USECA

Project Number	AC336
Project Title	**USECA: UMTS Security Architecture**
Deliverable Type	Report
Security Class	Public
Deliverable Number	D06
Title of Deliverable	**Intermediate report on UMTS security mechanisms**
Nature of the Deliverable	Intermediate deliverable
Document reference	AC336/DOC/SAG/001/WP22/F
Contributing WPs	WP2.2
Contractual Date of Delivery	February 1999 (Y01M12)
Actual Date of Delivery	01. April 1999
Editor	Klaus Müller, Siemens AG

Abstract	This report contains a description and evaluation of security mechanisms for UMTS phase 1. It includes the mechanisms discussed in ETSI SMG10 and 3GPP WG3 and documents the reasons for the decisions taken there. All mechanisms described are based on secret key cryptography as it was stated aim for UMTS phase 1 that it is an evolution from GSM and therefore the security architecture should be as compatible to the GSM architecture as possible.

Keywords	ACTS, USECA, UMTS, security, security mechanisms, authentication, key agreement, confidentiality, integrity, data origin authentication

Executive Summary

The objectives of this intermediate report of WP2.2 are to review security mechanisms available to realise the security features elaborated in WP1.2 and documented in deliverable D02, to evaluate these mechanisms with respect to the special UMTS requirements and to select a preliminary set of security mechanisms. It should be emphasised that this deliverable is the only document so far where the evaluation of the UMTS security mechanisms is laid down in a comprehensive and systematic fashion. This could be helpful when one later wishes to recall the reasons for certain decisions taken in the standardisation process.

Commercial UMTS Phase 1 services are expected to commence in Europe by 2002. Accordingly, standardisation work for phase 1 is progressing very fast and has to be completed by the end of this year. USECA has made major contributions to the ongoing standardisation work for UMTS phase 1 in the relevant standardisation bodies within ETSI and 3GPP. This intermediate report contains a description and evaluation of security mechanisms discussed for UMTS phase 1, within the standards bodies or within USECA. In particular, it contains the mechanisms discussed in ETSI SMG10 WPC and 3GPP WG3 and documents the reasons for the decisions taken there.

All mechanisms described are based on secret key cryptography. One reason for this is that UMTS, as specified by 3GPP [WWW.3GPP.ORG], is to be based on the evolved GSM core networks. Another reason is that the additional benefits of public key mechanisms do not seem to justify the additional effort involved in their deployment in the early phase of UMTS.

In chapters 4 and 5 the underlying threat model, on which the selection of the security mechanisms is based, and the appropriate countermeasures are discussed. Chapter 4 describes new attacks that were not possible when GSM was designed, but are now or are perceived to be possible in the near future, because intruders have more computational capabilities or new equipment has become available to attackers. The description of these attacks, together with the description of appropriate countermeasures are shifted to a confidential annex, due to the public nature of this deliverable and the necessity not to educate potential attackers.

Chapter 6 on access network security contains the discussion on four different items: On the confidentiality of the user identity and location, on authentication and key agreement protocols, on integrity protection of certain signalling messages and on user traffic confidentiality.

Subsection 6.1 discusses four methods for the provision of user identity and location confidentiality. A recommendation for one of the mechanisms is given, which seems to be the most appropriate one. It is based on the mechanism used in GSM today and defines an optional add-on mechanism in specific (failure) situations where today GSM transmits the user identity in the clear. A symmetric group key is introduced for distinct groups of users to provide identity confidentiality even in these cases.

Subsection 6.2 describes the authentication and key agreement (AKA) mechanisms which were under discussion for a UMTS security architecture. The section at first lists the protocol goals to be fulfilled by an AKA mechanism for UMTS in order to counter the threats discussed in the previous sections. It contains a detailed description of seven security protocols, namely of the existing systems GSM, DECT, TETRA, IS-41 but also three other proposals (RHUL, SEQ, TETRA-3) which were also discussed in the standardisation bodies as candidates for UMTS systems. Only the latter three are evaluated according to a set of selection criteria as they are the only ones which fulfil all the required protocol goals.

In subsection 6.3 and 6.4 mechanisms for integrity protection of certain signalling messages and for user traffic confidentiality are discussed. The work on integrity protection has not progressed as far as some of the other work, therefore only a number of general candidate mechanisms is discussed so far.

Chapter 7 provides a first analysis of the proposals for core network security. The goal of core network security is to protect the security related signalling messages (e.g. authentication vectors) which have to be sent within the SS7 network. Candidate security mechanisms have been proposed. Chapter 8 contains the conclusions of this intermediate report and an outlook on the future work of WP2.2. Besides the final decisions on security mechanisms for UMTS phase 1 the final report is aimed to contain a discussion of the advantages and consequences of utilising public key mechanisms in later UMTS phases.

1 Table of contents

2 Document management

2.1 Contributors

Klaus Müller (Editor)	Siemens AG ZT IK 3, D-81730, München, Germany Phone: +49 89 636 41126 / Fax: +49 89 636 48000 Email: Klaus.Mueller@mchp.siemens.de
Günther Horn	Siemens AG ZT IK 3, D-81730, München, Germany Phone: +49 89 636 41494 / Fax: +49 89 636 48000 Email: Guenther.Horn@mchp.siemens.de
Bart Vinck	Siemens Atea Atealaan 34, B-2200, Herentals, Belgium Phone: +32 14 252592 / Fax: +32 14 253339 Email: Bart.Vinck@ntnet.atea.be
Peter Howard	Vodafone Ltd The Courtyard, 2-4 London Road, Newbury, Berkshire, RG14 1JX, UK Phone: +44 1635 506206 / Fax: +44 1635 583464 Email: Peter.Howard@vf.vodafone.co.uk
Timothy Wright	Vodafone Ltd The Courtyard, 2-4 London Road, Newbury, Berkshire, RG14 1JX, UK Phone: +44 1635 506456 / Fax: +44 1635 31127 Email: Timothy.Wright@vf.vodafone.co.uk

2.2 Document history

Version	Date	Comment
A	24/02/1999	First draft
B	08/03/1999	Second draft
C	16/03/1999	Third draft
D	26/03/1999	Fourth draft
E	29/03/1999	Final draft for PMC approval
F	01/04/1999	Final Version

2.3 References

[3GPP 1(99)027] 3GPP Systems and Services TSG, Security WG (Source: Mannesmann Mobilfunk and T-Mobil): *Proposal for improved User Identity Confidentiality*; TSGS3#1(99)027, Docklands, UK, 2 – 4 February 1999.

[3GPP 1(99) 009] 3GPP Systems and Services TSG, Security WG (Source: Vodafone): *The Requirements for Identity Confidentiality*; TSGS3H#1(99)009 adhoc #1 on 3G security architecture Stockholm, 23rd February 1999.

[3GPP s3-99005] 3GPP Systems and Services TSG, Security WG, Tdoc s3-99005 (Source: ETSI SMG10): Draft UMTS 33.23 V0.2.0 (1999-01) – Security Mechanisms and Architecture.

[3GPP s3-99006] 3GPP Terminals TSG, Security WG, Tdoc s3-99006 (Source: ETSI SMG10 WPC); London, 2-4 February, 1999: *Record of strategic decisions taken by SMG10 (WPC) with regard to UMTS security specification.*

[DaPr] D.W. Davice, W.L. Price: *Security for computer networks*; John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore, 1984.

[DECT] ETSI DECT Part 7: *Security features*

[ETSI 99C050] ETSI 99C050 (Source Siemens Atea), Circulated at the Borehamwood meeting: *General authentication framework for UMTS*.

[ETSI 33.21] UMTS 33.21: *Security requirements*; Version 3.0.0 (February 1999).

[ETSI 33.22] UMTS 33.22: *Security features*; Version 1.0.0 (February 1999).

[ETSI 33.23] UMTS 33.23: *Security mechanisms and architecture*; Version 0.2.0 (January 1999).

[GSM03.20] GSM 03.20 Digital cellular telecommunications system (Phase 2+): *Security related network functions*; Version 6.0.1.

[GSM03.48] GSM 03.48 Digital cellular telecommunications system (Phase 2+): *Security Mechanisms for the SIM application toolkit*; stage 2, Version 6.1.0.

[GSM11.11] GSM 11.11 Digital cellular telecommunications system (Phase 2+): *Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface.*

[IS-41 (95)] TIA/EIA, PN 2991: *Cellular radio telecommunications intersystem operations IS-41 Rev. D*; May 4 1995.

[ISO SD6] ISO, SD 6, SC 27 N1954: *Glossary of IT security terminology*; March 1998.

[ISO 9797-1] ISO/IEC FDIS 9797-1: *Security techniques - Message authentication codes (MACs) - Part 1: Mechanisms using a block cipher*; November 11 1998, Revision of IS 9797 (1994).

[ISO 9797-2] ISO/IEC 9797-2: *Security techniques - Message authentication codes (MACs) - Part 1: Mechanisms using a hash function*.

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

[MeOoVa] A. Menezes, P. van Oorschot, S. Vanstone: *Handbook of Applied Cryptography*; CRC Press, Boca Raton, 1997

[Moh96] S. Mohan: *Privacy and Authentication protocols for PCS*; IEEE Personal Communications, Oct 1996.

[Pat97] S. Patel: *Weakness of North American Wireless Authentication protocol*; IEEE Personal Communications, June 1997.

[SG-169/96] ETSI SMG SG DOC 169/96: *General authentication framework for UMTS*; June 1996.

[SMG5 (94)] ETSI STC SMG 5 TD430/94, Source Vodafone: *A mutual authentication mechanism for UMTS*; Newbury, Nov., 14th (1994).

[TET-VD7] Terrestrial Trunked Radio (TETRA), Voice plus Data (V+D): *Part 7: Security*; Edition 2f, November 1998

[USE-D12] ACTS USECA AC336 deliverable 12: *Overview of UMTS architecture*; July 1998

2.4 Abbreviations

3GPP	Third Generation Partnership Project
ACTS	Advanced Communications Technologies and Services
AES	Advanced Encryption Standard
AKA	Authentication and Key Agreement
API	Application Programming Interface
ASPeCT	Advanced Security for Personal Communications Technologies
AuC	Authentication Centre
BER	Bit Error Rate
BTS	Base Transceiver Station
CN	Core Network
DECT	Digitally Enhanced Cordless Telecommunications
DES	Data Encryption Standard
ETSI	European Telecommunications Standardisation Institute
GSM	Global System for Mobile Communications
HE	Home Environment
HLR	Home Location Register
IETF	Internet Engineering Task Force
IMSI	International Mobile Subscriber Identity
IMUI	International Mobile User Identity
MAC	Message Authentication Code
MExE	Mobile Execution Environment
MO	Mobile Originated
MS	Mobile Station
PKI	Public Key Infrastructure
SAT	SIM Application Toolkit
SIM	Subscriber Identity Module
SN	Serving Network
TETRA	TErrestrial Trunked RAdio
TLS	Transport Layer Security
TMSI	Temporary Mobile Subscriber Identity
TMUI	Temporary Mobile User Identity
UMTS	Universal Mobile Telecommunication System
USIM	User Services Identity Module
USECA	UMTS SECurity Architecture
VASP	Value-Added Service Provider
VLR	Visited Location Register
W-CDMA	Wideband Code Division Multiple Access

3 Introduction

While current second generation mobile systems will continue to play an important role, the new third generation system UMTS is shortly to be introduced. UMTS needs to be capable of co-existing and working with existing second generation mobile communications technologies so that operators can choose to re-use their existing infrastructure assets and expertise. Commercial UMTS Phase 1 services are expected to commence in Europe by 2002. UMTS will provide a wider spectrum of services than today's systems, ranging from simple voice telephony to high speed, high quality multimedia services, regardless of physical location of the user, using radio frequency access to a convergent network of fixed, cellular and satellite components.

UMTS has so far been standardised by ETSI. Recently, the specification work has been taken over by 3GPP. Work is currently concentrated on UMTS Phase 1 security specifications to be released by the end of 1999.

Two areas can be distinguished in the security work: Security in the access network and security in the core network.

GSM security is based on the assumption that the core network is adequately secure and trustworthy for the transmission of confidential information in the clear from the HE to the SN. For future systems like UMTS this assumption may not be adequate anymore. In standardisation bodies within ETSI security measures related to authentication information transmitted across the SS7 network will be specified.

The threat analysis shows that new attacks in the access network that were not possible when GSM was designed are feasible now, because intruders have more computational capabilities or new equipment has become available to attackers. Therefore additional security features have to be offered by the mechanisms utilised in UMTS to protect the access network. These additional features include enhancements in user identity confidentiality mechanisms, enhancements in the authentication and key agreement mechanisms to assure the freshness of the agreed keys (used e.g. to provide confidentiality or integrity) also to the user, or to assure the integrity of certain signalling messages to prevent sophisticated attacks. Changes in the security mechanisms in the access network may also have to be introduced because of changes in technology, e.g. the introduction of CDMA requires mechanisms different to the ones in GSM systems.

4 Threat model

The whole chapter on threats can be found in the confidential annex of this deliverable. This also includes new threats that were not possible before but are perceived to be possible now or very soon, because intruders have more computational capacities, new equipment has become available to attackers, and the physical security of certain network elements is questioned.

5 Countermeasures

The chapter on countermeasures has a close relationship to the section on threats; e.g. for each class of threats different countermeasures are discussed and it is justified which one provides the best protection against certain threats. Therefore also this whole chapter is shifted to the confidential annex of this document.

6 Access network security

In this chapter several aspects of security in the access network are discussed. Besides mechanisms which provide user identity and location confidentiality a subchapter on authentication and key agreement (AKA) mechanisms is included. Further subsections deal with integrity protection mechanisms for several specific purposes and with mechanisms providing user traffic confidentiality.

6.1 User identity and location confidentiality mechanisms

6.1.1 Overview

The purpose of user identity and location confidentiality mechanisms is to avoid the possibility for an intruder to identify which user is using a given resource by listening to the signalling exchanges on the radio path. Two different kinds of attacks have to be prevented:

- That an attacker gets hold of the user identity or more general of any information allowing an attacker to derive the user identity easily;
- that an attacker is able to trace a user's location, even if he does not (yet) know which specific user he is tracing.

The provision of mechanisms to counter these attacks implies that the IMUI (International Mobile User Identity), or any information allowing a listener to derive the IMUI easily, should normally not be transmitted in the clear in any signalling message on the radio path.

Consequently, to obtain the required level of protection, it is necessary that:

- A protected identifying method is normally used instead of the IMUI on the radio path;
- the IMUI is normally not used as addressing means on the radio path;
- when the signalling procedures permit it, signalling information elements that convey information about the mobile user identity must be ciphered for transmission on the radio path.
  (E.g. if a signalling element would require to be encrypted for security reasons, but it would already be needed in a network node which physically lies in front of the decrypting node, encryption may not be feasible.)

Further on we will call all these requirements shortly requirements for the "protection of the user identity".

6.1.2 User identity and location confidentiality protection methods

Threats related to user identity confidentiality are discussed in 4. Section 5 mentions countermeasures against these kinds of attacks. The following four alternatives to provide protection of the user identity are discussed in the subsections below:

- Essentially using the mechanism specified for GSM;
- using group based symmetric key encryption;
- using two layer temporary identity schemes;
- using public key mechanisms.

In GSM systems instead of IMUI the abbreviation IMSI (International Mobile Subscriber Identity) is used for the parameter which identifies the user globally. Correspondingly temporary user identities are called TMSIs (Temporary Mobile Subscriber Identities) in GSM whereas they are called TMUIs (Temporary Mobile User Identities) in UMTS systems.

Smart cards used in GSM systems are called SIM (Subscriber Identity Module) and USIM (UMTS Subscriber Identity Module) within UMTS.

6.1.2.1 Protection of the user identity by essentially using GSM mechanisms

In GSM systems TMSIs are used to protect the user identity from disclosure to third parties [GSM03.20]. Each TMSI is established between a user and the SN in which he is roaming, temporarily. Normally the TMSI is sent on the air interface instead of the user's IMSI. To prevent tracing attacks, during the allocation process of a new TMSI, this new TMSI is sent over the air-interface in encrypted form. The user identity IMSI is only transmitted on the air-interface in specific situations, namely in the following cases:

- Very first authentication of a SIM;
- registration in a new network
  (i.e. a user for the first time enters a new network);
- registration in a new VLR area in networks that do not pass TMSIs from one VLR to another;
- networks that do not use TMSIs;
- VLR malfunction in the SN
  (e.g. loss of the actual TMSI in the SN because of a data base failure in the VLR);
- previous VLR not reachable
  (thereby making it impossible to pass the TMSI from the previous VLR to the VLR in the new location area)
- authentication failures

6.1.2.2 Protection of the user identity by group based symmetric key encryption

In this section a mechanism is discussed, which is an add-on to the mechanism described in section 6.1.2.1. The mechanism was first proposed in [3GPP 1(99)027], in [*3GPP 1(99)009*] an alteration was proposed in order to avoid possible regulatory problems. As in 6.1.2.1 usually temporary mobile user identities (TMUIs) are utilised here to protect the user identity. The add-on mechanism is invoked whenever a request for IMUI transmission in the clear would have to be performed (a list of such occasions is given in 6.1.2.1). It avoids the transmission of the user identity in the clear.This is facilitated in the following way.

Additional to the IMUI an International Mobile User Group Identity (IMUGI) has to be introduced. The mechanism is based on a permanent common symmetric key K_G shared between a limited group of users of a specific service provider and their HE. On the demand of the SN to the user to provide his IMUI, the user sends his IMUGI and the IMUI (together with some replay protection material) protected by the secret group key K_G. The SN forwards this message to the HE of the user. As the HE also possesses the group key K_G, it is able to reconstruct the IMUI from the received data. The HE then sends the IMUI to the SN.

It should be noted that the size of the group is critical. In case that the number of users per group is too small, the group identity will provide significant information on the user identity. In case that the number of users per group is too big, extracting a group key from at the HE may cause significant effort and may be considered worth the effort.

By the add-on mechanism passive as well as active attacks on the user identity are prevented.

6.1.2.3 Protection of the user identity by two layer temporary identity schemes

The aim of two layer temporary identity schemes is to introduce a second temporary identity (additionally to the one used e.g. in GSM systems today, cf. section 6.1.2.1) for the following purposes:

- To improve the protection of the user identity on the air-interface (compared to GSM systems) and
- to provide protection of the user identity towards the SN.

An AKA which uses two layer temporary identities is described in [SMG5 (94)].

Each user possesses two temporary identities, one is established between the MS and the SN (analogous to section 6.1.2.1), we call this one $TMUI_{MS-SN}$ here, and the other (additional) one is a temporary identity $TMUI_{MS-HE}$ valid between the user's MS and his HE.

$TMUI_{MS-HE}$ is aimed to be used instead of the user identity for authentication but also for accounting purposes. Only the HE is able to retrieve the real identity of the user which has utilised a specific $TMUI_{MS-HE}$ to get a specific service from the SN.

Below limitations are discussed for both of the purposes mentioned above. These limitations make it questionable, if the additional benefit justifies the additional effort compared to the mechanism used in GSM today.

Improvement of the protection of the user identity on the air-interface

In some of the cases listed in section 6.1.2.1 the transmission in the clear of the user identity is aimed to be avoided (e.g. in case of registration in a new network).

A limitation of the approach using two temporary identities is, that IMSI catching attacks would be still possible. E.g. in case of VLR malfunction in the SN with a loss of the currently valid TMUI of a user, it is proposed in [SMG5 (94)] that the SN should run the same protocol, which is usually applied if a user is not known to the serving network (i.e. as in a new registration scenario). This implies that the HE is involved in the authentication and key agreement. If additionally to the VLR malfunction, the connection between SN and HE is down or a HLR malfunction has occurred, then instead of the temporary identities the IMUI has to be used. Aside from the fact, that all these things are quite unlikely to happen simultaneously, a solution for this situation has to be provided, i.e. it must be possible to send the IMUI on the air-interface in the clear.

This fact could be abused by an attacker using a false base station. If the false base station claims to the user that the situation described above has occurred, then the user would have to sent his IMUI in the clear.

Provision of protection of the user identity towards the SN

One reason to provide protection of the user identity towards the SN is, that it is likely, that in a world-wide operating UMTS system there may be a lot of service providers and network operators. The trust relationship between some of these service providers may differ from that in 2nd generation systems today, where only a relatively few service providers are present, i.e. users may not be able to trust SN's that do not belong to their HE.

A limitation of the approach using two temporary identities is, that it is anyhow questionable, if protection of the user identity towards the SN is feasible, as with a set-up of a session by a roaming UMTS user, a signalling message seems to be necessary which contains both, the A-number and B-number of the user's MS. These numbers at least have to be available in the serving network in the clear as it has to be able to set-up the connection to the correct destination. From the A-number the user identity can be retrieved anyhow.

6.1.2.4 Protection of the user identity by public key mechanisms

Probably the most secure way to protect the user identity from disclosure can be provided by the use of public key cryptography. This kind of protection requires a public key infrastructure. A prerequisite of protocols using public key mechanisms for the protection of the user identity is, that the public key of the SN is already available at the user side or that it is sent in course of the protocol in a previous step. The user sends his IMUI encrypted by the public key of the SN. Only the SN is able to decrypt it, thereby recovering the user identity.

It was decided in the standardisation bodies not to support public key mechanisms to provide authentication in UMTS phase 1 (cf. [3GPP s3-99006]). Therefore the use of public key mechanisms only for the protection of the user identity is not worthwhile, because of the already mentioned necessity to set up a complex public key infrastructure.

This could be different in later UMTS phases, if it was decided to use public key systems for authentication and key agreement purposes. Then the protection of the user identity by public key systems would require no additional effort. Instead, the quite complex handling of the temporary identifiers would no longer be necessary.

6.1.3 Recommendations on location confidentiality protection methods

In [3GPP s3-99006] it is documented that two of the four mechanisms discussed in section 6.1.2 above, are not appropriate for UMTS phase 1, namely it was decided:

- UMTS Phase 1 shall not support the use of public key techniques for service related authentication.
  The gains from the use of public key techniques were not sufficient to justify the extra complexity of public key over secret key techniques. However, the possibility of using public key techniques in future phases of UMTS is not excluded.
- UMTS shall not support a two layer (SN and HE) temporary identity scheme.
  Such a scheme would not provide extra protection against active attacks, as it is presumed a facility for HE's to request the IMUI would still be required. This request could be spoofed by a false BTS. The two layer scheme gives limited extra protection against passive attack but this is not presumed to be the real threat – active attack is.

Taking into account these pre-decisions, it is recommended that the proposal using group based secret key encryption (cf. 6.1.2.2) should be implemented.

6.2 User authentication and key agreement mechanisms

6.2.1 Overview

In this section different mechanisms for authentication and key agreement (AKA) are described. An important characteristic of an AKA protocol are the goals the protocol achieves. Before we come to the goals a protocol is required to have for its usage in UMTS, we at first have to clarify what is meant by the different goals. We do not want to give formal definitions here, clarification shall only be provided by informal definitions and explanations for plausibility. (Cf. e.g. [ISO SD6] and [MeOoVa].)

- **Entity authentication:**

There are different definitions in the literature of what constitutes entity authentication. (Cf. [ISO SD6] and [MeOoVa].) Depending on the type of time variant parameter used - random number, sequence number or time-stamp - these definitions differ in the guarantees given to the verifier about the time the evidence received from the claimant was produced.

For entity authentication of an entity *A* to an entity *B*, a protocol using random challenges as time variant parameters, provides assurance to *B* that the evidence was generated during the current protocol run. A sequence number as time variant parameter only provides assurance to *B* that the evidence was not used in a previous protocol run. (As none of the AKA protocols discussed in the sections below does uses time-stamps we do not emphasis on time-stamps here.)

- **Implicit key authentication:**

Implicit key authentication of an entity *A* to an entity *B* gives assurance to *B* that only entity *A* can hold the agreed keys.

- **Assurance of key freshness:**

Assurance of key freshness to an entity *B* implicates that *B* can be sure that the keys derived in course of the AKA protocol were not used before the current protocol run.

Note that there are different definitions of key freshness in the literature. (Cf. e.g. [ISO SD 6] and [MeOoVa].)

- **Key (seed) confirmation:**

Key seed confirmation of an entity *A* to an entity *B* provides assurance to entity *B* that entity *A* is in possession of the correct parameter(s) to derive the agreed keys. This property provides key confirmation apart from possible errors in the local key derivation process as it provides assurance of the availability on both sides of the correct input parameters for key derivation.
Key confirmation of *A* to *B* is the stronger goal and provides assurance to entity *B* that entity *A* is in possession of the derived key itself.

After clarification of the meanings of the different goals we can now list the goals which are at least ***required for an AKA protocol for UMTS***:

- Entity authentication of the user to the SN;
- implicit key authentication of the network to the user;
- implicit key authentication of the user to the SN;
- assurance of key freshness to the SN;
- assurance of key freshness to the user;
- confidentiality of the user identity (and other user related information from which the user identity may be derived) on the interface between the user and the SN. Additionally, if the transmission of the identity is not part of the AKA and is instead provided by other means, the AKA must not prevent the provision of confidentiality of the user identity on the air interface.

Two other goals seem to be desirable for an AKA for UMTS, but they are however not really required to be fulfilled.

- Key (seed) confirmation from the user to the SN
- Key (seed) confirmation from the HE to the user

If a protocol does not provide key (seed) confirmation, then the usage of the agreed keys after a successful run of the AKA, will (in case of a failure in one of the agreed keys) lead to problems in the messages protected by these keys and therefore to a break in the transmission of user/signalling data which otherwise could have been detected beforehand.

6.2.2 Description of the mechanisms

In this section different mechanisms for authentication and key agreement (AKA) are described in a uniform way. Subsections on the following headings are provided:

(a) Prerequisites
The section contains the assumptions on the mechanism which have to be made in order to utilise it.

(b) Protocol goals achieved
All goals are mentioned here, including the additional goals which are not required for the use of the AKA in UMTS.

(c) Description of the protocol
The protocol is explained based on a verbal description, which is completed by figures containing the information flow of the AKA with the data elements exchanged in course of the protocol but also the computations made by the involved parties.

(d) How are the goals achieved?
For each of the goals it is explained by which of the parts of the exchanged message and by which of the calculations made by the appropriate party a specific goal is achieved.

Only those mechanisms are described here in full detail using all of these which are expected to be suitable for a UMTS system. For some other AKA mechanisms (like GSM, DECT, TETRA and IS-41)

most emphasis is put on the description of the way the mechanisms work and the goals which are achieved, as they are not expected to be used within UMTS. Nevertheless they are described here, because some of them form the basis of proposed AKA mechanisms for UMTS (e.g. GSM, DECT, TETRA and IS-41 mechanism), some are additionally also used in the selection criterion on compatibility with the security architecture of 2nd generation mobile systems (GSM and IS-41 mechanism, cf. criterion (6) and (7) below). For these reasons it seems to be helpful if the description is made within this document - in the same manner as for the other mechanisms which are meant as proposals for UMTS - instead of always only referring to standardisation papers.

The sections on (a) to (d) are followed by sections on selection criteria. The following selection criteria are used to assess the different AKA mechanisms:

(1) Are all goals required for an AKA for UMTS achieved?

(2) Limitations of the protocol
This section mainly focuses on the goals required for UMTS which are not achieved by the AKA, but also other limitations may be mentioned, e.g. possible attacks.

(3) Implications on the protocol in case of UMTS system failures
- (3.1) Compromise of SN authentication information
- (3.2) Resilence to USIM or HE data corruption
- (3.3) Resilence to breakdown of links between SN and HE

(4) Protocol efficiency
- (4.1) Number of passes (for new registration and for current registration)
- (4.2) Number of calculations (in USIM, SN, HE)
- (4.3) Precomputation of authentication information in the HE
- (4.4) Frequency of SN – HE signalling

(5) Implications on the mechanism on the UMTS infrastructure
- (5.1) Need for an AuC in the SN

(6) Compatibility with GSM security architecture
- (6.1) Ease of migration from GSM to UMTS
- (6.2) Ease of roaming between GSM and UMTS networks

In order to provide an evaluation for the two bullets it has to be assumed that the described AKA was the one chosen as the AKA standard for UMTS.

(7) Compatibility with IS-41 security architecture
- (7.1) Ease of migration from IS-41 to UMTS
- (7.2) Ease of roaming between IS-41 and UMTS networks

In order to provide an evaluation for the two bullets it has to be assumed that the described respective AKA was the one chosen as the AKA standard for UMTS.

(8) Need for a standard AKA algorithm

These selection criteria are used in the subsections below to assess the different mechanisms. In section 6.2.3 a comparison of the different mechanisms is made. It summarises the evaluation process in a table according to the selection criteria, referencing them according to the numbering above.

6.2.2.1 GSM

The discussion in this section is based on [GSM 03.20].

(a) Prerequisites

The GSM scheme is based on the use of a symmetric authentication algorithm *A3*, and a symmetric key generation algorithm *A8*, which are implemented in the USIM and in an HE/AuC. It requires a user-specific authentication key *Ki*, which is held in the USIM and the HE/AuC.

The authentication exchange with the user proves knowledge of authentication data given to an SN/VLR by an HE/AuC. The SN/VLR is assumed to be trusted by the HE to handle this authentication data

securely. It is also assumed that the intra-system interfaces linking the SN/VLR to the HE/AuC are adequately secure.

Cryptographic processing capabilities are required in the USIM and in the HE but not in the SN.

(b) Protocol goals achieved

- User authentication
- Cipher key agreement

(c) Description of protocol

When the user attempts to access a service, the SN recognises the HE of the user from the *IMUI* and signals a request for authentication data to the HE. The HE then instructs the AuC to produce the data and signals this back to the SN. The data consists of one or more triplets consisting of an authentication challenge *RAND*, an authentication response *XRES* and a cipher key *Kc*. The response *RES* is computed from the challenge *RAND* using the algorithm *A3* under control of the user specific key *Ki*. Similarly, the cipher key *Kc* is computed from *RAND* using the algorithm *A8* under control of *Ki*.

For each authentication exchange the SN sends *RAND* to the user and awaits a response *RES*. This response is computed in the USIM, from the challenge received from the SN using the algorithm *A3* under control of *Ki*, and then sent to the SN. Upon receipt, the SN checks that *RES* and *XRES* are the same. If they are not, authentication has failed and access is denied. The USIM also computes the cipher key *Kc* from *RAND* using the algorithm *A8* under control of *Ki*.

Typically, the SN will request more than one set of authentication data so that it does not have to signal back to the HE for every authentication exchange. Each (*RAND*, *XRES*, *Kc*) triplet is generally used once, but may be reused in failure situations where the serving network cannot obtain fresh triplets.

The general procedure for authentication and key agreement for new registrations is shown in Figure 6.2.2-1.

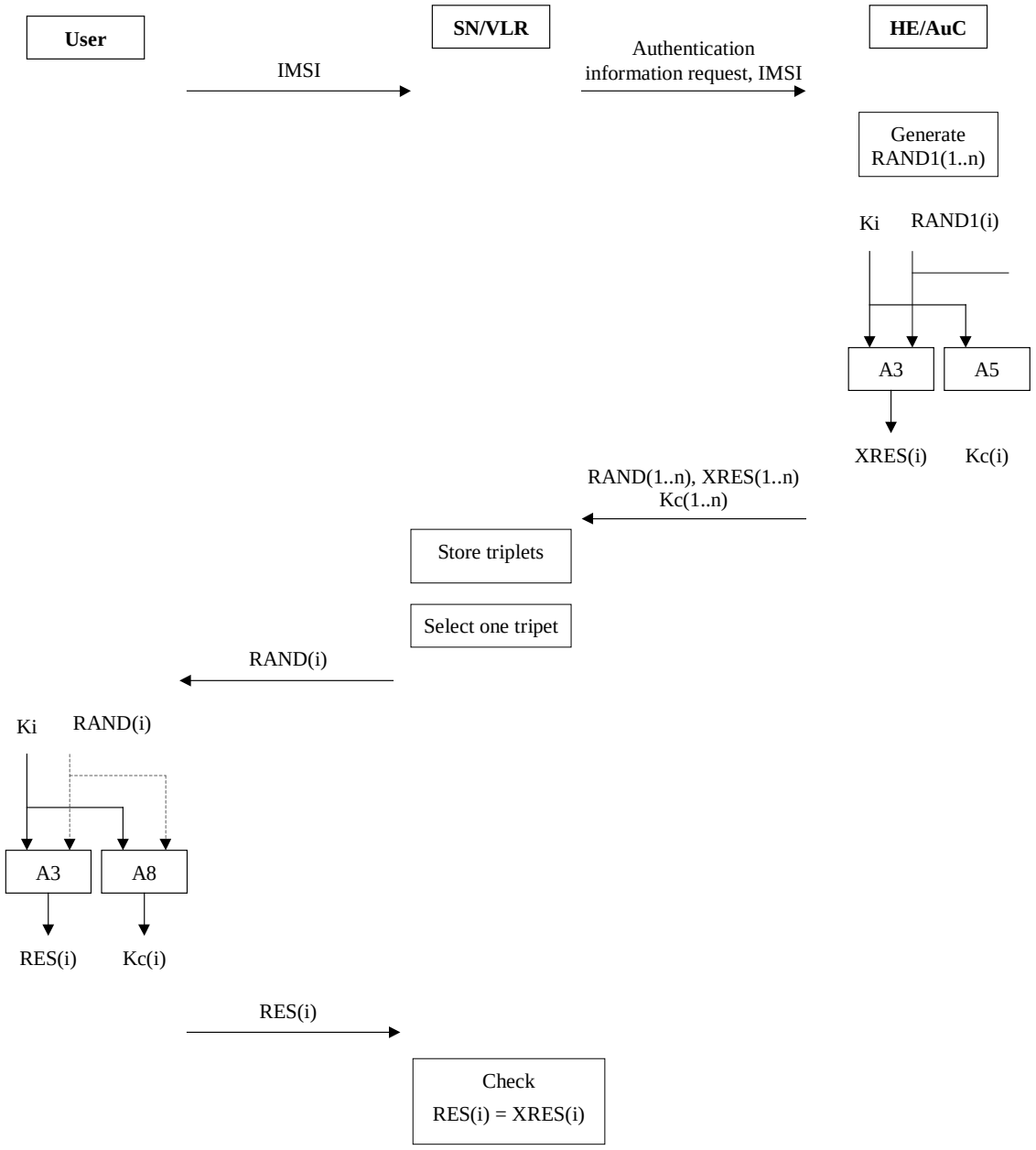

Figure 6.2.2-1: The GSM scheme

(1) Are all goals required for an AKA for UMTS achieved?

Not all of the goals required for an AKA for UMTS are achieved by the protocol. Further evaluation is therefore not carried out.

6.2.2.2 DECT

The discussion in this section is based on [DECT].

(a) Prerequisites

The scheme is based on the use of four symmetric algorithms *A11, A12, A21 and A22*. All four algorithms are implemented on the USIM. Algorithms *A11* and *A21* are also implemented in the HE/AuC while algorithms *A12* and *A22* are also implemented in the SN/AuC. The scheme requires a user-specific authentication key *K*, which is held in the USIM and in the HE/AuC.

The authentication exchange with the user proves knowledge of authentication data given to an SN/VLR by an HE/AuC. The SN/VLR is assumed to be trusted by the Home Environment to handle this authentication data securely. It is also assumed that the intra-system interfaces linking the SN/VLR to the HE/AuC, and linking SN/VLRs, are adequately secure.

Cryptographic processing capabilities are required in the USIM, the HE and in the SN.

(b) Protocol goals achieved

- Mutual authentication
- Cipher key agreement

(c) Description of protocol

The DECT scheme is a modification of the GSM approach which removes the need for the operator to request user authentication data from an HE for each authentication exchange. It also provides network authentication as well as user authentication.

When the user first attempts to access a service, it recognises the HE for the user from the *IMUI* and signals to that HE requesting authentication data. The HE then instructs the AuC to produce the authentication key data and signals this back to the SN. The key data consists of a key seed *RS* and a key value *KS* and a key value *KS'*.

The key *KS* is computed from the key seed *RS* using the algorithm *A11* under control of the user specific key *K*. This key is used to authenticate the user as follows. The network generates a challenge, *RAND1*, and computes a response *XRES1* from *RAND1* using algorithm *A12* under control of the key *KS*. The user is sent *RAND1* along with the seed *RS*, and the network awaits a response RES1. This response is computed in the USIM, by first computing *KS* from *RS* using algorithm *A11* under control of *K*, and then computing *XRES1* and the derived cipher key *DCK* from *RAND1* using algorithm *A12* under control of *KS*. Upon receipt, the network checks that *RES1* and *XRES1* are the same. If they are not, authentication has failed and access is denied. Note that the value of *RAND1* must change upon every use of this mechanism.

The key *KS'* is computed from the key seed *RS* using the algorithm *A21* under control of the user specific key *K*. This key is used to authenticate the network as follows. The user first generates *KS* using the key seed *RS* received from the network using *A21* and *K*. The user then generates a challenge, *RAND2*, and computes a response *XRES2* from *RAND2* using algorithm *A22* under control of the key *KS'*. The network is sent *RAND2*, and the user awaits a response *RES1*. This response is computed in the network, by computing *RES2* from *RAND2* using algorithm *A22* under control of *KS'*. Upon receipt, the user checks that *RES2* and *XRES2* are the same. If they are not, authentication has failed. Note that the value of *RAND2* must change upon every use of this mechanism.

The general procedure for authentication and key agreement for new registrations is shown in Figure 6.2.2-2.

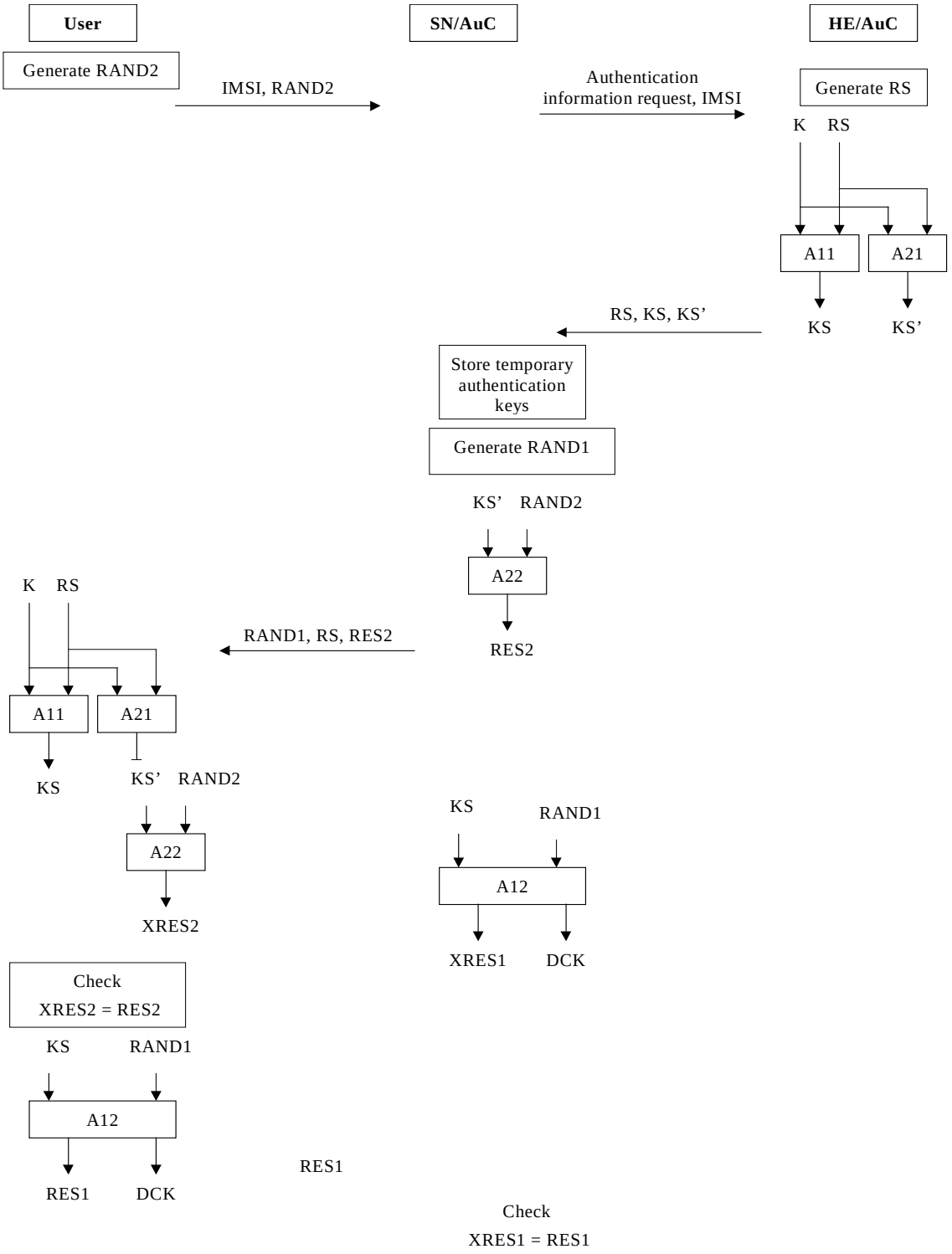

Figure 6.2.2-2: The DECT scheme

(1) Are all goals required for an AKA for UMTS achieved?

Not all of the goals required for an AKA for UMTS are achieved by the protocol. Further evaluation is therefore not carried out.

6.2.2.3 TETRA

The discussion in this section is based on [TET-VD7].

(a) Prerequisites

The scheme is based on the use of five symmetric algorithms *TA11, TA12, TA21, TA22 and TB4*. All five algorithms are implemented on the USIM. Algorithms *TA11* and *TA21* are also implemented in the HE/AuC while algorithms *TA12*, *TA22* and *TB4* are also implemented in the SN/AuC. The scheme requires a user-specific authentication key *K*, which is held in the USIM and in the HE/AuC.

The authentication exchange with the user proves knowledge of authentication data given to an SN/VLR by an HE/AuC. The SN/VLR is assumed to be trusted by the HE to handle this authentication data securely. It is also assumed that the intra-system interfaces linking the SN/VLR to the HE/AuC, and linking SN/VLRs, are adequately secure.

Cryptographic processing capabilities are required in the USIM, the HE and in the SN.

(b) Protocol goals achieved

- Mutual authentication
- Cipher key agreement

(c) Description of protocol

The TETRA scheme is a modification of the DECT approach which allows the user to check the freshness of the cipher key.

When the user first attempts to access a service, it recognises the HE for the user from the IMUI and signals to that HE requesting authentication data. The HE then instructs the AuC to produce the authentication key data and signals this back to the SN. The key data consists of a key seed *RS* and a key value *KS* and a key value *KS'*.

The key *KS* is computed from the key seed *RS* using the algorithm T*A11* under control of the user specific key *K*. This key is used to authenticate the user as follows. The network generates a challenge, RAND1, and computes a response *XRES1* from *RAND1* using algorithm T*A12* under control of the key *KS*. The user is sent *RAND1* along with the seed *RS*, and the network awaits a response *RES1*. This response is computed in the USIM, by first computing *KS* from *RS* using algorithm T*A11* under control of *K*, and then computing *RES1* and the derived cipher key *DCK1* from *RAND1* using algorithm T*A12* under control of *KS*. Upon receipt, the network checks that *RES1* and *XRES1* are the same. If they are not, authentication has failed and access is denied. Note that the value of *RAND1* must change upon every use of this mechanism.

The key *KS'* is computed from the key seed *RS* using the algorithm T*A21* under control of the user specific key *K*. This key is used to authenticate the network as follows. The user first generates *KS* using the key seed *RS* received from the network using T*A21* and *K*. The user then generates a challenge, *RAND2*, and computes a response *XRES2* from RAND2 using algorithm T*A22* under control of the key *KS'*. The network is sent *RAND2*, and the user awaits a response *RES1*. This response is computed in the network, by computing *RES2* from RAND2 using algorithm *TA22* under control of *KS'*. Upon receipt, the user checks that *RES2* and *XRES2* are the same. If they are not, authentication has failed. Note that the value of *RAND2* must change upon every use of this mechanism.

The general procedure for authentication and key agreement for new registrations is shown in Figure 6.2.2-3.

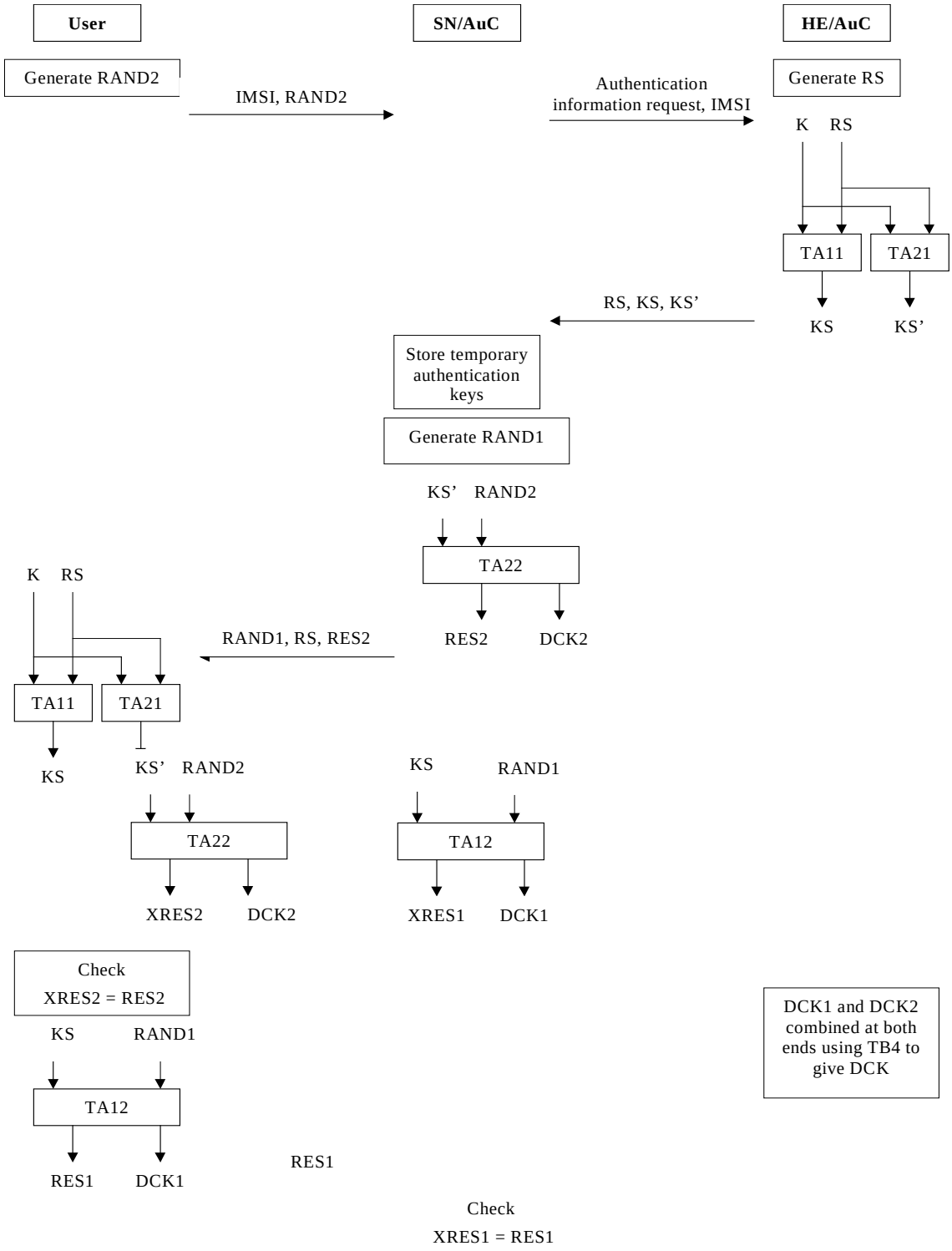

Figure 6.2.2-3: The TETRA scheme

(1) Are all goals required for an AKA for UMTS achieved?

Not all of the goals required for an AKA for UMTS are achieved by the protocol. Further evaluation is therefore not carried out.

6.2.2.4 IS-41

The discussion in this section is based on the AKA protocols for IS-41 standardised in [IS-41 (95)]. A description of selected variants of IS-41 security protocols can be found in [Moh96].

(a) Prerequisites

In the different protocols specified in IS-41 there are two or three parties involved in the communication. Here only three protocol variants are considered (*Initial Registration*, *SSD Update when SSD is shared* and *Call origination when SSD is shared*). There the involved parties are the AuC in the HE, the VLR in the SN and the MS of the user. The authentication and key establishment method described is of symmetric key type. In this method one secret, the authentication and key establishment key, *A-key*, shall be shared by two parties, the HE and the user.

Authentication and key establishment consists of two procedures: First, in an *Initial Registration* procedure authentication information is distributed to the SN by the HE. Second, an authentication exchange is run between the user and the SN.

There are two variants of these procedures which differ in the way the authentication information is provided to the SN by the HE. In the first variant, which is called *'SSD shared'* this information consists of a temporary user specific authentication and key agreement key *SSD* (shared secret data) which is derived from the *A-key*. In the second variant, which is called *'SSD* not shared', the information provided by the HE to the SN consists of the parameters necessary to carry out the authentication exchange and of the agreed keys. Only procedures for the first variant are described in this chapter.

The user trusts the HE in all respects concerning this protocol. The HE trusts the SN to handle authentication information, sent by the HE to the SN, securely. The HE distributes authentication information only to entities it trusts. The SN trusts the HE to send correct authentication information. The SN accepts authentication information only from entities it trusts. The intra-system interfaces linking the SN to the HE, and linking the SNs, are adequately secure.

The mechanisms utilised for the IS-41 AKA protocols are based on the use of the symmetric key based cryptographic function *CAVE*. It is implemented on the mobile station and the HE/AuC, respectively. (In the first variant (*SSD* shared) mentioned above, the algorithm also has to be implemented in the SN/VLR.) The method requires a permanent user-specific authentication key *A-key*, which is only held in the MS and in the HE/AuC.

The SN/VLR is assumed to be trusted by the Home Environment to handle authentication information securely. It is also assumed that the intra-system interfaces linking the SN/VLR to the HE/AuC, and linking SN/VLRs, are adequately secure.

The HE/AuC has a random number generator implemented.

The provision of user identity confidentiality is not a goal of IS-41.

(b) Protocol goals achieved

There is a multitude of different protocols within IS-41, see section (c) below. For this subsection, we limit ourselves to the protocols *Initial Registration*, *SSD Update* and *Call Origination*. The first two protocols correspond to initial registrations in the RHUL, TETRA, DECT or TETRA-3 protocols. The third protocol corresponds to a current registration in these protocols. Other forms of current registrations in IS-41 are listed in section (c).

The goals achieved at the end of a successful run of these protocols are:

- Entity authentication:

In case of *Initial Registration, SSD Update* and *Call Origination*:

- Entity authentication of the user to the SN.

In case of *SSD Update*:

- Entity authentication of the SN to the user.

As in case of *Initial Registration* no keys are agreed, the following goals do not apply to the initial registration case.

- Agreement between user and SN on a shared secret key.

  In case of *Call Origination*:

  - Agreement between user and SN on two shared secret keys VPMASK and SMEKEY (both to provide confidentiality)

  In case of *SSD Update* (with shared SSD):

  - Agreement between user and SN on the shared key SSD.

- Implicit key authentication:

  In case of *Call Origination* and *SSD Update* (with shared SSD):

  - Implicit key authentication of the network to the user.
  - Implicit key authentication of the user to the SN

- Assurance of key freshness:

  In case of *Call Origination* and *SSD Update* (with shared SSD):

  - Assurance of key freshness to the SN

  There is no assurance of key freshness to the user in the case of *SSD Update* and only a limited form of key freshness to the user in the case of *Call Origination* when new dialled digits are included.

- Key (seed) confirmation

  In case of *SSD Update* (with shared SSD):

  - Key confirmation from user to SN
  - Key confirmation from SN to user

  (It is assumed that the use of SSD_A also confirms the possession of SSD_B.

  In case of *Call Origination*:

  - Key seed confirmation from user to SN

(c) Description of the protocol

The general case is described where authentication information is obtained from the HE/AuC by the SN. We abstract from the fact that the SN may receive the authentication information from the HE via another SN.

There are a number of protocols specified in [IS-41 (95)], which are associated with authentication and key agreement purposes and which are to be applied in specific situations:

- Initial registration with authentication
  Is applied in a scenario when a MS initially registers in a visited system.
- Initial origination with authentication
  Is applied in a scenario when the initial access in the visited system is a call origination.

- Call origination when *SSD* is shared
  Is applied for a call origination when *SSD* is shared.
- Termination with authentication
  Is applied in a scenario when a call is terminated to a visiting MS roaming in the visited system.
- *SSD* update when *SSD* is shared/not shared
  These procedures are applied when the SSD must be updated. The decision for the need of such an update may be the result of administrative procedures at the HE/AuC, expiration of an authentication time interval at the HE/AuC, or the report of a security violation from a visited system.
- Authentication when *SSD* is shared with another system
  Procedure to support authentication when the HE/AuC is currently sharing *SSD* with another system. It applies to all scenarios when an authentication request is received in the HE/AuC.
- Authentication on voice channel only
  Is required for systems that support authentication only on the voice or traffic channel. It is introduced, because IS-41 may not necessarily support authentication upon system access.
- Unique challenge when SSD is shared/not shared
  The procedures are applied to support a unique challenge (in contrast to the so-called global challenge which is broadcast and used for authentication by every user e.g. roaming in a specific location area).
- Authentication on flash request
  Is applied in case of a that a user uses the flash request feature, the significance of which is that it can be used to establish a third-party call. It is stated in the standard that this procedure guards against hijacking of the voice/traffic channel and that it may not be needed if signalling message encryption is being used between the MS and the appropriate entity in the SN.
- Call history count update
  Procedure to support the update of the call history counter *COUNT* of a MS in a visited system. The aim of the call history counter is to simplify clone detection.

Only some of the basic procedures are described below. For the other procedures the reader is referred to the IS-41 standard [IS-41 (95)].

In IS-41 two temporary user specific session keys are established within the authentication and key agreement process. They are both derived from the second part SSD_B of the user specific temporary key *SSD* and used to provide confidentiality. While *VPMASK* is applied to the voice channel, *SMEKEY* is used to encrypt certain fields within the signalling messages.

- **Initial registration**

In the Initial Registration procedure (cf. Figure 6.2.2-4) the MS determines that a new SN has been entered and that authentication is required on all system accesses (by the broadcast parameter *AUTH=1*). The broadcasted "global challenge" random number R_G to be used for authentication may also be obtained by the MS at this time.

The authentication result *AUTHR* is computed by the MS from the random number R_G (and the electronical serial number *ESN* of the handset and the first part of the mobile identification number *MIN1*) using *CAVE* under control of the temporary user specific authentication key SSD_A. The MS then sends *MIN*, *ESN*, *AUTHR*, its Call History Counter *COUNT*, and RC_G (the first 8 significant bits of R_G) to the SN, which forwards these parameters to the HE/AuC, replacing RC_G by R_G.

The HE/AuC verifies *MIN* and *ESN*. It then calculates the expected authentication response *XAUTHR* analogous to the calculation in the MS, and checks if *AUTHR* and *XAUTHR* match. The HE/AuC then verifies that the *COUNT* received from the MS is consistent with the value currently stored at the HE/AuC.

If this is not the case the HE/AuC responds with "Deny Access". Otherwise the response includes *SSD* and if needed directives to update the *SSD* or the call history counter *COUNT* of the MS according to the SN-HE agreed local administrative practices. The *SSD* Update procedure is described below. The *COUNT* Update procedure is not described in this document.

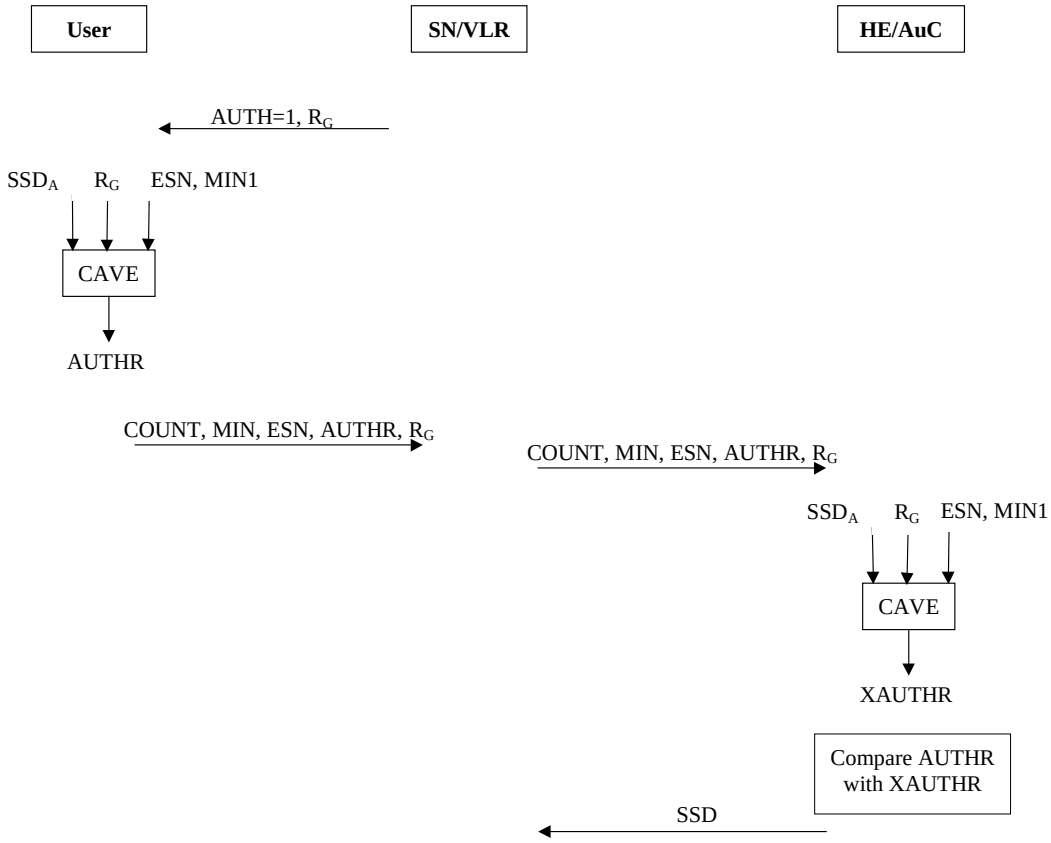

Figure 6.2.2-4: IS-41 protocol in case of initial registration when SSD is to be shared

- **SSD Update when SSD is shared**

In the SSD Update procedure with shared SSD (cf. Figure 6.2.2-5) the HE/AuC determines that the temporary user specific authentication and key agreement key *SSD* in the MS must be updated. This may be the result of administrative procedures at the HE/AuC, expiration of an authentication time interval at the HE/AuC, or the report of a security violation from a visited system.

The HE/AuC generates a random number R_{SSD}. A pending value of the *SSD* is computed at the HE/AuC from the random number R_{SSD} and the *ESN* of the MS 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 SN of the outcome of the updating procedure. The HE/AuC transmits the random number R_{SSD} and the new (pending) value of *SSD* to the SN.

The SN/VLR stores the pending *SSD* and forwards R_{SSD} to the MS (R_{SSD} may be sent over the control channel or over a voice or traffic channel). The pending *SSD* is now used in all subsequent calculations in this protocol run for *SSD* Update.

The pending value of the *SSD* is computed on the user side from the random number R_{SSD} and the *ESN* of the MS using *CAVE* under control of the user specific master key *A-key* in the same way as in the HE/AuC.

In order to authenticate the network the MS generates a random number $RAND_{BS}$ and sends it to the SN. The MS then computes the expected authentication result $XAUTH_{BS}$ using from *ESN*, *MIN1* and $RAND_{BS}$ using *CAVE* under control of the pending value of SSD_A.

The SN analogously computes the authentication result *AUTHBS*, and sends it to the MS. If the $AUTH_{BS}$ result provided by the SN matches the value $XAUTH_{BS}$ computed by the MS, the MS stores the new *SSD* value for use in future executions of *CAVE* and sends an *SSD* update confirmation message to the SN.

The SN generates a unique random number R_U - as opposed to a broadcast challenge as described above. The SN computes the expected authentication response $XAUTH_U$ from R_U, and the values *ESN*, *MIN1*, and *MIN2* associated with the MS using *CAVE* under control of the pending value of SSD_A. The SN now sends the generated unique challenge R_U to the MS. The MS computes the authentication response $AUTH_U$ analogously to the calculations for $XAUTH_U$ in the SN. and sends it to the SN.

The 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 HE/AuC. The current *SSD* is now updated by the pending *SSD*.

The HE/AuC stores the pending *SSD* value for use in future executions of *CAVE* for the MS. The HE/AuC sends an indication to the SN that service is to be provided to the MS.

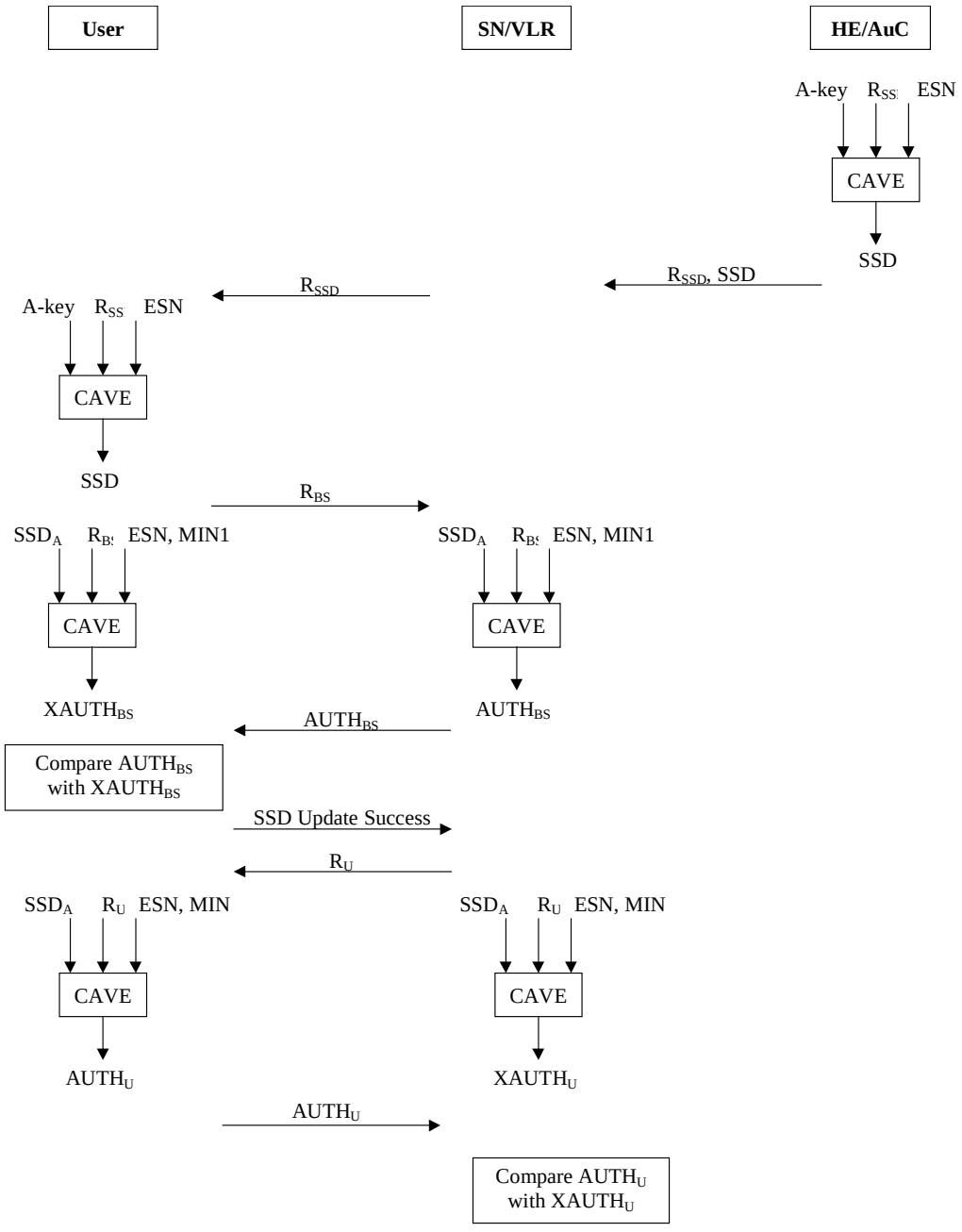

Figure 6.2.2-5: IS-41 protocol for SSD Update when SSD is shared

- **Call origination when SSD is shared**

In the *Call Origination* procedure with shared SSD (cf. Figure 6.2.2-6) the MS determines that authentication is required on all system accesses (by the broadcast parameter AUTH=1). The broadcasted "global challenge" random number R_G to be used for authentication may also be obtained by the MS at this time.

The authentication result *AUTHR* is computed by the MS from the random number R_G and the electronical serial number *ESN* of the handset and the dialled digits using *CAVE* under control of the temporary user specific authentication key SSD_A. Additionally the temporary cipher keys *VPMASK* (for later voice channel encryption) and *SMEKEY* (for later signalling channel encryption) are computed from R_G, *ESN* and the dialled digits under control of the temporary user specific key agreement key SSD_B. The MS then sends *MIN*, *ESN*, *AUTHR*, its Call History Counter *COUNT*, and RC_G (the first 8 significant bits of R_G) to the SN.

The SN/VLR verifies *MIN* and *ESN*. It then calculates the expected authentication response *XAUTHR* and the temporary cipher keys *VPMASK* and *SMEKEY* analogous to the calculation in the MS, and checks if *AUTHR* and *XAUTHR* match. The SN/VLR then verifies that the *COUNT* received from the MS is consistent with the value currently stored.

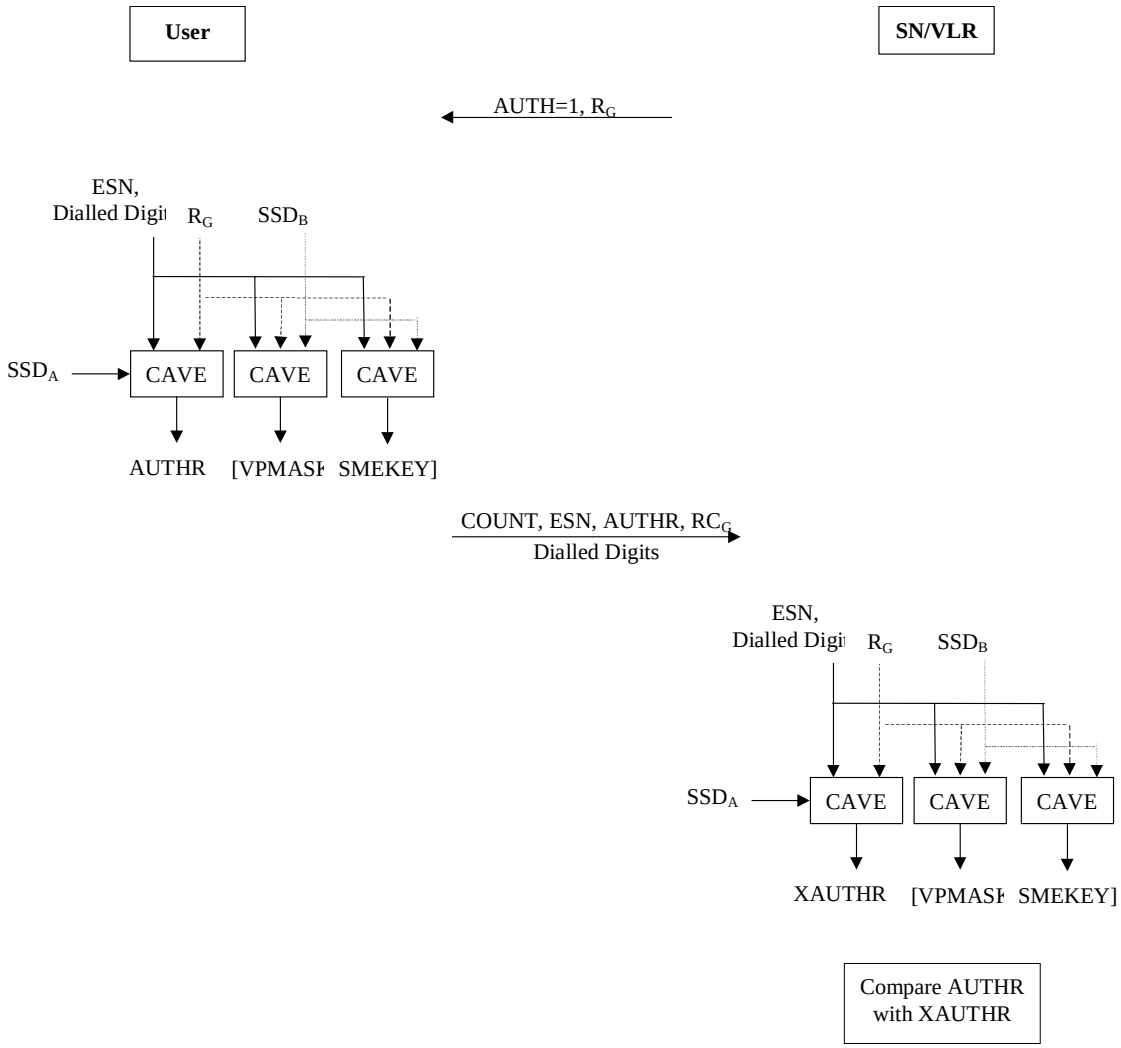

Figure 6.2.2-6: IS-41 protocol in case of call origination when SSD is shared

(d) How are the goals achieved?

- Entity authentication:

  Initial Registration: Entity authentication of the user to the SN is ensured by employing a "distributed" variant of the challenge-response mechanism according to [ISO 9798-4, 5.1.2], using R_G as a challenge and *AUTHR* as the response computed under control of the user specific key SSD_A. Here, HE trusts SN that R_G is a suitable challenge, and SN is told by HE (whom it trusts) that HE could verify *AUTHR.*

  SSD Update and *Call Origination:* Entity authentication of the user to the SN is ensured by employing a challenge-response mechanism according to [ISO 9798-4, 5.1.2], using R_G *(*R_U respectively*)* as a challenge and *AUTHR* (*AUTH*$_U$ respectively*)* as the response computed under control of the user specific key SSD_A. In case of *SSD* update, entity authentication of SN to user is ensured by the same mechanisms using the challenge R_{BS} and the response $AUTH_{BS}$.

As in case of initial registration no keys are agreed, the following goals do not apply to the initial registration case.

- Agreement between user and SN on a shared secret key.

  Call Origination: The two shared secret keys *VPMASK* and *SMEKEY* are derived in course of the protocol run for call origination, both to be used for confidentiality protection on the radio interface. The first one for encryption of the voice channel and the latter one to encrypt the signalling channel.

  SSD Update: A shared *SSD* is established in the course of the protocol run between SN and user.

- Implicit key authentication:

  Call Origination:

  - Implicit key authentication of the network to the user is ensured by the fact that the computation of the agreed keys *VPMASK* and *SMEKEY* requires knowledge of the user specific key SSD_B which, besides the user, only entities trusted by the HE have.
  - Implicit key authentication of the user to the SN: This is based on the fact that the computation of the agreed keys *VPMASK* and *SMEKEY* requires knowledge of the user specific key SSD_B which, besides trusted network entities, only the user has.

  SSD Update:

  - Implicit key authentication of the network to the user is ensured by the fact that the computation of the shared *SSD* requires knowledge of the user specific *A-key* which, besides the user, only the HE knows. The user trusts the HE that it will distribute *SSD* only to entities it trusts.
  - Implicit key authentication of the user to the SN: This is based on the fact that the computation of the shared *SSD* requires knowledge of the user specific *A-key* which, besides the HE, only the user has. The HE gives the corresponding assurance to the SN.

- Assurance of key freshness:

  Call Origination:

  - Assurance of key freshness to the SN is given by the fact that that SN can control the freshness of the random seed R_G used to derive the keys *VPMASK* and *SMEKEY*. *SSD update:*
  - Assurance of key freshness to the SN is given by the fact that that the HE can control the freshness of the random seed R_{SSD} used to derive *SSD*, and by the trust of SN in HE.

- Key (seed) confirmation:

  SSD Update: The possession of *SSD* is mutually confirmed by the use of SSD_A in the mutual entity authentication between SN and user.

  Call Origination: The SN is assured that the user has the correct key seed R_G because the user can have computed the correct response *AUTHR* only when knowing the correct R_G. The MS is assured

that the SN has the correct values of *ESN* and the dialled digits as *AUTHR* was equal *XAUTHR* which was also calculated using these values.

(1) Are all goals required for an AKA for UMTS achieved?

Not all of the goals required for an AKA for UMTS are achieved by the protocol. The missing goals are listed below. Further evaluation is therefore not carried out.

(2) Limitations of the protocol

The following goals required for utilising the protocol in UMTS systems are not achieved by the IS-41 protocol:

- Assurance of key freshness to the user and
- confidentiality of the user identity.

Attacks on the protocol are described in [Pat97]. The main threats mentioned their deal with false base station attacks as described in section 4 above.

Another attack described there is based on specific features of the mobile system, for some of which specific authentication variants are specified. E.g. the so-called "unique challenge" command is used in case that the user is roaming in an SN which does not have authentication capabilities. This command facilitates authentication between the user and his HE/AuC without involving the SN.

A further attack described in [Pat97] is based on the misuse of the so-called "flash request" feature, which facilitates that a user during a call is able to initiate a second call without terminating the first one.

6.2.2.5 Royal Holloway Protocol

The discussion in this section is based on [ETSI 99C050]. The protocol was proposed by the Royal Holloway University of London and therefore usually referred to as RHUL protocol.

(a) Prerequisites

The scheme is based on the use of five symmetric algorithms *Au*, *An*, *As*, *Ak* and *Cu*. All five algorithms are implemented on the USIM. Algorithms *An*, *As* and *Cu* are also implemented in the HE/AuC while algorithms *Au*, *Ak* and *Cu* are also implemented in the SN/AuC. The scheme requires a user-specific authentication key *Ksu*, which is held in the USIM and in the HE/AuC.

The authentication exchange with the user proves knowledge of authentication data given to an SN/VLR by an HE/AuC. The SN/VLR is assumed to be trusted by the HE to handle this authentication data securely. It is also assumed that the intra-system interfaces linking the SN/VLR to the HE/AuC, and linking SN/VLRs, are adequately secure.

Cryptographic processing capabilities are required in the USIM, the HE and in the SN.

(b) Protocol goals achieved

- Mutual authentication
- Cipher key agreement
- User anonymity towards SN

(c) Description of the protocol

The RHUL scheme is an extension of the TETRA scheme which provides anonymity towards the SN.

When the user first attempts to access a service, it sends a random challenge *RANDu* and the temporary identity *TMSIs* to the SN. The SN recognises the HE for the user from the *TMUIs* and signals to that HE

requesting authentication data and a new temporary identity *TMUIs'*. The HE then instructs the authentication centre to produce the relevant data and signals this back to the SN. The HE generates a random *RNDu* and a key offset *KO* and then calculates:

– $TMUI_S$' a new temporary identity

– $CIPH_S = C_U(K_{SU}, RND_U, KO)$. $CIPH_S$ will conceal the new temporary identity $TMUI_S$' when it is transmitted the first time to the USIM.

– $AUTH_S = A_S (K_{SU}, RND_U || TMUI_S')$

– $K_{NU} = A_N(K_{SU}, NOID, KO)$

The HE sends *TMUIs' xor CIPHs, Knu, XAUTHs, KO* to the SN. The SN generates *RNDu* and calculates:

– $TMUI_S$' a new temporary identity

– $AUTH_N = A_U (K_{NU},RND_N || RND_U || TMUI_N')$

– $CIPH_N = C_U (K_{NU}, RND_U)$. $CIPH_N$ will conceal the new temporary identity $TMUI_N$' when it is transmitted the first time to the USIM.

– $K_S = A_K (K_{NU} , RND_U || RND_N || TMUI_N')$.

– $XAUTH_U = A_U (K_{NU}, RND_U || RND_N)$.

The SN then sends *TMUIs' xor CIPHs, AUTHs, RNDn, TMUIn' xor CIPHn, KO* to the USIM.

On receipt the USIM calculates

– $CIPH_N$ and $CIPH_S$ in the same way as the SN and the HE did.

– $TMUI_S' = (TMUI_S' \text{ xor } CIPH_S) \text{ xor } CIPH_S$

– $TMUI_N' = (TMUI_N' \text{ xor } CIPH_N) \text{ xor } CIPH_N$

– $XAUTH_S$ and K_{NU} in the same way as the HE did.

– $AUTH_U$, $XAUTH_N$ and K_S in the same way as the SN did.

The USIM compares the received $AUTH_N$ and $AUTH_S$ with the calculated values *XAUTHn* and *XAUTHs*.

The USIM sends $AUTH_U$ to the SN. The SN compares $AUTH_U$ with the calculated value $XAUTH_U$.

The general procedure for authentication and key agreement for new registrations is shown in Figure 6.2.2-7.

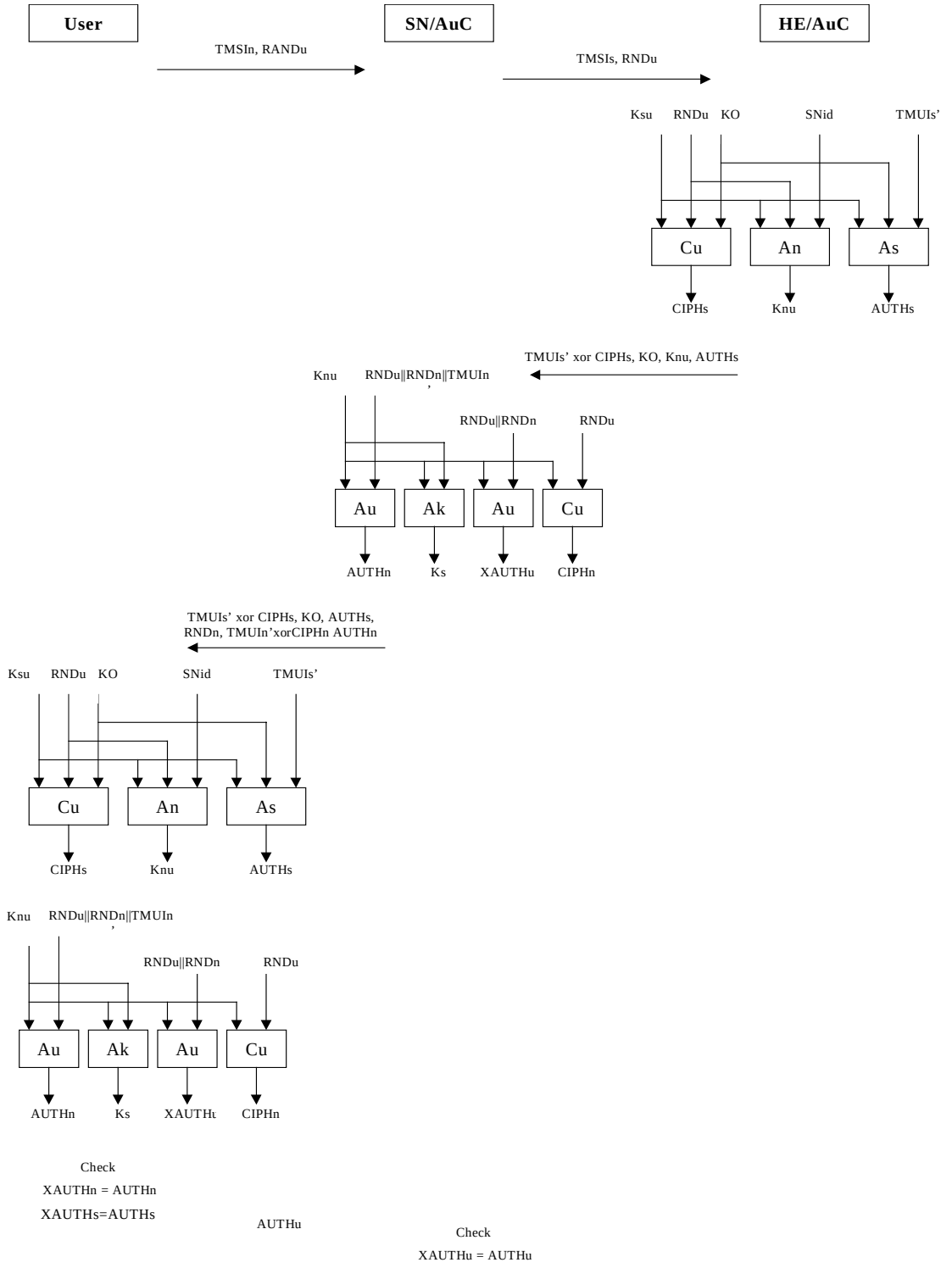

Figure 6.2.2-7: The RHUL scheme

(1) Are all goals required for an AKA for UMTS achieved?

yes

(2) Limitations of the protocol

(3) Implications on the protocol in case of UMTS system failures

(3.1) Compromise of SN authentication information

The key data may be obtained by eavesdropping the signalling links or by compromising SN/VLR security.

An attacker who obtains *the current* key data for a particular user registered on an SN/VLR can masquerade as that user until the current set of triplets used in the SN/VLR to authenticate users is replaced.

An attacker who obtains *any valid* key data can force the use of a known cipher key in order to eavesdrop calls. The attacker could do this by masquerading as a legitimate network to relay calls to and from the target. The attacker would be able to calculate the cipher key *Ks* and the authentication response *AUTHn*, and the ciphering string *CIPHn* using the key data.

(3.2) Resilience to USIM or HE data corruption

Security information stored in the HE/AuC is updated during the lifetime of the USIM. In particular the temporary identity shared between the user and the HE is updated on a regular basis. Thus in the event of a catastrophic database failure all information for a particular user cannot be recovered from a single backup. Instead more complex backup procedures are required.

Some security information stored on the USIM needs to be updated. In particular the temporary identity shared between the user and the HE. The fact that some security information needs to be updated during the lifetime of the USIM may make it more likely that the security information on the USIM is corrupted.

(3.3) Resilience to breakdown of links between SN and HE

Since key data is designed to be used for more than one authentication exchange, and each authentication exchange uses a different *RNDu* and *RNDn*, then the key data can be safely reused in the event of SN-HE link failure. Note however that it would be good practice to update *Knu* on a regular basis.

(4) Protocol efficiency

(4.1) Number of passes

The protocol is three pass over the radio interface for authentication at both new registrations and for subsequent authentication.

(4.2) Number of calculations

For each authentication during new registrations and for subsequent re-keying the USIM must calculate *Knu*, *CIPHs* and *AUTHs* based on Ks. For authentication during new registrations and for each subsequent registration the USIM must also calculate *AUTHn*, *Ks*, *XAUTHu* and *CIPHu* based on *Knu*.

For each authentication the SN must calculate *AUTHn*, *Ks*, *XAUTHn* and *CIPHn* based on *Knu*.

For each authentication during new registrations and for subsequent re-keying the HE must calculate *Knu*, *CIPHs* and *AUTHs* based on *Ksu*.

(4.3) Precomputation of authentication information in the HE

The HE/AuC is not able to compute all the key data before it is requested by the SN, since some of it depends on a random challenge from the user, *RNDu*.

(4.4) Frequency of SN - HE signalling

Key data sent to the SN in response to a request could potentially be used indefinitely by the SN. However, it is good practice to update *Knu* on a regular basis, thus the SN will need to signal requests for a new *Knu* to the HE.

(5) Implications on the mechanism on the UMTS infrastructure

The SN must support the handling of key data and the computation of various cryptographic parameters. The HE must support a new AuC.

(6) Compatibility with GSM security architecture

(6.1) Ease of migration from GSM to UMTS

A new AuC is required. New USIMs must be issued. The SN must be modified to support the handling of key data and the computation of various cryptographic parameters.

(6.2) Ease of roaming between GSM and UMTS networks

GSM users roaming into a UMTS network cannot use the UMTS security architecture in the serving network for authentication and key agreement. Additional functionality in the SN is required to support the handling of triplets.

UMTS users roaming into a GSM network cannot use the GSM security architecture in the serving network for authentication and key agreement. Additional functionality in the SN is required to support the handling of key data and to compute various cryptographic parameters.

(7) Compatibility with IS-41

(7.1) Ease of migration from IS-41 to UMTS

Major changes to the serving network required to support the handling of key data rather than shared secret data. Major changes to the home environment required to support a different Authentication Centre.

(7.2) Ease of roaming between UMTS and IS-41 networks

UMTS users roaming into an IS-41 network cannot use the IS-41 security architecture in the serving network for authentication and key agreement. Additional functionality in the SN is required to support the handling of key data and to compute various cryptographic parameters.

IS-41 users roaming into a UMTS network cannot use the UMTS security architecture in the serving network for authentication and key agreement. Additional functionality in the SN is required to support the handling of shared secret data and to compute various cryptographic parameters.

(8) Need for a standard AKA algorithm

An and *As* can be operator specific, but *Au*, *Ak* and *Cu* need to be standardised to allow USIMs to roam to different SNs.

6.2.2.6 SEQ protocol

(a) Prerequisites

There are three parties communicating in the protocol: the AuC in the HE, the VLR in the SN and the user represented by his USIM. The authentication and key establishment method described is of symmetric

secret key type. In this method one secret, the authentication and key establishment key, *K*, shall be shared by two parties, the HE and the user.

Authentication and key establishment consists of two procedures: First, authentication information is distributed to the SN by the HE. Second, an authentication exchange is run between the user and the SN. The authentication information consists of the parameters necessary to carry out the authentication exchange and in the agreed keys.

The user trusts the HE in all respects concerning this protocol. The HE trusts the SN to handle authentication information, sent by the HE to the SN, securely. The HE distributes authentication information only to entities it trusts. The SN trusts the HE to send correct authentication information. The SN accepts authentication information only from entities it trusts. The intra-system interfaces linking the SN to the HE, and linking SNs, are adequately secure.

The scheme is based on the use of five symmetric key based cryptographic functions *f1 to f5*. All five functions are implemented on the user and the HE, respectively. *f1* and f*2* are MAC functions, *f3*, *f4* and *f5* are key generating functions.

There is a random number generator in the HE.

In order to control the use of sequence numbers, counters SEQ_{US} and SEQ_{HE} need to be maintained by the user and by the HE respectively. In the general case, there is one counter SEQ_{HE} for each user.

It is assumed that the association of the messages of the AKA protocol with a certain user is done by other means. (E.g. the dedicated signalling channel over which the messages are sent is associated with an IMUI via a TMUI.)

(b) Protocol goals achieved

A protocol run consists of three steps: The distribution of authentication information for this run from the HE to the SN, an authentication request from the SN to the user and an authentication response from the user to the SN. The authentication information may be stored in the SN for some time. We abstract from the fact that the SN may receive the authentication information from the HE via another SN.

The goals achieved at the end of a successful protocol run are:

- Entity authentication:
  - Entity authentication of the user to the SN
  - Entity authentication of the HE to the user

  Note: There are different definitions in the literature of what constitutes entity authentication. Depending on the type of time variant parameter used, these definitions differ in the guarantees given to the verifier about the time the evidence received from the claimant was produced. For entity authentication of the user to the SN, the SEQ protocol, using random challenges as time variant parameters, provides assurance to the SN that the evidence was generated during the current protocol run. For entity authentication of the HE to the user, the SEQ protocol, using sequence numbers as time variant parameters, only provides assurance to the user that the evidence was not used in a previous protocol run.
- Agreement between user and SN on two shared secret keys.
- Implicit key authentication:
  - Implicit key authentication of the network to the user.
  - Implicit key authentication of the user to the SN

  Note: The later use of the agreed keys by the SN implicitly provides assurance to the user that the SN is authorised by the HE to deliver service to the user.
- Assurance of key freshness
  - Assurance of key freshness to the SN.

- Assurance of key freshness to the user.

- Key seed confirmation
  - Key seed confirmation from the user to the SN
  - Key seed confirmation from the HE to the user
- Confidentiality of the user identity related information on the interface between the user and the SN

  (An eavesdropper on the interface between the user and the SN cannot gain information on the user identity from the protocol.)

(c) Description of the protocol

- **Overview**

Figure 6.2.2-8 provides an overview of the authentication and key establishment method. Detailed definitions can be found in the subsections below. Figure 6.2.2-8 shows that, after receiving an authentication information request, the HE/AuC generates an ordered array of n authentication vectors. Each authentication vector consists of five components (and hence may be called a UMTS "quintuplet" in analogy to GSM "triplets"): A random number *RAND*, an expected response *XRES*, a cipher key *CK*, an integrity *IK* and an authentication token *AUTN*. This array of n authentication vectors is then sent from HE/AuC to SN/VLR. It is good for n authentication exchanges between the SN/VLR and the USIM. In an authentication exchange the SN/VLR first selects the next (the i-th) authentication vector from the array and sends the parameters *RAND(i)* and *AUTN(i)* to the user. The USIM checks whether *AUTN(i)* can be accepted and, if so, produces a response *RES(i)* which is sent back to the SN/VLR. The USIM also computes *CK(i)* and *IK(i)*. The SN/VLR compares the received *RES(i)* with *XRES(i)*. If they match SN/VLR considers the authentication exchange to be successfully completed. The established keys *CK(i)* and *IK(i)* will then be transferred by the USIM and the SN/VLR to the entities which perform ciphering and integrity functions.

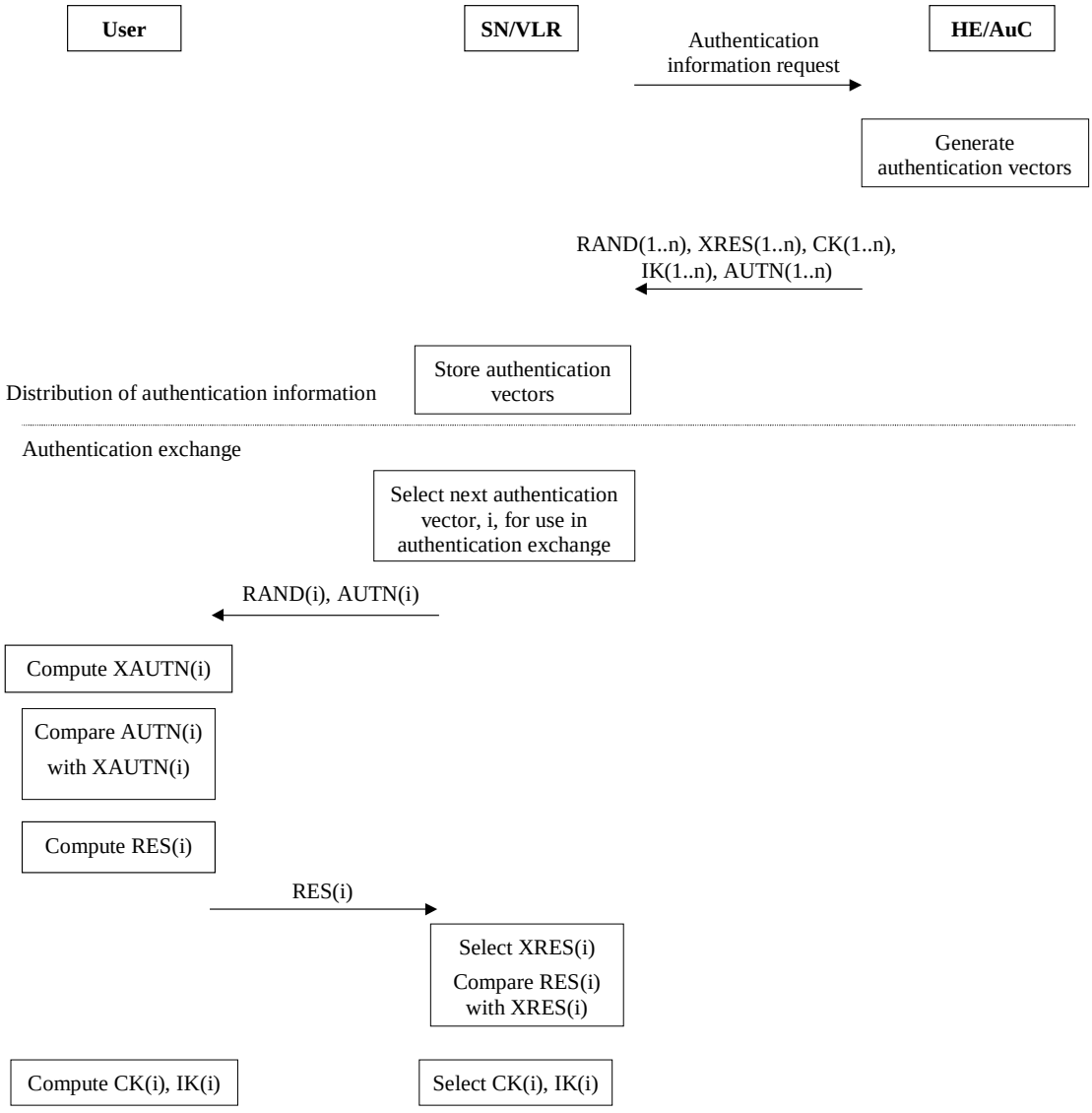

Figure 6.2.2-8: Overview of SEQ authentication and key establishment method

The presented protocol is valid under the general assumptions stated. Under certain additional assumptions simplifications of the above protocol are possible. These will be discussed in the notes further below.

- **General procedure for distributing authentication vectors ("quintuplets") to the SN/VLR**

When needed for each user, the SN/VLR sends a request for authentication information to the HE/AuC corresponding to the user. The HE/AuC generates n random challenges *RAND(1..n)* and n consecutive sequence numbers *SEQ(1..n)* from the appropriate HE/AuC counter, starting with $SEQ(1) = SEQ_{HE}+1$. The counter SEQ_{HE} is then reset to *SEQ(n)*. These parameters are used to generate an ordered array of authentication parameters as follows (we omit the reference to the i-th vector in the following):

$AUTN = SEQ \oplus Ka \,||\, f1_K\,(PAR1 \,||\, SEQ \,||\, RAND)$

where *f1* is a MAC function.

$XRES = f2_K\,(PAR2 \,||\, RAND)$ where *f2* is a MAC function.

CK = $f3_K$ (PAR3 || RAND) where *f3* is a key generating function.

IK = $f4_K$ (PAR4 || RAND) where *f4* is a key generating function.

Ka = $f5_K$ (PAR5 || RAND) where *f5* is a key generating function.

Here, *Ka* is an "anonymity" key used to conceal the sequence number as the latter may expose the identity and location of the user.

The need for *f5* to use a long-term key different from *K* is ffs.

The requirements on *f3*, *f4* and *f5* are ffs. It is also ffs in how far the functions *f1, ..., f5* need to differ and how they may be suitably combined.

PAR1, ..., PAR5 are pairwise different fixed initial values which may be used when similar or identical functions are used for *f1, ..., f5*. The need for the inclusion of *PAR1, ... PAR5* is ffs. When omitted they may be thought of as being integrated in the definition of the functions *f1, ..., f5* respectively.

These authentication parameters are used to construct an ordered array of authentication vectors for the user consisting of *RAND*, *XRES*, *CK*, *IK* and *AUTN*. This array is used by the SN/VLR in an authentication exchange with a user.

.

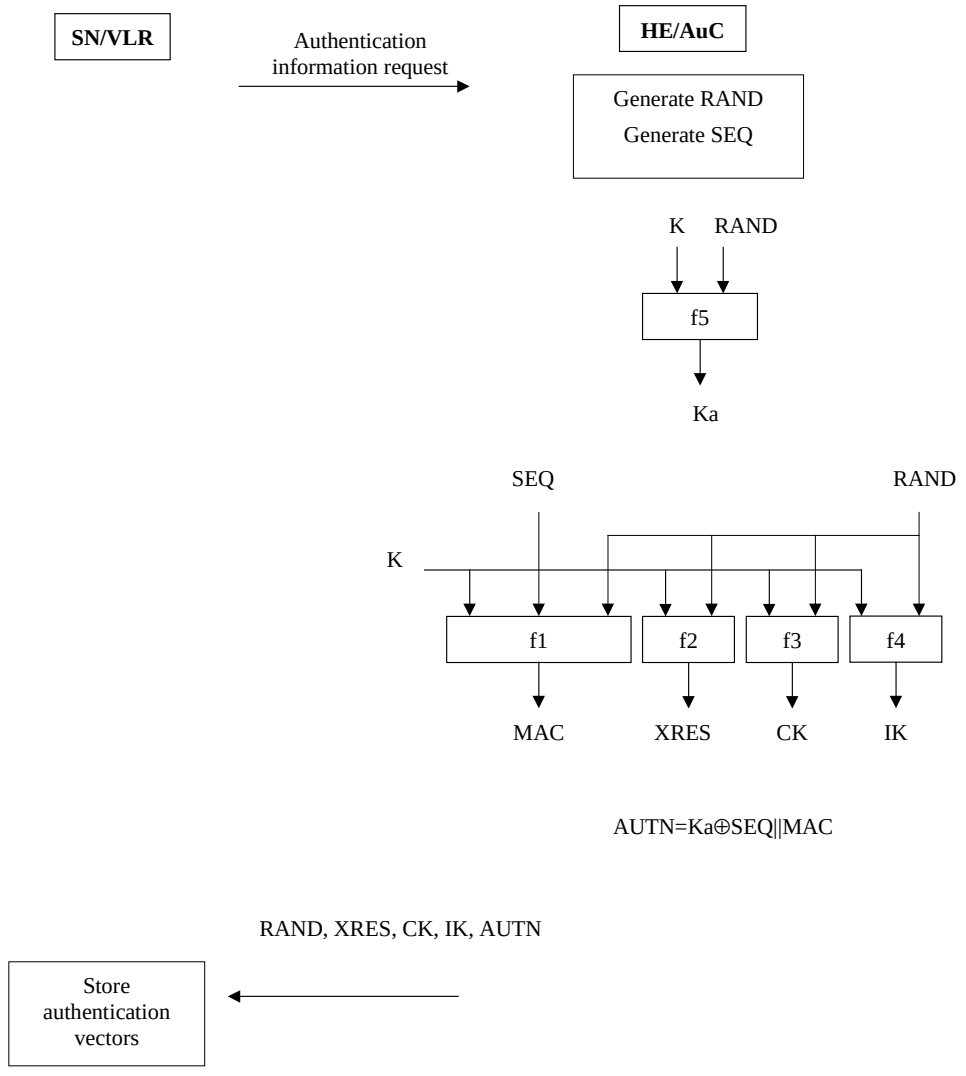

Figure 6.2.2-9: Procedure for distributing authentication vectors from the HE/AuC to the SN/VLR

- **The authentication exchange**

When the SN/VLR performs an authentication, it uses an ordered array of authentication vectors, corresponding to the user. The authentication exchange is described in Figure 6.2.2-10.

To authenticate a user, the SN/VLR selects the next vector from the ordered array of authentication vectors corresponding to the user. The SN/VLR sends to the user the random challenge *RAND* and an authentication token *AUTN* for network authentication from the selected authentication vector.

When the user receives *RAND* and *AUTN* it first verifies *AUTN* as follows:

The USIM first computes *Ka* from *RAND* and *K* using the function *f5*. It then computes *SEQ* from *SEQ⊕Ka* and *Ka*. Next the USIM computes $XMAC = f1_K\ (PAR1 \mid\mid SEQ \mid\mid RAND)$ from the available parameters and compares it with the received value. If there is a match and if *SEQ* is greater than the current value of the counter SEQ_{US}, the counter SEQ_{US} is set to *SEQ*. Otherwise, the authentication procedure is aborted indicating the cause of failure to the SN.

The user then computes *RES*, *CK* and *IK* from *PAR2* (*PAR3*, *PAR4* respectively), *RAND* and *K* using the function *f2* (*f3*, *f4* respectively). If this is more efficient, *RES*, *CK* and *IK* could also be computed earlier at any time after receiving *RAND*.

The user sends the response *RES* to the SN/VLR. The SN/VLR verifies *RES* by checking it against the expected response *XRES* from the selected authentication vector. If *XRES* equals *RES* then the authentication of the user has passed. The SN/VLR also selects the appropriate derived cipher key *CK* and derived integrity key *IK* from the selected authentication vector.

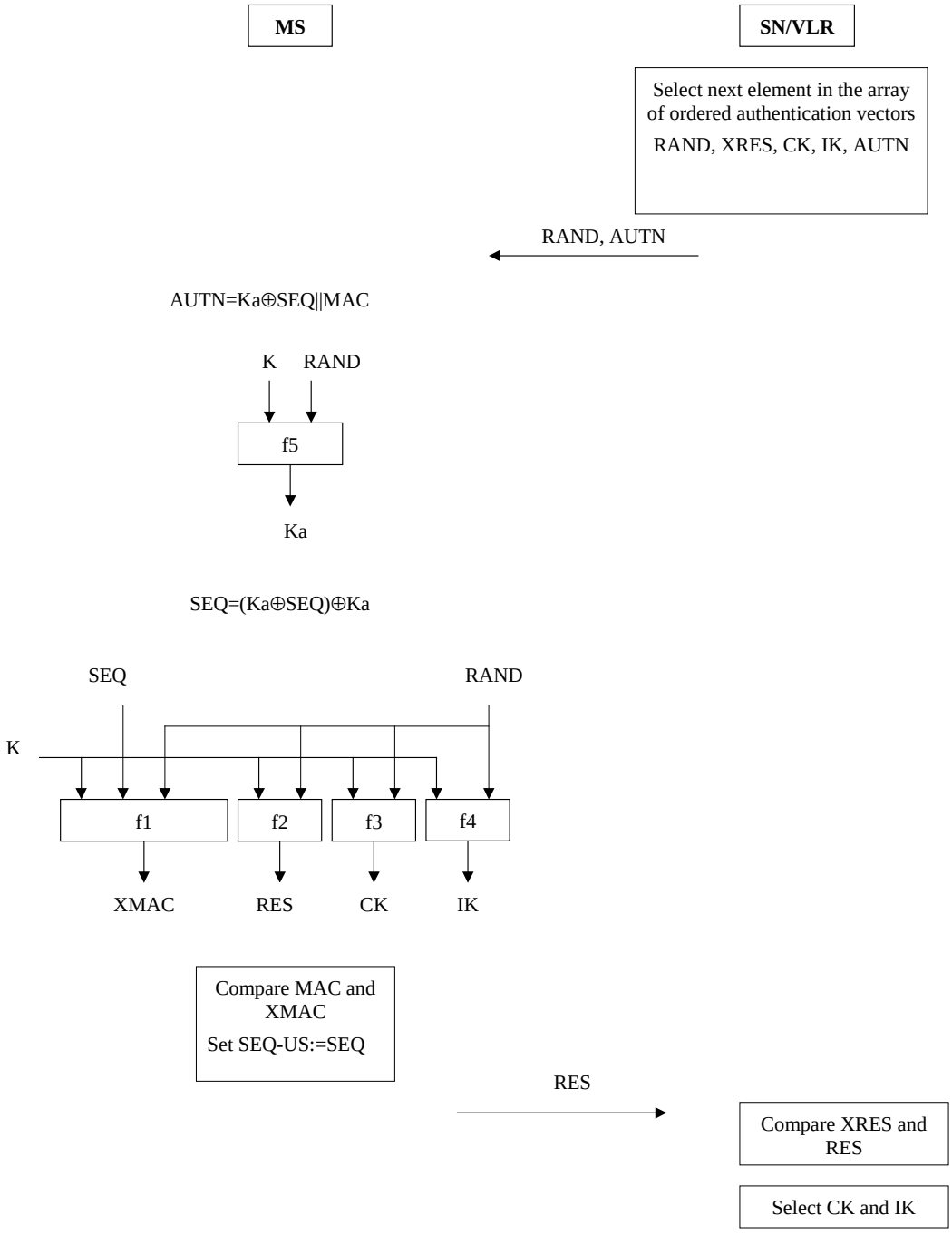

Figure 6.2.2-10: Overview of authentication exchange

Notes:

1) It is ffs whether the HE/AuC has to maintain separate counters SEQ_{HE} for each user, or whether the generation of sequence numbers as part of the generation of authentication vectors can be based on a global counter. This global counter may in turn be based on a clock with an appropriate granularity giving e.g. Universal Time Co-ordinates (UTC). It should be remembered in this context that sequence numbers have to be unique only on a per user basis, not across all users. This is important to determine the required granularity of the clock. Intermediate solutions are also possible where a sequence number would apply to a group of users.

2) If sequence numbers are derived from a global or group counter the information which can be derived from a sequence number on the identity of a user may be considered not a serious threat to the anonymity of the user. (If authentication vectors are generated in arrays using consecutive sequence numbers chaining of authentication attempts pertaining to the same user may still be possible.) If the residual threat to the user's anonymity is deemed negligible then the use of the anonymity key *Ka* to conceal the sequence number is no longer needed.

3) If *Ka* was no longer needed in order to conceal the sequence number then it could be considered to replace *RAND* with *SEQ*. It was decided, however, to keep *RAND* for maximum compatibility with GSM.

- **Conditions on the use of authentication information**

Using the procedure described in the previous subsections, an array of authentication vectors will have to be used in the specific order in which they were generated, otherwise the user will reject the authentication attempt. The SN/VLR shall use a quintuplet only once and, hence, shall send out each message *RAND* || *AUTN* only once no matter whether the authentication attempt was successful or not. A consequence is that authentication vectors cannot be reused. (Note: Re-use of authentication vectors is an insecure practice anyway.)

When a user leaves a VLR area the VLR shall delete any quintuplets possibly after forwarding any remaining quintuplets to the new VLR. Consequently, there is at most one VLR at a time holding authentication vectors for a particular user.

(d) How are the goals achieved?

The goals mentioned in (b) are achieved in the following way:

- Entity authentication:
  - Entity authentication of the user to the SN is ensured by employing a challenge-response mechanism according to [ISO 9798-4, 5.1.2], using *RAND* as a challenge and *RES* as the response computed under control of the user specific key *K*, and distributing *XRES* from the HE to the SN.
  - Entity authentication of the HE to the user is ensured by the use of the sequence number SEQ as a time variant parameter for the calculation of the parameter *AUTN* under control of the user specific key *K*, according to [ISO 9798-4, 5.1.1].
- Agreement between user and SN on two shared secret keys

  CK and *IK* are derived in course of the protocol run.
- Implicit key authentication:
  - Implicit key authentication of the network to the user is ensured by the fact that the computation of the agreed keys *CK* and *IK* requires knowledge of the user specific key *K* which, besides the user, only the HE has. The HE will distribute keys only to entities it trusts.
  - Implicit key authentication of the user to the SN: A corresponding assurance given by the HE to the SN is implied in the trust of the SN to the HE in the sending of correct authentication information. The HE can give this assurance based on the fact that the computation of the agreed keys *CK* and *IK* requires knowledge of the user specific key *K* which, besides the HE, only the user has.
- Mutual assurance of key freshness between SN and user:

  A corresponding assurance given by the HE to the SN is implied in the trust of the SN to the HE in the sending of correct authentication information. Both, the user and the HE can control the freshness of the random seed *RAND* used to derive the keys *CK* and *IK*. The HE can do this because it generates *RAND*, the user can do this because he can control the freshness of *SEQ*, and *SEQ* and *RAND* are jointly integrity protected in *AUTN*.

- Key seed confirmation between user and network:

  The SN is assured that the user has the correct key seed *RAND* because the user can have computed the correct response *RES* only when knowing the correct *RAND*. The user is assured that the HE has the correct key seed *RAND* because he knows from verifying *AUTN* that the HE knows *RAND*.

- Confidentiality of user identity related information on the interface between the user and the SN:

  The messages sent across the interface do not allow to derive any information about the user identity.

(1) Are all goals required for an AKA for UMTS achieved?

yes

(2) Limitations of the protocol

The protocol does not prevent pre-play attacks of the following kind: A number of AUTN messages may be obtained by an attacker beforehand and be later replayed to the user. Such an attacker cannot know the agreed keys, however. The use of one of these keys in the session rules out threats arising from this attack.

(3) Implications on the protocol in case of UMTS system failures

(3.1) Compromise of SN authentication information

If a number *n* of authentication vectors stored in the SN became known to an attacker he could perform the following attacks:

- He could impersonate the user towards the network in *n* authentications and the subsequent call set-ups based on the use of the integrity key;
- he could impersonate the network towards the user in *n* authentications based on the use of the integrity key;
- he could eavesdrop on n sessions in which the n compromised cipher keys were used.

n is typically small (in GSM, $n = 5$ is typical).

As soon as the legitimate user has performed a successful authentication based on a new authentication vector (with a new sequence number), the compromised authentication vector becomes worthless for the attacker.

(3.2) Resilience to USIM or HE data corruption

Below the process of re-synchronisation in case of failures is discussed.

Sequence numbers shall be sufficiently long so that they cannot wrap around during the lifetime of the USIM in normal operations.

If $SEQ \leq SEQ_{US}$ then authentication fails because the sequence number is not accepted by the user. In normal operations, as described in the preceding subsections, i.e. in the absence of failures or attacks, this does not happen. If *AUTN* can be verified, but $SEQ \leq SEQ_{US}$ then we speak of a synchronisation failure. The possible causes of synchronisation failures, and measures against it, are described below. The term ”synchronisation failure” shall also cover the case where no counter values are available in the AuC due to a database failure. A special re-synchronisation procedure may be used to recover from synchronisation failures. It is also described below.

What follows is based on per-user counters (not global counters) and applies to one user at a time.

Procedures to protect against synchronisation failure

1. **Back-Up:** There is a secure back-up method in the AuC which allows to determine the time of the last back-up and recover the value of the counter SEQ_{HE} at that time.

2. **Re-synchronisation procedure:** There is a re-synchronisation procedure RESYNC run between the HE/AuC and the USIM. It is initiated by the HE/AuC. The procedure is as follows:

   Upon receiving a re-synchronisation request, the user sends a re-synchronisation response *RAND* || *AUTN* with $SEQ = SEQ_{US}$ back to the HE/AuC.

   The USIM has several alternatives for producing *RAND* || *AUTN*. The choice does not affect other entities. The USIM may have stored the last successfully verified authentication message *RAND* || *AUTN* along with the new value of SEQ_{US}, or the USIM may have stored only the received *RAND* and recompute *AUTN*, or the USIM may generate a new *RAND* and compute *AUTN*. (The required properties of the function generating *RAND* are ffs.)

   The HE updates SEQ_{HE} only when the value of *SEQ* received is greater than SEQ_{HE}. In this case, the HE sets SEQ_{HE} to *SEQ*.

   Note: Of course, an attacker pretending to be a user can replay an old *RAND* || *AUTN* as a re-synchronisation message to the HE. However, this will not cause a denial of service attack because the HE will have seen the corresponding sequence number before and will decide that no update of SEQ_{HE} is needed.

Possible causes of synchronisation failure

The following possible causes of synchronisation failure may be distinguished:

1. **Failure during transmission of authentication information from HE/AuC to USIM:**

   This is expected to be the most common cause. It may be further subdivided as follows:

   1.1. An SN /VLR uses an authentication vector out of sequence.

   1.2. An attacker replays AUTN.

2. **Failure in the HE/AuC:**

   This may be further subdivided as follows:

   2.1. The value of the counter SEQ_{HE} is lost due to a database failure in the HE/AuC.

   2.2. The value of the counter SEQ_{HE} is erroneously set to a lower than expected value.

   2.3. The value of the counter SEQ_{HE} is erroneously set to a higher than expected value which is still less than the maximum value of *SEQ*.

   2.4. The value of the counter SEQ_{HE} used to produce a new *AUTN* equals the maximum value of *SEQ* ("wrap-around" of SEQ_{HE}). This case is expected to be very rare. It is ffs whether it has to be taken into account. If so, a modified handling of sequence numbers and re-synchronisation is needed.

3. **Failure in the USIM:**

   It is assumed that the USIM functions correctly. If there is an error in the USIM it has to be replaced. (Reason: Security of UMTS has to rely on the assumption that the USIM is working according to specification. The smart card industry indicates that secure counters can be implemented. Therefore, no particular measures related to a synchronisation failure caused by a USIM failure are believed to be needed.)

Determining a synchronisation failure event

1. The USIM determines that there is a synchronisation failure if it can verify *AUTN*, but if the value of *SEQ* is not acceptable.
2. The AuC may determine that there is a synchronisation failure as a result of internal procedures (e.g. by determining that there was a database failure).
3. The HE/AuC may determine that there is a synchronisation failure after a corresponding indication from the USIM.

Measures taken in case of a synchronisation failure event

1. If the USIM determines that there is a synchronisation failure then it indicates this fact to the SN/VLR in the authentication response which forwards the indication to the HE/AuC.
2. If the AuC determines that there is a synchronisation failure as a result of internal procedures (database failure) then it may reset the counter SEQ_{HE} in appropriate way using the secure back-up.

   One such way may be to reset SEQ_{HE} to the sum of the value stored in the back-up and a delta chosen so large that the increase of the value of SEQ_{HE} (number of authentication vectors generated) between the time of the last back-up and the time of reset will, in normal operations, be less than delta.
3. If the AuC determines that there is a synchronisation failure as a result of a corresponding indication from the USIM then it then it may reset the counter SEQ_{HE} in an appropriate way using the secure back-up or it may initiate the RESYNC procedure, according to its policy. After the completion of the procedure, the AuC generates a new batch of authentication vectors and sends it to SN/VLR.
4. Whenever the SN/VLR receives a synchronisation failure indication from the USIM or forwards a RESYNC request message to the USIM it deletes the authentication vectors for that user in the VLR.

   Note: If the USIM indicates that it could verify *AUTN* as in measure 1 then this would allow an attacker to test for a user's identity if the attacker knows previously used authentication information to be associated with a certain user. However, this is considered acceptable as the improved user confidentiality scheme based on encryption with a group key also allows this type of attack, so no new threat to user anonymity is created.

How do the measures address the possible causes of synchronisation failure?

1. Failure during transmission will result in failed authentication attempts and will cause the USIM to determine that there is a synchronisation failure. Then measures 1, 3 and 4 apply.
2. Failure in the HE/AuC
   2.1. Database failure is addressed by measure 2.
   2.2. If the value of SEQ_{HE} was already used before then this will result in a failed authentication attempt and will cause the USIM to determine that there is a synchronisation failure. Then measures 1, 3 and 4 apply.
   2.3. If the value of SEQ_{HE} is higher than it should be, but less than the maximum no action is required.

What has to be standardised?

As a minimum the following has to be standardised:

- the RESYNC procedure at the interfaces AuC/VLR and VLR/USIM consisting of a void re-synchronisation request and a re-synchronisation response containing *RAND* || *AUTN*.
- the synchronisation failure indications from the USIM to the SN/VLR and from the SN/VLR to the AuC.

(3.3) Resilience to breakdown of links between SN and HE

When the link between SN and HE breaks down at a time when new authentication vectors would be required then secure connection set-up based on the use of the integrity key is used up to the specified maximum number of such connection set-ups, until the link between SN and HE becomes available again.

(4) Protocol efficiency

(4.1) Number of passes

On the air interface:
For one authentication exchange, always two passes are required.

On the SN / HE interface:
For n authentication exchanges, one pass is needed if the size of a batch of authentication vectors is n.

(4.2) Number of cryptographic computations

Cryptographic computations are performed in the USIM and in the AuC. For one authentication exchange the following computations are required:

At the AuC:
Generation of a random number *RAND*, computation of one *MAC* over random number *RAND* and sequence number *SEQ*, one *MAC* over random number *RAND*, three key derivations using *RAND*.

At the USIM:
Computation of one *MAC* over random number *RAND* and sequence number *SEQ*, one *MAC* over random number *RAND*, three key derivations using *RAND*.

(4.3) Precomputation of authentication information in the HE

Precomputation of the whole AKA information is possible as the AKA information is not SN–specific.

(4.4) Frequency of SN – HE signalling

The frequency of SN_{HE} signalling is determined by the following factors: the size *n* of a batch of authentication vectors and the specified maximum number *m* of connection set-ups based on the use of the integrity key. One batch of authentication vectors sent in one signalling message is then good for *nm* connection set-ups.

(5) Implications on the mechanism on the UMTS infrastructure

(5.1) Need for an AuC in the SN

An AuC in the SN is not needed.

(6) Compatibility with GSM security architecture

(6.1) Ease of migration from GSM to UMTS

A new AuC is required. New USIMs must be issued. The SN must be slightly modified for the purposes of the AKA to handle the new format of authentication vectors.

(6.2) Ease of roaming between GSM and UMTS network

UMTS user roaming into GSM:

The AuC would issue triplets as a subset of UMTS authentication vectors, namely the triplet would be (RAND*, RES*, CK*) where the * denotes possible truncation to arrive at the parameter size required in GSM. The same functions could be used in both, AuC and USIM for both networks.

It should be noted, however, that, by roaming into GSM, the user has to accept the lower security standards of GSM. This should be made transparent to the user. A false base station could make a UMTS MS believe that it was in a GSM environment and could perform attacks on this basis. This is ffs.

GSM user roaming into UMTS:

It is not clear what this means. If it means SIM-roaming (plastic roaming) and if UMTS terminals are to accept GSM-SIMs then it is ffs in how far this is to be permitted. If so then the GSM user may obtain only GSM-type service. The ME and the SN, respectively, would have to be adapted in such a way that they can handle GSM triplets, and map them onto UMTS parameters and vice versa, possibly extending RAND(GSM), SRES and Kc to match the formats of RAND(UMTS), RES and CK. The algorithm for encryption would have to be able to accept shorter key lengths giving GSM grad security. Integrity of signalling messages could not be provided.

(7) Compatibility with IS-41security architecture

(7.1) Ease of migration from IS-41 to UMTS

A USIM functional module is required in the MS. A new AuC is required at the HE. The SN must be modified for the purposes of the AKA.

(7.2) Ease of roaming between IS-41 and UMTS networks

UMTS user roaming into IS-41:
An IS-41 application would be required on the USIM, similarly in the UMTS AuC.

IS-41 user roaming into UMTS:
A UMTS application would be required in the IS-41 terminal, similarly in the IS-41 AuC.

(8) Need for a standard AKA algorithm

No standard AKA algorithm is needed.

6.2.2.7 TETRA-3

(a) Prerequisites

There are three parties communicating in the protocol: the AuC in the HE, the VLR in the SN and the user represented by his USIM. The authentication and key establishment method described is of symmetric secret key type. In this method one permanent secret, the authentication and key establishment key, *K*, shall be shared by two parties, the HE and the user.

Authentication and key establishment consists of two procedures: In a new registration, authentication information is distributed to the SN by the HE, followed by an authentication exchange run between the

user and the SN. In a current registration, the authentication information distributed to the SN during new registration, is used in an authentication exchange run between the user and the SN.

The user trusts the HE in all respects concerning this protocol. The HE trusts the SN to handle authentication information, sent by the HE to the SN, securely. The HE distributes authentication information only to entities it trusts. The SN trusts the HE to send correct authentication information. The SN accepts authentication information only from entities it trusts. The intra-system interfaces linking the SN to the HE, and linking SNs, are adequately secure.

The scheme is based on the use of five symmetric key based cryptographic functions *f1 to f5*. All five functions are implemented on the USIM and the HE/AuC, respectively. (In an alternative version, which is not described here, algorithms *f4* and *f5* may be implemented in each SN/AuC.) The scheme requires a user-specific authentication key *K*, which is only held in the USIM and in the HE/AuC. A temporary user specific authentication and key agreement key *KT*, which is derived from the long-term key *K*, is sent to the SN by the HE/AuC.

Both, the user and HE/AuC have a random number generator implemented.

(b) Protocol goals achieved

The goals achieved at the end of a successful protocol run are:

- Entity authentication:

  In case of new registration:

  - Entity authentication of the HE to the user.
  - Entity authentication of the user to the SN (based on authentication data given to the SN/VLR by the HE/AuC).

  In case of current registration:

  - Entity authentication of the user to the SN.

- Agreement between user and SN on a shared secret key.

  In case of new registration:

  - agreement on a temporary authentication key *KT* and on cipher key *CK* and integrity key *IK*.

  In case of current registration

  - agreement on cipher key *CK* and integrity key *IK*.

- Implicit key authentication:

  In case of first registration:

  - Implicit key authentication of the network to the user for the keys *KT*, *CK* and *IK*.
  - Implicit key authentication of the user to the SN for the keys *KT*, *CK* and *IK*.

    (It is based on temporary user specific key data given to the SN/VLR by the HE/AuC).

  In case of current registration:

  - Implicit key authentication of the network to the user for the keys *CK* and *IK*.
  - Implicit key authentication of the user to the SN for the keys *CK* and *IK*.
  - (based on temporary user specific key data given to the SN/VLR by the HE/AuC).

- Key seed confirmation

  In case of new registration:

  - Key seed confirmation from user to SN for the keys *KT*, *CK* and *IK*.
  - Key seed confirmation from HE to user for the keys *KT*, *CK* and *IK*.

In case of current registration:

- Key seed confirmation from user to SN for the keys *CK* and *IK*.

- Assurance of key freshness.

  Assurance is given to both, user and SN for all keys in both protocols.

- Compatible with mechanisms used for the provision of confidentiality of the user identity on the air interface.

(c) Description of the protocol

The general case is described where authentication vectors are obtained from the HE/AuC. We abstract from the fact that the SN may receive the authentication information from the HE via another SN.

- **New registration**

The following authentication and key agreement protocol is performed when a MS enters a new network, i.e. a network which does not contain an entry in the VLR corresponding to this MS. It is depicted in Figure 6.2.2-11.

The user at first sends a random seed *RSu* to the SN/VLR, which forwards it in an authentication information request to the HE/AuC corresponding to the user. The HE/AuC generates a random seed component *RSh*. The temporary authentication key *KT* is then computed from the random seeds *RSu* and *RSh* using the cryptographic function f1 under control of the user specific authentication key *K*.

This key *KT* is subsequently used for user and network authentication and for session key derivation. For authentication the random seeds are regarded as random challenges. The HE/AuC computes the response *RES1* using function *f2* under control of the key *K*, as well as the expected response *XRES2* to the network's challenge *RSh* using function *f3* under control of the key *K*.

The following two calculation can be performed in the HE/AuC but also in the AuC of the requesting SN. In the first case the temporary cipher key *CK* is computed by the HE/AuC from the random seeds *RSu* and *RSh* using function *f4* under control of key *KT*. Analogously the temporary integrity key *IK* is computed using function *f5*.

The random seed of the network *RSh*, the response *RES1*, the expected response *XRES2* and the temporary authentication key *KT* are passed to the requesting SN/VLR. In case that *CK* and *IK* are computed by the HE/AuC, they are also sent to the requesting SN/VLR.

The SN/VLR forwards the received values *RSh* and *RES1* to the user. The USIM of the user then computes all the values computed in the HE/AuC analogously, namely *KT*, *XRES1*, *RES2*, *CK* and *IK*. Then the USIM compares its computed expected authentication response *XRES1* with the received one *RES1*. In case that these values do not match authentication has failed and an appropriate indication is made to the SN. Otherwise the USIM responds to the SN/VLR by sending its calculated authentication response parameter *RES2*. SN/VLR compares *XRES2* received from the HE/AuC with *RES2* and in case of a matching the user is regarded as authentic.

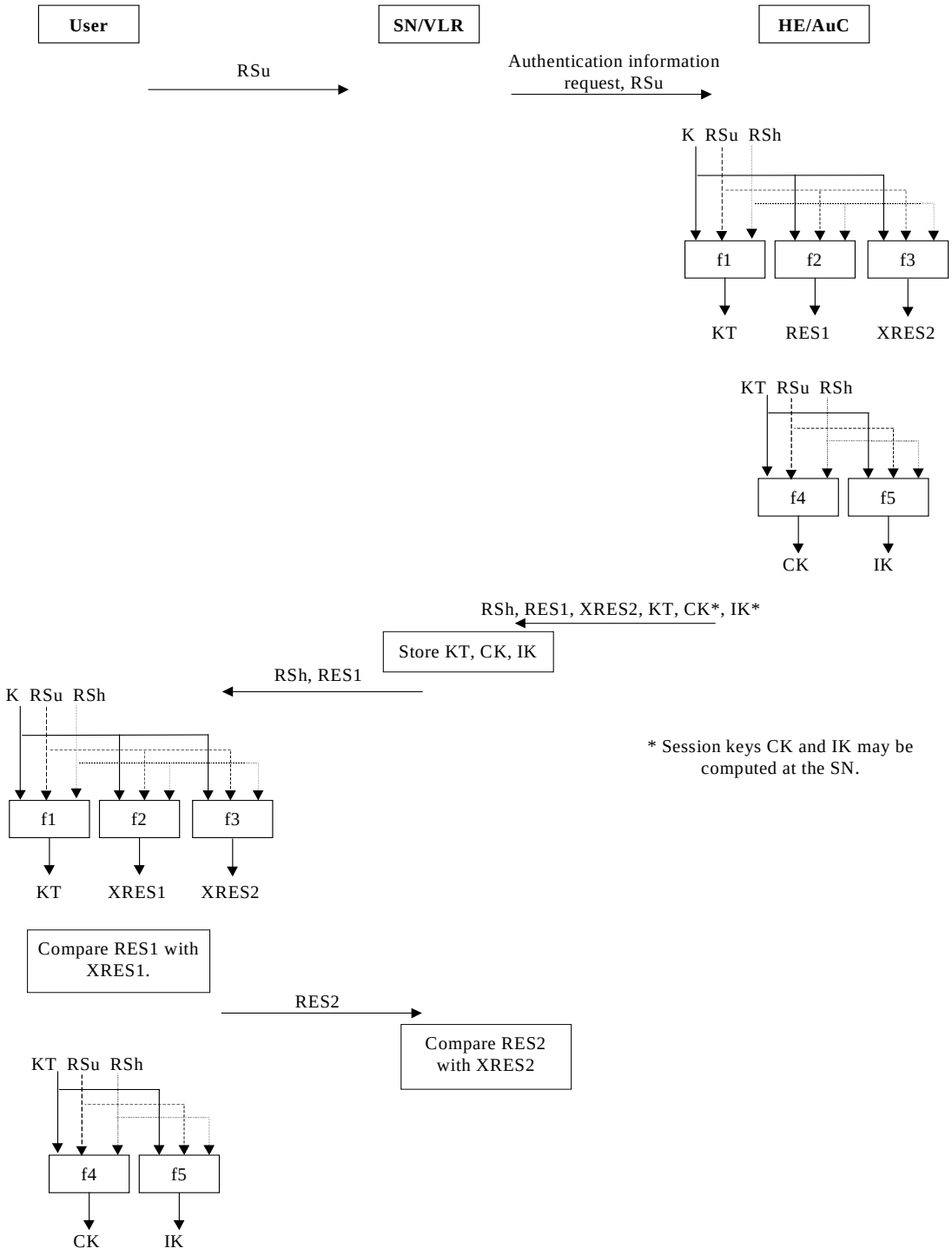

Figure 6.2.2-11: TETRA-3 protocol in case of new registration

- **Current registration**

The following authentication and key agreement protocol is performed when there is an already existing relationship between the MS and the SN, i.e. the SN contains an entry in the VLR corresponding to this MS which includes the current temporary authentication key *KT*. It is depicted in Figure 6.2.2-12.

The protocol starts on the network side, in contrast to the one above used for new registration. The SN at first generates a random seed *RSsn* and sends it to the user. After reception the user also generates a random seed *RSu*.

The user utilises the random seeds *RSu* and *RSsn* to calculate the authentication response *RES* using function *f3* under control of his temporary authentication key *KT*. The user also uses *RSu* and *RSsn* to calculate the temporary keys *CK* and *IK* using the function *f4* and *f5* under control of the key *KT*.

The user then sends his random seed *RSu* and his authentication response *RES* to the network. The network subsequently calculates the expected authentication result *XRES* and the temporary keys *CK* and *IK* analogously to the calculations performed by the user before. The SN then compares the values *XRES* and *RES*. In case of a match the user is regarded as authentic.

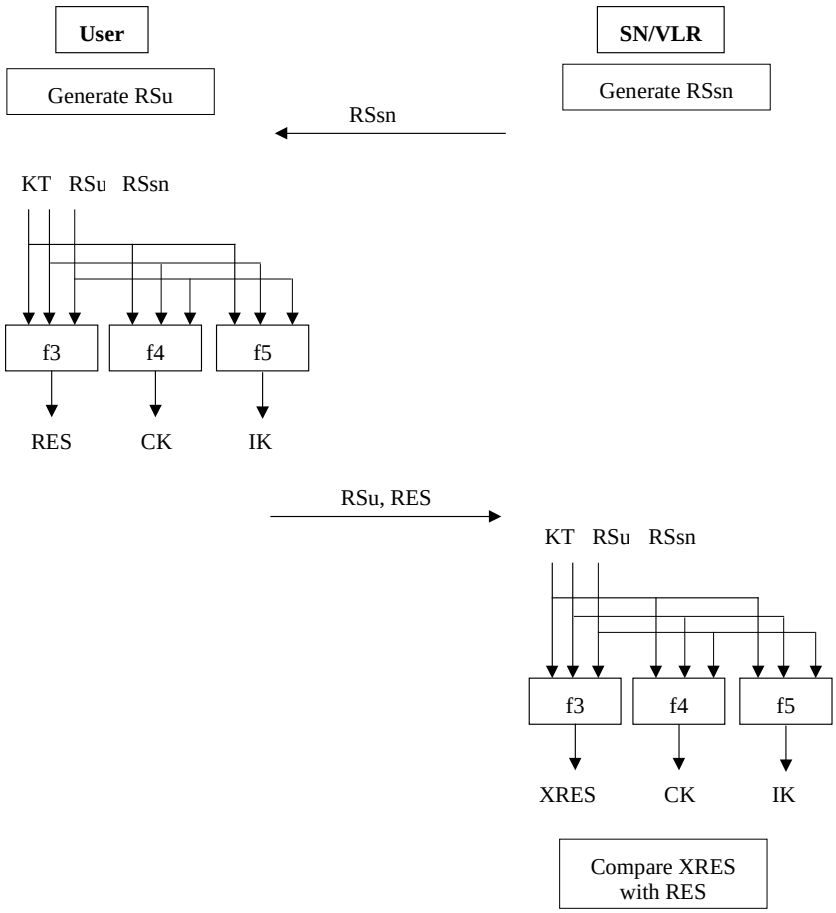

Figure 6.2.2-12: TETRA-3 protocol in case of current registration

(d) How are the goals achieved?

The goals mentioned in (b) are achieved in the following way:

- Entity authentication:

  In case of new registration:

  - Entity authentication of the HE to the user is ensured by the usage of the random seed *RSu* as nonce under control of the user specific key *K* for the calculation of the authentication responses *RES1*.

- Entity authentication of the user to the SN is ensured by the usage of the random seed *RSh* as nonce under control of the user specific key K for the calculation of the authentication responses *RES2*.

In case of current registration:

- Entity authentication of the user to the SN is ensured by the usage of the random seed *RSn* as nonce under control of the user specific temporary key *KT* for the calculation of the authentication responses *RES*. It is based on authentication data given to the SN/VLR by the HE/AuC.

- Agreement between user and SN on a shared secret key:

  In case of new registration: *KT*, *CK* and *IK* are calculated in course of the protocol run.

  In case of current registration: *CK* and *IK* are calculated in course of the protocol run.

- Implicit key authentication:

  In case of new registration:

  - Implicit key authentication of the network to the user is ensured by the following facts: The computation of the key *KT* requires knowledge of the user specific key *K* which, besides the user, only the HE has. The HE will distribute keys *KT* only to entities it trusts. The computation of the keys *CK* and *IK* requires knowledge of the user specific key *K* which, besides the user, only trusted networks entities have.
  - Implicit key authentication of the user to the SN: A corresponding assurance given by the HE to the SN for the keys *KT*, *CK* and *IK* is implied in the trust of the SN to the HE in the sending of correct authentication information. The HE can give this assurance based on the following facts: The computation of the key *KT* requires knowledge of the user specific key *K* which, besides the HE, only the user has. The computation of the agreed keys *CK* and *IK* requires knowledge of the user specific key *KT* which, besides trusted network entities, only the user has.

  In case of current registration:

  - Implicit key authentication of the network to the user is ensured by the fact that the computation of the keys *CK* and *IK* requires knowledge of the user specific key *KT* which, besides the user, only trusted networks entities have.
  - Implicit key authentication of the user to the SN is ensured by the fact that the computation of the keys *CK* and *IK* requires knowledge of the user specific key *KT* which, besides trusted networks entities, only the user has.

- Mutual assurance of key freshness:

  In case of new registration:

  This is achieved, because both, the user and the HE/AuC contribute to the calculated temporary keys *KT*, *CK* and *IK* by their random seeds *RSu* and *RSh*, respectively. The SN trusts the HE to distribute only fresh keys.

  In case of current registration:

  This is achieved, because both, the user and the HE/AuC contribute to the calculated temporary keys *CK* and *IK* by their random seeds *RSu* and *RSh*, respectively.

- Compatible with mechanisms used for the provision of confidentiality of the user identity on the air interface:

  The transmission of the user identity is not part of the protocol, but the protocol does not prevent the provision of this property.

- Key seed confirmation

  In case of new registration:

  Mutual key seed confirmation is ensured by the verification of the correctness of the received *RES1* and *RES2*.

In case of current registration:

Key seed confirmation of the user to the SN is ensured by the verification of the correctness of the received *RES*.

(1) Are all goals required for an AKA for UMTS achieved?

yes

(2) Limitations of the protocol

No limitations are known.

(3) Implications on the protocol in case of UMTS system failures

(3.1) Compromise of SN authentication information

If the temporary key *KT* stored in the SN became known to an attacker he could perform the following attacks:

- He could impersonate the user towards the network as long as *KT* is valid.
- He could impersonate the network towards the user as long as *KT* is valid.
- He could eavesdrop on sessions in which the derived cipher keys were used as long as KT is valid.

As soon as the legitimate user has performed a successful *New Registration* procedure, the temporary key becomes worthless for an attacker.

(3.2) Resilience to USIM or HE data corruption

All security information stored in the HE/AuC is static for the lifetime of the USIM. In the event of a catastrophic database failure all information for a particular user could be recovered from a single backup.

All security information stored on the USIM is static and read only. If the *IMUI*, *K* or *f1*, *f2*, *f3*, *f4* or *f5* are corrupted then the USIM must be reissued. Since security information does not need to be updated during the lifetime of the USIM the possibility of corruption is reduced.

(3.3) Resilience to breakdown of links between SN and HE

A break-down of the link between SN and HE is only critical, during a *New Registration* procedure. After that, as long as the user stays in this SN no contact to the HE for authentication and key agreement purposes is required.

(4) Protocol efficiency

(4.1) Number of passes (for new registration and for current registration)

In case of new registration:

- On the air interface: Three passes are required for each authentication exchange.
- On the SN/HE interface: Two passes are required for each authentication exchange.

In case of current registration:

- On the air interface: Two passes are required for each authentication exchange.

(4.2) Number of calculations

Cryptographic computations are performed in the USIM, in SN/VLR and in the new registration case also in the HE/AuC.

In case of new registration:

- **At the HE/AuC:** Generation of a random number *RSh*, computation of the temporary authentication key *KT*, computation of the authentication results *RES1* and *XRES2* and if this is not carried out by the SN/VLR computation of the temporary encryption and integrity protection keys *CK* and *IK*.
- **At the SN/VLR:** If this is not carried out by the HE/AuC, computation of the temporary encryption and integrity protection keys *CK* and *IK*.
- **At the USIM:** Generation of a random number *RSu*, computation of the temporary authentication key *KT*, computation of the authentication results *XRES1* and *RES2* and computation of the temporary keys *CK* and *IK*.

In case of current registration:

- **At the SN/VLR:** Generation of a random number *RSn*, computation of the authentication results *RES1* and *XRES2* computation of the temporary encryption and integrity protection keys *CK* and *IK*.
- **At the USIM:** Generation of a random number *RSu*, computation of the authentication result *RES* and computation of the temporary keys *CK* and *IK*.

(4.3) Precomputation of authentication information in the HE

As the random number *RSu* of the user is not known before the start of the protocol run a precomputation of authentication information is not possible.

(4.4) Frequency of SN – HE signalling

Key data sent to the SN in response to a request could potentially be used indefinitely by the SN. However, it is good practice to update *KT* on a regular basis (e.g. expiration of a predefined authentication time interval), thus the SN will need to signal requests for a new *KT* to the HE.

(5) Implications on the mechanism on the UMTS infrastructure

(5.1) Need for an AuC in the SN

The SN requires an AuC for the handling of key data and the computation of the cryptographic parameters.

(6) Compatibility with GSM security architecture

(6.1) Ease of migration from GSM to UMTS

A new AuC is required at the HE. New USIMs must be issued. The SN must be modified and needs an AuC to support the handling of key data and the computation of cryptographic parameters.

(6.2) Ease of roaming between GSM and UMTS networks

UMTS user roaming into GSM:

The SN needs to be provided with triplets. It is ffs how this could be best done. It seems likely that the UMTS-AuC or a gateway have to provide GSM-AuC functionality. The USIM must be able to carry out the GSM protocols.

GSM user roaming into UMTS:

It is not clear what this means. If it means SIM-roaming (plastic roaming) and if UMTS terminals are to accept GSM-SIMs then it is ffs in how far this is to be permitted. If so then the GSM user may obtain only GSM-type service. The ME and the SN, respectively, would have to be adapted in such a way that they can handle GSM triplets, and map them onto UMTS parameters and vice versa. The algorithm for

encryption would have to be able to accept shorter key lengths giving GSM grade security. Integrity of signalling messages could not be provided.

(7) Compatibility with IS-41security architecture

(7.1) Ease of migration from IS-41 to UMTS

A USIM functional module is required in the MS. A new AuC is required at the HE. The SN must be modified for the purposes of the AKA.

(7.2) Ease of roaming between IS-41 and UMTS networks

UMTS user roaming into IS-41:
An IS-41 application would be required on the USIM, similarly in the UMTS AuC.

IS-41 user roaming into UMTS:
A UMTS application would be required in the IS-41 terminal, similarly in the IS-41 AuC.

(8) Need for a standard AKA algorithm

Standardised algorithms are needed for *f3*, *f4* and *f5* respectively.

6.2.3 Concluding remarks on authentication and key agreement mechanisms

We analysed seven different AKA protocols in the above sections regarding their suitability for use in UMTS. Four of them – GSM, DECT, TETRA and IS-41 – are protocols standardised for second generation mobile systems. The analysis showed that they could not be used for UMTS as none of these protocols met the additional requirements for a third generation mobile system which are reflected in the required protocol goals described in subsection 6.2.1.

The three remaining protocols were all specifically developed for third generation mobile systems, and have not been standardised or been in use elsewhere. These three protocols all seem to satisfy the required goals. Two of them will be considered further in USECA as well as in standardisation in 3GPP, the SEQ protocol and the TETRA-3 protocol. The RHUL is not considered any further because it is felt that the additional complexity involved with two tiers of temporary user identities was not justified by the gain of this mechanism.

The decision on the selection between the two remaining protocols will have to take into account questions of compatibility with security architectures of second generation systems, ease of roaming and interoperability with various second generation systems and other emerging third generation systems.

6.3 Integrity protection

6.3.1 Overview

The analysis of the threats to UMTS security yields that there is a requirement to protect the integrity of certain signalling messages. In this section, elements of a mechanism to guarantee the integrity of these signalling messages are described. In the first subsection, the signalling messages are listed for which a requirement for integrity protection has been identified. The next subsection briefly deals with periodic in-call signalling messages introduced to prevent channel hijacking. Following that, the mechanism for integrity protection is described, including a mechanism for replay protection. In the next subsection, it is shown how the integrity mechanism can be used to provide local authentication when the authentication and key agreement procedure is not run.

The description does not represent the final state of the discussion as there are still a number of open issues which have to be dealt with before a final decision on the integrity mechanism can be taken. These open issues include:

- Precisely which signalling messages are to be integrity protected?
- How are these signalling messages embedded in which procedures (call set-up, location update, location registration, other)?
- How should the periodic signalling messages to prevent channel hijacking be defined?
- Where is the integrity mechanism handled on the user and on the network side?
- What function split between USIM and ME is appropriate for handling the integrity mechanism?

A number of integrity mechanisms are discussed below. The resolution of the above open issues will have an impact on the selection of the appropriate mechanism(s).

6.3.2 Integrity protection of critical data elements

The integrity protection which is dealt with in this section is end-to-network (MSC or RNC), not end-to-end.

The critical signalling elements that need integrity protection are listed below.

It is also clear that not all signalling messages need to be integrity protected in their entirety.

It was decided not to provide integrity protection of user data. There are two reasons for this: there seems to be little benefit, and there is serious doubt about its feasibility. Most applications which require integrity protected user data require this protection end-to-end. End-to-end protection is outside the scope of UMTS security standardisation. Regarding feasibility, it has to be borne in mind that the radio interface is more error-prone than fixed lines, and therefore that the probability of a message being corrupted is higher. It would be undesirable if a connection had to be torn down because the integrity of messages could not be verified due to bit errors on the radio path. In particular, this would not be acceptable for error-tolerant applications such as voice.

The following signalling elements have so far been identified as requiring integrity protection.

Signalling elements sent by the MS to the SN:

- The MS capabilities, including authentication mechanism, ciphering and integrity capabilities.
- The security mode accept/reject message.
- The called party number in a mobile originated call.
- Periodic integrity protected in-call signalling messages (see next subsection).
- Detach message (ffs)

Signalling elements sent by the SN to the MS:

- The security mode command, including whether ciphering is enabled or not and the ciphering and integrity algorithm to be used.
- Periodic integrity protected in-call signalling messages (see next subsection).

The list may be not complete.

6.3.3 Periodic integrity protected in-call signalling messages

The new threat by “false networks” also conjures up the possibility of the hijacking of channels. Hijacking means that a connection set up by a bona fide user is taken over by an intruder who continues to use the connection in place of the user. Mutual authentication is not a countermeasure as the take over could take place after the mutual authentication was complete. Ciphering is an effective countermeasure, but ciphering is not applied in all networks.

Therefore, it was proposed to use periodic integrity protected in-call signalling messages to prevent the hijacking of channels. It should be noted, however, that this measure is only effective for circuit-switched connections, not for packet-switched services.

The idea is that the network (termination of integrity mechanism) periodically issues an integrity protected message which is answered by an integrity protected message sent by the user. It has not been determined, however, over which channel these messages should be carried, nor what their format should be.

It is open at this point whether protection against hijacking of channels by periodic integrity protected in-call signalling messages is to be a mandatory or an optional feature.

6.3.4 Integrity protection method

Definitions:

In order to be able to describe in some more detail what the feature "integrity" means we first give a few definitions (based on [MeOoVa, sec. 9.6]).

Data integrity is the property whereby data has not been altered in an unauthorised manner since the time it was created, transmitted, or stored by an authorised source.

Data origin authentication is a type of authentication whereby a party is corroborated as the (original) source of specified data created at some (typically unspecified) time in the past.

The term *message authentication* is used analogously with data origin authentication.

Assurances are typically required both that data actually came from its reputed source and that its state is unaltered. These issues cannot be separated. Integrity mechanisms thus implicitly provide data origin authentication, and vice versa.

Additional guarantees may be required:

Protection against replay is certainly required for critical messages in UMTS, such as the security mode command.

It is ffs whether also *protection against the reordering of messages* is required.

Protection against deletion of messages is not a goal for UMTS. Corresponding measures would make the handling of incomplete protocol runs (messages sent, but not received due to errors) more difficult.

Integrity protection as discussed here is meant to protect against malicious threats, not just channel errors.

Mechanisms:

The most commonly used method to provide data integrity is the use of *message authentication codes* (MACs), see e.g. [MeOoVa, Def. 9.7]). The use of MACs is also proposed for UMTS. The key to be used with the MAC is the integrity key IK derived in the authentication and key agreement protocol.

Replay protection can be provided by the use of sequence numbers in the following way (see e.g. [DaPr, sec. 5.7]):

For exchanging integrity protected messages with replay protection, the MS and the SN maintain two counters each for two sequence numbers, n_{MS} for messages M_{MS} sent from MS to SN and n_{SN} for messages sent from the SN to the MS. The two counters are initialised to zero on both sides whenever a new integrity key is agreed between the MS and the SN. When the MS wants to send a message M_{MS} the MS increases the value of the counter corresponding to n_{MS} by one, computes a MAC over M_{MS} and n_{MS} and sends the concatenation of M_{MS} and n_{MS} to the SN, as shown in the following Figure 6.3.4-1. When M_{MS} has to be resent due to a protocol failure, the MS again updates n_{MS} before computing the MAC. (This is because it is assumed that the sender cannot know whether the receiver received the previous message.) The SN accepts the message only if the MAC is correct and if the received sequence number n_{MS} is higher than the corresponding current counter value. If this is the case, the SN updates the counter value to n_{MS}. The corresponding counters at MS and SN for the sequence numbers n_{SN} are managed in an analogous fashion. n_{MS} and n_{SN} are uncorrelated. Here, DOWN and UP are direction indicators to prevent reflection of a message with its MAC.

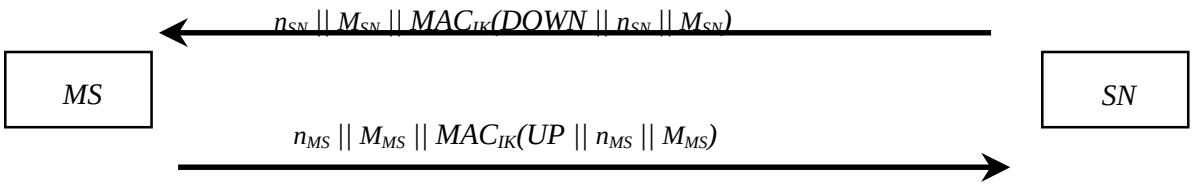

Figure 6.3.4-1: General integrity with replay protection

This mechanism also provides *protection against the reordering of messages.*

It does not provide protection against intercepting an arbitrary number of response messages from the MS and replaying them later as a response to quite different request, hence it not suitable for local user-to-network authentication. (One could also say instead that the mechanism does not provide any linking of the request and the response messages.) Local authentication will be dealt with in the next section.

> For illustration of what could happen: Let us assume for the sake of the argument that an exchange of signalling messages between the MS and the SN consists of only one message from the SN to the MS and another message going back. Then an attacker can let the message from the SN go through to the MS, but intercept the answer from the MS. This will cause the n_{SN}–counter at the SN to increase by one while the n_{MS}–counter at the SN for messages received from the MS will remain unchanged. In this way, the attacker can collect arbitrarily many answers from the MS and replay them later to the SN. (Hence the use of the term "preplay".) They will be accepted by the SN as long as the MS does not get a new message through to the SN in the meantime. (The SN cannot know whether the messages previously sent did not get through to the MS or whether the answer did not get through.)

Messages sent only once during the lifetime of the integrity key (e.g. MS capabilities) do not need replay protection by sequence numbers, but providing this protection for uniform handling of messages does not harm.

Cautionary note: In general, it is advisable that a length indicator be prepended to the message before a MAC is computed. Note, however, that the initial parts of two messages are always different due to the use of sequence numbers so that a class of well-known attacks on MACs does not work here. If the fields M_{MS} and M_{SN} contain parameters indicating the type of signalling message and if the integrity protected signalling messages of one type are either of fixed length or contain a prepended length indicator then prepending a length indicator certainly is not necessary. One may also consider prepending of the length as part of the MAC algorithm, cf. [ISO 9797-1]. The general case is ffs.

6.3.5 Providing authentication by using the integrity mechanism (local authentication)

The integrity mechanism may be used to provide authentication between MS and SN without invoking the AKA procedure. Such a local authentication mechanism is especially useful when no encryption is provided in the network to ensure continued authentication of the communication parties. It may then be used in conjunction with location updates, attach procedures, service requests and responses to paging requests.

In UMTS there may be two types of signalling messages sent by the MS, those which come in pairs of request from the SN and response from the MS, and those which do not (single messages).

Let us go through the list of messages in Figure 6.3.5-1: The security mode accept/reject message responds to the security mode command. The periodic integrity protected in-call signalling messages also come in request/response pairs. The called party number in a mobile originated call does not seem to fit in this category, however, it is ffs whether it can be combined with a security mode accept/reject message in a request for a mobile originated call. The MS capabilities message does not need replay protection, but

the MAC on the MS capabilities message can only be computed when the security mode is known, so the MAC could be combined with an security mode accept/reject message

A message sent by the MS which does not respond to a message from the network is the detach message. The need to integrity protect it is ffs. If it is to be protected it is ffs whether it needs replay protection or whether the AKA procedure should be run in conjunction with attach.

It is thus ffs whether there are signalling messages in UMTS which do not respond to a message from the network.

For request / response pairs of signalling messages M_{req} / M_{resp} , integrity with replay protection can be combined with local authentication in the following way:

The sequence number used by the MS for replay protection of a response messages is now to be equal to the sequence number n_{SN} used on the corresponding request from the network. In this way, a challenge–response authentication mechanism is provided, as shown in figure Figure 6.3.5-1below. The MS sends the response only after it verified the MAC and sequence number on the request from the network. Therefore, the challenge only has to be unique, not unpredictable, i.e. a sequence number can be used as a challenge.

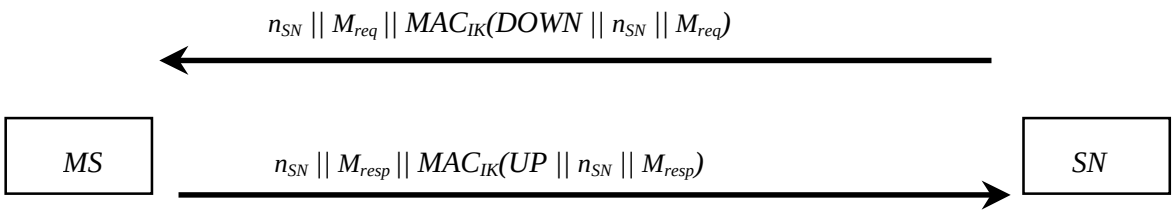

Figure 6.3.5-1: Local authentication by integrity

The local authentication protocol has the following properties:

It provides entity authentication of the user to the SN (in the same sense as for the SEQ AKA protocol , cf. section 6.2.2.6). It does not provide entity authentication of the SN to the user, as the user cannot verifiably link the identity of SN to the key used. However, it provides assurance to the user that the origin of the M_{req} message is in possession of the integrity key *IK*, hence is authorised by the HE to provide service to the user. It also provides assurance to the user that the M_{req} message is not a replay from a previous protocol run. It also provides data origin authentication and replay protection for the M_{resp} message to the SN/RNC.

If all integrity protected signalling messages are part of a request/response pair then the mechanism also providing local authentication described in this subsection is the only mechanism required.

If this is not the case then those signalling messages not being part of a request/response pair will have to be replay protected by the general method described in the preceding subsection 6.3.4. This would create two unrelated mechanisms for replay protection for the two sets of messages of different type, and the sequence numbers used in these would be unrelated. In particular, nothing could be inferred by a recipient about the relative order in which two messages of different type were sent.

6.3.6 Protection against the reordering of messages

In the general case, when both types of signalling messages, as described in the preceding subsection, do occur and when also protection against the reordering of messages is required over all integrity protected signalling messages then the two methods described in the two preceding subsections 6.3.4 and 6.3.5 may be combined in the following way: The two types of signalling messages – the ones that come in pairs and the ones that do not – are handled differently. There is one sequence number n_{MS} which is incremented by the MS when the MS sends a message of either type, and there is a sequence number n_{SN} which is incremented by the SN when the SN sends a message.

For messages being part of a request/response pair, the mechanism is as shown in the

Figure 6.3.6-1 below:

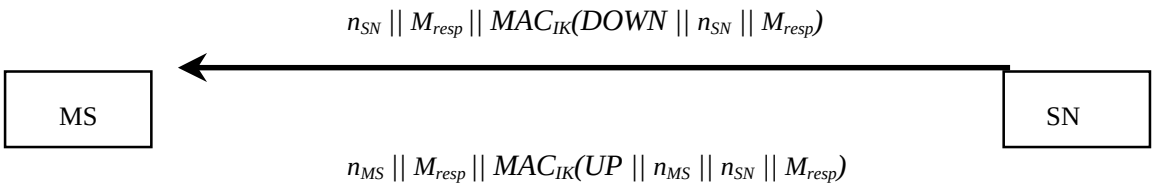

Figure 6.3.6-1: **Integrity with protection against replay and reordering for paired messages**

For single messages from the MS, the mechanism is as described in the next

Figure 6.3.6-2:

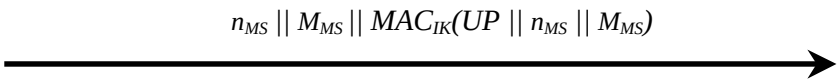

Figure 6.3.6-2: **Integrity with protection against replay and reordering for single messages**

Protection against reordering over all integrity protected signalling messages is provided by using the same sequence number n_{MS} in both mechanisms.

Conclusion: We discussed several mechanisms which may be used for integrity protection of signalling messages in UMTS. The mechanism discussed in this subsection appears to be applicable quite generally, however, it is possible that this generality is not needed. Before a decision on a mechanism can be taken the requirements on integrity protection (which messages in which procedures, which type of protection) have to be known in more detail.

6.4 User traffic confidentiality

"User traffic" will be taken to mean both user traffic, as in voice or data traffic, and also user-related signalling, such as location information, temporary identities, and certain call set up information such as the required b-number for an MO call.

User traffic confidentiality is the security feature designed to meet the requirement for privacy of user traffic and user-related signalling. [ETSI 33.21] contains the following relevant requirements:

R2a It shall be possible to protect the confidentiality of user traffic, particularly on radio interfaces, including protection against eavesdropping from relay nodes.

R2b It shall be possible to protect the confidentiality of user identity data, particularly on radio interfaces, including protection against eavesdropping from relay nodes.

R2c It shall be possible to protect the confidentiality of location data about users, particularly on radio interfaces, including protection against eavesdropping from relay nodes.

This is to be met by the features “confidentiality of user data” and “confidentiality of signalling data” described in section 5.2.3 and 5.2.4 in [ETSI 33.22].

6.4.1 Overview

A brief description of GSM (including GPRS) ciphering is given in section 6.4.2, along with the likely ways that ciphering for UMTS will differ from that of GSM.

In mobile communications, confidentiality of user traffic across radio interfaces is traditionally performed using a stream cipher, as this prevents error propagation. However, block ciphers can be used in a variety of modes, with a variety of advantages and disadvantages, depending on the residual bit error rate (BER, the residual BER is the BER after error correction has been performed). This issue is discussed in section 6.4.3.

The W-CDMA radio interface used in UMTS will allow for variable data rates on one channel. The impact of this on ciphering requirements is dealt with in section 6.4.4.

Stream ciphers and block ciphers used in stream cipher mode must be synchronised between the two endpoints of the communication. In GSM circuit switched calls, the physical layer frame number was used for synchronisation as ciphering was performed at the physical layer. For GPRS, ciphering is performed at the LLC layer within layer 2, and an LLC frame number is used as synchronisation. The decision as to which layer to use in UMTS has not yet been taken. Whichever layer must is used, the variable must have sufficient range to prevent certain types of cryptanalysis. This issue, and other issues of synchronisation are examined in section 6.4.5.

6.4.2 Ciphering in GSM and likely differences in UMTS

A full description of GSM ciphering can be obtained in [GSM 03.20]. A brief summary is given below.

Ciphering takes place on the physical layer between the mobile and BTS only. Physical layer frames of 114 bits of plaintext are individually XOR’ed with 114 bit cipher stream string at the encrypting endpoint to produce the ciphertext that is transmitted across the radio interface. At the receiving endpoint, the ciphertext is XOR’ed with the same cipher stream string to produce, bar any residual bit errors, the original plaintext.

A 64 bit key (see section 6.4.6) derived from authentication of an MS (for the call in progress or a previous call), is used, along with the physical layer frame number, as inputs to a stream cipher to produce the cipher stream for each frame.

The algorithm used to produce the cipher stream is termed “A5” within the GSM specifications. There are, in fact, two version of A5. A5/1 was produced first. Later, when it was apparent that GSM was going to be used globally and not just in Europe, a second algorithm, cryptically called A5/2, was developed for countries to which export of A5/1 was politically undesirable.

Ciphering in UMTS will be very different to that in GSM. The differences will/may be:

Variable Data Frame length:	The UMTS radio interface, being W-CDMA allows for a variable data rate on a channel (from 20 to 640 bits per frame, increasing in powers of 2). This is achieved by reducing the spreading factor as the data rate increases. Thus the cipher stream generator must be able to produce a variable amount of cipher stream each frame and not a fixed amount, as in GSM.
Extent of ciphering	Ciphering in UMTS will extend further towards (and possibly into) the core network than in GSM.
Layer for ciphering	The extent of the ciphering will dictate which layer the ciphering will be performed on. It is likely not to be the physical layer as is used in GSM.
Key length	It is likely that the ciphering key will be longer than the 64 bits in GSM, so as to defeat the exhaustive key search attacks that are possible with present-day computing power.

Ciphering method	A block cipher may be used to produce cipher text instead of the stream cipher used in GSM.
Combined voice/data ciphering	In GSM, the architecture for GPRS ciphering is separate from the circuit switched ciphering. The intention in UMTS is to have a combined circuit switched/packet switched ciphering architecture.

6.4.3 Stream or block ciphering?

In GSM, bitwise stream ciphers are used to generate a keystream which is XOR'ed with the plaintext to give the cipher text.

There are problems with the use of stream ciphers:

a) The use of a bitwise stream cipher means that the bit positions of data within the cipher text are the same as in the plaintext. Therefore, if the bit position of a particular data bit is known, it can be toggled by intruders. Checksum bits may also have to be changed to make the manipulation undetectable. Such attacks are only possible where the data bit positions are predictable. However, as UMTS is designed as much for data services as for voice, such predictability is likely for a significant amount of ciphered material. In addition, there will be sensitive signalling messages for both voice and data services that will have predictable formats

   However, many of these sensitive messages should also have integrity protection to defeat manipulation in the absence of ciphering. A non-linear checksum (where the attacker must know the content of all the material covered by the checksum to know which bits in the checksum to toggle) will also provide adequate protection against toggling, unless messages have no variability at all. (Alternatively, frames can be permuted or contain a random amount of data at the start of each frame. However, the former method has a processing overhead and the latter, a bandwidth overhead.)

b) Stream ciphers are almost exclusively used for radio communications. Much of this radio communication use is by the military who keep their algorithms secret. In GSM, the algorithms are also kept secret. There are therefore very few publicly known stream ciphers. It is therefore likely that the use of a stream cipher for UMTS will required a newly designed algorithm, which is obviously slower than using an existing, proven algorithm.

c) There has been less public work in general, both in design and in cryptanalysis, of stream ciphers than on block ciphers. The chances of a badly designed algorithm or a sudden advance in cryptanalytic capabilities may therefore be higher in using a stream cipher than using a block cipher.

However, stream ciphers have the clear advantage over most modes of block cipher use of zero error propagation. That is, if there is one residual bit error in the ciphertext, this will only cause one error in the decrypted plaintext. With block ciphers in ECB, CBC and CFB mode, a residual error in the ciphertext may result in several errors in the plaintext (for CBC mode, a single residual bit error will propagate through the rest of the communication).

Such error propagation is not present if a block cipher in OFB mode is used. This "block cipher" is effectively a stream cipher as the block cipher is used to generate the keystream which is bitwise XOR'ed with the plaintext as for an ordinary stream cipher. Problem (a) above would therefore remain (and would remain with any solution with zero error propagation, by definition). However, problems (b) and (c) would be removed. With regard to (b) and (c), which relate to the lack of public design and cryptanalytic work on stream ciphers, the AES winner could be used in OFB mode for UMTS, and an internationally recognised cipher thereby used. Though the AES winner is to be announced sometime in 2000, and so could not be part of the first release of the UMTS specifications (which must be complete by the end of 1999), this might not have too much effect, as the algorithm can be abstracted from much of the rest of the design of UMTS systems and components.

"Traditional" stream ciphers tend to require less processing power and memory space (for lookup tables) than block ciphers. With advances in chip design and capability, this may not be a crucial issue if ciphering is performed on the terminal (which is assumed).

6.4.4 Plaintext block length

The UMTS radio interface, being W-CDMA, allows for a variable data rate on a channel (from 20 to 640 bits per frame, increasing in powers of 2). This is achieved by reducing the spreading factor as the data rate increases. Thus the keystream generator must be able to produce a variable amount of keystream each frame and not a fixed amount (two 114 bit blocks), as in GSM.

The stream cipher will therefore be more complex, and have at least one more input (length of keystream required) than in GSM.

Attention may also have to be paid to cryptanalytic issues. Stream ciphers which are formed by combining the outputs of a number of shift register sequences and are therefore subject to correlation attacks, will be increasingly subject to such attacks as the length of keystream produced increases.

The stream cipher will also have to work considerably faster than in GSM. In GSM, 428 bits (it is believed) are produced in 4.62ms (the length of a frame, though the text to be ciphered is produced in 0.577ms, the duration of a timeslot within the 8 timeslots in a frame). For UMTS, up to 1280 bits must be produced in 0.625ms. 0.625ms is the duration of a frame in UMTS, but as a mobile in FDD mode will broadcast on all timeslots in a frame (that is, the channel is not time multiplexed among more than one mobile), keystream text must be produced every timeslot and not every frame. UMTS keystream generators will therefore have to work up to approximately 2244 times faster (2048 bits per ms as oppose to 92.6 bits per ms) in UMTS than in GSM.

The above discussion assumes that data is ciphered before it is spread. If the spread data were ciphered, i.e. all 2560 chips per timeslot, this would remove the need for a variable rate keystream generator. However, it is believed that the resulting redundancy in the plaintext (much of the plaintext, especially for low data rates, would be redundant and merely code-dependant repetition of the data) would provide a significant platform from which to mount cryptanalytic attacks on the cipher. The keystream generator would also be running faster than required in most cases, and producing more keystream than is required, possibly providing a further weakening of the cipher.

6.4.5 Synchronisation

We assume that, for reasons of error propagation, a stream cipher or a block cipher in OFB mode will be used. Synchronisation of ciphering will therefore be required. Synchronisation here does not mean simple time alignment of bits, but that for a particular block of data, the setup of the keystream generator must be the same at both ends or the plaintext will not be retrievable.

In GSM this is achieved by using the physical layer frame number as an input for the generation of keystream for each frame as well as the cipher key.

A similar method of providing synchronisation and diversity in the input to the keystream generator will be required for UMTS. The layer on which ciphering is performed must therefore be able to provide such a frame number (in reality, frame and timeslot number, but the combination of frame number and timeslot number within frame (range 1...16) may suffice.

The range of the frame number produced is important. If the frame number loops around its range and begins again during the use of a single cipher key (this use may be within one call or spread over several calls, where the cipher key is re-used), the cryptanalytic resistance of the cipher falls dramatically. This phenomenon is loosely referred to as “the cipher stream repeating”. In GSM, an attempt was made to avoid this problem by defining the hyperframe. For radio resource reasons, the maximum aggregate of frames was the superframe, of 1326 (=26*51) frames. A maximum frame number of 1326 would have meant that the cipher stream would have begun to repeat after 1326*4.62ms = 6.13s of call – a disaster in cryptanalytic terms. The hyperframe of 2048 superframes, giving a maximum frame number (“FN_MAX”) of 2715647 (numbering from 0 to FN_MAX-1) was therefore introduced. With the hyperframe, the keystream repeats after approximately 3 hours 29 minutes. Very few calls indeed will cause such a keystream to repeat.

In GPRS a frame number at the LLC layer is used for synchronisation, the range of this is 2^{32}. This will cause a keystream repeat after over 2 years! Though this may seem overkill, it would not be proper for UMTS to be seen to be less secure than GPRS. A frame number with a range of at least 2^{32} should

therefore be used for UMTS. Prior to the establishment of UMTS, SMG2 UMTS Layer 2/3 followed SMG10 WPC guidance and indicated that a frame number of 2^{32} would be provided for UMTS.

6.4.6 Length of ciphering key

The length of the ciphering key cannot properly be discussed in isolation from discussions on the strength of the algorithm, which are outside of the scope of this document.

7 Core network security

This section considers issues surrounding the choice of mechanisms for securing communications within the UMTS core network (CN).

The security of signalling information only will be considered within this section. The security of user traffic within the core network will not be considered as within the SMG10 and 3GPP Security, it is signalling information that has priority. The security of user traffic within the CN network is generally thought of in terms of end to end security, rather than separate CN security. At this stage, it is not clear that there will be end to end security solutions other than at the application layer.

The security of user traffic CN security is seen as desirable to secure the sensitive user related security information transmitted within and between network operators (e.g. ciphering keys and authentication data) and to secure the use of the powerful commands (e.g. “HLR Reset”) that exist within CN signalling systems.

7.1 Overview

The signalling system used within CN is primarily SS7 for GSM and other 2G systems. It is likely that, initially, SS7 will also be predominantly used within UMTS CN but migration to IP based systems is likely, especially for data traffic.

SS7 was defined in an age when telecommunications was fixed only and was restricted to a small group of national operators, communicating over their own transport networks. In such an environment, the need for security services within the signalling system does not seem great. SS7 contains no security services

Telecommunications is now a highly deregulated industry. There are many operators, with the majority of them being in private hands, and some of them very small, serving only niche markets. Most of these operators do not have their own transport networks but use those of dedicated network companies, of which there are also many. In addition, there are also “airtime resellers” who lease spare capacity on transport networks and this is also used to transport signalling and traffic on behalf of operators. The deregulation of the industry is such that The Economist has made the conjecture that a “spot market” for telecommunication transport services is not far off.

7.2 Threats

In such a deregulated environment, where a signalling message may pass through the facilities of several organisations, the number of points at which the message can be accessed are many. As such points, messages are vulnerable to eavesdropping or manipulation by corrupt operatives of these organisations. In addition, intruders outside participating organisations may access messages at the surface access points of the network.

The threats can be divided into the following groups:

(a) Eavesdropping on secret information, with a view to either posing as the user and so obtaining services fraudulently, or obtaining the cipher key for a user’s traffic and so being able to eavesdrop on the user’s traffic. An example of secret information that could be so used would be GSM authentication triplets.

(b) Unauthorised requests for such information

(c) Serious denial of service attacks by the unauthorised generation and transmission or manipulation of powerful messages such as “HLR Reset” and “VLR Reset”.

(d) Lower level denial of service/disruption attacks by unauthorised generation and transmission or manipulation of user and non-user related signalling traffic.

Though these threats are serious, it should be noted that it would be much easier to enact them in a “random” way as oppose to one targeted at a particular user. The unpredictability of the route to be taken by signalling traffic would make it difficult to target a particular user.

7.3 Features

As this document is concerned with security mechanisms, and not requirements and features, we will proceed from threats to the required security features and miss out the intermediate stage of security requirements.

The required features are given for each of the four threat types given in the previous section.

(a) Confidentiality of the secret information.

(b) Data origin authentication of the requests for secret information

(c) Data origin authentication and integrity of the signalling messages

(d) Data origin authentication and integrity of the signalling messages

If integrity is added to the (b) (a reasonable step, there is the threat of requests for secret information being manipulated to increase the number of triplets (for instance) requested and "stealing" the surplus number in the response), the features required can be divided into two groups:

Confidentiality for those messages containing secret information

Data origin authentication and **integrity** for messages that can be manipulated or generated to disrupt or deny service.

7.4 Mechanisms

7.4.1 Criteria for evaluation of security mechanisms for CN security

All the criteria for selection of security mechanisms in [ETSI 33.23] apply to the selection process for CN security. However, attention may be drawn to the following criteria which apply in particular to CN security:

(a) **Interoperability**. As one network may wish to communicate with any other, the security mechanisms must be fully interoperable.

(b) **Exportability**. As the signalling will often be between two countries, it is undesirable (from a security management point of view) to have a solution that is acceptable in one country but not in another (as is the case with GSM traffic ciphering in having an algorithm, A5/1, that can only be used in certain countries). In addition, a fully exportable mechanism will ensure that a HE can securely send information for its subscribers wherever they are roaming in the world.

(c) **Simplicity**. CN security may be seen many operators as an overhead and an unnecessary inconvenience. Any solution must therefore represent as minimal an overhead as possible.

(d) **Trust in mechanism**. As the mechanism is to be used globally, and not within a certain area with its own cryptographic traditions and opinions, such as Europe or the United States, a mature, tested and globally trusted mechanism should be used.

The criterion of the minimisation of bilateral agreements warrants some discussion. In principle, this is sensible criterion, as the agreement of a bilateral agreement is an involved process, and the minimisation of it is a worthwhile goal. However, bar automatic establishment of roaming relations, operators already have a bilateral agreement with those operators with which they have a roaming agreement. A separate bilateral agreement for security purposes is not required. This has impact for the choice of key management technology. Where two operators have no previous security relationship, it would appear that a public key based solution is appropriate (though the two operators would have to agree on standards for public key certification). However, if the two operators already have a roaming agreement, which is a commercial, and therefore involved process to agree, it would not significantly increase the complexity of this agreement to include in it the exchange of a secret key.

7.4.2 Mechanisms for Authentication, Integrity and Confidentiality

Bearing in mind criteria (a), (b) and (d) given in section 7.4.1, it is clear that a block cipher should be chosen for confidentiality provision rather than a stream cipher. Block ciphers have been the subject of much more publicity, study and standardisation than stream ciphers, which traditionally, have been used mainly for military purposes. The only advantage stream ciphers have is their lack of error propagation, which is not important over the relatively error free fixed signalling links.

Clearly, the use of encryption also provides integrity and data origin authentication to the granularity of the key distribution (i.e. if each operator only has one key, the origin can only be authenticated to a particular operator. If, however, individual nodes have their own key, origin can be authenticated down to this level). However, encryption is a sensitive matter, and therefore, in order to minimise the amount of material encrypted, it might be desirable to provide integrity and data origin authentication other than using encryption.

A first question to answer in providing integrity and data origin authentication without confidentiality is whether to use public or secret key techniques to do this. This question will be answered by a swift recourse to criterion (c) above, simplicity. If we assume that confidentiality is provided using secret key techniques (leaving aside key management) at this stage, then the pair of operators communicating already have a shared secret key. Therefore, in the interests of simplicity, we assume that this key can be re-used to provide integrity protection and data origin authentication. Therefore secret key techniques would provide integrity protection and data origin authentication without an increase in the key management task beyond that required for confidentiality.

Assuming secret key techniques are used to provide integrity protection and data origin authentication using secret key techniques, there is the question of whether to use a block cipher (see [ISO 9797-1]) or a hash function (see [ISO 9797-2]).

It would seem from a simplicity point of view, that re-using the block cipher is the best option, as it minimises the number of algorithms that must be standardised and agreed. The requirements specification for the PNO algorithm [PNO], devised by ETSI SAGE specifically for use by public network operators, suggests that the algorithm is used to provide integrity protection and data origin authentication as well as confidentiality. However, there may be performance or other considerations in using hash functions (perhaps related to the shortness of the messages to be protected) instead.

7.4.3 Algorithms for Authentication, Integrity and Confidentiality

Specific algorithms will not be recommended here. Instead some general comments will be made.

The criteria in section 7.4.1 are important again. A well established algorithm would be:

- Known and trusted by operators and industry in general
- Known and trusted by the “cryptographic community”
- Already be available in commercial software implementations

Were it stronger, DES would fulfil the criteria of 7.4.1. However, its key length is too short to have global confidence, even though the protection it provides (56 bit key search is still a very significant task) might well suffice for many uses within CN security.

Triple DES therefore seems a logical substitute for DES, in the absence of the official replacement for DES, the Advanced Encryption Standard, AES. Alternatively, depending on when the AES winner is announced, it could be used.

8 Conclusions and outlook

Commercial UMTS Phase 1 services are expected to commence in Europe by 2002. Accordingly, standardisation work for phase 1 is progressing very fast and has to be completed by the end of this year. USECA has made major contributions to the ongoing standardisation work for UMTS phase 1 in the relevant standardisation bodies within ETSI and 3GPP.

The discussions and analyses of security mechanisms which formed the basis for standards contributions by USECA partners were documented here for later reference. As we report on ongoing work not all parts of the security architecture are equally well developed.

The focus of the report is on access network security. Four different topics are discussed: Mechanisms providing user identity and location confidentiality, authentication and key agreement protocols, integrity protection of specific signalling messages and user traffic confidentiality. Concerning user identity and location confidentiality the proposed mechanisms are fairly stable. Concerning authentication and key agreement protocols, the choice could be narrowed down to just two candidates. Important issues for user traffic confidentiality such as synchronisation could also be resolved. Mechanisms for integrity are still subject of intensive discussion.

For the provision of user identity and location confidentiality a recommendation is given for the mechanism which seems to be the most appropriate one, taking into account pre-decision made in the standardisation bodies. It is based on the mechanism used in GSM today and defines an add-on mechanism in specific (failure) situations where today GSM transmits the user identity in the clear. A symmetric group key is introduced for distinct groups of users to provide identity confidentiality even in this cases.

From the different authentication and key agreement (AKA) mechanisms discussed and evaluated two protocols, which were also discussed in the standardisation bodies as candidates for UMTS systems, are found to be the most appropriate ones for UMTS. One of these protocols, the so-called SEQ-protocol, is likely to be more compatible with the GSM security architecture. It is expected that it may ease migration from existing GSM systems to UMTS but also to simplify roaming between both systems. The other protocols, the so-called TETRA-3 protocol, is optimised with respect to the frequency the HE has to be contacted for authentication purposes by an SN in which the user is roaming.

Discussion on mechanisms for core network security is still in its beginnings in the standards body, it has not been in the focus of USECA work so far.

The future work of WP2.2 will (besides the discussion and evaluation of security mechanisms and protocols made here) help prepare and document the final decisions on security mechanisms for UMTS phase 1 taken in the standardisation bodies. In particular, the mechanisms for core network security will be elaborated in more detail.

For later UMTS phases it may be desirable to have a public key infrastructure in order to provide additional services (e.g. incontestable charging) or to simplify the key management among a growing number of service providers and network providers world-wide. The final report is aimed to contain a discussion of the advantages and consequences of utilising public key mechanisms in later UMTS phases.

9 Confidential Annex

A. Threat model

A.1 Introduction

Enhancements and modifications in the security of mobile communications systems, when migrating from second generation systems to third generation systems, have either one of the following justifications:

1) A new type of service is offered which requires new security features. The main difference between 2G and 3G mobile communications systems is that data applications will be much more important in 3G systems than in 2G systems.

2) The same type of service is offered in a different way. This includes all modification due to the modified system architecture, e.g., the new ciphering algorithm which is required because of the changed radio interface.

3) The security requirements of users, network operators and regulators have changed. This category is empty as the ambitious objectives that were included in the 3G security requirements for a long time have been removed as the scope of 3G security was downgraded in view of the modified view on the 3G system in general.

And the category that is the subject of this chapter and forms the major justification of the mechanisms in the subsequent chapters:

4) New attacks on the 2G security architecture that were not possible before. This includes attacks that were not possible before but are or are perceived to be possible now or very soon, because intruders have more computational capacities, new equipment has become available to attackers, and the physical security of certain network elements is questioned.

The enhancements are justified almost exclusively by these threats. The following evolutions are key to our analysis of the 3G security architecture:

- *The availability of false base stations.* The current GSM security architecture can be said to provide good subscriber authentication, user identity confidentiality and user data and signalling confidentiality on a radio access link subject to passive attacks (see A.2). While GSM security is performing well, it is believed that UMTS security should also protect against so-called active attacks on the radio access link (see A.2) as the equipment to perform such attacks has become available on the market.

- *Compromise of authentication data.* Although the current wireline links between network entities are considered to be secure, a concern exists whether this will remain to be the case in the future, especially in view of a possible migration to routed signalling networks shared with other applications. Currently the GSM security architecture applies no protection (data integrity or encryption) of authentication vectors (triplets) transmitted between network elements. A workgroup within SMG10 has recently started to work on the “network security” work item. The combination of a false base station and the possible compromise of an authentication vector is detrimental to the GSM security architecture, as the compromised authentication data may be forced upon the user by a false base station an indefinite number of times.

- *Flexibility of authentication mechanism.* A further design option for the 3G security architecture is to prepare for future evolution. This is reflected by the inclusion of the negotiation of the ciphering and integrity algorithm, and of the negotiation of the authentication mechanism. This then also allows the possible future introduction of public-key authentication schemes and to facilitate interoperability with GSM equipment and UIMs from other IMT-2000 family members.

A.2 Capabilities of the attacker

In order to perform the attacks the intruder has to possess one or more of the following capabilities:

C1. **Eavesdropping.** This is the capability that the intruder eavesdrops signalling and data connections associated to other users. The required equipment is a modified MS.

C2. **Impersonation of a user.** This is the capability whereby the intruder sends signalling and/or user data to the network, in an attempt to make the network believe they originate from the target user. The required equipment is again a modified MS.

C3. **Impersonation of the network.** This is the capability whereby the intruder sends signalling and/or user data to the target user, in an attempt to make the target user believe they originate from a genuine network. The required equipment is a modified BS.

C4. **Man-in-the-middle.** This is the capability whereby the intruder puts itself in between the target user and a genuine network and has the ability to eavesdrop, modify, delete, re-order, replay, and spoof signalling and user data messages exchanged between the two parties. The required equipment is a modified BS in conjunction with a modified MS.

The first capability is the easiest to achieve, the following capabilities are gradually more complex and require more investment. Therefore, in general, an intruder having a certain capability is assumed also to have the capabilities of the list with a lower number.

The first two capabilities were acknowledged also in the design of GSM security and the current GSM security architecture can be said to provide good user authentication, identity confidentiality and user data and signalling confidentiality on a radio access link subject to these attacks requiring these capabilities. 3G security however should thwart all four types of attacks.

In addition to these four, there is another element which may be regarded as a fifth capability that enlarges the possibilities of intruders when combined with each of the previous capabilities.

C5. **Compromised authentication data.** The intruder possesses authentication and ciphering data.

A.3 Attacks

The attacks are sorted by objectives first.

A.3.1 Denial of service

We distinguish between the following denial of service attacks:

T1. **User de-registration request spoofing.** An attack that requires a false MS and exploits the weakness that the network cannot authenticate the messages it receives over the radio interface. The intruder spoofs a de-registration request (IMSI detach) to the network. The network de-registers the user from the visited location area and instructs the HLR to do the same. The user is subsequently unreachable for mobile terminated services.

T2. **Location update request spoofing.** An attack that requires a false MS and exploits the weakness that the network cannot authenticate the messages it receives over the radio interface. Instead of the de-registration request, the user spoofs a location update request in a different location area from the one in which the user is roaming. The network registers in the new location area and the target user will be paged in that new area. The user is subsequently unreachable for mobile terminated services.

T3. **Camping on a false BS.** An attack that requires a false BS and exploits the weakness that a user can be enticed to camp on a false base station. Once the target user camps on the radio channels of a false base station, the target user is out of reach of the paging signals of the serving network in which he is registered. This is surely the case for a stand-alone false BS.

T4. **Camping on a false BS/MS.** An attack that requires a false BS/MS and exploits the weakness that a user can be enticed to camp on a false base station. A false BS/MS can act as a repeater for some time and some requests in between the network and the target user, but subsequently modify or ignore certain service requests and/or paging messages related to the target user.

A.3.2 Identity catching

We identify the following types of attacks against the user identity confidentiality:

T5. **Passive identity catching.** An passive attack that exploits the weakness that the network sometimes requests the user to send its identity in cleartext, e.g. upon first registration. An eavesdropper may identify a user related to a service provided over the radio access link, by eavesdropping on signalling channels exchanged between target user and network.

T6. **Active identity catching.** An active attack that requires a false BS and exploits the weakness that the network may request the MS to send its permanent user identity in cleartext. An intruder entices the target user to camp on its false BS subsequently requests the target user to send its permanent user identity in cleartext. The attack may be generalised to active attacks that entice the user to send anything that allows the intruder to derive information on the user identity or to link several calls to the same user, as the latter might help to identify the user.

A.3.3 Eavesdropping on the called party number

We identify the following attacks with the objective to eavesdrop certain signalling elements:

T7. **Called party number catching by suppressing encryption between the target user and the intruder.** An attack that requires a false BS and that exploits the weakness that the MS cannot authenticate messages received over the radio interface. The target user is enticed to camps on the false BS. When he attempts to make a call, the intruder does not enable encryption and is able to capture the called party number sent in cleartext by the target user. After the called party number is obtained, the call set-up procedure is abandoned. This attack does not require the ability for the intruder to set-up user data channels.

T8. **Called party number catching by suppression of encryption between the target user and the true network.** An attack that requires a false BS/MS and that exploits the weakness that the network cannot authenticate messages received over the radio interface. The target user is enticed to camp on the false BS/MS. At call set-up the false BS/MS modifies the ciphering capabilities of the MS to make it appear to the network that a genuine incompatibility exists. The network may establish an unciphered connection. When the user attempts to make a call, the intruder is able to capture the called party number sent in cleartext by the target user. Besides the modification of the ciphering capabilities the intruder acts as a relay station.

T9. **Called party number catching by forcing the use of a compromised cipher key.** An attack that requires a false BS and the possession by the intruder of a compromised authentication triplet and thus exploits the weakness that the user has no control upon the cipher key. The target user is enticed to camp on the false BS/MS. At call set-up the false BS/MS forces the use of a compromised cipher key on the mobile user. When the user attempts to make a call, the intruder is able to capture the called party number which is encrypted with the compromised cipher key.

A.3.4 Impersonation of the network

We identify the following attacks with the objective to impersonate a genuine network. The ultimate aim of such attacks eventually is to eavesdrop on information that the user entrusts genuine networks, or to send the user information that he subsequently believes to originate from a genuine network or user with whom he is connected through that network.

T10. **Impersonation of the network by suppressing encryption between the target user and the intruder.** An attack that requires a false BS and that exploits the weakness that the MS cannot authenticate messages received over the radio interface. The target user is enticed to camp on the false BS. When the intruder or the target user initiate a service, the intruder does not enable encryption by spoofing the cipher mode command. The intruder maintains the call as long as it is required or as long as his attack remains undetected.

T11. **Impersonation of the network by suppressing encryption between the target user and the true network.** An attack that requires a false BS/MS and that exploits the weakness that the network cannot authenticate messages received over the radio interface. The target user is enticed to camp on the false

BS/MS. When a call is set-up the false BS/MS modifies the ciphering capabilities of the MS to make it appear to the network that a genuine incompatibility exists between the network and the mobile station ciphering capabilities. The network may then decide to establish an unciphered connection. After the decision not to cipher has been taken, the intruder cuts the connection with the network and impersonates the network to the target user.

T12. **Impersonation of the network by forcing the use of a compromised cipher key.** An attack that requires a false BS and the possession by the intruder of a compromised authentication triplet and thus exploits the weakness that the user has no control upon the cipher key. The target user is enticed to camp on the false BS/MS. When a call is set-up the false BS/MS forces the use of a compromised cipher key on the mobile user. The intruder maintains the call as long as it is required or as long as his attack remains undetected.

A.3.5 Eavesdropping on user data

We identify the following attacks with the objective to eavesdrop on user data which is actually transmitted through the genuine network to the intended recipient.

T13. **Eavesdropping on user data by suppressing encryption between the target user and the intruder.** An attack that requires a false BS/MS and that exploits the weakness that the MS cannot authenticate messages received over the radio interface. The target user is enticed to camp on the false BS. When the target user or the intruder initiates a call the network does not enable encryption by spoofing the cipher mode command. The intruder however builds op a (probably ciphered) connection between the genuine network and itself using its own subscription. The intruder may subsequently eavesdrop on the transmitted user data.

T14. **Eavesdropping on user data by suppression of encryption between the target user and the true network.** An attack that requires a false BS/MS and that exploits the weakness that the network cannot authenticate messages received over the radio interface. The target user is enticed to camp on the false BS/MS. When the target user or the genuine network set-up a service, the false BS/MS modifies the ciphering capabilities of the MS to make it appear to the network that a genuine incompatibility exists between the network and the mobile station ciphering capabilities. The network may then decide to establish an unciphered connection. After the decision not to cipher has been taken, the intruder may eavesdrop on the user data of the user.

T15. **Eavesdropping on user data by forcing the use of a compromised cipher key.** An attack that requires a false BS/MS and the possession by the intruder of a compromised authentication triplet and thus exploits the weakness that the user has no control the cipher key. The target user is enticed to camp on the false BS/MS. When the target user or the intruder set-up a service, the false BS/MS forces the use of a compromised cipher key on the mobile user while it builds up a connection with the genuine network using its own subscription.

A.3.6 Impersonation of the user

T16. **Impersonation of the user through the use of by the network of a compromised authentication vector.** An attack that requires a false MS and the possession by the intruder of the compromised authentication triplet which is used by the network (the intruder nor the user has any control over that) and thus exploits the weakness that an authentication vector sometimes is used several times. The intruder has obtained an authentication vector which is used by the network for authentication and key agreement. The intruder uses that data to impersonate the target user towards the network and the other party.

T17. **Impersonation of the user through the use by the network of an eavesdropped authentication response.** An attack that requires a false MS and exploits the weakness that an authentication vector sometimes is used several times. The intruder eavesdrops on the authentication response sent by the user and uses that when the same challenge is sent later on. Subsequently, ciphering has to be avoided by any of the mechanisms described here. The intruder uses the eavesdropped response data to impersonate the target user towards the network and the other party.

A.3.7 Access to services at the target user's account

The objective of these attacks is to receive services on the target user's account.

The first two attack are similar to the impersonation attacks:

T18. **Access to services at the target user's account through the use by the network of a compromised authentication vector.** An attack that requires a false MS and the possession by the intruder of the compromised authentication triplet which is used by the network (the intruder nor the user has any control over that) and thus exploits the weakness that an authentication vector sometimes is used several times. The intruder has obtained an authentication vector which is used by the network for authentication and key agreement.

T19. **Access to services at the target user's account through the use of an eavesdropped authentication response.** An attack that requires a false MS and exploits the weakness that an authentication vector sometimes is used several times. The intruder eavesdrops the authentication response sent by the user and uses that when the same challenge is sent later on. Subsequently, ciphering has to be avoided by any of the mechanisms seen so far. The intruder uses the eavesdropped response data to impersonate the target user towards the network and the other party.

In those cases where the intruder does not possess the authentication vector which is used by the network for authentication and key agreement, the intruder has to act as a relay between the target user and the network until authentication is completed. Then he conducts the following attacks:

T20. **Hijacking outgoing calls in networks with encryption disabled.** While the target user camps on the false base station, the intruder pages the target user for an incoming call. The user then initiates the call set-up procedure, which the intruder allows to occur between the serving network and the target user, modifying the signalling elements such that for the serving network it appears as if the target user wants to set-up a mobile originated call. The network does not enable encryption. After authentication the intruder cuts the connection with the target user, and subsequently uses the connection with the network to make fraudulent calls on the target user's subscription.

T21. **Hijacking outgoing calls in networks with encryption enabled.** In addition to the previous attack this time the intruder has to attempt to suppress encryption by modification of the message in which the MS informs the network of its ciphering capabilities as in T8.

T22. **Hijacking incoming calls in networks with encryption disabled.** While the target user camps on the false base station, an associate of the intruder makes a call to the target user's number. The intruder acts as a relay between the network and the target user until authentication and call set-up has been performed between target user and serving network. The network does not enable encryption. After authentication and call set-up the intruder releases the target user, and subsequently uses the connection to answer the call made by his associate. The target user will have to pay for the roaming leg.

Hijacking incoming calls in networks with encryption enabled. In addition to the previous attack this time the intruder has to suppress encryption T8.

B. Countermeasures

B.1 Exploited weaknesses

All attacks on the GSM security listed in section A.3 exploit a number of weakness of the GSM security architecture.

W1. ***Enticing users to camp on a false base station.*** Users cannot distinguish between false base stations and genuine base stations.

W2. ***The network may request the user to send his permanent user identity in cleartext.*** In GSM the network needs and has the ability to request the user to transmit his permanent user identity. Using a false BS an intruder may spoof such a request.

W3. ***Encryption is not everywhere and not always applied.*** In some countries network operators are by law prohibited to cipher. Also in networks where ciphering is usually applied, network operators have the ability to set-up not-ciphered connections. Of course, when ciphering is not applied, there is no confidentiality of user data or signalling data, nor of the user identity. However, ciphering is relied upon not only for confidentiality requirements, but also to authenticate the communicating parties throughout the call after call set-up is completed. Therefore, when encryption is not applied, the connection is open for hijacking of channels.

W4. ***The user and the serving network cannot authenticate messages they receive over the radio interface.*** The MS cannot verify the data origin nor the data integrity of signalling messages received over the radio interface. This can be exploited by an intruder to modify and spoof signalling data, such as the mobile station ciphering capabilities, the cipher mode command, and others.

W5. ***The confidentiality of the authentication vectors is questioned.*** In GSM authentication vectors (so-called triplets) are sent from the subscriber's Home Location Register (HLR) to the Visited Location Register (VLR), possibly in a different network. There are some concerns about the security of these links in view of future migration to shared routed transmission media.

B.2 Countermeasures

B.2.1 Denial of service

Any system that makes use of radio channels will be subject to some extent to denial of service attacks, including methods conceptually as simple as jamming. Therefore it is not required of a security architecture that it protects against all denial of service attacks. However, some denial of service attacks have consequences that last longer than those of other denial of service attacks.

We make the distinction between:

1. ***Instantaneous denial of service attacks:*** the targets are denied service for as long as the attack lasts, once the attack is over, the target users resume to have full service, or are likely to do so very soon. As these attacks are comparable to the effect of jamming, there is no need to prevent them.
2. ***Long term denial of service attack:*** these are attacks that cause denial of service for longer than the intervention of the attacker lasts. This type of attacks should be prevented by the 3G security architecture.

The threats T3 and T4 are instantaneous and cannot be thwarted.

However, the threats T1 have longer term consequences, and in addition, they do not require a false BS.

The countermeasures under consideration for this attack are:

M1a. **Data origin authentication and replay inhibition of the de-registration request.** This mechanism allows the serving network to verify that the de-registration request originates from the genuine user.

M1b. **Data origin authentication and replay inhibition of the location update request.** This mechanism allows the serving network to verify that the location update request originates from the genuine user.

Both the countermeasures M1a and M1b need to be implemented to thwart the denial of service attacks T1 and T2.

B.2.2 Identity catching

The following mechanisms have been suggested to provide user identity confidentiality (cf. also section 6.1.2):

M2a **A mechanism using a single temporary user identity.** In this mechanism the user is in most cases identified by a temporary user identity chosen by the serving network and sent to the mobile station using symmetric key encryption during a previous connection. In case the user registers for the first time in a serving network (then the network and the mobile station do not share a temporary user identity) or in case of a database malfunctioning in the serving network, the network can requests the mobile station to send its permanent user identity in cleartext.

M2b **A mechanism using two temporary identities.** This mechanism is an extension on the previous mechanism, and differs from it in case of first registrations and in case of a database malfunctioning in the serving network. In those cases the user is identified on the radio access interface by a second temporary user identity that is chosen by the user's home environment and sent to the mobile station by the home environment in a previous authentication between mobile station and home environment, again using symmetric key encryption. However, there is still a chance of a database malfunctioning in the home environment, therefore the serving network is still able to request the mobile station to send its permanent user identity in cleartext.

M2c **A mechanism using a secret group key.** In this mechanism the HE and its user's share a number of permanent secret group keys that are used to encrypt the permanent user identity. The user includes the group identity and home environment identity in clear text and the user identity encrypted by his group key when requested for identification by a permanent user identity. The serving network forwards all identity data to the home environment.

M2d **A mechanism using asymmetric key encryption of the user's permanent user identity.** The serving network provides the mobile station (either through broadcast or in the message requesting the identification) his certified public key, and after verification of the certificate, the mobile station sends the network its permanent user identity encrypted with the serving network's public key. The serving network is then able to decrypt the received message.

M2e **A mechanism using symmetric key encryption of the user's permanent user identity, after key establishment using asymmetric techniques.** Serving network and mobile station exchange certified copies of their public keys and establish a secret key. That key is subsequently used by the mobile station to send the network its permanent user identity.

Both mechanism M2d and mechanism M2e prevent active as well as passive eavesdropping, but are ruled out by the decision that UMTS Phase 1 will not incorporate public key technology.

Mechanism M2a, as compared with always sending the user identity in cleartext, makes passive identity catching inefficient, as the user identity is less frequently sent in cleartext.

Mechanism M2b gives even better protection against passive attacks, as it further reduces the frequency the user identity is sent in cleartext. However, active attacks, for those with the right equipment, are much more effective than passive eavesdropping of user identities, in which the eavesdropper must has to wait for a malfunctioning in the serving network database or for users registering in a serving network for the first time, for user identities to be sent in cleartext. Furthermore, the scenario of somebody passively listening out (in hope) for (effectively random) user identities, seems unlikely, if these user identities

cannot be used for any other attack (and the other measures in this document should ensure that is the case). Therefore it does not seem worthwhile to implement M2b.

Mechanism M2c prevents active identity catching without the introduction of a second layer of temporary identities, only static information is used to retrieve the permanent user identity form the information the user sends. This removes the problem related to a possible loss of data in the HLR database. In addition, the users are divided into groups which can be kept relatively small, such that their value will be insufficient to be the object of cryptographic attacks by intruders. The group size should on the other hand not be too small, as the group identity would otherwise be a compromise of the user identity itself.

In order to thwart the threats T5 and T6 the countermeasures M2a in combination with the add-on mechanism M2c should be implemented.

B.3 Suppression of encryption between target and intruder

A number of attacks listed in A.3 make use of the capability of an intruder that masquerades as a genuine network to decide on the encryption mode. This can be exploited for several objectives: Eavesdropping on the called party number (T7), impersonation of a genuine network (T10), non-covert eavesdropping of user data (T13).

The following countermeasures were considered:

M5a **Mandatory confidentiality of user and signalling data.** In this mechanism encryption is started by a mandatory command sent by the serving network. The user will not send data or signalling elements on unencrypted channels.

M5b **Mandatory cipher mode command with message authentication.** In this mechanism call set-up includes a mandatory command (referred to as the cipher mode command) that may enable encryption and defines the ciphering algorithm to be used. To the cipher mode command a message authentication code is appended such that the user can verify that the command originates from a proper network and has not been modified on the radio access link.

M5c **Default confidentiality for user and signalling data combined with an optional cipher mode command with message authentication.** In this mechanism encryption is started by default, without that the need for a cipher mode command to be sent. The ciphering algorithm is negotiated in a foregoing authentication exchange. The network can however switch off encryption by sending a cipher mode command. The user should be able to verify that that message comes from a proper serving network and has not been modified on the radio access link.

M5d **Network-to-user authentication.** In this mechanism, the network explicitly authenticates towards the user.

M5e **User control of encryption.** This mechanism alerts the user when encryption is not used on the call. The user may then decide to drop the call. This may also be done automatically by the user equipment depending on the user's choice.

Mechanism M5a suffers from the disadvantage that it prevents world-wide operability of the system as it may infer with national regulatory requirements in some countries.

Mechanism M5b counters all threats mentioned.

In mechanism M5c the absence of a mandatory cipher mode command may make the synchronisation required for ciphering more involved. Also, it is certainly not sufficient to rely on the encryption of the data channel to prevent modification of the cipher mode command, as that encryption mechanism will most probably be a stream cipher. Therefore, additional message authentication of the cipher mode command is a requirement here too. In short, there seems to be no real advantage of M5c as compared to M5b.

Mechanism M5d thwarts the attacks of a stand-alone false BS, but not the attacks that involve a man-in-the-middle attack which let authentication be conducted between. Therefore, it is concluded that network-to-user authentication on its own is not sufficient to thwart these attacks.

Mechanism M5e requires user interaction which is not wanted. Without doubt, most users will find the alarm most of the time disturbing and switch it off. The mechanism relies on the user to correctly assert

the probability that the lack of encryption is due to a false base station attack or not. The user is in no position to do so. However, the feature may be of interest to some users and can be implemented independently form any other feature.

The proposed countermeasure is therefore M5b.

B.4 Compromise of authentication data

The following countermeasures have been presented:

M6a **User control of cipher key generation.** The user participates in the generation of the cipher key. As long as the user inputs different or unpredictable values, he is able to prevent replay of old cipher keys.

M6b **Cipher key freshness towards the user.** The user does not necessarily participate in the generation of the cipher key, but has assurance that the cipher key has not been used before. This can be achieved by the use of sequence numbers. However, also user control of cipher key generation implies cipher key freshness.

M6c **Authentication data freshness towards the user.** The user is not only assured that the cipher key is fresh, but also that the authentication data sent by the home environment to the serving network and used by the serving network to proof to the user that he is allowed to offer him services (and possibly to generate cipher and integrity keys) is fresh.

M6d **Limited authentication data freshness towards the user.** If freshness in a strict sense means that authentication data has to be new for each authentication protocol run, limited authentication data freshness means that authentication data need not be new, but should only be valid for a certain period of time or for a limited number of protocol runs.

M6e **Cipher key updating without authentication data updating.** This mechanism allows to derive new cipher and/or integrity keys on the basis of the same authentication data.

Both M6a to M6b provide cipher key freshness towards the user, the only difference is that M6b details how that freshness is achieved, whereas M6b leaves that open. As the requirement is all what counts, the measure to include is M6b. Whether M6a will be used to accomplish M6b, will depend on the final decision on the mechanism.

The threats however, relate to the compromise of authentication data. If authentication data transfers the ability to the serving network to generate “fresh” cipher keys derived from one set of authentication data, cipher key freshness alone does not thwart the threat at all.

M6c limits the risks related to a compromise of authentication data to a minimum. M6c might however raise a concern on possible service interruption in cases where the serving network runs out of valid authentication data (which can only be used once). The serving network is then unable to start another authentication protocol run. In those cases, a serving network that does not want to disrupt services, may postpone authentication and both user and serving network may rely on the implicit authentication provided by ciphering and of the mandatory use of message authentication on some signalling elements. Whereas correct deciphering may be more error prone than verifying an authentication response, verification of message authentication code should be equally reliable. And unlike encryption algorithms, message authentication algorithms may be strong without incurring problems with export regulations or national regulatory restrictions. Nevertheless, if no mechanism exists to update the cipher key, long usage, might possibly compromise the cipher key. There might therefore be a requirement for M6e.

The weaker requirement M6d is also proposed. Authentication data may be re-used during a limited period of time, but as the USIM does not have a clock it is unlikely to use M6d, or until the occurrence of a particular event, such as location update, but that can easily be avoided, or a limited number of times, in which case the USIM keeps a counter. A new authentication based on the same authentication also updates the ciphering and authentication keys, and therefore implies M6e. However, this only temporarily avoids disruption of services, as the network eventually reaches the same situation as when M6c is implemented.

So, the essential difference is that with M6d, as compared to M6c, less authentication data may be sent to the serving network, but that authentication data provides the serving network with more autonomy and in

case of compromise, the intruder with more valuable material. The same autonomy can be accomplished using M6c, by providing larger bulks of authentication data. Whereas the losses due to the loss of public trust are hard to measure, the probability of running out of security data can be limited by sending a sufficient amount of authentication data, and how much authentication data needs to be sent can be determined by a study applying queuing theory. Of course also the value of these bulks of authentication data, in case they come in the hands of intruders, is then larger. The least that can be said is that the smaller the value of individual authentication data, the better serving networks and home environment can fine tune the doses sent transferred. M6c increases the signalling load on the wire line links between serving network and home environment, but reduces the security functionality in the serving network, eliminates real-time constraints at the network side and removes the need for standardised authentication algorithms. Whereas the losses due to the loss of public trust are hard to measure, the disturbance of services offered to roaming users can also be avoided in other ways: by acquiring sufficient authentication data.

Currently several proposals are on the table. One is a combination of M6a and M6d. The other combines M2b and M6c.

B.5 Manipulation of the encryption mode between the target and the serving network

While the target camps on the false base station, the intruder this time does not intervene in user-network authentication between the MS and the true serving network. However, the following threats are possible:

It should be noted that the GSM standards prevent this attack. In GSM 02.09 (v 5.1.1) it is stated clearly that a network should deny service to all user equipment that does not support both the algorithms A5/1 and A5/2. Therefore, since no other algorithms are currently in use, a genuine incompatibility cannot exist.

The proposed countermeasures are:

M7a **A set of mandatory ciphering algorithms.** The serving network should decide not to accept a call if the user equipment appears not to support a minimal set of ciphering algorithms. Interoperability is then achieved as soon as a serving is able to implemented at least one of these algorithms.

M7b **Terminal capability message with message authentication.** The user appends to the terminal capability message a message authentication code, such that the serving network can verify that is was sent by the users and has not been modified on the radio access link.

M7c **Mandatory encryption.** The network or the user may then refuse the connection in case of incompatibility.

Mechanism M7c again suffers from the disadvantage that it prevents the world-wide operability of the system as it may infer with national regulatory requirements in some countries.

Mechanism M7b on its own thwarts both attacks.

Mechanism M7a in its own only prevents the attack T14 and is in principle not required to meet the requirement. However, world-wide interoperability would only benefit from this requirement.

The proposed measures are thus M7a and M7b.

B.6 Hijacking of services

The following countermeasures may be introduced to thwart hijacking attacks:

M8a **Mandatory encryption of user data.** This mechanism forbids unencrypted calls.

M8b **User data with message authentication.** User and serving network append a message authentication code to the user data that is transmitted such that the receiving party can verify that the messages arrive from the expected source.

M8c **User entity authentication during service delivery.** While a call is in progress user and serving network may periodically run a challenge-response type re-authentication run of the other end during service delivery.

M8d **Periodic authentication of signalling messages with message authentication.** User and serving network may periodically exchange signalling messages to be authenticated by a message authentication code. The receiver should be able to verify that these messages are fresh, for instance through the use of sequence numbers, in order to prevent replay. Therefore (because of the freshness parameters needed) the messages used could not only be usual signalling messages.

M8e **Called party number with message authentication.** During call set-up in a mobile originated call the called party number is sent to the serving network with a message authentication code appended such that the serving network is able to verify that the called party number was sent by the user and not by an intruder.

Mechanism M8a suffers from the disadvantage that it prevents the world-wide operability of the system as it may infer with national regulatory requirements in some countries.

Mechanism M8b requires extra bandwidth on the radio access link. In addition, radio access links are typically error prone. Care should be taken that the detection of transmission errors due to the atmospheric conditions does not disrupt services.

In mechanism M8c and M8d the frequency of these messages is an issue. If the check is performed every 30s, then channels can be hijacked for calls less than 30s. This may be sufficient to deter the hijacking of circuit switched calls. Packet switched services however, will probably require individual protection of each packet or burst.

Mechanism M8e prevents the intruder to set-up a connection to a party of his own choice. This reduces the value of the attack considerably. However, it does not prevent T22.

The proposed countermeasures are therefore M8d and M8e.