Technical Specification Group Services and System Aspects **TSGS#7(00)0048**

Meeting #7, Madrid, Spain, 15-17 March 2000

Source: **SA WG3**

Title: **Miscellaneous CRs to 33.105**

Document for: **Approval**

Agenda Item: **5.3.3**

Miscellaneous CRs to 33.105

Introduction:

This document contains 4 CRs to **33.105** for Release 1999 which is submitted to SA#7 for approval.

SA WG3 TD	Spec	CR	Rev	Phase	Subject	Cat	Current Version	Comments
S3-000084	33.105	006		R99	Authentication and key agreement	F	3.2.0	
S3-000082	33.105	007		R99	Editorial changes to Terminology	F	3.2.0	
S3-000134	33.105	009		R99	Ciphering	C	3.2.0	Agreed by e-mail
S3-000135	33.105	010		R99	Data integrity	D	3.2.0	Agreed by e-mail

3GPP TSG SA 3 Meeting #10
Antwerp, Belgium, 19-21 January 2000

***Document* S3-000084**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

DRAFT CHANGE REQUEST

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

33.105	CR	006	Current Version:	3.2.0

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

For submission to:	TSG SA #7	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

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: *(at least one should be marked with an X)* (U)SIM [X] ME [ ] UTRAN / Radio [ ] Core Network [X]

Source: Siemens Atea **Date:** 21-01-00

Subject: Authentication and key agreement

Work item: Security

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

F	Correction	X
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change: Bring TS 33.105 in line with the decisions taken in TSG SA and the CRs to TS 33.102 that have been approved. This includes the replacement of MODE by AMF, the change as regards input parameters to the computation of the re-synchronisation token AUTS, the deletion of the alternative mechanism (annex B) and the deletion of annex C on unspecified values, as all these value have in the mean time been specified.

Clauses affected: 5.1, annex B, annex C

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.

5.1 Authentication and key agreement

5.1.1 Overview

The mechanism for authentication and key agreement described in clause 6.3 of [1] requires the following cryptographic functions:

f0	the random challenge generating function;
f1	the network authentication function;
f1*	the re-synchronisation message authentication function;
f2	the user authentication function;
f3	the cipher key derivation function;
f4	the integrity key derivation function;
f5	the anonymity key derivation function.

5.1.1.1 Generation of quintets in the AuC

To generate a quintet the HLR/AuC

- computes a message authentication code for authentication MAC-A = $f1_K$(SQN || RAND || AMF), an expected response XRES = $f2_K$ (RAND), a cipher key CK = $f3_K$ (RAND) and an integrity key IK = $f4_K$ (RAND) where f4 is a key generating function.
- If SQN is to be concealed, in addition the HLR/AuC computes an anonymity key AK = $f5_K$ (RAND) and computes the concealed sequence number SQN ⊕ AK = SQN xor AK. Concealment of the sequence number is optional.
- Finally, the HLR/AuC assembles the authentication token AUTN = SQN [⊕ AK] || AMF || MAC-A and the quintet Q = (RAND, XRES, CK, IK, AUTN).

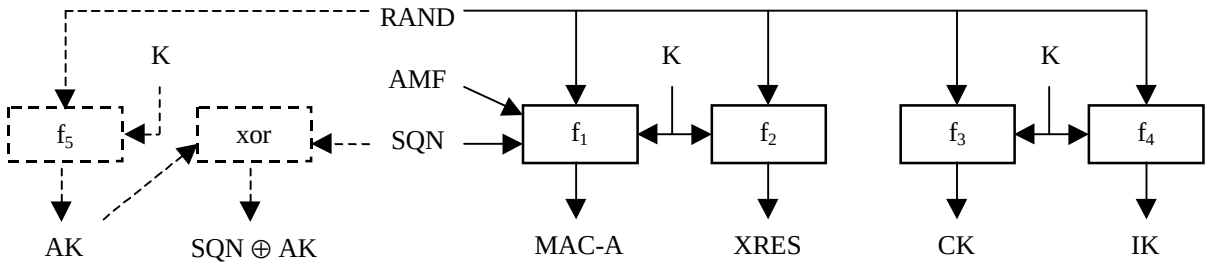

Figure 1: Generation of quintets in the AuC

5.1.1.2 Authentication and key derivation in the USIM

Upon receipt of a (RAND, AUTN) pair the USIM acts as follows:

- If the sequence number is concealed, the USIM computes the anonymity key AK = $f5_K$(RAND) and retrieves the unconcealed sequence number SQN = (SQN ⊕ AK) xor AK.

The USIM computes XMAC-A = $f1_K$ (SQN || RAND || AMF), the response RES = $f2_K$(RAND), the cipher key CK

= $f3_K$ (RAND) and the integrity key IK = $f4_K$ (RAND).

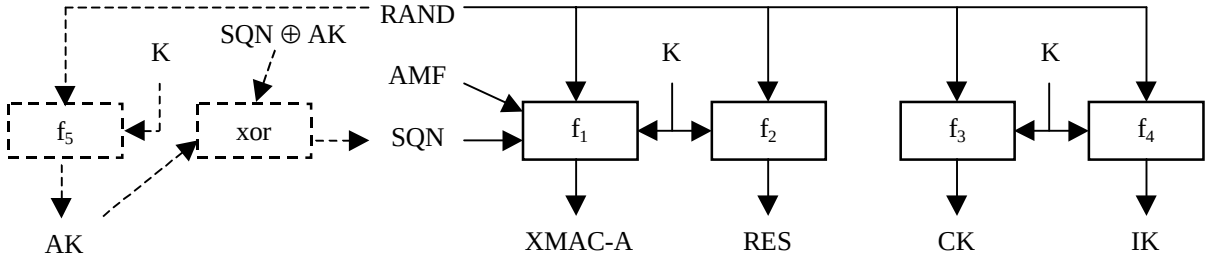

Figure 2: Authentication and key derivation in the USIM

5.1.1.3 Generation of re-synchronisation token in the USIM

Upon the assertion of a synchronisation failure, the USIM generates a re-synchronisation token as follows:

a) The USIM computes MAC-S = $f1^*_K(SQN_{MS} \| RAND \| AMF^*)$, whereby AMF* is a default value for AMF used in re-synchronisation.

b) If SQN_{MS} is to be concealed with an anonymity key AK, the USIM computes AK = $f5_K$(MAC-S || 0…0), whereby MAC-S forms the 12 most significant octets and 32 zeros form the 4 least significant octets of the required 16 octet input parameter, and the concealed counter value is then computed as $SQN_{MS} \oplus AK$.

c) The re-synchronisation token is constructed as AUTS = SQN_{MS} [⊕ AK] || MAC-S.

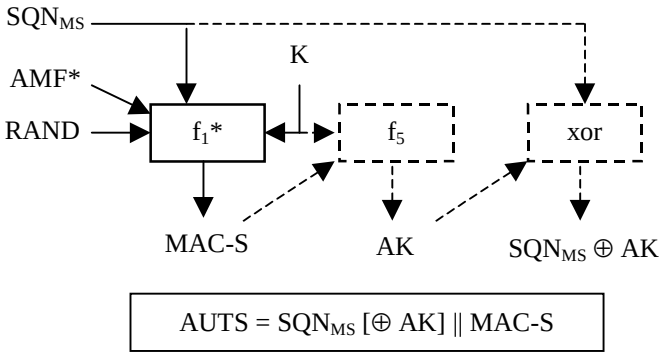

Figure 3: Generation of re-synchronisation token in the USIM

5.1.1.4 Re-synchronisation in the HLR/AuC

Upon receipt of an indication of synchronisation failure and a (AUTS, RAND) pair, the HLR/AuC may perform the following cryptographic functions:

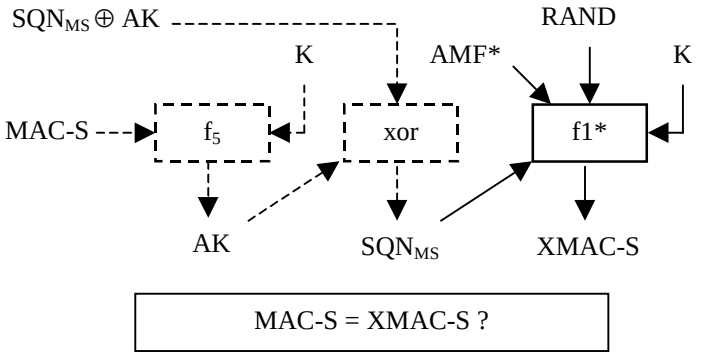

Figure 4: Re-synchronisation in the HLR/AuC

a) If SQN_{MS} is concealed with an anonymity key AK, the HLR/AuC computes AK = $f5_K$(MAC-S || 0…0), whereby MAC-S forms the 12 most significant octets and 32 zeros form the 4 least significant octets of the required 16 octet input parameter and retrieves the unconcealed counter value as SQN_{MS} = (SQN_{MS} ⊕ AK) xor AK.

b) If SQN generated from SQN_{HE} would not be acceptable, then the HLR/AuC computes XMAC-S = $f1^*_K$(SQN_{MS} || RAND || AMF*), whereby AMF* is a default value for AMF used in re-synchronisation.

5.1.2 Use

The functions f0—f5 shall only be used to provide mutual entity authentication between USIM and AuC, derive keys to protect user and signalling data transmitted over the radio access link and conceal the sequence number to protect user identity confidentiality. The function f1* shall only be used to provide data origin authentication for the synchronisation failure information sent by the USIM to the AuC.

5.1.3 Allocation

The functions f1—f5 and f1*are allocated to the Authentication Centre (AuC) and the USIM. The function f0 is allocated to the AuC.

5.1.4 Extent of standardisation

The functions f0—f5 and f1*are proprietary to the home environment. Examples of the functions f1, f1* and f2 are CBC-MACs or H-MACs [3].

5.1.5 Implementation and operational considerations

The functions f1—f5 and f1* shall be designed so that they can be implemented on an IC card equipped with a 8-bit microprocessor running at 3.25 MHz with 8 kbyte ROM and 300byte RAM and produce AK, XMAC-A, RES, CK and IK in less than 500 ms execution time.

5.1.6 Type of algorithm

5.1.6.1 f0

f0: the random challenge generating function

f0: (internal state) → RAND

f0 should be (pseudo) random number generating function.

5.1.6.2 f1

f1: the network authentication function

f1: (K; SQN, RAND, AMF) → MAC-A (or XMAC-A)

f1 should be a MAC function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND, SQN, AMF and MAC-A (or XMAC-A).

5.1.6.3 f1*

f1*: the re-synchronisation message authentication function

f1*: (K; SQN, RAND, AMF) → MAC-S (or XMAC-S)

f1 should be a MAC function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND, SQN, AMF and MAC-S (or XMAC-S).

5.1.6.4 f2

f2: the user authentication function

f2: (K; RAND) → RES (or XRES)

f2 should be a MAC function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND and RES (or XRES).

5.1.6.5 f3

f3: the cipher key derivation function

f3: (K; RAND) → CK

f3 should be a key derivation function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND and CK.

5.1.6.6 f4

f4: the integrity key derivation function

f4: (K; RAND) → IK

f4 should be a key derivation function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND and IK.

5.1.6.7 f5

f5: the anonymity key derivation function

f5: (K; RAND) → AK

f5 should be a key derivation function. In particular, it shall be computationally infeasible to derive K from knowledge of RAND and AK.

The use of f5 is optional.

5.1.7 Interface

5.1.7.1 K

K: the subscriber authentication key

K[0], K[1], …, K[127]

The length of K is 128 bits. The subscriber authentication key K is a long term secret key stored in the USIM and the AuC.

5.1.7.2 RAND

RAND: the random challenge

RAND[0], RAND[1], …, RAND[127]

The length of RAND is 128 bits.

5.1.7.3 SQN

SQN: the sequence number

SQN[0], SQN[1], …, SQN[47]

The length of SQN is 48 bits. The AuC should include a fresh sequence number in each authentication token. The verification of the freshness of the sequence number by the USIM constitutes to entity authentication of the network to the user.

5.1.7.4 AMF

AMF: the authentication management field

AMF[0], AMF[1], …, AMF[15]

The length of AMF is 16 bits. The use of AMF is not standardised. Example uses of the AMF are provided in annex F of TS 33.102.

5.1.7.6 MAC-A (equivalent for XMAC-A)

MAC-A: the message authentication code used for authentication of the network to the user

MAC-A[0], MAC-A[1], …, MAC-A[63]

The length of MAC-A is 64 bits. MAC-A authenticates the data integrity and the data origin of RAND, SQN and AMF. The verification of MAC-A by the USIM constitutes to entity authentication of the network to the user.

5.1.7.7 MAC-S (equivalent for XMAC-S)

MAC-S: the message authentication code used to provide data origin authentication for the synchronisation failure information sent by the USIM to the AuC.

MAC-S[0], MAC-S[1], …, MAC-S[63]

The length of MAC-S is 64 bits. MAC-S authenticates the data integrity and the data origin of RAND, SQN and AMF. MAC-S is generated by the USIM and verified by the AuC.

5.1.7.8 RES (or XRES)

RES: the user response

RES[0], RES[1], …, RES[31…127]

The maximum length of RES and XRES is 128 bits and the minimum is 32 bits. RES and XRES constitute to entity authentication of the user to the network.

5.1.7.9 CK

CK: the cipher key

CK[0], CK[1], …, CK[127]

The length of CK is 128 bits. In case the effective key length should need to be made smaller than 128 bits, the most significant bits of CK shall carry the effective key information, whereas the remaining, least significant bits shall be set zero.

5.1.7.10 IK

IK: the integrity key

IK[0], IK[1], …, IK[127]

The length of IK is 128 bits. In case the effective key length should need to be made smaller than 128 bits, the most significant bits of IK shall carry the effective key information, whereas the remaining, least significant bits shall be set zero.

5.1.7.11 AK

AK: the anonymity key

AK[0], AK[1], …, AK[47]

The length of AK is 48 bits. It equals the length of SQN.

[1] RSn and/or RSu can be a random number or a counter

DRAFT 3G CHANGE REQUEST

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

TS 33.105 **CR** **007** Current Version: V3.2.0

3G specification number ↑ *↑ CR number as allocated by 3G support team*

For submission to TSG SA#6 *list TSG meeting no. here ↑* for approval [X] for information [ ] *(only one box should be marked with an X)*

Form: 3G CR cover sheet, version 1.0 *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/3GCRF-xx.rtf*

Proposed change affects: *(at least one should be marked with an X)* USIM [X] ME [X] UTRAN [ ] Core Network [X]

Source: TSG SA WG3 **Date:** 21-01-2000

Subject: Enhanced user confidentiality

3G Work item: Security

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

F	Correction	[X]
A	Corresponds to a correction in a 2G specification	[ ]
B	Addition of feature	[ ]
C	Functional modification of feature	[ ]
D	Editorial modification	[ ]

Reason for change: Align with TS33.102 (Security Architecture), descriptions and figures on the enhanced user confidentiality have been corrected.

Clauses affected: 3.3, Annex A

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other 2G 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:

3.3 Abbreviations

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

3GPP	3rd Generation Partnership Project
AK	Anonymity key
AuC	Authentication Centre
AUTN	Authentication token
CK	Cipher key
EMSUI	Encrypted Mobile SubscriberUser Identity
GK	User group key
IK	Integrity key
IMSUI	International Mobile SubscriberUser Identity
IPR	Intellectual Property Right
MAC	Medium access control (sublayer of Layer 2 in RAN)
MAC	Message authentication code
MAC-A	MAC used for authentication and key agreement
MAC-I	MAC used for data integrity of signalling messages
PDU	Protocol data unit
RAND	Random challenge
RES	User response
RLC	Radio link control (sublayer of Layer 2 in RAN)
RNC	Radio network controller
SEQ_UIC	Sequence for user identity confidentiality
SDU	Signalling data unit
SQN	Sequence number
UE	User equipment
USIM	User Services Identity Module
XMAC-A	Expected MAC used for authentication and key agreement
XMAC-I	Expected MAC used for data integrity of signalling messages
XRES	Expected user response

Annex A (informative): User identity confidentiality

A.1 Overview

Figure A**Error! Reference source not found.** illustrates the use of the encryption function f6 to encrypt the IMSUI and the sequence for user identity confidentiality (SEQ_UIC) into an EMSUI and the use of the decryption function f7 to decrypt the EMSUI and retrieve the SEQ_UIC and the IMSUI.

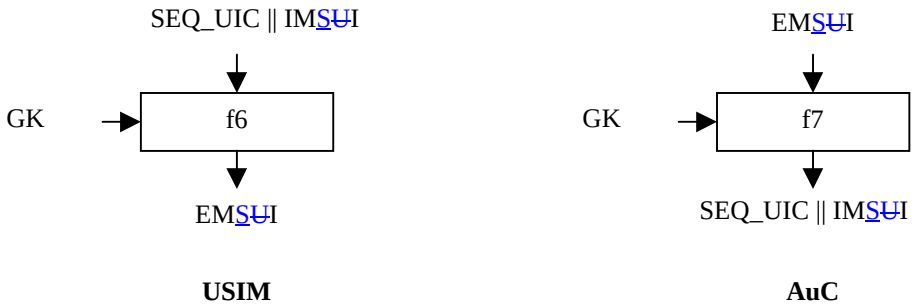

Figure A: Encryption and decryption of the permanent user identity

The mechanism for user identity confidentiality that is described in annex B of [1] requires the following cryptographic functions:

f6 the user identity encryption function;
f7 the user identity decryption function.

A.2 Use

The functions f6 and f7 shall only be used to protect the confidentiality of the user identity when transmitted from USIM to AuC.

A.3 Allocation

The function f6 is allocated to the USIM. The function f7 is allocated to the Authentication Centre.

A.4 Extent of standardisation

The functions f6 and f7 are proprietary to the home environment.

A.5 Implementation and operational considerations

The function f6 shall be designed so that it can be implemented on an IC card equipped with a X1-bit microprocessor running at X2 MHz and with X3 kbits of memory and produce EMUI in less than X11 ms.

The functions f7 shall be designed so that they can be implemented in software in the AuC on a X6-bit microprocessor running at X7 MHz and X8 kbits of memory and produce SEQ_UIC || IMUI in less than X12 ms.

A.6 Type of algorithm

A.6.1 f6

f6: the user identity encryption function

f6: (GK; SEQ_UIC || IMSUI) → EMSUI

f6 should be a block cipher.

A.6.2 f7

f7: the user identity decryption function

f7: (GK; EMSUI) → SEQ_UIC || IMSUI

f7 should be a block cipher and the inverse function of f6, in the sense that

x = f7(y; f6(y; x)), for all valid x = SEQ_UIC || IMSUI and all valid y = GK.

A.7 Interface

A.7.1 GK

GK: the user group key

GK[0], GK[1], …, GK[X13-1]

The maximum length of the group key GK is X13 bits. The user group key GK is a long term secret key stored in several USIMs and in the AuC.

A.7.2 SEQ_UIC

SEQ_UIC: the sequence for user identity confidentiality

SEQ_UIC[0], SEQ_UIC[1], …, SEQ_UIC[X14-1]

The length of SEQ_UIC is X14 bits. The SEQ_UIC is generated by the USIM and should be different each time so as to prevent traceability of a user.

A.7.3 IMSUI

IMSUI: the international mobile subscriber~~user~~ identity

IMSUI[0], IMSUI[1], …, IMSUI[X15-1]

The length of the IMSUI is X15bits. The IMSUI is the permanent identity of the user, stored in the USIM and in the AuC.

A.7.4 EMSUI

EMSUI: the encrypted mobile subscriber~~user~~ identity

EMSUI[0], EMSUI[1], …, EMSUI[X16-1]

The length of the EMSUI is X16 bits.

3GPP TSG SA 3 meeting #11
Mainz, Germany, 22-24 February 2000

***Document* S3-000134**
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.

33.105	CR	009	Current Version:	3.2.0

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

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

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: *(at least one should be marked with an X)* (U)SIM [ ] ME [ ] UTRAN / Radio [ ] Core Network [ ]

Source: SA WG3 **Date:**

Subject: Ciphering

Work item: Security

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

F	Correction	
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	X
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change: The structure of CK when the effective key length is smaller than 128 bits is changed. In addition, some editorial changes are proposed.

Clauses affected: 5.2

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.

5.2 Data confidentiality

5.2.1 Overview

The mechanism for data confidentiality of user data and signalling data that is described in 6.6 of [1] requires the following cryptographic function:

f8 UMTS encryption algorithm.

Figure 5.2.1 illustrates the use of f8 to encrypt plaintext by applying a keystream using a bitwise XOR operation. The plaintext may be recovered by generating the same keystream using the same input parameters and applying it to the ciphertext using a bitwise XOR operation.

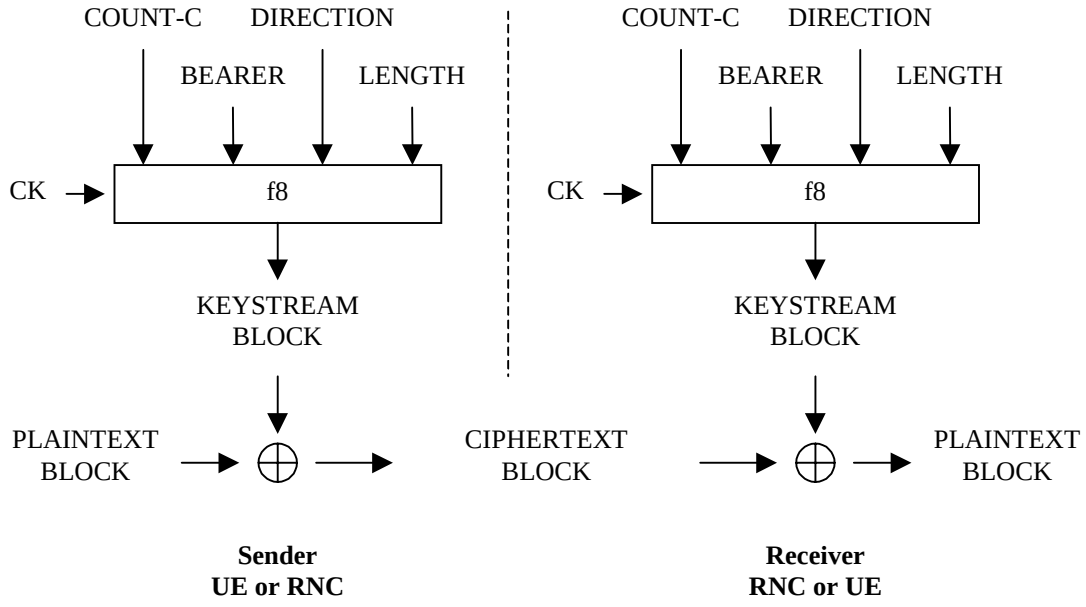

Figure 5.2.1: Ciphering user and signalling data transmitted over the radio access link

The input parameters to the algorithm are the Cipher Key (CK), a time dependent input (COUNT-C), the bearer identity (BEARER), the direction of transmission (DIRECTION) and the length of the keystream required (LENGTH). Based on these input parameters the algorithm generates the output keystream block (KEYSTREAM) which is used to encrypt the input plaintext block (PLAINTEXT) to produce the output ciphertext block (CIPHERTEXT).

The input parameter LENGTH shall affect only the length of the KEYSTREAM BLOCK, not the actual bits in it.

5.2.2 Use

The function f8 shall only be used to protect the confidentiality of user data and signalling data sent over the radio access link between UE and RNC.

5.2.3 Allocation

The function f8 is allocated to the UE and the RNC.

Encryption will be applied in the Medium Access Control (MAC) sublayer and in the Radio Link Control (RLC) sublayer of the data link layer (Layer 2).

5.2.4 Extent of standardisation

The function f8 shall be fully standardized.

5.2.5 Implementation and operational considerations

The algorithm should be designed to accommodate a range of implementation options including hardware and software implementations. For hardware implementations, it should be possible to implement one instance of the algorithm using less than 10,000 gates (working assumption).

A wide range of UE with different bearer capabilities is expected, so the encryption throughput requirements on the algorithm will vary depending on the implementation. However, based on the likely maximum user traffic data rates, it must be possible to implement the algorithm to achieve an encryption speed in the order of 2Mbit/s on the downlink and on the uplink.

1. RLC-transparent mode:
   - New keystream block required every physical layer frame (10ms)
   - Maximum number of bits per physical layer frame of 5114 bits
   - Minimum number of bits per physical layer frame of 1 bit.
   - Granularity of 1 bit on all possible intermediate values
2. For UM RLC mode:
   - New keystream block required every RLC frame (minimum 156μs)
   - Maximum number of bits per UM RLC frame of 1016 bits (ongoing specification work in TSG-R2 could extend this to 5000 bits)
   - Minimum number of bits per UM RLC frame of 16 bit.
   - Granularity of 8 bit on all possible intermediate values
3. For AM RLC mode:
   - New keystream block required every RLC frame (minimum 156μs)
   - Maximum number of bits per AM RLC frame of 1024 bits (ongoing specification work in TSG-R2 could extend this to 5000 bits)
   - Minimum number of bits per AM RLC frame of 24 bit.
   - Granularity of 8 bit on all possible intermediate values

The encryption throughput requirements should be met based on clock speeds upwards of 20MHz (typical clock speeds are expected to be much greater than this).

5.2.6 Type of algorithm

The function f8 should be a symmetric synchronous stream cipher.

5.2.7 Interfaces to the algorithm

5.2.7.1 CK

CK: the cipher key

CK[0], CK[1], …, CK[127]

The length of CK is 128 bits. In case the effective key length k is smaller than 128 bits, the most significant bits of CK shall carry the effective key information, whereas the remaining, least significant bits shall repeat the effective key information:

$CK[n] = CK[n \bmod k]$, for all n, such that $k \leq n < 128$.

5.2.7.2 COUNT-C

COUNT-C: the cipher sequence number.

COUNT-C[0], COUNT-C[1], …, COUNT-C[31]

The length of the COUNT-C parameter is 32 bits.

Synchronisation of the keystream is based on the use of a physical layer (Layer 1) frame counter combined with a hyperframe counter introduced to avoid re-use of the keystream. This allows the keystream to be synchronised every 10ms physical layer frame. The exact structure of the COUNT-C is specified in TS 33.102.

5.2.7.3 BEARER

BEARER: the bearer identifier.

BEARER[0], BEARER[1], …, BEARER[3]

The length of BEARER is 4 bits.

The same cipher key may be used for different bearers simultaneously associated with a single user which are multiplexed onto a single 10ms physical layer frame. To avoid using the same keystream to encrypt more than one bearer, the algorithm shall generate the keystream based on the identity of the bearer.

5.2.7.4 DIRECTION

DIRECTION: the direction of transmission of the bearer to be encrypted.

DIRECTION[0]

The length of DIRECTION is 1 bit.

The same cipher key may be used for uplink and downlink channels simultaneously associated with a UE, which are multiplexed onto a single 10ms physical layer frame. To avoid using the same keystream to encrypt both uplink and downlink transmissions, the algorithm shall generate the keystream based on the direction of transmission.

An explicit direction value is required in preference to splitting the keystream segment into uplink and downlink portions to allow for asymmetric bearer services.

5.2.7.5 LENGTH

LENGTH: the required length of keystream.

LENGTH[0], LENGTH[1], …, LENGTH[15]

The length of LENGTH is 16 bits. For a given bearer and transmission direction the length of the plaintext block that is transmitted during a single physical layer frame may vary. The algorithm shall generate a keystream block of variable length based on the value of the length parameter.

The input parameter LENGTH shall affect only the length of the KEYSTREAM BLOCK, not the actual bits in it.

The format of LENGTH cannot be specified at present since the number and sizes of RLC PDUs / MAC SDUs in each 10ms physical layer frame have not yet been fully specified. However, a maximum RLC PDU / MAC SDU size in the region of 1000 bits has been informally indicated by 3GPP TSG RAN2. The range of values of the length parameter will depend not only on the RLC PDU / MAC SDU size but also the number of RLC PDUs / MAC SDUs which may be sent in a single physical layer 10ms frame for a given bearer and transmission direction.

Not all values between the maximum and minimum values shall be required but it is expected that the ability to produce length values of whole numbers of octets between a minimum and a maximum value will be required.

5.2.7.6 KEYSTREAM

KEYSTREAM: the output keystream.

KS [0], KS [1], …, KS [LENGTH-1]

The length of a keystream block equals the value of the input parameter LENGTH.

5.2.7.7 PLAINTEXT

PLAINTEXT: the plaintext.

PT[0], PT[1], …, PT[LENGTH-1]

The length of a keystream block equals the value of the input parameter LENGTH.

This plaintext block consists of the payload of the particular RLC PDUs / MAC SDUs to be encrypted in a single 10ms physical layer frame for a given bearer and transmission direction. It may consist of user traffic or signalling data. The structure of the plaintext block cannot be specified at present.

5.2.7.8 CIPHERTEXT

CIPHERTEXT: the ciphertext.

CT[0], CT[1], …, CT[LENGTH-1]

The length of a keystream block equals the value of the input parameter LENGTH.

3GPP TSG SA 3 meeting #11
Mainz, Germany, 22-24 February 2000

***Document* S3-000135**
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.

33.105	CR	010	Current Version:	3.2.0

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

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

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: *(at least one should be marked with an X)* (U)SIM [X] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: SA WG3 **Date:**

Subject: Data integrity

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	X		Release 99	X
				Release 00	

Reason for change: The provision for shorter integrity keys is deleted from the text.
Additional editorial changes.

Clauses affected: 5.3

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.

5.3 Data integrity

5.3.1 Overview

The mechanism for data integrity of signalling data that is described in 6.6 of [1] requires the following cryptographic function:

f9 UMTS integrity algorithm.

Figure 3 illustrates the use of the function f9 to derive a MAC-I from a signalling message.

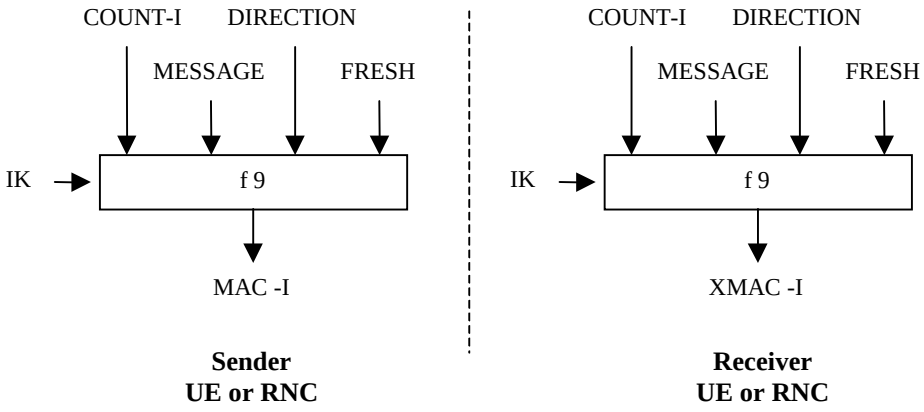

Figure 5.3.1: Derivation of MAC-I (or XMAC-I) on a signalling message

The input parameters to the algorithm are the Integrity Key (IK), a time dependent input (COUNT-I), a random value generated by the network side (FRESH), the direction bit (DIRECTION) and the signalling data (MESSAGE). Based on these input parameters the user computes with the function f9 the message authentication code for data integrity (MAC-I) which is appended to the message when sent over the radio access link. The receiver computes XMAC-I on the messages received in the same way as the sender computed MAC-I on the message sent.

5.3.2 Use

The MAC function f9 shall be used to authenticate the data integrity and data origin of signalling data transmitted between UE and RNC.

5.3.3 Allocation

The MAC function f9 is allocated to the UE and the RNC.

Integrity protection shall be applied at the RRC layer.

5.3.4 Extent of standardisation

The function f9 is fully standardized.

5.3.5 Implementation and operational considerations

The algorithm should be designed to accommodate a range of implementation options including hardware and software implementations.

5.3.6 Type of algorithm

The function f9 shall be a MAC function.

5.3.7 Interface

5.3.7.1 IK

IK: the integrity key

IK[0], IK[1], …, IK[127]

The length of IK is 128 bits.

5.3.7.2 COUNT-I

COUNT-I: a frame dependent input.

COUNT-I[0], COUNT-I[1], …, COUNT-I[31]

The length of COUNT-I is 32 bits.

The input parameter COUNT-I protects against replay during a connection. It is a value incremented by one for each integrity protected message. COUNT-I consists of two parts: the HYPERFRAME NUMBER (HFN) as the most significant part and a RRC Sequence Number as the least significant part. The initial value of the hyperframe number is sent by the user to the network at connection set-up. The user stores the greatest used hyperframe number from the previous connection and increments it by one. In this way the user is assured that no COUNT-I value is re-used (by the network) with the same integrity key.

5.3.7.3 FRESH

FRESH: a random number generated by the RNC.

FRESH[0], FRESH[1], …, FRESH[31]

The length of FRESH is 32 bits.

The same integrity key may be used for several consecutive connections. This FRESH value is an input to the algorithm in order to assure the network side that the user is not replaying old MAC-Is.

5.3.7.4 MESSAGE

MESSAGE: the signalling data.

MESSAGE[0], MESSAGE[1], …, MESSAGE[X19-1]

The maximum length of MESSAGE is X19.

5.3.7.5 DIRECTION

DIRECTION: the direction of transmission of signalling messages (user to network or network to users).

DIRECTION[0]

The length of DIRECTION is 1 bit.

The same integrity key may be used for uplink and downlink channels simultaneously associated with a UE.

5.3.7.6 MAC-I (and equivalently XMAC-I)

MAC-I: the message authentication code for data integrity authentication

MAC-I[0], MAC-I[1], …, MAC-I[31]

The length of MAC-I is 32 bits.