC when the VLR receives a MAP service indication from the MSC concerning a location registration, call set-up, operation on a supplementary service or a request from the MSC to initiate authentication. The service is a confirmed service and consists of four service primitives. The service primitives are shown in Table 8-2.

Table 8-2: Modification to the MAP Authenticate parameters

Parameter name	Request	Indication	Response	Confirm
Invoke id	M	M (=)	M (=)	M (=)
Authentication mechanism	M	M (=)	M (=)	M (=)
~~CKSN~~ KSI	M	M (=)		
~~SRES~~ XRES			M	M (=)
RAND	M	M (=)		
AUTN	C	C (=)		
Provider error				O

The modifications are the following:

- **Authentication mechanism.** A new parameter is added that differentiates between UMTS and GSM authentication. The parameter AM can take two values: UMTS and GSM.
- **CKSN → KSI.** The name of the parameter is changed as it now refers both to a ciphering and an integrity key and to avoid confusion with the sequence number in the authentication mechanism.
- **SRES → XRES.** The name of the parameter has changed.
- **AUTN.** This parameter is required when AM = UMTS.

8.2.4 MAP Send Authentication Info service

This service is used between the VLR (or SGSN) and the HLR for the VLR (or SGSN) to retrieve authentication information from the HLR. The VLR requests arrays of GSM authentication vectors (RAND, SRES, Kc) or arrays of UMTS authentication vectors (RAND, XRES, CK, IK, AUTN). The service is a confirmed service and consists of four service primitives. The service primitives are shown in Table 8-3:

Table 8-3: Modifications to the MAP Send Authentication Info parameters

Parameter name	Request	Indication	Response	Confirm
Invoke id	M	M (=)	M (=)	M (=)
AM	M	M (=)	M	M (=)
EUIC	C	C (=)		
SF	C	C (=)		
IMSI	C-M	C-M (=)		
IMUI	C	C (=)	C	C (=)
HE-message	C	C (=)		
GSM-AV			C	C (=)
UMTS-AV			C	C (=)
AUTS	C	C (=)		
RAND	C	C (=)		
RAND_MS	C	C (=)		
User error			C	C (=)
Provider error				O

The modifications are the following:

- **AM (Authentication mechanism).** A new parameter is added that differentiates between UMTS and GSM authentication. The parameter AM can take two values: UMTS and GSM.
- **EUIC (Enhanced user identity confidentiality).** A new parameter that has to be present when the authentication mode is UMTS. It indicates whether the user is identified by means of an IMUI or an HE-message.
- **SF (Synchronisation failure).** A new parameter that has to be present when the authentication mode is UMTS. It indicates whether the authentication data request followed a rejection by the user due to a synchronisation failure.
- **IMSI.** This parameter is required when the authentication mode is GSM.

- **IMUI.** This parameter is required when the authentication mode is UMTS and the EUIC indicates that the user is identified by means of an IMUI. The parameter is required in the response/confirm messages when the EUIC indicates that the user is identified by means of an HE-message.

- **HE-message.** This parameter is required when the authentication mode is UMTS and the EUIC indicates that the user is identified by means of an HE-message.

- **GSM-AV.** The GSM authentication vectors (RAND, SRES, Kc) that are included when in the response/confirm AM = GSM.

- **UMTS-AV.** The UMTS authentication vectors (RAND, XRES, CK, IK, AUTN) that are mandatorily included when in the response/confirm AM = UMTS.

- **AUTS.** An authentication token for re-synchronisation that contains the current sequence number SQN_{MS} of the user. This parameter shall be included when SF indicates a synchronisation failure.

- **RAND.** The RAND parameter that was sent along with the AUTN parameter that cause the synchronisation failure in the USIM.

- **RAND_MS.** The RAND parameter that was sent along with the AUTN parameter that contained the sequence number SQN_{MS}.

8.2.5 MAP Set Ciphering Mode

This service is used in GSM between the VLR and the MSC to set the ciphering mode and to start ciphering if applicable. It is called when another service requires that information is to be sent on the radio path in encrypted form. This service must also used to set the integrity mode so that the integrity function can be started in UTRAN. The service is a non-confirmed service and consists of two service primitives. The service primitives are shown in Table 8-4:

Table 8-4: Modifications to the MAP Set Ciphering Mode parameters

Parameter name	Request	Indication
Invoke id	M	M (=)
UEA-CN	M	M (=)
UIA-CN	M	M (=)
Kc	O	O (=)
CK	O	O (=)
IK	O	O (=)

The modifications are the following:

- **UEA-CN (UMTS encryption algorithms – Core Network).** A parameter that indicates a set of UMTS encryption algorithms acceptable to the VLR. The GSM ciphering algorithms should be a subset of the UMTS ciphering algorithms. In this way the VLR can indicate in one parameters which GSM (when attached to GSM BSS) and which UMTS (when attached to UTRAN) algorithms are acceptable to the CN.

- **UIA-CN (UMTS integrity algorithms – Core Network).** A parameter that indicates a set of UMTS encryption algorithms acceptable to the VLR. This parameters is required when AM = UMTS.

- **Kc.** The GSM cipher key. This parameter is required when the UMTS VLR supports GSM BSS or supports handover to GSM networks.

- **CK.** The UMTS cipher key. This parameter is required when the UMTS VLR supports UTRAN or-supports handover to UTRAN.

– **IK.** The UMTS integrity key. This parameter is required when the UMTS VLR supports UTRAN or supports handover to UTRAN.

8.2.6 MAP Send Identification service

The MAP Send Identification service is used between a VLR and a previous VLR to retrieve IMSI and authentication sets for a subscriber registering afresh in that VLR. The service is a confirmed service using the service primitives. The service primitives are shown in Table 8-3:

Table 8-5: Modifications to the MAP Send Identification parameters

Parameter name	Request	Indication	Response	Confirm
Invoke id	M	M (=)	M (=)	M (=)
TMSI/TMUI	M	M (=)		
IMSI/IMUI			O	O (=)
GSM AV/UMTS AV			U	U (=)
User error			O	O (=)
Provider error				O

The messages should be modified such that they allow to send over UMTS identities and UMTS authentication vectors.

8.3 Mobility management protocol

8.3.1 Outline

To describe the modifications to the MM protocol, the GSM document GSM 04.08 [31] is taken as a base. In [31] there is a group of MM messages which are used for the GPRS services and there is a group of MM messages which are used for non-GPRS services. Also in 3GPP there will be a group of MM messages for CS services and a group of MM messages for PS services. The MM messages for CS services will be based on the MM messages defined for non-GPRS services in [31].

Open issues concerning MM in 3GPP (handled in 3GPP TSG CN WG-N1) can be summarised as follows:

– *Definition of MM messages for PS services*: The MM messages for PS services will be different from the ones defined for GPRS services in [GSM04.08] because the LLC-layer will not be used in 3GPP.

– *Value to be used for the protocol discriminator information element in all 3GPP MM messages:* In [31] one has a protocol discriminator value for MM of non-GPRS services and another value for MM of GPRS services. What will this be in 3GPP ? Use a new protocol discriminator value or re-use the existing ones?

– *Value to be used for the message type information element in all 3GPP MM messages:* Will 3GPP define new message types or will the existing messages be modified?

– *Move the Paging Response message from the RR to the MM layer*: this is a proposal from Fujitsu which will probably be accepted.

Only the 3GPP MM messages for CS services are described because of the open issue concerning MM messages for PS services.

Following table gives an overview of which Mobility Management messages are modified for which 3GPP security features.

Table 8-6: Modified MM messages

	AKA	DC	DI	EUIC
Authentication Request	ρ	ρ	ρ	
Authentication Response	ρ	ρ	ρ	
Identity Response				ρ
Location Updating Request	ρ	ρ	ρ	
CM Service Request	ρ	ρ	ρ	
CM Re-establishment Request	ρ	ρ	ρ	
Paging Response[4]	ρ	ρ	ρ	

The message description of [GSM04.08] (which is described in GMS04.07) is used and can be summarised as follows:

A standard IE may have the following parts, in that order:

- an information element identifier (IEI);
- a length indicator (LI);
- a value part.

A standard IE has one of the formats shown in Table 8-7:

Table 8-7: Formats of information elements

Format	Meaning	IEI present	LI present	Value part present
T	Type only	yes	no	no
V	Value only	no	no	yes
TV	Type and Value	yes	no	yes
LV	Length and Value	no	yes	yes
TLV	Type, Length and Value	yes	yes	yes

The relevant protocol specification may define three different presence requirements (M, C, or O) for a standard IE:

M **Mandatory.** The information element shall be included;

C **Conditional.** There are conditions for the receiver to expect that the IE is present and/or conditions for the receiver to expect that the IE is not; these conditions depend only on the content of the message itself, and not for instance on the state in which the message was received, or on the receiver characteristics; they are known as static conditions;

O **Optional.** The information element can be included in the message.

8.3.2 General modifications to the MM messages

A first overall modification to many MM messages is the replacement of the information element CKSN (Ciphering Key Sequence Number) with the information element KSI (Key Set Identifier). The KSI is not only an identifier for the ciphering key but also for the integrity key. It remains to be assigned in a circular fashion, but it is no longer called sequence number to avoid confusion with the sequence number in the authentication mechanism.

[4] Paging Response is a RR message in GSM, but for UMTS this message will be moved to the MM-layer

MM messages with parameter CKSN:

- Location Updating Request
- Authentication Request
- CM Service Request
- CM Re-establishment Request
- Paging Response

This modification is described in detail for the Authentication Request message because this message is also modified for other purposes.

A second overall modification is the addition of a new information element that indicates whether the user requires that the serving network initiates an execution of the authentication and key agreement mechanism. The user asks for such a new authentication and key agreement procedure if the he detects that his access link keys have been used too many times. Note that also when the user indicates that an authentication is not required, the network can decide to initiate the authentication and key agreement procedure.

MM messages in which this new information element should be included:

- Location Updating Request,
- CM Service Request,
- CM Re-establishment Request,
- Paging Response.

8.3.3 Authentication Request message

This message is sent by the SN to the MS to initiate authentication and key agreement between the MS and the SN for UMTS users and GSM subscribers.

The spare bits is used to indicate whether the network initiates UMTS or GSM authentication and key agreement. The AUTN parameter is included when AM = UMTS.

Table 8-8: Modification to the Authentication Request message

Information element	Presence	Format	Length
~~CKSN~~ KSI	M	V	½
~~Spare Octet~~ AM	M	V	½
RAND	M	V	16
AUTN	C	TLV[5]	1+1+(13—17)

8.3.4 Authentication Response message

This message is sent by the MS to the SN in response to an Authentication Request message.

The response (parameter RESULT) can take three possible values:

- ACCEPTED. The information element XRES is included.
- REJECTED. No information element is included.
- FAILURE. The information elements AUTS and RAND_MS are included.

[5] Optional parameters always need to be of type TLV.

Note: Alternatively one may define three MM messages: Authentication Response, for the case ACCEPTED, Authentication REJECT, for the case ACCEPTED and Authentication FAILURE for the case FAILURE.

Table 8-9: Modifications to the Authentication Response message

Information element	Presence	Format	Length
Result	M	V	½
Spare Bits	M	V	½
~~SRES~~ XRES	C	TLV	1+1+(4—16)
AUTS	C	TLV	1+1+(13—17)
RAND_MS	C	TV	1+16

8.3.5 Identity Response message

This message is sent by the mobile station to the network in response to an Identity Request message providing the requested identity.

The message should be changed such that it can include an IMUI or an HE-message. The HE-identity is sent by users with enhanced user identity confidentiality. The minimum and maximum length of the HE-message has not yet been defined by SA-3. **Table 8-10: Modifications to the Identity Response message**

Information element	Presence	Format	Length
Identity Type	M	V	½
Spare Bits	M	V	½
IMUI	C	TV	1+2
HE-message	C	TLV	1+1+(variable)

8.4 Radio access network application protocol

8.4.1 Overview

To describe the necessary modifications to RANAP, the April release version of 3G TS 25.413 is taken as a base and modifications for 3GPP security features are indicated. Table 8-11 shows the modified RANAP messages.

Table 8-11: Modified RANAP messages

RANAP messages	AKA	DC	DI	EUIC
~~CIPHER~~ SECURITY MODE COMMAND		ρ	ρ	
~~CIPHER~~ SECURITY MODE COMPLETE		ρ	ρ	
~~CIPHER~~ SECURITY MODE REJECT		ρ	ρ	

Essentially the modifications relate to the security features access link data confidentiality and access link data integrity.

8.4.2 Cipher mode command

The CIPHER MODE COMMAND message is sent from the CN to the RNC to indicate the encryption and integrity protection parameters.

Table 8-12: Modifications to the CIPHER MODE COMMAND message

Information element	Presence
Cipher Mode	M
Integrity Mode	M
Cipher Information	M
Integrity Information	M
Cipher Response Mode	O[6]

The modifications are the following:

- The IE "Cipher Mode" and "Integrity Mode" define the sets of UMTS encryption algorithms (UEA) and UMTS integrity algorithms (UIA) that are allowed by the CN. It is expected that up to 16 UEA and 16 UIA will be allowed. The length of the data field of "Cipher Mode" and "Integrity Mode" is thus 2 bytes each.
- The IE "Cipher Information" and "Integrity Information" contain the cipher key and the integrity key respectively. The length of the data field of both is thus 16 bytes each.

  Note: It is doubtful whether it is required that the CN sends "Cipher Mode" and "Integrity Mode" to the RNC. As an alternative the RNC selects a ciphering and integrity function from set of algorithms supported by the UE and the RNC.

8.4.3 Cipher mode complete

The CIPHER MODE COMPLETE message is sent from the RNC to the CN to notify the successful establishment of the selected ciphering and integrity mode between UE and RNC.

Table 8-13: Modifications to the CIPHER MODE COMPLETE message

Information element	Presence
Selected Encryption Mode	M
Selected Integrity Mode	M

The modifications are the following:

- The IE "Selected Encryption Mode" indicates the selected UEA (4 bits).
- The IE "Selected Integrity Mode" indicates the selected UIA (4 bits).

8.4.4 Cipher mode reject

The CIPHER MODE REJECT message is sent from the RNC to the CN to indicate that none of the RNC and the UE do not support either any of the ciphering or any of the integrity algorithms that are specified in the CIPHER MODE COMMAND.

[6] This element is used by the CN to indicate whether the MS should include the IMEI in the CIPHER MODE COMPLETE message. The need for this parameter is ffs.

Table 8-14: Modifications to the CIPHER MODE REJECT message

Information element	Presence
Possible UEA	M
Possible UIA	M

The IE "Possible UEA" (2 bytes) and "Possible UIA" (2 bytes) provide the CN information on which UEA and UIA are supported by both the UE and the RNC.

8.5 RRC protocol

8.5.1 Overview

To describe the 3GPP RR messages, the RRC protocol as it is specified in the April release version of 3G TS 25.331 is taken as a base and modifications for 3GPP security features are indicated.

Following table gives an overview of which RRC messages are modified for which 3GPP security features.

Table 8-15: Modified RRC messages

RANAP messages	AKA	DC	DI	EUIC
UE CAPABILITY INFORMATION		ρ	ρ	
~~CIPHER~~ SECURITY MODE COMMAND		ρ	ρ	
~~CIPHER~~ SECURITY MODE COMPLETE		ρ	ρ	

Note: SECURITY MODE COMMAND and SECURITY MODE COMPLETE are not yet available in [32].

8.5.2 UE capability information

This message is sent by the UE to the RNC to indicate the UE's capabilities, amongst which the ciphering and integrity capabilities.

Table 8-16: Modification the UE CAPABILITY INFORMATION

Information element	Presence	Length
Power Control Capability	M	
Code Resource Capability	M	
UE Mode Capability	M	
Transport Channel Support Capability	O	
UMTS Encryption Algorithm Capability	M	2
UMTS Integrity Algorithm Capability	M	2
Macro Diversity Capability	M	

The modification consists in the inclusion of two parameters: UEA_{MS} (2 bytes) and UIA_{MS} (2 bytes) that indicate the security capabilities of the MS.

8.5.3 Security mode command

This message is sent on the main DCCH from the RNC to the UE:

- to indicate that the network has started deciphering and that enciphering and deciphering shall be started in the mobile station, or to indicate that ciphering will not be performed,
- to provide information that is needed for the data integrity function .

Table 8-17: Modification the security mode command

Information element	Presence	Length
Selected UMTS Encryption Algorithm	M	½
Selected UMTS Integrity Algorithm	M	½
Fresh	M	4
Data integrity code for integrity	M	4

8.5.4 Security mode complete

This message is sent on the main DCCH from the UE to the RNC:

- to indicate that enciphering and deciphering has been started in the mobile station,
- to provide authenticate the UMTS encryption and integrity capabilities of the UE that were transmitted in the UE CAPABILITY INFORMATION message.

Table 8-18: Modification the SECURITY MODE COMMAND

Information element	Presence	Length
UMTS Encryption Algorithm Capabilities	M	½
UMTS Integrity Algorithm Capabilities	M	½
Data integrity code for integrity	M	4

8.6 Example of signalling procedures

In TS 25.931 examples for signalling procedures are given. For the non access stratum signalling connection establishment two examples are foreseen:

1. UE initiated signalling connection establishment,
2. CN initiated signalling connection establishment.

Only for the UE initiated signalling connection establishment a figure is available in. This figure is taken as a base to add the signalling procedures for authentication and key agreement, ciphering and integrity.

The consecutive steps are the following:

1. An RRC Connection is established.
2. The UE the RRC message UE Capability Information to the RNC (see 8.5.2). In this message the UE sends
   a) UEA_{MS}: the ciphering capabilities of the UE;
   b) UIA_{MS}: the integrity capabilities of the UE;
   c) $COUNT_C$: the initialisation value for the counters for encryption of user and signalling channels;
   d) $COUNT_{I/UP}$: the initialisation value for the counter for data integrity of signalling channels in the up-link direction;
   e) $COUNT_{I/DOWN}$: the initialisation value for the counters for data integrity of signalling channels in the down-link direction;

Note: To speed up the transfer of the initial non-access stratum message the RRC message "UE Capability Information" can also be transferred after the initial non-access stratum message (step 3). In 3GPP TSG RAN this is currently ffs.

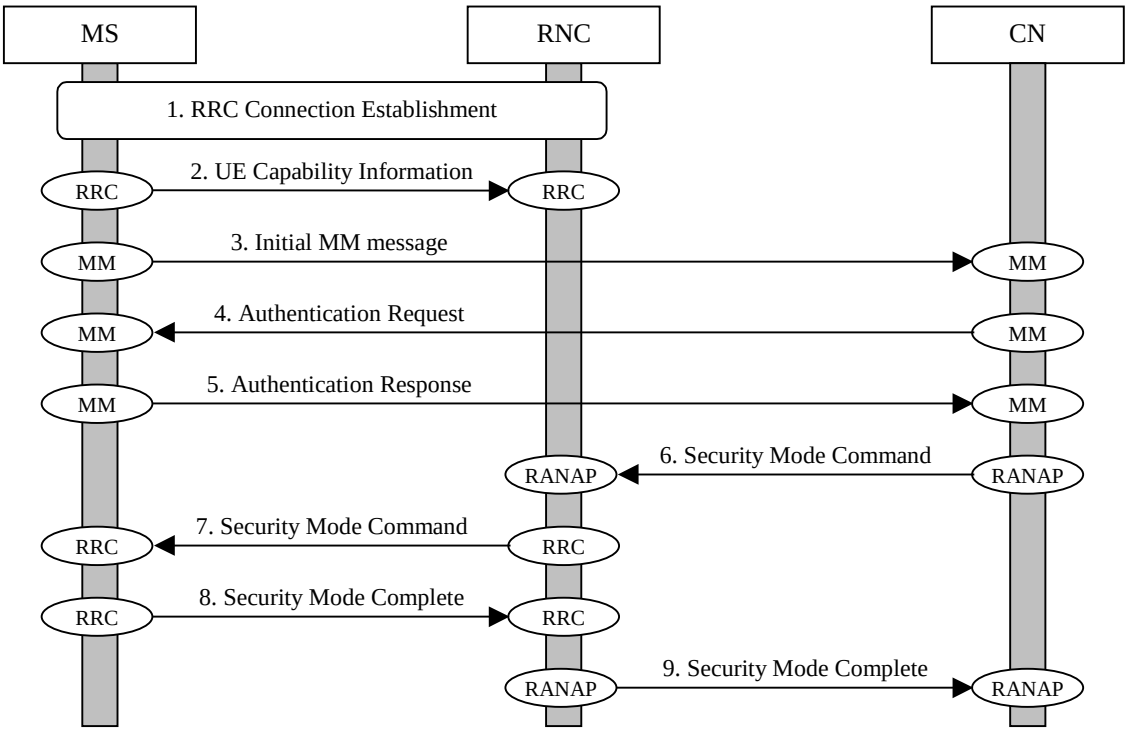

Figure 8-1: Message flow for connection set-up

3. The UE sends the initial MM message to the VLR (or SGSN). This message is send via an RRC/Direct Transfer message and the RANAP/Initial UE message. The parameters included are:
   a) AUTH: a Boolean parameter that indicates whether the user requires an execution of the authentication and key agreement protocol.
   b) KSI: the key set identifier;
4. [Optional] The CN sends the user an MM/Authentication request message. This includes the security parameters:
   a) RAND: a random challenge that allows the user to derive the user response RES, the cipher key CK and the integrity key IK.
   b) AUTN: an authentication token for network-to-user authentication that provides assurance of key freshness to the user
5. [Optional, only when 4. was present] The user sends back the MM/Authentication Response to the VLR (or SGSN). This include the parameters:
   a) RESULT (= ACCPETED): an indication that the user has verified the integrity and the freshness of the AUTN parameter included in the MM/Authentication request;
   b) RES: the user response to the authentication request.
6. The VLR (or SGSN) sends the RANAP/security mode command to the RNC. This includes the following parameters:
   a) UEA-CN, UIA-CN: The encryption and integrity capabilities that are allowed by the CN;
   b) CK, IK: the cipher and the integrity key.

7. The RNC sends the RRC/security mode command message to the UE. It contains:
   a) UEA, UIA: The selected encryption and integrity capability;
   b) FRESH: A network challenge for the data integrity mechanism.
8. The MS sends the RRC/security mode complete message to the RNC. It contains:
   a) UEA-MS, UIA-MS: The encryption and integrity capabilities of the MS;
   b) MAC-I: The data integrity code
9. The RNC sends the RANAP/security mode complete message to the VLR (or SGSN). It contains:
   a) UEA, UIA: the selected encryption and integrity mechanism.

9. Conclusions and outlook

The first part of this deliverable describes and analyses the current 3GPP security architecture and points to some open issues in it. The identification of these open issues then leads to a number of proposals for improvements to the mechanisms. These enhancements are a major direct result of the work done for this deliverable. Some of the improvements have already been presented and agreed at a 3GPP TSG SA-3 meeting, e.g. those on re-synchronisation and sequence number management. The formal analysis of the sequence number management mechanism was especially appreciated by the SA-3 group, as it is expected to raise the user's confidence in the security and reliability of the UMTS. It is unlikely that such analysis would have been conducted without the support of the USECA project. Other enhancements have been presented and had a significant impact on what the group believe to be feasible, e.g. secure interoperation between UMTS and GSM. Further contributions are yet to be presented in SA-3 but are expected to be in the near future. USECA WP2.3 plans to continue to contribute to the standards-making process in 3GPP, and to work on the design of further mechanisms and solutions arising from new concepts coming from other members of SA-3.

The latter part of this deliverable was devoted to the integration of the security mechanisms into the overall network architecture. The preliminary security architecture provided here is seen as an initial foundation for the comprehensive specification that must undertaken once the mechanisms are sufficiently stable. WP2.3 plans to participate in this task, but this will be dependent on the progress achieved in SA-3.

In parallel, WP2.3 plans to contribute to the architecture for future releases of UMTS: mainly on a public key architecture for user-network authentication and on the MAP over SS7 security mechanisms.

In summary:

- the work so far has made valuable contributions to the standards making process for UMTS;
- on-going tasks for WP2.3 comprise:
  - continued contribution to 3GPP security;
  - work on integration of mechanisms into the developing architecture;
  - future needs of the architecture for public key cryptography.