3GPP TSG SA WG 3 (Security) meeting #11
Mainz, 22—24 February, 2000

Document **S3-000163**
(Rev. of S3-000113)

e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

33.102 **CR** **053r1** Current Version: 3.3.1

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: SA #7 — for approval ☒ — for information □ — strategic □ — non-strategic □ *(for SMG use only)*
list expected approval meeting # here ↑

Form: CR cover sheet, version 2 for 3GPP and SMG The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: (U)SIM □ ME □ UTRAN / Radio □ Core Network ☒
(at least one should be marked with an X)

Source: S3 **Date:** 2000-02-23

Subject: Removal of EUIC from 'Authentication Data Request' procedure.

Work item: Security

Category: *(only one category shall be marked with an X)*

F	Correction	☒	**Release:**	Phase 2	□
A	Corresponds to a correction in an earlier release	□		Release 96	□
B	Addition of feature	□		Release 97	□
C	Functional modification of feature	□		Release 98	□
D	Editorial modification	□		Release 99	☒
				Release 00	□

Reason for change:

Decryption of EMSI will only be performed within a "User Identity Request Procedure" described in chapter 6.2 of TS 33.102 (MAP_SEND_IMSI operation will be used for this purpose). The rest of operations will have to wait until decryption of EMSI is performed.

The use of EMSI ('*HLR message*') is therefore removed from 'Authentication Data Request' procedure. IMSI is also removed from 'Authentication data response'.

The terms "SN/VLR" and "IMUI" have been replaced by "VLR/SGSN" and "IMSI" respectively.

Clauses affected: 6.3.2

Other specs affected:

Other 3G core specifications	□	→ List of CRs:
Other GSM core specifications	□	→ List of CRs:
MS test specifications	□	→ List of CRs:
BSS test specifications	□	→ List of CRs:
O&M specifications	□	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

6.3.2 Distribution of authentication data from HE to SN

The purpose of this procedure is to provide the VLR/SGSN with an array of fresh authentication vectors from the user's HE to perform a number of user authentications.

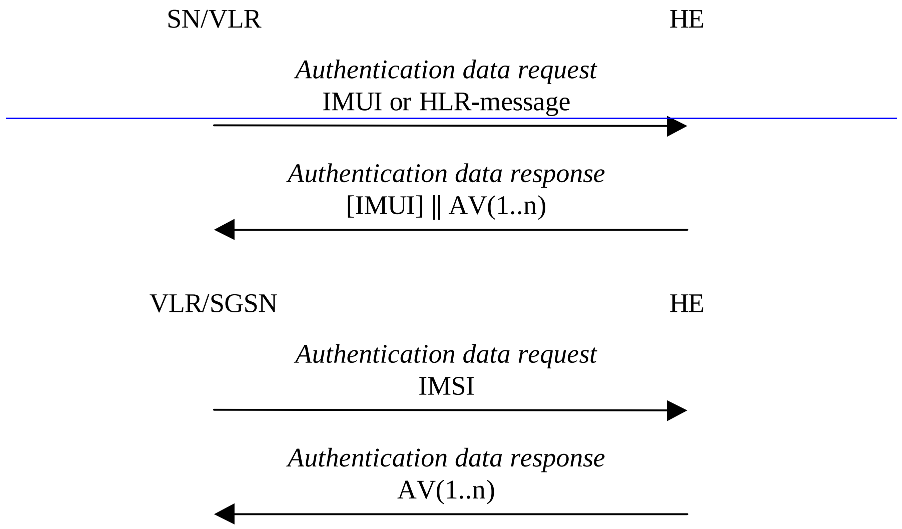
SN/VLR
HE
Authentication data request
IMUI or HLR-message
Authentication data response
[IMUI] || AV(1..n)
VLR/SGSN
HE
Authentication data request
IMSI
Authentication data response
AV(1..n)

Figure 6: Distribution of authentication data from HE to VLR/SGSN

The VLR/SGSN invokes the procedures by requesting authentication vectors to the HE/AuC.

The *authentication data request* shall include~~ a user~~ the IMSI~~identity. If the user is known in the VLR/SGSN by means of the IMUI, the *authentication data request* shall include the IMUI~~. ~~However, if the user is identified by means of an encrypted permanent identity (see 6.2), the HLR-message from which the HE can derive the IMUI is included instead. In that case, this procedure and the procedure *user identity request to the HLR* are integrated.~~

Upon the receipt of the *authentication data request* from the VLR/SGSN, the HE may have pre-computed the required number of authentication vectors and retrieve them from the HLR database or may compute them on demand. The HE/AuC sends an authentication response back to the VLR/SGSN that contains an ordered array of n authentication vectors AV(1..n).

Figure 7 shows the generation of an authentication vector AV by the HE/AuC.

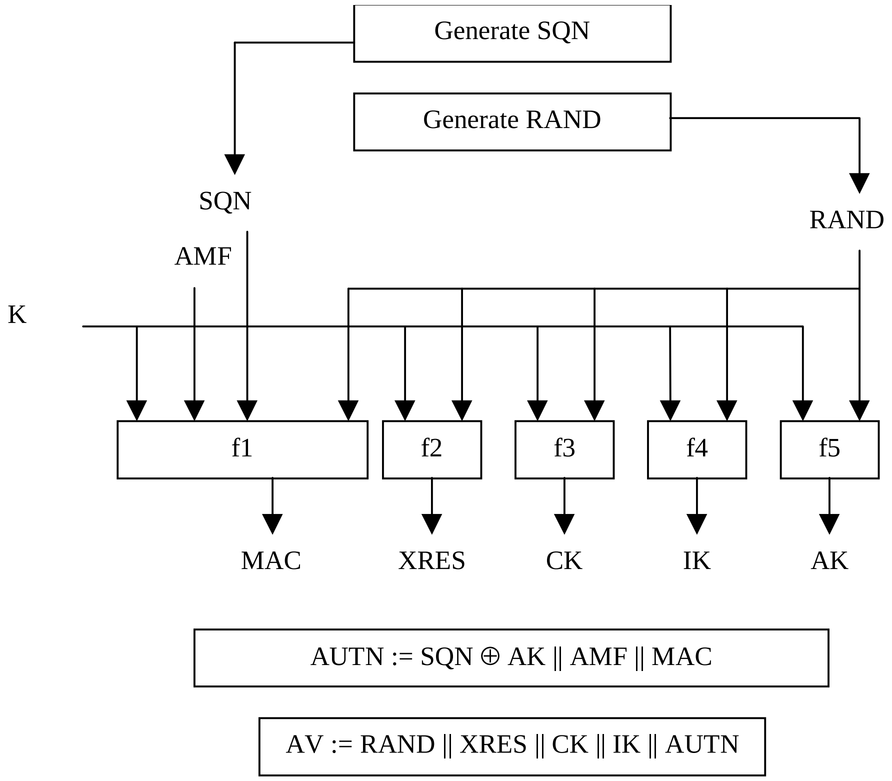
Generate SQN
Generate RAND
SQN
RAND
AMF
K
f1
f2
f3
f4
f5
MAC
XRES
CK
IK
AK
AUTN := SQN ⊕ AK || AMF || MAC
AV := RAND || XRES || CK || IK || AUTN

Figure 7: Generation of authentication vectors

The HE/AuC starts with generating a fresh sequence number SQN and an unpredictable challenge RAND.

For each user the HE/AuC keeps track of a counter: SQN_{HE}

The HE has some flexibility in the management of sequence numbers, but some requirements need to be fulfilled by the mechanism used:

a) The generation mechanism shall allow a re-synchronisation procedure in the HE described in section 6.3.5

b) The SQN should be generated in such way that it does not expose the identity and location of the user.

c) In case the SQN may expose the identity and location of the user, the AK may be used as an anonymity key to conceal it.

d) The generation mechanism shall allow protection against wrap around the counter in the USIM.
A method how to achieve this is given in informative Annex C.2.

The mechanisms for verifying the freshness of sequence numbers in the USIM shall to some extent allow the out-of-order use of sequence numbers. This is to ensure that the authentication failure rate due to synchronisation failures is sufficiently low. This requires the capability of the USIM to store information on past successful authentication events (e.g. sequence numbers or relevant parts thereof). The mechanism shall ensure that a sequence number can still be accepted if it is among the last x = 50 sequence numbers generated. This shall not preclude that a sequence number is rejected for other reasons such as a limit on the age for time-based sequence numbers.
The same minimum number x needs to be used across the systems to guarantee that the synchronisation failure rate is sufficiently low under various usage scenarios, in particular simultaneous registration in the CS- and the PS-service domains, user movement between VLRs/SGSNs which do not exchange authentication information, super-charged networks.

The use of SEQ_{HE}~~HE~~ is specific to the method of generation sequence numbers. A method is specified in Annex C.1 how to generate a fresh sequence number. A method is specified in Annex C.2 how to verify the freshness of a sequence number.

An authentication and key management field AMF is included in the authentication token of each authentication vector. Example uses of this field are included in Annex F.

Subsequently the following values are computed:

- a message authentication code MAC = $f1_K(SQN \| RAND \| AMF)$ where f1 is a message authentication function;
- an expected response XRES = $f2_K$ (RAND) where f2 is a (possibly truncated) message authentication function;
- a cipher key CK = $f3_K$ (RAND) where f3 is a key generating function;
- an integrity key IK = $f4_K$ (RAND) where f4 is a key generating function;
- an anonymity key AK = $f5_K$ (RAND) where f5 is a key generating function or $f5 \equiv 0$.

Finally the authentication token AUTN = $SQN \oplus AK \| AMF \| MAC$ is constructed.

Here, AK is an anonymity key used to conceal the sequence number as the latter may expose the identity and location of the user. The concealment of the sequence number is to protect against passive attacks only. If no concealment is needed then $f5 \equiv 0$.