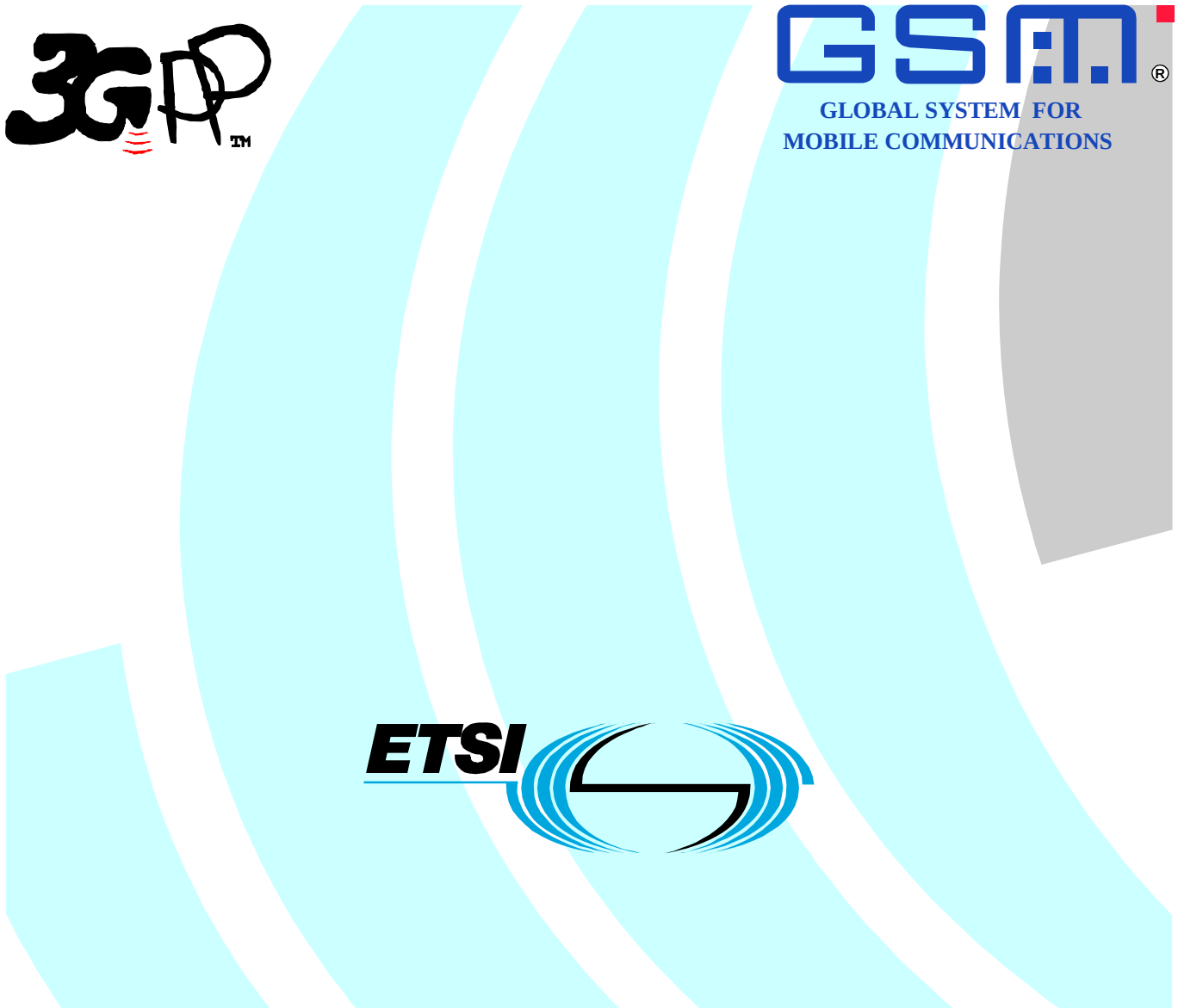
ETSI TS 133 401 V8.1.1 (2009-01)

Technical Specification

Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
3GPP System Architecture Evolution (SAE);
Security architecture
(3GPP TS 33.401 version 8.1.1 Release 8)



Reference
DTS/TSGS-0333401v811

Keywords
GSM, SECURITY, UMTS

ETSI

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

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

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

Intellectual Property Rights

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

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

Foreword

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

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

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

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document specifies the security architecture, i.e., the security features and the security mechanisms for the evolved packet system and the evolved packet core, and the security procedures performed within the evolved packet system (EPS) including the evolved packet core (EPC) and the evolved UTRAN (E-UTRAN)..

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 23.401: "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access".

[3] 3GPP TS 23.003: "Numbering, addressing and identification".

[4] 3GPP TS 33.102: "3G security; Security architecture".

[5] 3GPP TS 33.210: "3G security; Network Domain Security (NDS); IP network layer security".

[6] 3GPP TS 33.310: "Network Domain Security (NDS); Authentication Framework (AF)".

[7] IETF RFC 4303: "IP Encapsulating Security Payload (ESP)".

[8] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic bootstrapping architecture".

[9] 3GPP TS 24.301: "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3".

[10] IETF RFC 2104 (1997): "HMAC: Keyed-Hashing for Message Authentication".

[11] ISO/IEC 10118-3 (2004): "Information Technology - Security techniques - Hash-functions - Part 3: Dedicated hash-functions".

[12] 3GPP TS 36.323: "Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification"

[13] 3GPP TS 31.102: "Characteristics of the Universal Subscriber Identity Module (USIM) application".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1], in TS 33.102 [4] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Access Security Management Entity: entity which receives the top-level keys in an access network from the HSS. For E-UTRAN access networks, the role of the ASME is assumed by the MME

Authentication data: Security context and/or authentication vectors.

Cached EPS security context: A security context shared between a user and a serving network domain which was not deleted in EPS although the most recent registration is not in EPS.

Chaining of K_{eNB}: derivation of a new K_{eNB} from another K_{eNB} (i.e., at cell handover)

Current security context : The security context which has been taken into use by the network most recently.

EPS-Authentication Vector: $K_{ASME,}$ RAND, AUTN, XRES

EPS security context: A state that is established between a user and a serving network domain as a result of the execution of EPS AKA. At both ends "EPS security context data" is stored, that consists at least of the K_{ASME}, the key set identifier KSI_{ASME} and the NAS uplink and downlink COUNTs. After successful NAS SMC-execution also selected NAS algorithms are stored and the UE security capabilities.

Legacy security context: A security context which has been established according to TS 33.102 [4].

Mapped EPS security context: A security context that has been converted from a security context received from a system differently from EPS.**Re-derivation of NAS keys:** derivation of new NAS keys from the same K_{ASME} but including different algorithms (and no freshness parameter)

Refresh of K_{eNB}: derivation of a new K_{eNB} from the same K_{ASME} and including a freshness parameter

Re-keying of K_{eNB}: derivation of a new K_{eNB} from a new K_{ASME} (i.e., after an AKA has taken place)

Re-keying of NAS keys: derivation of new NAS keys from a new K_{ASME}

3.2 Symbols

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

|| Concatenation

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

AS	Access Stratum
ASME	Access Security Management Entity
CP	Control Plane
eNB	Evolved Node-B
EPC	Evolved Packet Core
EPS	Evolved Packet System
E-AV	EPS authentication vector
E-UTRAN	Evolved UTRAN
GUTI	Globally Unique Temporary Identity
MAC	Medium Access Control

MME	Mobility Management Entity
NAS	Non Access Stratum
PDCP	Packet Data Convergence Protocol
RAN	Radio Access Network
RRC	Radio Resource Control
SMC	Security Mode Command
S-TMSI	S-Temporary Mobile Subscriber Identity
UE	User Equipment
UP	User Plane

3.4 Conventions

All data variables in the present document are presented with the most significant substring on the left hand side and the least significant substring on the right hand side. A substring may be a bit, byte or other arbitrary length bitstring. Where a variable is broken down into a number of substrings, the leftmost (most significant) substring is numbered 0, the next most significant is numbered 1, and so on through to the least significant.

4 Overview of Security Architecture

Figure 4-1 gives an overview of the complete security architecture.

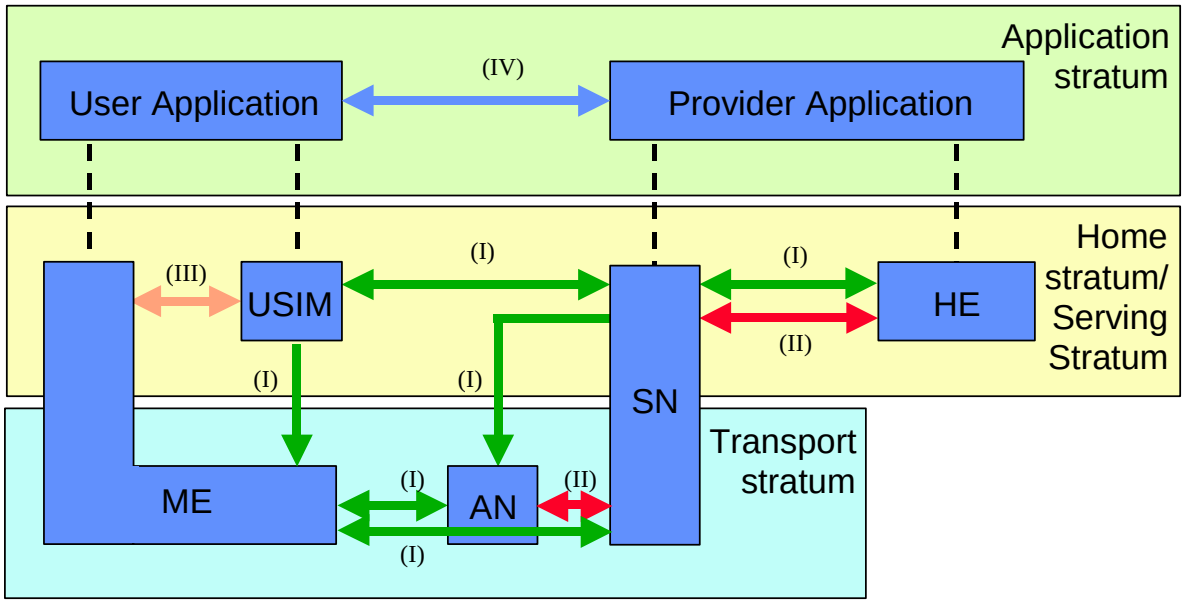


Figure 4-1: Overview of the security architecture

Five security feature groups are defined. Each of these feature groups meets certain threats and accomplishes certain security objectives:

- **Network access security (I):** the set of security features that provide users with secure access to services, and which in particular protect against attacks on the (radio) access link.
- **Network domain security (II):** the set of security features that enable nodes to securely exchange signalling data, and protect against attacks on the wireline network.
- **User domain security (III):** the set of security features that secure access to mobile stations.
- **Application domain security (IV):** the set of security features that enable applications in the user and in the provider domain to securely exchange messages.

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

5 Security Features

5.1 User-to-Network security

5.1.1 User identity and device confidentiality

User identity confidentiality is as defined by TS 33.102 [4] subclause 5.1.1

From subscriber's privacy point of view, the MSIN (also IMEI) should be confidentiality protected.

The UE shall provide its equipment identifier IMEI(SV) to the network, if the network asks for it.

The IMEI shall be securely stored in the terminal.

The UE shall not send IMEI(SV) to the network on a network request before the NAS security has been activated.

The IMEI(SV) shall be sent in the NAS protocol.

NOTE: In some cases, e.g., the very first attach procedure, MSIN has to be sent to network in cleartext. When NAS confidentiality protection is beyond an operator option, IMEI (SV) can not be confidentiality protected.

5.1.2 Entity authentication

Entity authentication is as defined by TS 33.102 [4] subclause 5.1.2

5.1.3 User data and signalling data confidentiality

5.1.3.1 Ciphering requirements

Ciphering may be provided to RRC-signalling to prevent UE tracking based on cell level measurement reports, handover message mapping, or cell level identity chaining. RRC signalling confidentiality is an operator option.

The NAS signalling may be confidentiality protected.

NOTE 1: RRC and NAS signalling confidentiality protection is recommended to be used.

Editor's NOTE: It is for ffs which of the NAS and RRC messages cannot be confidentiality protected.

User plane confidentiality protection shall be done at PDCP layer and is an operator option.

NOTE 2: User plane confidentiality protection is recommended to be used.

NOTE 3: Confidentiality protection for RRC and UP is applied at the PDCP layer, and no layers below PDCP are confidentiality protected. Confidentiality protection for NAS is provided by the NAS protocol.

5.1.3.2 Algorithm Identifier Values

All algorithms specified in this subclause are algorithms with a 128-bit input key.

NOTE: Deviations from the above requirement have to be indicated explicitly in the algorithm identifier list below.

Each EPS Encryption Algorithm (EEA) will be assigned a 4-bit identifier. Currently, the following values have been defined for NAS, RRC and UP ciphering:

"0000_2" 128-EEA0 Null ciphering algorithm

"0001_2" 128-EEA1 SNOW 3G

"0010_2" 128-EEA2 AES

The remaining values have been reserved for future use.

UEs and eNBs shall implement 128-EEA0,128-EEA1 and 128-EEA2 for both RRC signalling ciphering and UP ciphering.

UEs and MMEs shall implement 128-EEA0,128-EEA1 and 128-EEA2 for NAS signalling ciphering.

Editor's Note: The modes of operation of Snow 3G and AES need FFS

5.1.4 User data and signalling data integrity

5.1.4.1 Integrity requirements

Integrity protection, and replay protection, shall be provided to NAS and RRC-signalling.

User plane packets between the eNB and the UE shall not be integrity protected.

NOTE: Integrity protection for RRC is applied at the PDCP layer, and no layers below PDCP are integrity protected. Integrity protection for NAS is provided by the NAS protocol.

5.1.4.2 Algorithm Identifier Values

All algorithms specified in this subclause are algorithms with a 128-bit input key.

NOTE: Deviations from the above requirement have to be indicated explicitly in the algorithm identifier list below.

Each EPS Integrity Algorithm (EIA) will be assigned a 4-bit identifier. Currently, the following values have been defined:

"0001_2" 128-EIA1 SNOW 3G

"0010_2" 128-EIA2 AES

The remaining values have been reserved for future use.

UEs and eNBs shall implement 128-EIA1 and 128-EIA2 for RRC signalling integrity protection.

UEs and MMEs shall implement 128-EIA1 and 128-EIA2 for NAS signalling integrity protection.

Editor's Note: The modes of operation of Snow 3G and AES need FFS

5.2 Security visibility and configurability

Although in general the security features should be transparent to the user, for certain events and according to the user's concern, greater user visibility of the operation of following security feature shall be provided:

- indication of access network encryption: the property that the user is informed whether the confidentiality of user data is protected on the radio access link, in particular when non-ciphered calls are set-up;

Configurability is the property that the user can configure whether the use or the provision of a service should depend on whether a security feature is in operation. A service can only be used if all security features, which are relevant to that service and which are required by the configurations of the user, are in operation. The following configurability features are suggested:

- enabling/disabling user-USIM authentication: the user should be able to control the operation of user-USIM authentication, e.g., for some events, services or use.

5.3 Security requirements on eNodeB

Editor's Note: the suitable location for these requirements and, possibly, implementation guidelines, is ffs.

5.3.1 Requirements for eNB setup and configuration

Setting up and configuring eNBs shall be authenticated and authorized so that attackers shall not be able to modify the eNB settings and software configurations via local or remote access.

1. Security associations are required between the EPS core and the eNB and between adjacent eNBs, connected via X2. These security association establishments shall be mutually authenticated and used for communication between the entities.
2. Communication between the remote/local O&M systems and the eNB shall be mutually authenticated.
3. The eNB shall be able to ensure that software/data change attempts are authorized
4. The eNB shall use authorized data/software.
5. Sensitive parts of the boot-up process shall be executed with the help of the secure environment.
6. Confidentiality of software transfer towards the eNB shall be ensured.

Editor's Note: The definition of secure environment is FFS

5.3.2 Requirements for key management inside eNB

The EPS core network provides subscriber specific session keying material for the eNBs, which also hold long term keys used for authentication and security association setup purposes. Protecting all these keys is important.

1. Keys stored inside eNBs shall never leave a secure environment within the eNB except when done in accordance with this or other 3GPP specifications.

5.3.3 Requirements for handling User plane data within the eNB

It is eNB's task to cipher and decipher user plane packets between the Uu reference pointand the S1/X2 reference points.

1. User plane data ciphering/deciphering shall take place inside the secure environment where the related keys are stored.
2. The transport of user data over S1-U and X2-U shall be integrity, confidentially and replay-protected from unauthorized parties. If this is to be accomplished by cryptographic means, clause 12 shall be applied.

NOTE: The use of cryptographic protection on S1-U and X2-U is an operator's decision. In case the eNB has been placed in a physically secured environment then the 'secure environment' may include other nodes and links beside the eNB.

Editor's Note: SA3 aims for a single set of high level security requirements for all types of eNodeB (i.e. femto, pico and macro eNB). However, SA3 recognizes that different deployment environments dictate that different security solutions are needed to meet these requirements. SA3 has not yet agreed whether the requirements on the solutions for different deployment environments will be documented by 3GPP.

5.4 Other security features

Editor's Note: in TS 33.102, section covers other topics, such as User domain security (e.g. user-UICC interaction) and application security (e.g. USIM toolkit). Currently, nothing new is expected here compared to UMTS, so it is ffs whether to include corresponding material here. Maybe a reference would be useful here?

6 Security Procedures between UE and EPC Network Elements

Editor's Note: the core network elements under consideration in this section are ASME and HSS including Authentication Centre and, if applicable, AAA server.

Editor's Note: by definition, the ASME for E-UTRAN is the MME. Security procedures involving the MME, but no other core network elements, are specified in section 7.

6.1 Authentication and key agreement

6.1.1 AKA procedure

EPS AKA is the authentication and key agreement procedure that shall be used over E-UTRAN.

A Rel-99 or later USIM application on a UICC shall be sufficient for accessing E-UTRAN. Access to E-UTRAN with a 2G SIM or a SIM application on a UICC shall not be granted.

An ME that has E-UTRAN radio capability shall support the USIM-ME interface as specified in TS 31.102 [8]

EPS AKA shall produce keying material forming a basis for user plane (UP), RRC, and NAS ciphering keys as well as RRC and NAS integrity protection keys.

NOTE 1: Key derivation requirements of AS and NAS keys can be found in subclause 7.2.1

During the authentication, the USIM shall verify the freshness of the authentication vector that is used. The MME sends to the USIM via ME the random challenge RAND and an authentication token AUTN for network authentication from the selected authentication vector. At receipt of this message, the USIM shall verify whether AUTN can be accepted and if so, produces a response RES. USIM shall compute CK and IK.

An ME accessing E-UTRAN shall check during authentication that the "separation bit" in the AMF field of AUTN is set to 1 and reject authentication otherwise with a CAUSE value. The "separation bit" is bit 0 of the AMF field of AUTN.

UE shall compute K_{ASME} from CK, IK, and serving network's identity (SN id) using the KDF as specified in Annex A. SN id binding implicitly authenticates the serving network's identity when the derived keys from K_{ASME} are successfully used.

NOTE 2: This separation bit in the AMF can not be used anymore for operator specific purposes as described by TS 33.102 [4], Annex F

NOTE 3: The HSS needs to ensure that the MME requesting the authentication data is entitled to use the SN id used to calculate K_{ASME}. The exact details of how to achieve this are not covered in this specification.

The UE shall store CK, IK resulting from a run of EPS AKA in a non-volatile part of the ME memory. The UE shall not store CK, IK resulting from a run of EPS AKA on the USIM.

NOTE 4: It is not possible to store the EPS security context on the USIM because the required fields, e.g. for the NAS COUNT values, cf. clause 7.2, are not available. Furthermore, if the keys CK, IK resulting from an EPS AKA run were stored in the fields already available on the USIM for storing keys this could lead to overwriting keys resulting from an earlier run of UMTS AKA. This would lead to problems when EPS security context and UMTS security context were held simultaneously (as is the case when security context is cached e.g. for the purposes of Idle Mode Signaling Reduction). Therefore, 'plastic roaming' where a UICC is inserted into another ME will necessitate an EPS AKA authentication run. This is in contrast to UMTS where the security context stored in the USIM may be used in plastic roaming to avoid another UMTS AKA run.

UE shall respond with User authentication response message including RES in case of successful AUTN verification as described in TS 33.102[4] and successful AMF verification as described above. Otherwise UE shall send User authentication reject message with a proper CAUSE value.

Figure 6.1.1-1 describes EPS AKA procedure, which is based on UMTS AKA (see TS 33.102[4]). The following keys are shared between UE and HSS:

- **K** is the permanent key stored on the USIM on a UICC and in the Authentication Centre AuC.
- **CK, IK** is the pair of keys derived in the AuC and on the USIM during an AKA run. CK, IK shall be handled differently depending on whether they are used in an EPS security context or a legacy security context, as described in subclause 6.1.2.

As a result of the authentication and key agreement, an intermediate key $\mathbf{K_{ASME}}$ shall be generated which is shared between UE and ASME. How this is done is described in subclause 6.1.2.

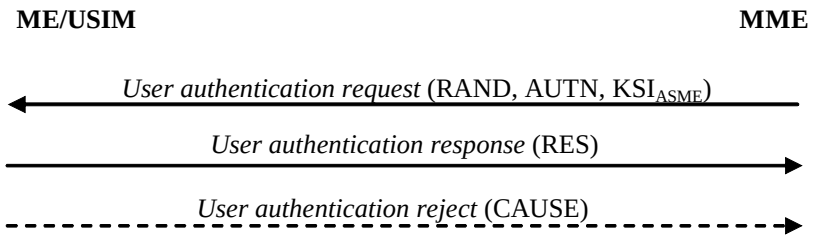


Figure 6.1.1-1: EPS user authentication (EPS AKA)

6.1.2 Distribution of authentication data from HSS to serving network

The purpose of this procedure is to provide the MME with one or more EPS authentication vectors (RAND, AUTN, XRES, K_{ASME}) from the user's HE (HSS) to perform a number of user authentications.

NOTE: It is recommended that the MME fetch only one EPS authentication vector at a time as the need to perform AKA runs has been reduced in EPS through the use of a more elaborate key hierarchy. In particular, service requests can be authenticated using a stored K_{ASME} without the need to perform AKA. Furthermore, the sequence number management schemes in TS 33.102, Annex C [4], designed to avoid re-synchronisation problems caused by interleaving use of batches of authentication vectors, are only optional. Re-synchronisation problems in EPS can be avoided, independently of the sequence number management scheme, by immediately using an authentication vector retrieved from the HSS in an authentication procedure between UE and MME.

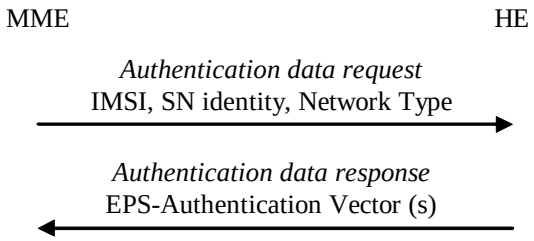


Figure 6.1.2-1: Distribution of authentication data from HE to MME

An EPS authentication vector is derived from the authentication vector defined in TS 33.102 [4] clause 6.3.2. To derive the key K_{ASME} in the HE, the KDF as specified in Annex A is used which shall contain following mandatory input parameters: CK, IK and SN identity.

If the Network Type equals E-UTRAN then the "separation bit" in the AMF field of AUTN shall be set to 1 to indicate to the UE that the authentication vector is only usable for AKA in an EPS context, if the "separation bit" is set to 0, the vector is usable in a non-EPS context only (e.g. GSM, UMTS). For authentication vectors with the "separation bit" set to 1, the secret keys CK and IK generated during AKA shall never leave the HSS.

The MME invokes the procedures by requesting authentication vectors from the HE (Home environment).

The *authentication data request* shall include the IMSI, the Serving Network identity i.e. MCC + MNC, and the Network Type (I.e. E-UTRAN)

Upon the receipt of the *authentication data request* from the MME, the HE may have pre-computed the required number of EPS authentication vectors and retrieve them from the HSS database or may compute them on demand.

NOTE: For K_{ASME} the possibilities for pre-computation are restricted due to the PLMN-binding.

The HE sends an authentication response back to the MME that contains the requested information. If multiple EPS authentication vectors had been requested then they are ordered based on their sequence numbers.

6.1.3 User identification by a permanent identity

The user identification mechanism should be invoked by the serving network whenever the user cannot be identified by means of a temporary identity (GUTI). In particular, it should be used when the serving network cannot retrieve the IMSI based on the GUTI by which the user identifies itself on the radio path.

The mechanism described in figure 6.1.3-1 allows the identification of a user on the radio path by means of the permanent subscriber identity (IMSI).

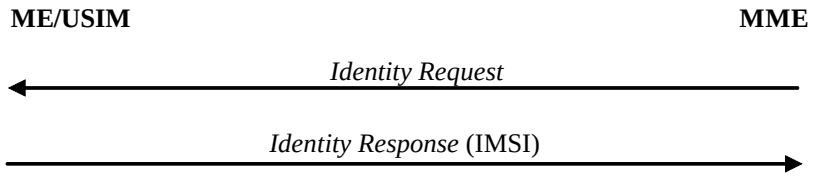


Figure 6.1.3-1: User identity query

The mechanism is initiated by the MME that requests the user to send its permanent identity. The user's response contains the IMSI in cleartext. This represents a breach in the provision of user identity confidentiality.

6.1.4 Distribution of IMSI and authentication data within one serving network domain

The purpose of this procedure is to provide a newly visited MME with authentication data from a previously visited MME within the same serving network domain.

The procedure is shown in Figure 6.1.4-1

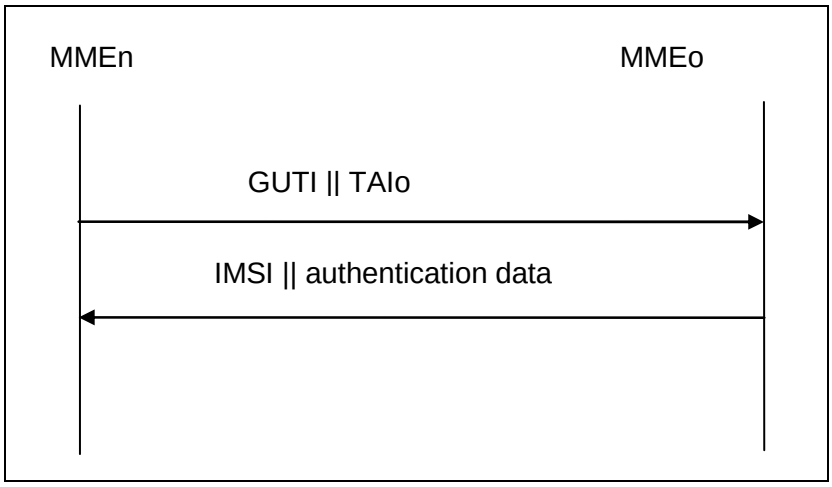


Figure 6.1.4-1: Distribution of IMSI and authentication data within one serving domain

The procedure shall be invoked by the newly visited MMEn after the receipt of a Tracking Area update request from the user wherein the user is identified by means of a temporary user identity GUTIo and the Tracking area identity TAIo under the jurisdiction of a previously visited MMEo that belongs to the same serving network domain as the newly visited MMEn.

The protocol steps are as follows:

a) The MMEn sends a *user identity request* to the MMEo, this message contains GUTIo and TAIo.

b) The MMEo searches the user data in the database.

   If the user is found, the MMEo shall send a *user identity response* back that:

   i) shall include the IMSI,

   ii) may include a number of unused EPS-authentication vectors ordered on a first-in / first-out basis, and

   iii) may include the current EPS security context data

   The MMEo subsequently deletes the EPS-authentication vectors which have been sent and the data elements on the EPS current security context.

   If the user cannot be identified the MMEo shall send a *user identity response* indicating that the user identity cannot be retrieved.

c) If the MMEn receives a *user identity response* with an IMSI, it creates an entry and stores any EPS-authentication vectors and any data on the current EPS security context that may be included.

   If the MMEn receives a *user identity response* indicating that the user could not be identified, it shall initiate the user identification procedure described in clause 6.1.3.

6.1.5 Distribution and use of authentication vectors between different serving network domains

The distribution of authentication data (unused authentication vectors and/or current security context data) between MME's of the same service network domain is described according to subclause 6.1.4.

The following three cases are distinguished related to the distribution of authentication vectors between SGSNs and MME's:

a) MME to MME

   EPS authentication vectors shall not be distributed between MME's belonging to different serving domains (PLMNs)

   UMTS authentication vectors may be forwarded between MME's.

   An MME should not use an UMTS authentication vector.

   NOTE 1: The impossibility to use is due to the fact that the separation bit in the AMF-field may be set for EPS access.

b) SGSN to MME

   An MME should not use (Cfr NOTE 1 above) forwarded UMTS authentication vectors from an SGSN, but may forward them back to the SGSN later.

c) MME to SGSN

   UMTS AVs which were stored in the MME can be forwarded from an MME towards an SGSN.

   EPS authentication vectors shall not be forwarded from an MME towards an SGSN.

NOTE 2: This is due to the fact that in an EPS-AV the CK and IK are not available for the MME and hence also not for the SGSN when an EPS-AV would be forwarded.

6.2 EPS key hierarchy

Requirements on EPC and E-UTRAN related to keys:

a) The EPC and E-UTRAN shall allow for use of encryption and integrity protection algorithms for AS and NAS protection having keys of length 128 and for future use the network interfaces shall be prepared to support 256 bit keys.

b) The keys used for UP, NAS and AS protection shall be dependent on the algorithm with which they are used.

c) As part of the initial attach request from the UE, ME shall signal security capabilities to the MME, i.e. the ME supported EPS key derivation algorithms, integrity protection algorithms and encryption algorithms.

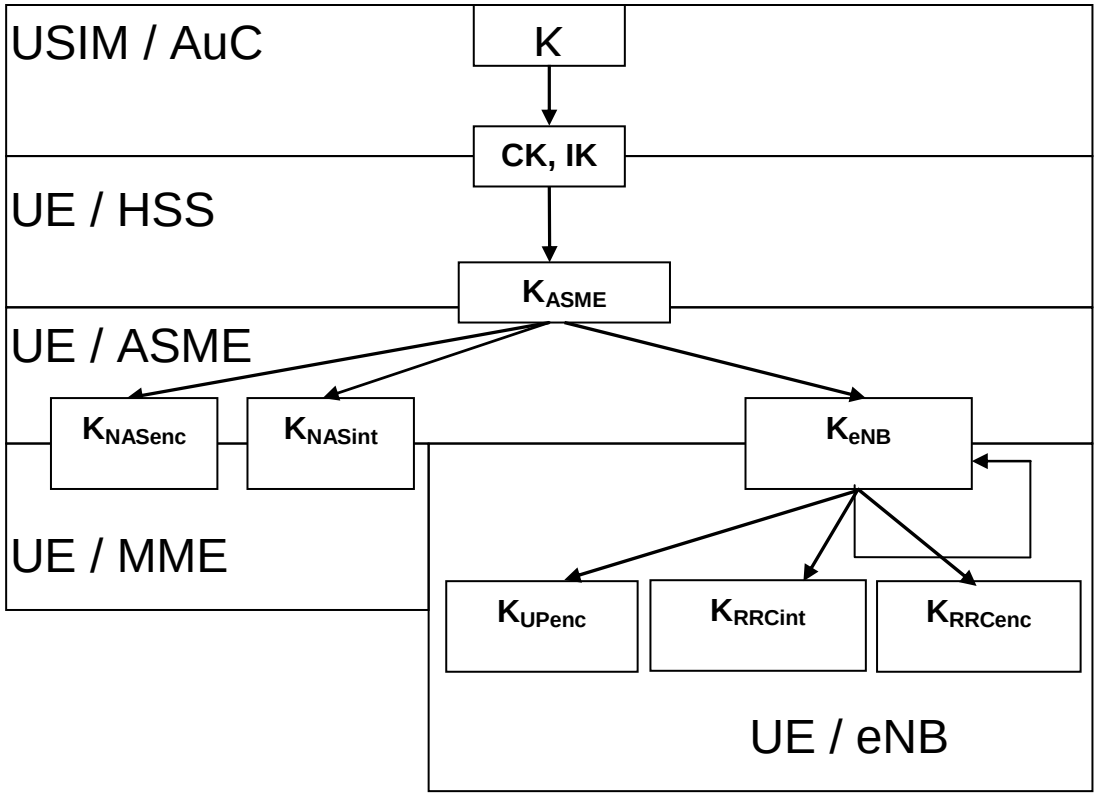


Figure 6.2-1: Key hierarchy in E-UTRAN

The key hierarchy (see Figure 6.2-1) includes following keys: $\mathbf{K_{eNB}}$, $\mathbf{K_{NASint}}$, $\mathbf{K_{NASenc}}$, $\mathbf{K_{UPenc}}$, $\mathbf{K_{RRCint}}$ and $\mathbf{K_{RRCenc}}$

- $\mathbf{K_{eNB}}$ is a key derived by UE and MME from K_{ASME} when the UE goes to ECM-CONNECTED state or by UE and target eNB during eNB handover.

Keys for NAS traffic:

- $\mathbf{K_{NASint}}$ is a key, which shall only be used for the protection of NAS traffic with a particular integrity algorithm This key is derived by UE and MME from K_{ASME} , as well as an identifier for the integrity algorithm using the KDF as specified in Annex A.

- $\mathbf{K_{NASenc}}$ is a key, which shall only be used for the protection of NAS traffic with a particular encryption algorithm. This key is derived by UE and MME from $K_{ASME,}$,as well as an identifier for the encryption algorithm using the KDF as specified in Annex A.

Keys for UP traffic:

- $\mathbf{K_{UPenc}}$ is a key, which shall only be used for the protection of UP traffic with a particular encryption algorithm. This key is derived by UE and eNB from K_{eNB}, as well as an identifier for the encryption algorithm using the KDF as specified in Annex A.

Keys for RRC traffic:

- $\mathbf{K_{RRCint}}$ is a key, which shall only be used for the protection of RRC traffic with a particular integrity algorithm. $\mathbf{K_{RRCint}}$ is derived by UE and eNB from K_{eNB}, as well as an identifier for the integrity algorithm using the KDF as specified in Annex A.

- **K_{RRCenc}** is a key, which shall only be used for the protection of RRC traffic with a particular encryption algorithm. K_{RRCenc} is derived by UE and eNB from K_{eNB} as well as an identifier for the encryption algorithm using the KDF as specified in Annex A.

Figure 6.2-2 shows the dependencies between the different keys, and how they are derived from the network nodes point of view. Figure 6.2-3 shows the corresponding relations and derivations as performed in the ME.

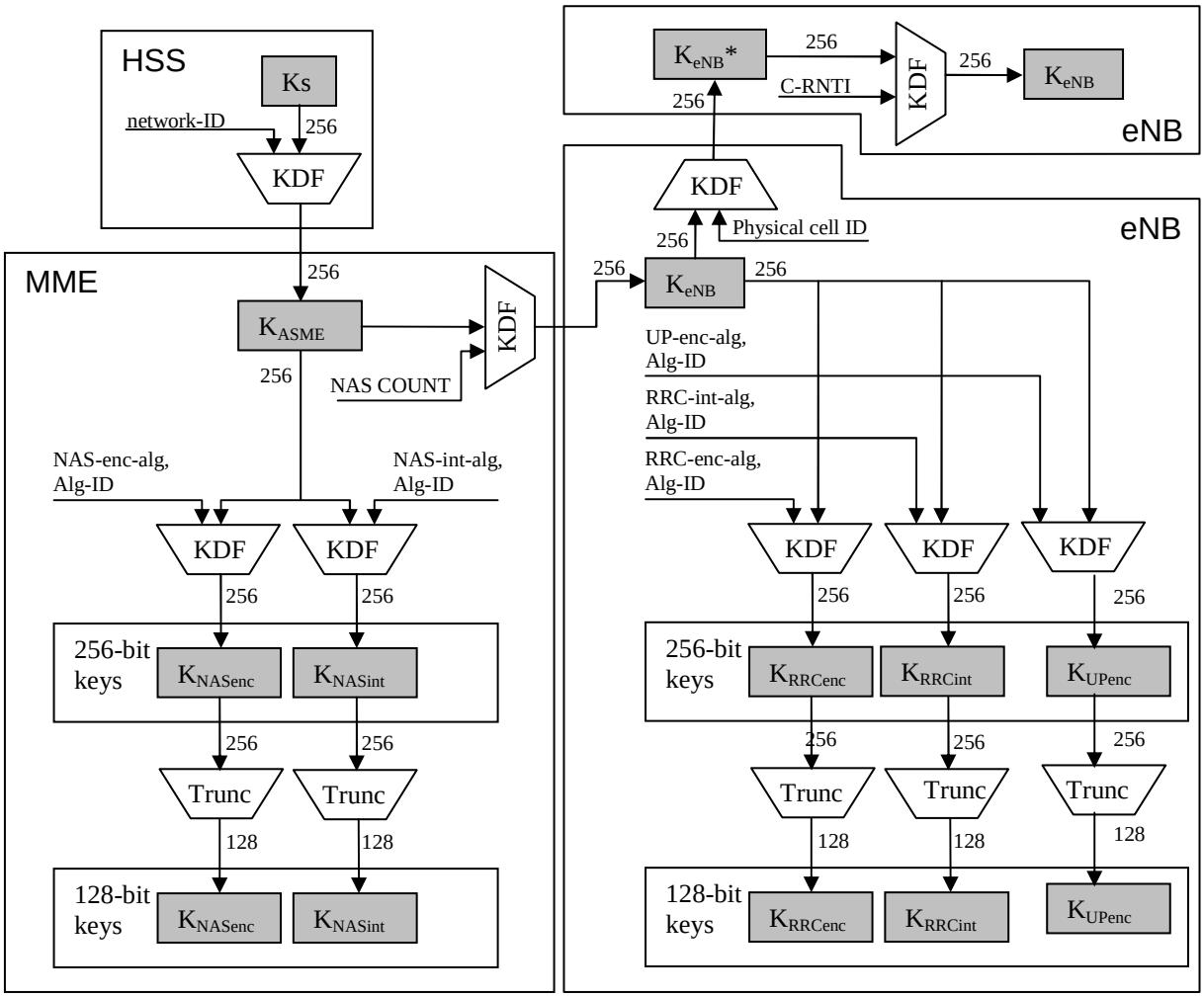


Figure 6.2-2: Key distribution and key derivation scheme for EPS (in particular E-UTRAN) for network nodes. The basic derivations are covered in the figure, but derivations performed at, e.g. inter-RAT mobility is not shown

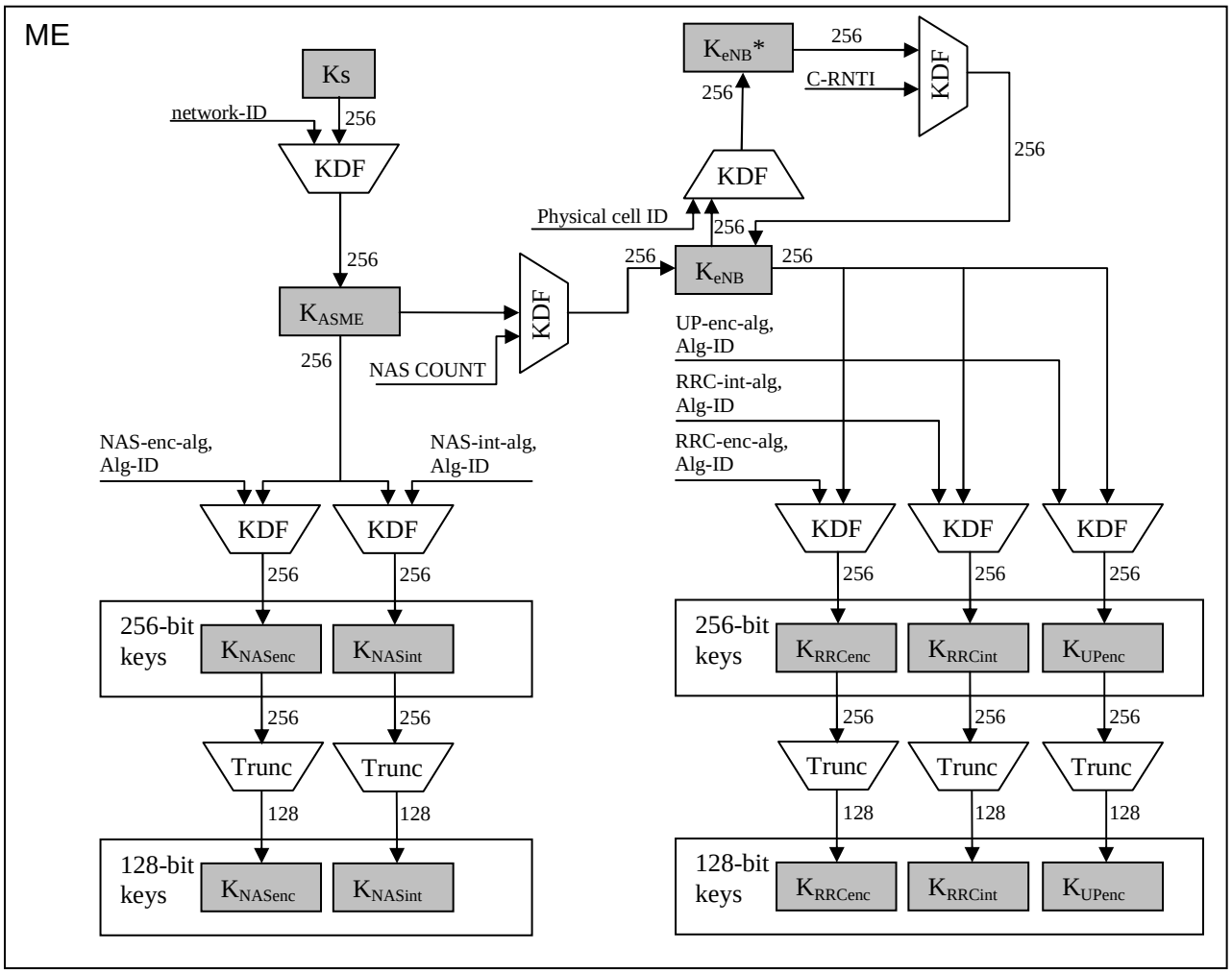


Figure 6.2-3: Key derivation scheme for EPS (in particular E-UTRAN) for the ME. The basic derivations are covered in the figure, but derivations performed at, e.g. inter-RAT mobility is not shown

As the figures 6.2-2 and 6.2-3 show, the length of K_{ASME} and K_{eNB} is 256 bits, and 256-bit NAS, UP and RRC keys are always derived from K_{ASME} and K_{eNB} respectively. In case the encryption or integrity algorithm used to protect NAS, UP or RRC requires a 128-bit key as input, the key is truncated and the 128 least significant bits are used.

The function Trunc takes as input a 256-bit string, and returns the 128 least significant bits of that string as output. The input Ks, in the derivation of the K_{ASME} , is for a Rel8 or earlier USIMs, the concatenation of CK and IK.

6.3 EPS key identification

The EPS ASME key identifier (KSI_{ASME}) is a number which is associated with the K_{ASME} derived during authentication. The key set identifier is allocated by the network and sent with the authentication request message to the mobile station where it is stored together with the K_{ASME}.

The identifier of EPS key originated from SGSN (KSI_{SGSN}) is a number which is associated with the mapped K_{ASME} that derived from the SGSN keys during inter-RAT mobility. The KSI_{SGSN} is stored together with the mapped K_{ASME}.

The purpose of the KSI_{ASME} is to make it possible for the network to identify the K_{ASME} which is stored in the mobile station without invoking the authentication procedure. This is used to allow re-use of the K_{ASME} during subsequent connection set-ups.

KSI_{ASME} and KSI_{SGSN} have the same format. KSI_{ASME} is three bits. Seven values are used to identify the key set. A value of '111' is used by the mobile station to indicate that a valid K_{ASME} is not available for use. At deletion of the K_{ASME}, the KSI_{ASME} is set to '111'. The value '111' in the other direction from network to mobile station is reserved.

Editor's Note: It is FFS whether another KSI is needed for NAS keys due to the key refresh procedures.

6.4 EPS key properties

If K_{ASME} is stored in the ME, it should be deleted when the ME is powered down or when the UICC is removed.

K_{ASME} shall never leave the EPC.

7 Security Procedures between UE and EPC Access Network Elements

7.1 Mechanism for user identity confidentiality

The MME shall allocate a GUTI to a user in order to support the subscriber identity confidentiality. S-TMSI, the shortened form of the GUTI, is used to support the subscriber identity confidentiality with more efficient radio signalling procedures (e.g. paging and Service Request). The structure of the GUTI is specified in TS 23.401 [2]. The GUTI allocation procedure should be performed after the initiation of NAS ciphering.

GUTI Reallocation procedure is described in TS 23.401 [2].

7.2 Handling of user-related keys in E-UTRAN

7.2.1 E-UTRAN key setting during AKA

Authentication and key setting are triggered by the authentication procedure. Authentication and key setting may be initiated by the network as often as the network operator wishes. Key setting can occur as soon as the identity of the mobile subscriber (i.e. GUTI or IMSI) is known by the MME. Key K_{ASME} is stored in the MME and key K_{eNB} is derived using the KDF as specified in Annex A from the key K_{ASME} and transferred to the UE's serving eNB when needed. K_{ASME} is stored in the ME and MME and updated with the next authentication procedure.

The RRC and UP keys are derived from the K_{eNB} using the KDF as specified in Annex A when needed.

If an authentication procedure is performed during a connection, the new K_{ASME}, NAS, RRC and UP keys shall be taken in use in both the eNB and the ME as part of the security mode set-up procedure (see subclause 7.2.4).

7.2.2 E-UTRAN key identification

Editor's Note: cf. TS 33.102, sections 6.4.1 and TR 33.821, section 7.4.6

7.2.3 E-UTRAN key lifetimes

All E-UTRAN keys are derived based on a K_{ASME}. The key hierarchy does not allow, as is, explicit key updates, but fresh RRC and UP keys are derived based on a fresh $K_{eNB,}$ which is bound to certain dynamic parameters (like PCI) and fresh key derivation parameter(s) in state transitions (like NAS uplink COUNT). This results as fresh RRC and UP keys in the eNB between inter-eNB handovers and state transitions (see subclauses 7.2.6 to 7.2.8). The K_{eNB} shall be deleted in the eNB while UE is in ECM-IDLE state.

If RRC keys are corrupted (e.g. RRC level deciphering and or integrity protection check fails repeatedly on the receiver side beyond some retransmission threshold, keys are missing in UE/eNB, contained bit errors, etc.) UE will have to restart radio level attachment procedure (e.g. similar radio level procedure to ECM-IDLE to ECM-CONNECTEDtransition or initial attachment).

In case K_{ASME} is invalid, a KSI_{ASME} with value "111" shall be sent to the network, which then can initiate (re-)authentication procedure to get a new K_{ASME} based on a successful AKA authentication.

7.2.4 Security mode command procedure and algorithm negotiation

7.2.4.1 Requirements for algorithm selection

a) An active UE and a serving network shall agree upon algorithms for

- RRC ciphering and RRC integrity protection (to be used between UE and eNB)
- UP ciphering (to be used between UE and eNB)
- NAS ciphering and NAS integrity protection (to be used between UE and MME)
- KDF (key derivation function for the EPS key hierarchy)

Editor's note: The following issues must be studied w.r.t. KDF negoatiation:

Which ones of the following KDFs should be negotiable (if any):

HSS-KDF, MME-KDF, eNB-KDF

If an eNB-KDF is negotiated, how are the handover cases solved when two different eNBs uses different eNB-KDFs.

If it can be shown that KDF negotiation can be introduced in Rel-9 and there is not time to introduce it in Rel-8, then it may be introduced in Rel-9 instead (then it would be removed from Rel-8),

b) The serving network shall select the algorithms to use dependent on

- the UE security capabilities of the UE,
- the UE security capabilities of the currently serving network entity

c) UE Security capabilities shall include the supported ciphering and integrity protection algorithms for E-UTRAN, but also for UTRAN/GERAN if supported by the UE, as well as the supported KDFs for EPS key hierarchy derivations. The same set of ciphering and integrity algorithms shall be supported by the UE both for AS and NAS level.

d) Each selected algorithm shall be acknowledged to the UE in an integrity protected way such that the UE is ensured that the algorithm selection was not manipulated ("bidding down protection of networks choice") that the UE security capabilities were not bidden down.

e) The UE security capabilities the ME sent to the network shall be repeated in an integrity protected NAS level message to the ME such that "bidding down attacks" against the UE's security capabilities can be detected by the ME. The UE security capabilities apply to both AS and NAS level security.

f) Separate AS and NAS level security mode command procedures are required. AS level security mode command procedure configures AS security (RRC and UP) and NAS level security mode command procedure configures NAS security.

   a. Both integrity protection and ciphering for RRC are activated within the same AS SMC procedure, but not necessarily within the same message.

   b. User plane ciphering is activated at the same time as RRC ciphering.

g) It shall be possible that the selected AS and NAS algorithms are different at a given point of time.

7.2.4.2 Procedures for AS algorithm selection

7.2.4.2.1 Initial AS security context establishment

Each eNB shall be configured via network management with lists of algorithms which are allowed for usage. There shall be one list for integrity algorithms, and one for ciphering algorithms. These lists shall be ordered according to a priority decided by the operator. When AS security context is established in the eNB, the MME shall send the UE's security capabilities to the eNB, which contains the algorithms supported by the UE. The eNB shall choose the

ciphering algorithm wich has the highest priority from its configured list and is also present in the UE's security capabilities. The eNB shall choose the integrity algorithm wich has the highest priority from its configured list and is also present in the UE's security capabilities. The chosen algorithms shall be indicated to the UE in the AS SMC. The ciphering algorithm is used for ciphering of the user plane and RRC traffic. The integrity algorihtm is used for integrity protection of the RRC traffic.

7.2.4.2.2 X2-handover

At handover from a source eNB over X2 to a target eNB, the source eNB shall include the UE security capabilities in the handover request message. The target eNB shall select the algorithm with highest priority from the UE security capabilities according to the prioritized locally configured list of algorithms (this applies for both integrity and ciphering algorithms). In the path-switch message, the target eNB shall send the UE security capabilities received from the source eNB to the MME. The MME shall verify that the UE security capabilities received from the eNB are the same as the UE security capabilities that the MME has stored. If there is a mismatch, the MME may log the event and may take additional measures, such as raising an alarm.

7.2.4.2.3 S1-handover

At handover from a source eNB to a target eNB over S1 (possible including an MME change), the source eNB shall forward the UE security capabilities to the target eNB in the transparent container which is sent in the handover required and handover request S1-AP messages. The target eNB shall select the algorithm with highest priority from the UE security capabilities according to the prioritized locally configured list of algorithms (this applies for both integrity and ciphering algorithms). In the handover notify message, the target eNB shall send the UE security capabilities received from the source eNB to the MME. The MME shall verify that the UE security capabilities received from the eNB are the same as the UE security capabilities that the MME has stored. If there is a mismatch, the MME may log the event and may take additional measures, such as raising an alarm.

7.2.4.3 Procedures for NAS algorithm selection

7.2.4.3.1 Initial NAS security context establishment

When the NAS security context is established, e.g., by a TAU Accept, Attach Accept or by NAS SMC message, the MME shall choose one NAS ciphering algorithm, one NAS integrity protection algorithm, and one KDF, and indicate them in the corresponding integrity protected message to UE and shall also include the UE security capabilities into that message. UE shall reply with an integrity protected NAS message, protected by the integrity algorithm selected by the MME in case the algorithms or the KDF change from the previous TAU procedure or if the NAS message (e.g., TAU Request or Attach Request) that carried the UE security capabilities to the MME was not protected. This NAS message shall contain the UE security capabilities so that the MME can verify that the UE security capabilities are the same as the ones sent by the UE in the unprotected initial NAS message. This enables detection of attacks where an attacker has modified the UE security capabilities in the initial NAS message. The MME shall select the NAS algorithms which have the highest priority according to an ordered list which can be configured in the MME.

Editor's NOTE: The following was obserced at SA3#52 output document handling, and needs further study. In case the MME runs a NAS SMC procedure as part of the TAU/Attach procedure, the NAS SMC will echo back the UE security capabilities to the UE, and the fact that the UE sends any integrity protected message to the MME indicates to the MME that the UE has verified that the UE security capabilities has not been tampered with. This would have the same properties of as the UMTS SMC procedure. The text, as currently written reverses the replay-mechanism for UE security capabilities compared to UMTS, and forces the use of the TAU/Attach COMPLETE message unnecessarily,

7.2.4.3.2 MME change

In case there is a change of MMEs, the target MME shall indicate in the TAU accept message which integrity and ciphering algorithm is selected for NAS protection as well which KDF to use from now on in case they are different from the previously selected algorithms and KDF for NAS. In this case the TAU Accept message shall also include the UE security capabilities from the TAU Request so that UE can be sure that the MME had the right set of UE security capabilities to base its decision on. UE shall reply with an integrity protected NAS message containing the UE security capabilities in case the algorithms or the KDF change from the previous TAU procedure or if the TAU Request was not protected. The MME shall select the NAS algorithms which have the highest priority according to an ordered list which can be configured in the MME.

NOTE: After an S1-handover with MME change a TAU procedure is executed. The same is true for an inter-RAT handover to E-UTRAN and for both inter- and intra-RAT idle mode mobility resulting in a change of MMEs.

7.2.4.4 NAS security mode command procedure

The NAS SMC procedure consists of a roundtrip of messages between MME and UE. The MME sends the NAS security mode command to the UE and the UE replies with the NAS security mode complete message.

The NAS security mode command message from MME to UE contains the replayed UE security capabilities of the UE (including the security capabilities with respect to NAS, RRC and UP ciphering as well as integrity, and other target network security capabilities i.e. UTRAN/GERAN if UE included them in the message to MME), the selected NAS algorithms, and the KSI_{ASME} for identifying K_{ASME}. This message is integrity protected with NAS integrity key based on K_{ASME} indicated by the KSI_{ASME} in the message. See figure 7.2.4.4-1.

UE verifies the integrity of the NAS security mode command message. If successfully verified, UE starts NAS integrity protection and ciphering/deciphering and sends the NAS security mode complete message to MME ciphered and integrity protected with the selected NAS algorithm indicated in the NAS security mode command message and NAS keys based on K_{ASME} indicated by the KSI_{ASME} in the NAS security mode command message.

NAS uplink and downlink ciphering at the MME starts after sending the NAS security mode command message. NAS uplink and downlink ciphering at the UE starts after receiving the NAS security mode command message. The NAS security mode complete message includes IMEI in case MME requested it in the NAS SMC Command message.

If any verification of the NAS security mode command is not successful, the procedure ends in the ME [see TS 33.102 section 6.4.5] and ME shall not send NAS security mode complete message.

Only after EPS AKA the NAS security mode command message resets NAS uplink and downlink COUNT values. Both the NAS security mode command and NAS security mode complete messages are protected based on reset COUNT values (zero). NAS SMC always changes the NAS keys (i.e. due to EPS AKA with new K_{ASME} and KSI_{ASME} or due to the algorithms change).

UE shall add the UE security capabilities to the NAS Security Mode Complete message so that in case the UE security capabilities were sent unprotected to the network the MME can verify that they were not modified.

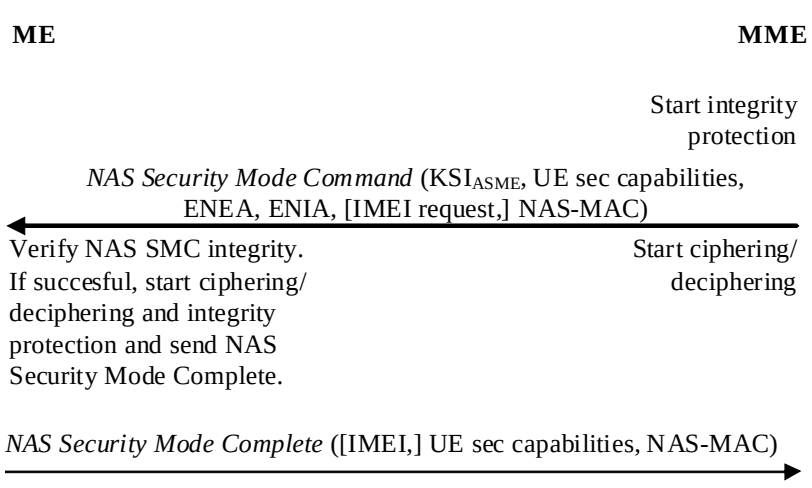


Figure 7.2.4.4-1: NAS security mode command procedure

7.2.4.5 AS security mode command procedure

The AS SMC procedure consists of a roundtrip of messages between eNB and UE. The eNB sends the AS security mode command to the UE and the UE replies with the AS security mode complete message. See figure 7.4.2.3-1.

The AS security mode command message from eNB to UE shall contain the selected AS algorithms and the KSI_{ASME} for K_{ASME}. This message shall be integrity protected with RRC integrity key based on K_{ASME} indicated by the KSI_{ASME}.

The AS security mode complete message from UE to eNB shall be integrity protected with the selected RRC algorithm indicated in the AS security mode command message and RRC integrity key based on K_{ASME} indicated by the KSI_{ASME}.

RRC and UP downlink ciphering (encryption) at the eNB shall start after sending the AS security mode command message. RRC and UP uplink deciphering (decryption) at the eNB shall start after receiving the AS security mode complete message.

RRC and UP uplink ciphering (encryption) at the UE shall start after sending the AS security mode complete message. RRC and UP downlink deciphering (decryption) at the UE shall start after receiving the AS security mode command message

If any control of the AS security mode command is not successful, the procedure ends in the ME [see TS 33.102 section 6.4.5] and ME shall not send AS security mode complete message.

For the case when mapped security context in use and cached security context needs to be activated, the Service Request or TAU Request message NAS uplink COUNT value can not be used as the K_{eNB} key derivation parameter as the security context are different. For this reason the MME shall provide 4 LSBs of the cached security context NAS uplink COUNT value and the corresponding KSI_{ASME} for the serving eNB and the eNB shall include them into the AS SMC message. This way the possible desynchronization of cached security context NAS uplink COUNT value e,g. due to lost NAS messages is identified and fixed in the ME.

AS security mode command always changes the AS keys.

Editor's Note: It is for further study, whether the SMC procedure does not affect the AS (PDCP) uplink and downlink SN or HFN values (RAN2). As when keys are fresh, it does not matter from ciphering and integrity protection point of view whether AS COUNTs are reset or not. In case they are not reset, the threshold value needs to be taken care of.

Editor's Note: It is for further study, whether the RRC and UP ciphering algorithms are combined, i.e. no separate algorithm identifier for RRC and UP but one common AS level ciphering algorithm id.

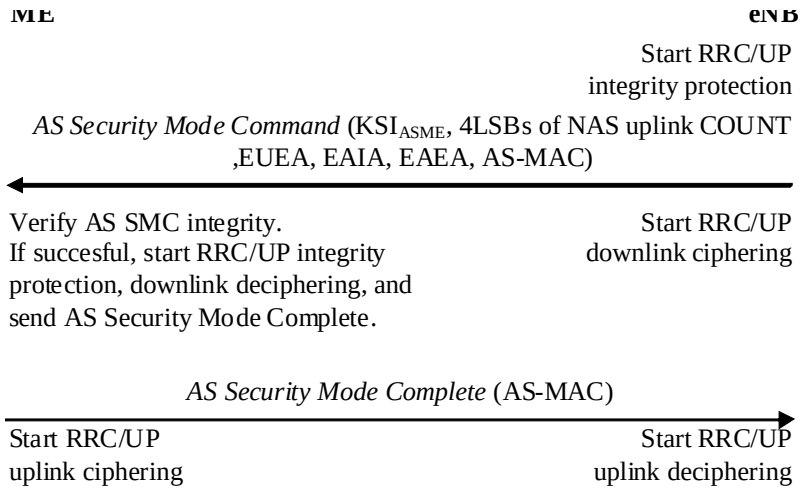


Figure 7.2.5.3-1: AS security setup

7.2.5 Key handling at state transitions to and away from EMM-DEREGISTERED

7.2.5.1 Transition to EMM-DEREGISTERED

There are different reasons for transition to the EMM-DEREGISTERED state. Key handling for each of these cases are given below. As these are NAS messages, they will be integrity protected when a security context exists between the UE and MME.

1. Attach reject: All authentication data shall be removed from the UE and MME
2. Detach:

a. UE-initiated

   i. If the reason is switch off then all authentication data and related information shall be removed from the UE and MME.

   ii. If the reason is not switch off then MME and UE shall keep all authentication data.

b. MME-initiated

   i. Explicit: authentication data shall be kept in the UE and MME if the detach type is re-attach

   ii. Implicit: authentication data shall be kept in the MME

c. HSS-initiated: If the message is "subscription withdrawn" then authentication data is removed from the UE and MME.

Editors Note: Handover to non-3GPP needs to be considered based on SA2 work.

3. TAU reject: There are various reasons for TAU reject. The action to be taken shall be as given in TS 24.301 subclause 5.5.3.5.

7.2.5.2 Transition away from EMM-DEREGISTERED

7.2.5.2.1 General

When the UE transits from EMM-DEREGISTERED to EMM-REGISTERED/ECM-CONNECTED, there are two cases to consider, either a complete NAS security context exists, or it does not.

7.2.5.2.2 With existing NAS security context

If there is a NAS security context, the UE transmits a NAS Attach Request message. This message is integrity protected, and the NAS COUNT of the Attach Request message is used to derive the K_{eNB} with the KDF as specified in Annex A. As a result of the NAS Attach Request, the eNB sends an AS SMC to the UE with the current KSI_{ASME} indicating the current K_{ASME}.

When the UE receives the AS SMC, it uses the NAS COUNTof the Attach/Service Request message (i.e. the uplink NAS COUNT) that triggered the AS SMC to be sent as freshness parameter in the derivation of the K_{eNB}. From this K_{eNB} the RRC protection keys and the UP protection keys are derived as described in subclause 7.2.1.

Editor's Note: Whether the UE can be certain which Attach Request triggered a given AS SMC is pending verification from RAN2/RAN3/SA2/CT1. If the UE cannot be certain, the uplink NAS COUNTused by the MME must be echoed back to the UE from the MME.

The same procedure for refreshing K_{eNB} can be used regardless of the fact if the UE is connecting to the same MME to which it was connected previously or to a different MME. In case UE connects to a different MME and this MME supports different NAS algorithms, the NAS keys have to be re-derived in the MME with the new algorithm IDs as input using the KDF as specified in Annex A.

In addition, there is a need for the MME to send a NAS SMC to the UE to indicate the change of NAS algorithms and to take the re-derived NAS keys into use. The UE shall assure that the NAS keys used to verify the integrity of the NAS SMC are derived using the algorithm ID specified in the NAS SMC. The NAS SMC Command and NAS SMC Complete messages are protected with the new keys.

7.2.5.2.3 No existing NAS security context or after AKA

In the case that there is an AKA run (either because there is no NAS security context, or the network decides to run an AKA after the Attach Request but before the corresponding AS SMC), the NAS (uplink and downlink) COUNTs are reset to start values, and the start value of the uplink NAS COUNTshall be used as freshness parameter in the K_{eNB} derivation from the fresh K_{ASME} (after AKA) when UE receives AS SMC Command with the new KSI_{ASME} indicating that the fresh K_{ASME} is used to derive the K_{eNB} The KDF as specified in Annex A shall be used to derive the K_{eNB}.

NOTE: Using the start value for the uplink NAS COUNTin this case cannot lead to the same combination of K_{ASME} and NAS COUNTbeing used twice. This is guaranteed by the fact that the first integrity protected NAS message the UE sends to the MME after AKA is the NAS SMC complete message.

The NAS SMC complete message will include the start value of the NAS COUNTthat is used as freshness parameter in the K_{eNB} derivation and the K_{ASME} is fresh. After an AKA, a NAS SMC needs to be sent from the MME to the UE in order to take the new NAS keys into use. Both NAS SMC Command and NAS SMC Complete messages are protected with the new NAS keys.

7.2.6 Key handling in ECM-IDLE to ECM-CONNECTED and ECM-CONNECTED to ECM-IDLE transitions when in EMM-REGISTERED state

7.2.6.1 General

As a general principle, on ECM-IDLE to ECM-CONNECTED transitions when in EMM-REGISTERED state, RRC protection keys and UP protection keys shall be generated as described in subclause 7.2.1 while K_{ASME} is assumed to be already available in the MME.

K_{ASME} may have been established in the MME as a result of an AKA run, or as a result of a security context transfer from another MME during handover or idle mode mobility. On ECM-CONNECTED to ECM-IDLE transitions, eNBs shall delete the keys they store such that state in the network for ECM-IDLE state UEs will only be maintained in the MME.

7.2.6.2 ECM-IDLE to ECM-CONNECTED transition

The procedure the UE uses to transit from ECM-IDLE to ECM-CONNECTED when in EMM-REGISTERED state is initiated by a NAS Service Request message from the UE to the MME. As the UE is in EMM-REGISTERED state, a EPS security context exists in the UE and the MME, and this EPS security context further contains uplink and downlink NAS COUNTs. The NAS Service Request message sent in EMM-REGISTERED shall be integrity protected and contain the uplink NAS sequence number.

As a result of the NAS Service Request, radio bearers are established, and the eNB sends an AS SMC Command to the UE. When the UE receives the AS SMC Command including the current KSI_{ASME}, it shall use the NAS uplink COUNT of the NAS Service Request message that triggered the AS SMC as freshness parameter in the derivation of the K_{eNB} The KDF as specified in Annex A shall be used for the K_{eNB} derivation. From this K_{eNB} the RRC protection keys and the UP protection keys are derived as described in subclause 7.2.1.

Editor's Note: Whether the UE can be certain which NAS Service Request triggered a given AS SMC is pending verification from RAN2/RAN3/SA2/CT1. If the UE cannot be certain, the uplink NAS COUNTused by the MME must be echoed back to the UE from the MME.

If the ECM-IDLE to ECM-CONNECTED procedure contains an AKA run (which is optional), the NAS uplink and downlink COUNTshall be reset to the start values, and the reset value of the uplink NAS COUNTshall be used as freshness parameter in the K_{eNB} derivation from fresh K_{ASME} when executing an AS SMC including the new $KSI_{ASME.}$ The KDF as specified in Annex A shall be used for the K_{eNB} derivation also in this case..

Editor's Note: If an AKA is run and the UE immediately transits back into ECM-IDLE, and after this again transits into ECM-CONNECTED, it may happen that the uplink NAS COUNTused for the previous K_{eNB} derivation is used also for this second one (e.g., if AKA messages do not carry NAS sequence numbers). It is FFS whether this can happen.

On transitions to ECM-CONNECTED, the MME should be able to check whether a new authentication is required, e.g. because of prior inter-provider handover.

7.2.6.3 ECM-CONNECTED to ECM-IDLE transition

On ECM-CONNECTED to ECM-IDLE transitions the eNB does no longer need to store state information about the corresponding UE. In particular eNB shall delete the current AS keys from its memory.

In particular, on ECM-CONNECTED to ECM-IDLE transitions:

- The eNB and the UE shall delete K_{eNB} , K_{RRCenc} and K_{RRCint} and K_{UPenc}
- MME and the UE will keep K_{ASME}, NAS uplink and downlink COUNTs, selected NAS algorithms, and UE security capabilities stored.

7.2.7 Key handling in ECM-IDLE mode mobility

If the "active flag" is not set in the TAU request, the TAU procedure does not establish any RRC or UP level security. Because of this, there is no need to derive any K_{eNB} in this case. If the "active flag" is set in the TAU request message, radio bearers will be established as part of the TAU procedure. In this case a K_{eNB} derivation is necessary, and the uplink NAS COUNTof the TAU request message sent from the UE to the MME is used as freshness parameter in the K_{eNB} derivation using the KDF as specified in Annex A. The TAU request shall be integrity protected..

In the case an AKA is run successfully as part of the TAU procedure, the uplink and downlink NAS COUNTshall be set to the start values.

In the case an AKA is not run as part of the TAU procedure and source and target MME use different NAS algorithms, the target MME re-derives the NAS keys from K_{ASME} with the new algorithms-Ids as input and provides the new algorithm identifiers within a NAS SMC. The UE shall assure that the NAS keys used to verify the integrity of the NAS SMC are derived using the algorithm ID specified in the NAS SMC.

7.2.8 Key handling in handover

7.2.8.1 General

AS level (RRC, UP) algorithms can be changed during inter-eNB handovers, ECM-IDLE to ECM-CONNECTED state transitions, EMM-DEREGISTERED to EMM-REGISTERED/ECM-CONNECTED state transitions.

On initial context setup (e.g. attach, transition to ECM-CONNECTED, or intersystem handover to E-UTRAN), the MME shall derive key $K_{eNB}[0]$ from K_{ASME} and the current NAS uplink COUNT. The MME shall also calculate a next key $K_{eNB}[1] = KDF(K_{eNB}[0], K_{ASME})$. The MME shall communicate these two key values along with the corresponding indexes (0 and 1) to the Serving eNB. See figure 7.2.8.3-1.

RRC and UP keys are refreshed during eNB handovers. Source eNB creates K_{eNB}* key from the current K_{eNB} key if the index does not increase from the previous handover, or from the Next Hop (NH) parameter if the index increases from the previous handover (i.e. fresh NH). If the NH is used, the current K_{eNB} is deleted. The physical cell ID (PCI) of the target cell is bound to the K_{eNB}* using the KDF as specified in Annex A. Target eNB creates new K_{eNB} based on K_{eNB}* and C-RNTI using the KDF as specified in Annex A if the index from the previous handover increases. Otherwise the target eNB uses the K_{eNB}* as the new K_{eNB}. UE and target eNB derive RRC and UP keys from the new K_{eNB}.

The target eNB algorithm identifiers and key purpose identifiers are used in the AS level key derivations as input parameters with K_{eNB} to the key derivation function KDF (see Annex A).

At an eNB handover with MME relocation, the K_{eNB} is chained in the same way as if it was a regular intra-MME eNB handover. However, there is the possibility that the source MME and the target MME do not support the same set of NAS algorithms or have different priorities regarding the use of NAS algorithms. In this case, the target MME re-derives the NAS keys from K_{ASME} using the NAS algorithm identities as input to the NAS key derivation functions (see Annex A) and sends NAS SMC. All inputs, in particular the K_{ASME}, will be the same in the re-derivation except for the NAS algorithm identity.

It is essential that the NAS COUNTs are not reset to the start values unless a new K_{ASME} is taken into use. This prevents that, in the case a UE moves back and forth between two MMEs the same NAS keys will be re-derived time and time again resulting in key stream re-use. Since K_{ASME} only changes when a new AKA has been run successfully, the NAS COUNTs shall only be reset to the start value when there is a new AKA run followed by the activation of the corresponding K_{ASME}. In case the target MME decides to use NAS algorithms different from the ones used by the source MME, a NAS SMC including KSI_{ASME} (new or current value depending on whether AKA was run or not) shall be sent from the MME to the UE.

This NAS Key and algorithm handling also applies to other MME changes e.g. TAU with MME changes.

The procedures for key derivation during handovers are described in the sections 7.2.8.3 below. The key management common to these handovers are described with key management handling blocks described in section 7.2.8.2.

NOTE: It is per operator's policy how to configure selection of handover types. Depending on an operator's security requirements, the operator can decide whether to have S1 1-hop secure handovers for a particular eNB according to its security characteristics.

Editors Note: Some of the procedures in clauses 7.2.8.2 to 7.2.8.4 may have a complex handling; further CRs to simplify these procedures may be necessary.

7.2.8.2 Key management handling blocks

7.2.8.2.1 Next Hop (NH) handling blocks (Proc-NH0, Proc-NH1, and Proc-NH2)

Proc-NH0 – create initial NH value from K_{ASME} and fresh K_{eNB} – run in UE and MME

Proc-NH0:

NCC = 0; NH = KDF(K_{ASME}, K_{eNB}), see S_6 in Annex A.6

K_{eNB}* = K_{eNB}

Proc-NH1 – synchronize NH (NH*) – run in UE and MME

Proc-NH1:

Temp-NCC = NCC; NH* = NH

*In UE repeatedly update NH if NH needs to be updated (e.g. Temp-NCC < Received-NCC). UE shall try at least **MIN-NH-UPDATE** times before giving up with an error:*

NH* = KDF(K_{ASME}, NH*), see S_4 in Annex A.4

++Temp-NCC

The variable **MIN-NH-UPDATE** shall be at least 0x04. This means that the loop shall be run in the UE at least **MIN-NH-UPDATE** times before giving up with an error.

Proc-NH2 – update current NH

Proc-NH2:

NH = NH*; NCC = Temp-NCC

Delete NH*; Delete Temp-NCC

7.2.8.2.2 K_{eNB} key handling blocks (Proc-KeNB1 and Proc-KeNB2)

Proc-KeNB1 – create target eNB key (K_{eNB}*) – run in UE, eNB, and MME

Proc-KeNB1:

If index increases from previous HO:

K_{eNB} = NH*

K_{eNB}* = KDF(K_{eNB}, PCI), see S_5 in Annex A.5

Proc-KeNB2 – create new K_{eNB}, RRC/UP keys, and delete temporary K_{eNB} key – run in UE and eNB

Proc-KeNB2:

If index increases from previous HO:

$K_{eNB} = K_{eNB}^*$

If index does not increase from previous HO:

$K_{eNB} = KDF(K_{eNB}^*$, target C-RNTI); See S_6 in Annex A.6

Derive RRC and UP keys from K_{eNB}; See S_8 in Annex A.8

Delete K_{eNB}^*

7.2.8.3 K_{eNB} and Next Hop (NH) parameter handling during initial NAS messages

Fig. 7.2.8.3-1 describes the NH parameter creation during initial attachment, intersystem mobility, and ECM-IDLE to ECM-CONNECTED state transition. Whenever a fresh K_{eNB} is calculated from the K_{ASME}, the NH parameter is also calculated from the K_{ASME} and the fresh K_{eNB}. The yellow key handling boxes are defined in section 7.2.8.2.

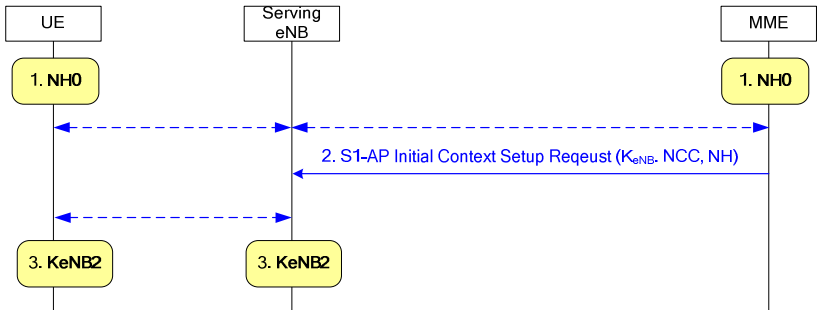


Figure 7.2.8.3-1 Initialization of NH key derivation parameter and K_{eNB} in UE and MME and transfer to eNB

1. Both UE and MME run Proc-NH0
2. MME provides K_{eNB}, NCC, and NH in the AS security context for the serving. Serving eNB stores NCC and NH as NH* and K_{eNB} as K_{eNB}^*.
3. Both UE and serving eNB run Proc-KeNB2

7.2.8.4 K_{eNB}, Next Hop (NH), and NH Chaining Count (NCC) parameter handling during handovers

The mechanism is described in Figure 7.2.8.4-1 and includes a NH parameter from MME to the target eNB within the path switch acknowledgement message. Feeding the K_{ASME} with the previous NH and incremented NCC to the NH KDF in the MME results in a cryptographically separate parameter for the target eNB compared to the parameter in the source eNB.

NOTE 1: Because the path switch message is transmitted after the radio link handover, it can only be used to provide keying material for the next handover procedure and target eNB. Thus, key separation happens only after two hops because the source eNB knows the target eNB keys (the fresh key derivation parameter, NH, for target eNB is incorporated in the key derivations by the source eNB).

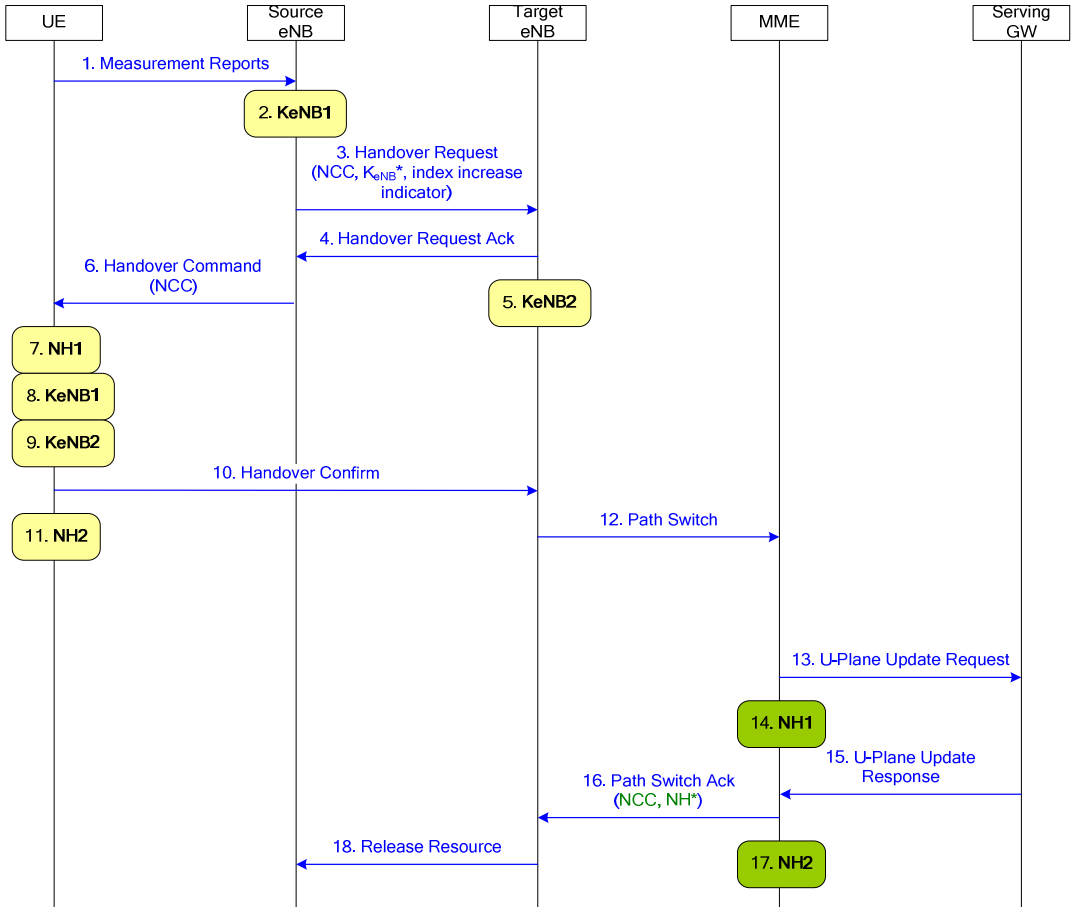


Figure 7.2.8.4-1 NH based key refresh with intra-MME handover

1. Measurement report

2. Source eNB runs Proc-KeNB1

3. Source eNB provides security context (NCC, index increase indicator, K_{eNB}*, …) for target eNB. Source eNB knows whether the index increases or not and indicates that in the security context for the target eNB.

4. Handover request Ack

5. Target eNB runs Proc-KeNB2

6. Handover Command includes the NCC.

7. UE runs Proc-NH1 that updates the NH depending on UE"s current NCC value and the Received-NCC value in the HO Command

8. UE runs Proc-KeNB1

9. UE runs Proc-KeNB2

10. UE sends Handover Confirm to target eNB

11. UE runs Proc-NH2

12. Target eNB sends path switch to MME

13. MME sends U-Plane update request to S-GW

14. MME runs Proc-NH1 and may update NCC and NH*

15. S-GW sends U-Plane update response to MME

16. MME sends path switch acknowledgement and includes NCC and NH*. Target eNB updates the currently stored NCC and NH* with NCC and NH* from the path switch acknowledgement message.

17. MME runs Proc-NH2

18. Target eNB sends release resource message to source eNB

In Fig. 7.2.8.3-2 key derivations are described for handover procedure, where the NCC and NH parameters are the same as in previous handover. This can happen for X2 and S1 handovers (e.g., if the MME does not provide NCC and NH values to the eNB), but the figure only describes the intra-eNB handover.

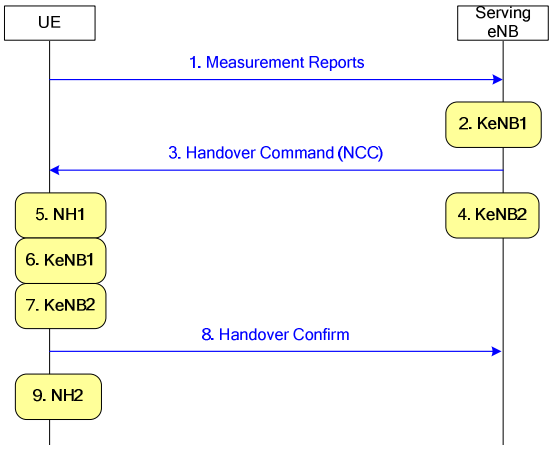


Figure 7.2.8.4-2 Key re-fresh with intra-eNB handover

1. Measurement report

2. Source (serving) eNB runs KeNB1

3. Handover Command includes the NCC.

4. Target (serving) eNB runs KeNB2

5. UE runs Proc-NH1 that updates the NH depending on UE''s current NCC value and the Received-NCC value in the HO Command

6. UE runs Proc-KeNB1

7. UE runs Proc-KeNB2

8. UE sends Handover confirmation for serving eNB

9. UE runs Proc-NH2

Similarly to procedure in Figure 7.2.8.4-1, the procedure in Fig. 7.2.8.4-3 describes key derivation steps for inter-MME handover procedure.

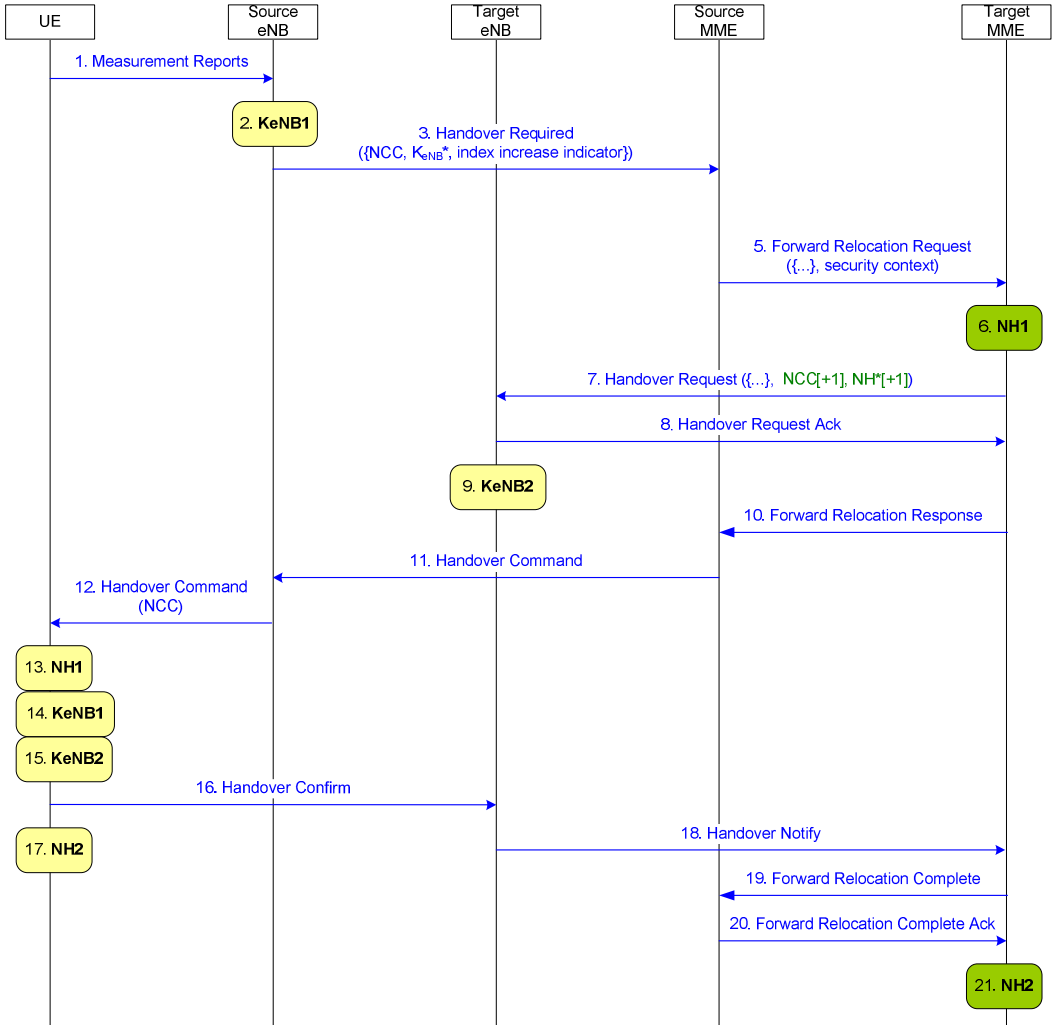


Figure 7.2.8.4-3: NH based key refresh with inter-MME handover with 2-hop security

1. Measurement report

2. Source eNB runs Proc-KeNB1

3. Source eNB sends Handover Required message including AS level security context (NCC, K_{eNB}*, index increase indicator, …) for the source MME as in X2 handover. Source eNB knows whether the index increases or not and indicates that in the security context for the target eNB.

5. Source MME sends Forward Relocation Request message including the AS level security context from source eNB to the target MME. Source MME also sends the NH and NCC values from its memory to the target MME along with the NAS level security context.

6. Target MME updates NH by running Proc-NH1

7. Target MME sends Handover Request to Target eNB including the the AS level security context from the source eNB (K_{eNB}*, NH, index increase indicator, …). MME includes fresh NH* and NCC into the Handover Request message. Target eNB updates the security context NCC and NH with fresh NH* and corresponding NCC in the Handover Request message if MME included them in addition to the security context from the source eNB (i.e. instead of using the parameters from source eNB the target eNB uses the parameters from the MME).

8. Target eNB sends Handover Request Ack for target MME

9. Target eNB runs Proc-KeNB2 based on the context received from the source eNB

10. Target MME sends Forward Relocation Response to source MME

11. Source MME sends Handover Command to source eNB including the NCC and NH* if updated

12. Source eNB sends Handover Command including the NCC from the Handover Command message from the MME if present, otherwise using the NCC that it has in the memory and that it sent to the target eNB within the transparent container.

13. UE runs Proc-NH1 that updates the NH depending on UE"s current NCC value and the Received-NCC value in the HO Command

14. UE runs Proc-KeNB1

15. UE runs Proc-KeNB2

16. UE sends Handover Confirm to target eNB

17. UE runs Proc-NH2

18. Target eNB sends Handover Notify for target MME

19. Target MME sends Forward Relocation Complete to source MME

20. Source MME sends Forward Relocation Complete Ack to target MME

21. Source MME runs Proc-NH2

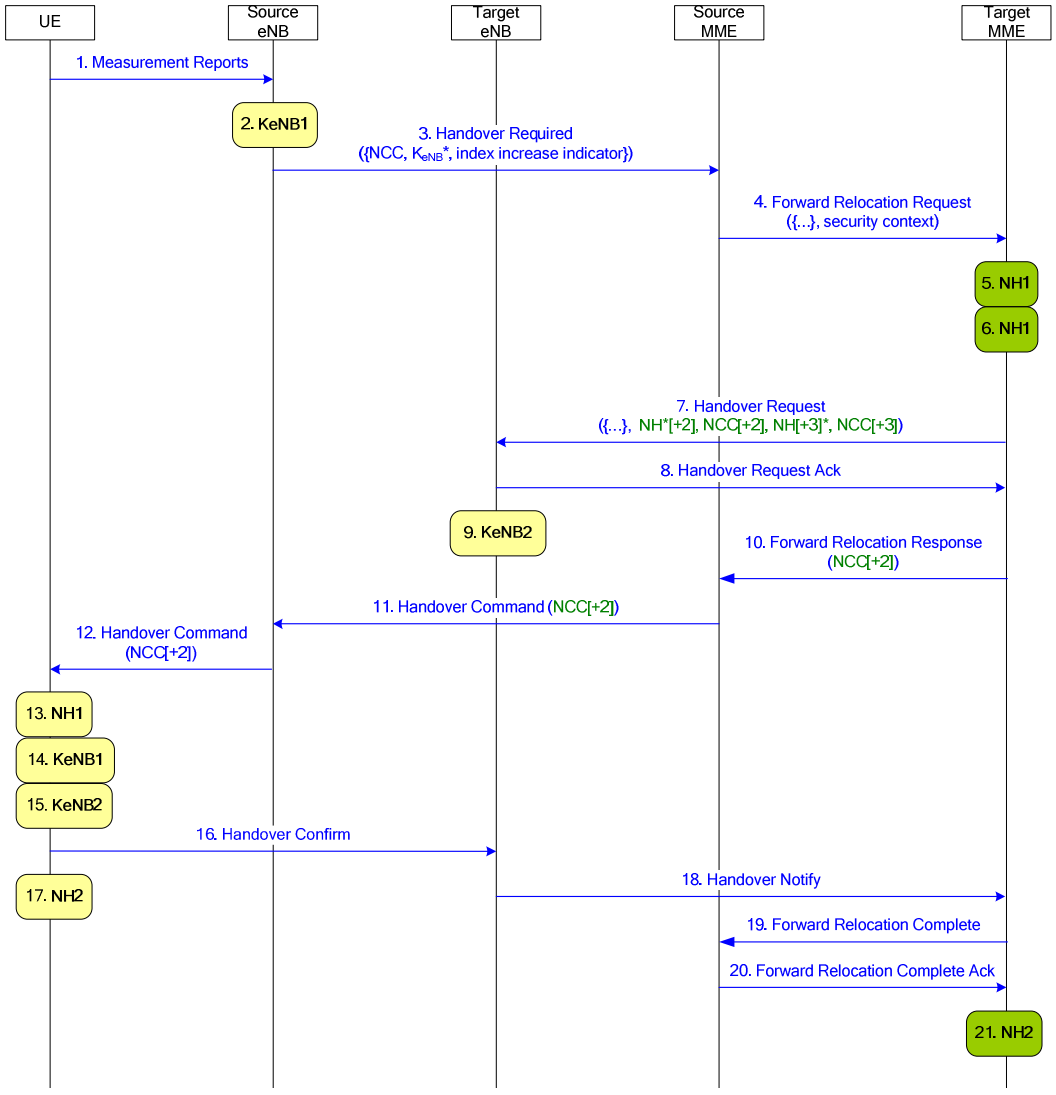


Figure 7.2.8.4-4: NH based key refresh with inter-MME handover with 1-hop security

1. Measurement report

2. Source eNB runs Proc-KeNB1

3. Source eNB sends Handover Required message including AS level security context (NCC, K_{eNB}*, index increase indicator, K_{eNB}*, …) for the source MME as in X2 handover. Source eNB knows whether the index increases or not and indicates that in the security context for the target eNB.

4. Source MME sends Forward Relocation Request message including the AS level security context from source eNB to the target MME. Source MME also sends the NH and NCC values from its memory to the target MME along with the NAS level security context.

5.-6. Target MME runs Proc-NH1 twice (1-hop S1 handover security).

7. Target MME sends Handover Request to Target eNB including the the AS level security context from the source eNB (K_{eNB}*, NCC, handover index increase indicator,…). MME includes two new fresh NH*s and NCCs into the Handover Request message. Target eNB updates the security context, NCC and NH with fresh NH* and corresponding NCC in the Handover Request message if MME included them in addition to the security context from the source eNB (i.e. instead of using the parameters from source eNB the target eNB uses the +2 parameters from the MME and stores the +3 parameters).

8. Target eNB sends Handover Request Ack for target MME

9. Target eNB runs Proc-KeNB2

10. Target MME sends Forward Relocation Response to source MME

11. Source MME sends Handover Command to source eNB including the NCC and NH* if updated

12. Source eNB sends Handover Command including the NCC from the Handover Command message from the MME if present, otherwise using the NCC that it has in the memory and that it sent to the target eNB within the transparent container.

13. UE runs Proc-NH1 that updates the NH depending on UE"s current NCC value and the Received-NCC value in the HO Command

14. UE runs Proc-KeNB1

15. UE runs Proc-KeNB2

16. UE sends Handover Confirm to target eNB

17. UE runs Proc-NH2

18. Target eNB sends Handover Notify for target MME

19. Target MME sends Forward Relocation Complete to source MME

20. Source MME sends Forward Relocation Complete Ack to target MME

21. Source MME runs Proc-NH2

7.2.9 Key-change-on-the fly

7.2.9.1 General

AS Key change on-the-fly is accomplished using a procedure based on intra-cell handover. The following AS key changes on-the-fly shall be possible: local K_{eNB} refresh (performed when PDCP COUNTs are about to wrap around), K_{eNB} re-keying performed after an AKA run.

Editor's Note: It is FFS whether there is a need for change of AS keys derived from K_{ASME} but with a different freshness parameter.

7.2.9.2 K_{eNB} re-keying

The procedure is initiated by the MME after a successful AKA run with the UE. The MME derives the new K_{eNB} using the same key derivation function as is used for ECM-IDLE to ECM-CONNECTED state transitions (see Annex A) using the new K_{ASME} and the NAS COUNT zero as input. The K_{eNB} is sent to the eNB in a S1-AP message triggering the eNB to perform the re-keying.

The eNB runs the key change on-the-fly procedure with the UE. During this procedure the eNB indicates to the UE which KSI_{ASME} was used to generate the K_{eNB} that shall be the basis for derivation of the K_{eNB} to be used after the procedure. The procedure used is based on an intra-cell handover, and hence the same K_{eNB} derivation steps shall be taken as in a normal handover procedure.

If the UE receives an indication that a different K_{ASME} than the one currently in use, the UE derives a temporary K_{eNB} by applying the same key derivation function as is used in ECM-IDLE to ECM-CONNECTED state transitions (see Annex A) using a NAS COUNT of zero and the new K_{ASME} as input. From this temporary K_{eNB} the UE derives the K_{eNB}* as normal (see Annex A). The eNB takes the K_{eNB} it received from the MME, which is equal to the temporary $K_{eNB,}$ as basis for its K_{eNB}* derivations. From this step onwards, the key derivations continue as in a normal handover.

In case the eNB has scheduled the UE for a handover when the re-keying message is received from the MME, the eNB and the UE shall perform the same key derivation steps as if it was a intra-cell handover with the sole purpose of a K_{eNB} re-keying.

7.2.9.3 KeNB rerefresh

This procedure is initiated by the eNB when the PDCP COUNTs are about to wrap around. It is based on an intra-cell handover. The K_{eNB} chaining that is performed during a handover ensures that the K_{eNB} is re-freshed w.r.t. the RRC and UP COUNT after the procedure.

Editors NOTE: It is for ffs if the UE should be prepared first when receiving a key change on the fly request before preparing the neighbouring cell for RLFs. Security issues of sending the same KeNB to different eNB needs to be considered.

7.2.9.4 NAS key re-keying

After an AKA has taken place, new NAS keys from a new K_{ASME} shall be derived, i.e. NAS key re-keying. NAS SMC always changes the NAS keys (i.e. due to EPS AKA with new K_{ASME} and KSI_{ASME} or due to the algorithms change).

7.3 UP security mechanisms

7.3.1 UP confidentiality mechanisms

The user plane data is ciphered by the PDCP protocol between the UE and the eNB as specified in TS 36.323 [12]..

The input parameters to the 128-bit EEA algorithms as described in Annex B are an 128-bit cipher key K_{UPenc} as KEY, an 8-bit bearer identity BEARER which corresponds to the radio bearer identity, the 1-bit direction of transmission DIRECTION, the length of the keystream required LENGTH and a bearer specific, time and direction dependent 32-bit input COUNT which corresponds to the 32-bit PDCP COUNT.

7.4 RRC security mechanisms

7.4.1 RRC integrity mechanisms

RRC integrity protection is provided by the PDCP layer between UE and eNB.

The input parameters to the 128-bit EIA algorithms as described in Annex B are an 128-bit integrity key K_{RRCint} as KEY,, an 8-bit bearer identity BEARER corresponding to the signalling bearer identity, the 1-bit direction of transmission DIRECTION and a bearer specific, time and direction dependent 32-bit input COUNT which corresponds to the 32-bit PDCP COUNT.

7.4.2 RRC confidentiality mechanisms

RRC confidentiality protection is provided by the PDCP layer between UE and eNB.

The input parameters to the 128-bit EEA algorithms as described in Annex B are an 128-bit cipher Key K_{RRCenc} as KEY, an 8-bit bearer identity BEARER which corresponds to the radio bearer identity, the 1-bit direction of transmission DIRECTION, the length of the keystream required LENGTH and a bearer specific, time and direction dependent 32-bit input COUNT which corresponds to the 32-bit PDCP COUNT.

NOTE: Since the BEARER identities for Signalling Radio Bearers (SRBs) that are used for RRC messages are different from the BEARER identities used for radio bearers used to carry user plane data there will not be collisions between the input parameters.

7.5 Signalling procedure for periodic local authentication

The following procedure is used optionally by the eNB to periodically perform a local authentication. At the same time, the amount of data sent during the AS connection is periodically checked by the eNB and the UE for both up and down streams. If UE receives the Counter Check request, it shall respond with Counter Check Response message.

The eNB is monitoring the PDCP COUNT values associated to each radio bearer. The procedure is triggered whenever any of these values reaches a critical checking value. The granularity of these checking values and the values themselves are defined by the visited network. All messages in the procedure are integrity protected.

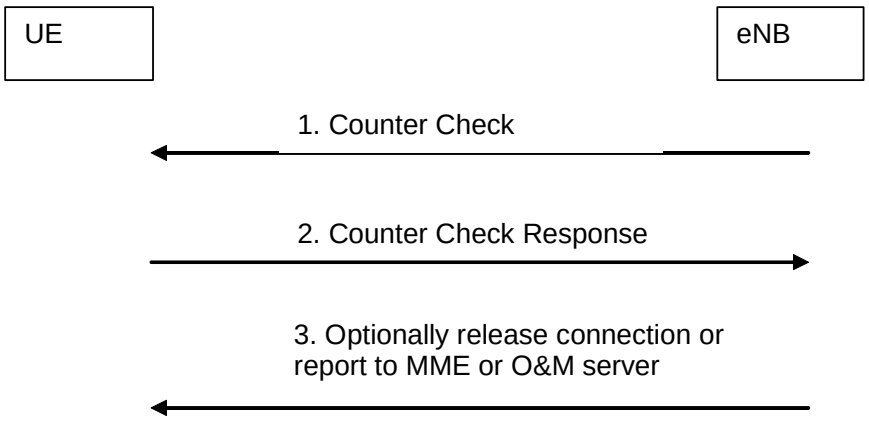


Figure 7.5-1: eNB periodic local authentication procedure

1. When a checking value is reached (e.g. the value in some fixed bit position in the hyperframe number is changed), a Counter Check message is sent by the eNB. The Counter Check message contains the most significant parts of the PDCP COUNT values (which reflect amount of data sent and received) from each active radio bearer.

2. The UE compares the PDCP COUNT values received in the Counter Check message with the values of its radio bearers. Different UE PDCP COUNT values are included within the Counter Check Response message.

3. If the eNB receives a counter check response message that does not contain any PDCP COUNT values, the procedure ends. If the eNB receives a counter check response that contains one or several PDCP COUNT values, the eNB may release the connection or report the difference of the PDCP COUNT values for the serving MME or O&M server for further traffic analysis for e.g. detecting the attacker.

8 Security mechanisms for non-access stratum signalling

8.1 NAS integrity mechanisms

Input parameters to the 128-bit EIA algorithms as described in Annex B are an 128-bit integrity key K_{NASint} as KEY, an 8-bit bearer identity BEARER which shall equal the constant value 0x00, the direction of transmission DIRECTION, and a bearer specific, time and direction dependent 32-bit input COUNT which is constructed as follows:

COUNT := 0x00 || NAS OVERFLOW || NAS SQN

Where

- the leftmost 8 bits are padding bits including all zero's.

- NAS OVERFLOW is a 16-bit value which is incremented each time the NAS SQN is incremented from the maximum value.

- NAS SQN is the 8-bit sequence number carried within each NAS message.

NOTE: The BEARER identity is not necessary since there is only one NAS signalling connection per pair of MME and UE, but is included as a constant value so that the input parameters for AS and NAS will be the same, which simplifies specification and implementation work.

8.1.1 NAS integrity activation

NAS integrity shall be activated with the help of the NAS SMC procedure immediately after authentication. NAS integrity stays activated until the EPS security context is deleted. The EPS security context may only be deleted if UE is in EMM-DEREGISTERED. While the EPS security context exists, all NAS messages shall be integrity protected. In particular the NAS service request shall always be integrity protected and the NAS attach request message shall be integrity protected if the EPS security context is not deleted while UE is in EMM-DEREGISTERED. The length of the NAS MAC is 32 bit. The full NAS MAC shall be appended to all integrity protected messages except for the NAS service request. Only the 16 least significant bits of the 32 bit NAS MAC shall be appended to the NAS service request message.

8.2 NAS confidentiality mechanisms

The input parameters for the NAS ciphering algorithms shall be the same as the ones used for NAS integrity protection as described in clause 8.1, with the addition that the length of the key stream to be generated by the encryption algorithms.

9 Security interworking between E-UTRAN and UTRAN

Editor's Note: Any impacts in this section due the feature of per user activation of UP ciphering are for ffs and dependent on an agreement on this feature which is for ffs.

9.1 Idle mode mobility

9.1.1 From E-UTRAN to UTRAN

The E-UTRAN NAS Attach Request and TAU Request messages shall include the UTRAN and GERAN security capabilities of UE. All UE security capabilities are sent back to the UE in the corresponding integrity protected response messages for verification that they were not changed.

Cached context

If UE has valid cached SGSN security context it shall send RAU Request with P-TMSI and with a KSI indicating the cached KSI in the old SGSN. In this case the UE may include P-TMSI signature into the RAU Request. Keys from the old SGSN shall overwrite keys in the target SGSN if any. If the network does not have valid cached security context it shall run AKA.

Mapped context

If UE does not have valid cached SGSN security context, it will send RAU Request with P-TMSI mapped from GUTI and with a KSI indicating KSI_{ASME}. UE shall include NAS-token into the P-TMSI signature IE. The MME shall transfer UE's UTRAN and GERAN security capabilities and CK' || IK' with KSI_{ASME} to SGSN with Context Response/SGSN Context Response message. The SGSN and UE shall assign the value of KSI_{ASME} to KSI ,i.e KSI =KSI_{ASME}. The MME and UE derive CK' and IK' from K_{ASME} and the current NAS downlink COUNT value indicated by the NAS-token received by the MME from SGSN as specified in Annex A. Keys from the old MME shall replace the keys in the target SGSN if any.

SGSN shall include the allowed security algorithm and transfer to RNC with SMC message. RNC will select security algorithms and indicate to UE whenever needed.

The available P-TMSI signature field bits (at minimum 16 bits) shall be filled with a NAS-token (i.e. the x least significant bits of the KDF output):

NAS-token is derived as specified in Annex A. Both K_{ASME} and current NAS downlink COUNT are mandatory input parameters.

SGSN forwards the P-TMSI signature to the old MME, which compares it with a NAS-token (truncation of most significant bits if needed), for the UE identified within the context request. If they match, the context request message is authenticated and authorized and MME will provide the needed information for the SGSN. Old MME responds with an appropriate error cause if it does not match the value stored in the old MME. This should initiate the security functions in the new SGSN.

To avoid possible race condition problems, the MME shall be able to compare the received NAS-token with NAS-tokens generated from the current NAS SN downlink value down to current NAS SN-L downlink values, i.e. [current NAS downlink SQN - L, current NAS downlink SQN].. A suitable value for the parameter L can be configured by the network operator. The default value for the parameter L is 5 and maximum value 10. MME shall not accept the same NAS-token for the same UE twice except in retransmission cases happening for the same mobility event.

9.1.2 From UTRAN to E-UTRAN

SGSN shall transfer CK || IK to MME in the Context Response/SGSN Context Response message. MME shall derive K'_{ASME} from CK || IK as described in Appendix A.

SGSN shall be informed by the UE about its EPC/E-UTRAN security capabilities. This happens via the MS Network Capability IE that is extended to include also E-UTRAN capabilities, in Attach Request and RAU Request.

Editor's NOTE: Pre-rel8 SGSN may not include the EPC/E-UTRAN security capabilities. In this case UE capabilities from the TAU Request are used.

SGSN shall also transfer UE's EPC/E-UTRAN security capabilities to the new MME in the Context Response/SGSN Context Response message. MME shall select security algorithms and will indicate them to the UE by e.g. with NAS SMC or in the TAU procedure.

Cached context

UE uses the E-UTRAN cached security context if available in the UE to protect the TAU Request and include the corresponding temporary identity and KSI_{ASME} value. UE uses the cached security context algorithms to protect the TAU Request message. UE shall also include KSI_{SGSN} with corresponding source system temporary identity to point to the right source SGSN and key set. there. This allows the network to choose the mapped security context if cached security context is not available in the network. UE shall include 32bit $NONCE_{UE}$ into the TAU Request message independent of whether a cached context is available or not.In case MME has the cached security context it verifies the TAU Request message and replies with TAU Accept message protected with the cached security context. In case the TAU Request had the active flag set or there is pending downlink UP data, the NAS uplink COUNT from the TAU

Request is used to derive the K_{eNB} as specified in Annex A. MME delivers the K_{eNB} to the target eNB on the S1 interface.

If the MME changes the algorithms, they shall be indicated to the UE in an integrity protected message, which shall also include the UE security capabilities. The algorithm identifiers shall not be ciphered. UE shall reply with integrity protected message based on the new selected algorithms.

Mapped context

If no cached context is available in the UE the UE shall send the TAU request unprotected. UE shall include $NONCE_{UE}$ and KSI_{SGSN} with corresponding source system temporary identity to point to the right source SGSN and key set there

In case MME does not have the cached context indicated by the UE in the TAU request, or the TAU request was received unprotected, the MME shall use the mapped security context. In this case, the MME shall generate a 32bit $NONCE_{MME}$ and use the received $NONCE_{UE}$ with the $NONCE_{MME}$ to generate a fresh mapped $K"_{ASME}$ from CK and IK identified by the KSI_{SGSN} in the TAU Request. See Appendix A for more information on how to derive the fresh $K"_{ASME}$.

The selected algorithms and keys with the KSI_{SGSN} shall be indicated to the UE in an integrity protected message, which shall also include the UE security capabilities and both the nonces. The algorithm identifiers shall not be ciphered. UE shall reply with integrity protected message based on the selected algorithms and fresh $K"_{ASME}$ and shall include the UE security capabilities so that the MME can be sure that they were not modified in the TAU Request message by an outsider.

TAU Accept shall be protected using the NAS keys based on the fresh $K"_{ASME}$.

9.2 Handover

9.2.1 From E-UTRAN to UTRAN

If UP ciphering in E-UTRAN is activated, it shall remain activated after handover in UTRAN as well.

Integrity protection on NAS signalling is mandatory for both E-UTRAN and UTRAN

MME shall derive a confidentiality key CK', and an integrity key IK' from K_{ASME} and the current NAS downlink COUNT value with the help of a one-way key derivation function KDF in the following way:

$CK'||IK' = KDF(K_{ASME,} S_{10})$ MME shall transfer CK' || IK' with KSI_{ASME} to SGSN, UE and SGSN shall assign the value of KSI_{ASME} to KSI.. The KDF returns a 256-bit output, where the 128 most significant bits are identified with CK and the 128 least significant bits are identified with IK. S_{10} is the string where the input parameters are defined (see Annex A).

MME will also provide the 4 LSB of the current NAS downlink COUNT value to the source eNB, which then includes the bits to the HO Command to the UE.

Editor's Note: FFS whether 4 LSB is ok for RAN2.

The E-UTRAN NAS Attach Request and TAU Request messages shall also include the UTRAN and GERAN security capabilities of UE. All UE security capabilities are sent back to the UE in an integrity protected message for verification that they were not changed in the message that informs about the selected NAS level security algorithms (e.g. NAS SMC Command, Attach Accept, or TAU Response).

MME shall transfer UE's UTRAN and GERAN security capabilities to SGSN. SGSN shall include the allowed security algorithms in the relocation request to RNC. RNC shall select the algorithms and include the corresponding identifiers in the relocation request acknowledgement. The selected algorithms shall be indicated to UE in the handover command message.

9.2.2 From UTRAN to E-UTRAN

9.2.2.1 Procedures

If UP ciphering in UTRAN is activated, then UP ciphering shall remain activated in E-UTRAN after handover as well.

Integrity protection on NAS is mandatory for both E-UTRAN and UTRAN.

SGSN shall transfer CK and IK to MME in the relocation request message. MME and UE shall derive K'_{ASME} from CK and IK with the help of a one-way key derivation function as defined in Annex A.MME and UE shall derive the NAS keys and K_{eNB} from K'_{ASME}.

SGSN shall be informed by the UE about its EPS security capabilities. This happens via the MS Network Capability IE, that is extended to include also E-UTRAN security capabilities, in Attach Request and RAU Request.

SGSN transfers UE's EPC/E-UTRAN security capabilities to MME in the relocation request message. MME shall select the NAS security algorithms and include K_{eNB}in the relocation request to the target eNB. The target eNB shall select the RRC and UP algorithms and indicate them in the relocation request acknowledgement. MME shall include the selected NAS, UP, and RRC algorithms in the relocation response sent to SGSN. SGSN shall include them in the relocation command and RNC shall indicate them to UE in the handover from UTRAN command. MME shall send a $NONCE_{MME}$ and the KSI (KSI_{ASME} or KSI_{SGSN}) to the target eNB to be put in the transparent container created by the eNB and to be sent towards the UE via the source RNC

MME shall send KSI_{ASME} and KSI_{SGSN} to the target eNB for inclusion into the transparent container if it has cached security context, otherwise the KSI_{SGSN} of the mapped security context. UE shall select cached security context if it has it, otherwise mapped security context and indicate the selected context by including the corresponding KSI (KSI_{ASME} or KSI_{SGSN}) into the HO Confirm message to the target eNB. For this purpose the target MME shall provide both mapped and cached context based K_{eNB} for the target eNB along with KSI_{ASME} and KSI_{SGSN} if cached security context exists in the MME. The same algorithms, indicated to the UE inside E-UTRAN to UTRAN the transparent container, are used regardless of whether cached or mapped K_{eNB} is used.

If the UE has cached security context and the network indicated that it has cached security context the TAU Request message shall be integrity protected and created as described in section 9.1.2, except that the UE does not include $NONCE_{UE}$ into the TAU Request. Otherwise the UE shall use the mapped security context and protects the TAU Request with the same algorithms as selected for RRC.

If EPS does not have the cached EPS security context for the UE indicated in the TAU Request, or if EPS does not activate the cached EPS security context indicated in the TAU Request, the newly generated keys or the cached EPS security context shall be taken into operation as soon as possible.

9.2.2.2 Key Derivation during Handover

SGSN shall transfer CK and IK to MME in the relocation request message. MME and UE shall derive K'_{ASME} from CK and IK and a 32-bit $NONCE_{MME}$ generated by the MME with the help of a one-way key derivation function KDF as specified in Annex A.

MME and UE shall derive the NAS keys and K_{eNB} from K'_{ASME} as specified in Annex A.

During UTRAN to E-UTRAN HO, MME sets NAS COUNT equal to zero and uses it with K'_{ASME} to derive $\mathbf{K_{eNB}}$ by applying the KDF defined in Annex A for IDLE to CONNECTED transition. The MME distributes the mapped security context based $\mathbf{K_{eNB}}$ and the KSI_{SGSN} to the eNB in the Handover Request message.

MME shall also include cached security context based K_{eNB} and the KSI_{ASME} to the eNB in the Handover Request message in case MME included KSI_{ASME} into the transparent container.The UE derives the mapped security context based $\mathbf{K_{eNB}}$ in the same way as the MME did (if MME included the KSI_{SGSN} into the transparent container) and cached security context based K_{eNB} if it has it and if MME included KSI_{ASME} into the transparent container. The cached context based K_{eNB} shall be derived by replacing the NAS uplink COUNT with the $NONCE_{MME}$ in the KDF in Appendix A.

When mapped context is used after mobility to E-UTRAN and cached security context is available both in UE and the MME, it may be activated within the next idle to active state transition with AS level SMC or with intra-cell handover with HO Command. In both cases the AS SMC and the HO Command message shall include the KSI_{ASME} and 4 least significant bits of the cached context NAS uplink COUNT value. MME shall not activate the cached context if the NAS uplink COUNT has not increased from the previous K_{eNB} derivation from the same cached context.

9.3 Recommendations on AKA at IRAT-mobility to E-UTRAN

After a handover from GERAN or UTRAN into E-UTRAN, it is strongly recommended to run an AKA and perform a key change on-the-fly of the entire key hierarchy as soon as possible after the handover if there is no cached security context in E-UTRAN.

When a UE moves in IDLE mode from GERAN or UTRAN into E-UTRAN, it is strongly recommended to run an AKA if there is no cached security context in E-UTRAN, either after the TAU procedure that establishes an EPS security context in the MME and UE, or when the UE transits into ECM-CONNECTED state.

10 Security interworking between E-UTRAN and GERAN

Editor's Note: Any impacts in this section due the feature of per user activation of UP ciphering are for ffs and dependent on an agreement on this feature which is for ffs.

10.1 Idle mode mobility

10.1.1 From E-UTRAN to GERAN

The E-UTRAN NAS Attach Request and TAU Request messages shall also include the UTRAN and GERAN security capabilities of UE. All UE security capabilities are sent back to the UE in the corresponding integrity protected response messages for verification that they were not changed.

Cached context

If UE has valid cached SGSN security context it shall send LAU Request with P-TMSI and with a KSI indicating the cached KSI in the old SGSN. UE may include P-TMSI signature into the LAU Request. Keys from the old SGSN shall overwrite keys in the target SGSN if any. If the network does not have valid cached security context it shall run AKA.

Mapped context

If UE does not have valid cached SGSN security context, it will send LAU Request with P-TMSI mapped from GUTI and with a KSI indicating KSI_{ASME}. UE shall include NAS-token into the P-TMSI signature IE. MME shall transfer UE's UTRAN and GERAN security capabilities and the CK' and IK' with KSI_{ASME} to SGSN. UE and SGSN shall assign the value of KSI_{ASME} to CKSN,i.e CKSN=KSI_{ASME}. SGSN shall select the encryption algorithm to use in GERAN when needed. Key from the old MME shall replace the keys in the target SGSN if any.

The available P-TMSI signature field bits (16 bits) shall be filled with a NAS-token (i.e. the x least significant bits of the KDF output). The key derivation function of NAS-token is specified in Annex A. The same NAS downlink COUNT as was used to create the NAS-token is used as a key derivation parameter to derive CK" and IK".

SGSN forwards the P-TMSI signature to the old MME, which compares it with a NAS-token (truncation of most meaningful bits if needed), for the UE identified within the context request. If they match, the context request message is authenticated and authorized and MME will provide the needed information for the SGSN. Old MME responds with an appropriate error cause if it does not match the value stored in the old MME. This should initiate the security functions in the new SGSN.

To avoid possible race condition problems, the MME shall be able to compare the received NAS-token with NAS-tokens generated from the current NAS SN downlink value down to current NAS SN-L downlink values, i.e. [current NAS downlink SQN - L, current NAS downlink SQN]. A suitable value for the parameter L can be configured by the network operator. The default value for the parameter L is 5 and maximum value 10. MME shall not accept the same NAS-token for the same UE twice except in retransmission cases happening for the same mobility event.

10.1.2 From GERAN to E-UTRAN

See section 9.1.2.

10.2 Handover

10.2.1 From E-UTRAN to GERAN

If UP ciphering in E-UTRAN is activated, it shall remain activated after handover in GERAN as well.

MME shall derive a confidentially key CK' and an integrity IK' from K_{ASME} as described for the E-UTRAN to UTRAN handover case. MME shall transfer CK' and IK' to the SGSN. SGSN shall derive Kc from CK' and IK' with the help of the key conversion function c3 of TS 33.102.

Editor's Note: The assumption here is that an SGSN that supports E-UTRAN to GERAN handover is capable of the key conversion functions c3, c4, c5 of TS 33.102. If this assumption falls the key derivation and conversion described above may have to be adopted.

MME will also provide the 4 LSB of the current NAS downlink COUNT value to the source eNB, which then includes the bits to the HO Command to the UE.

The E-UTRAN NAS Attach Request and TAU Request messages shall also include the UTRAN and GERAN security capabilities of UE. All UE security capabilities are sent back to the UE in an integrity protected message for verification that they were not changed in the message that informs about the selected NAS level security algorithms (e.g. NAS SMC Command, Attach Accept, or TAU Response).

MME shall transfer UE's UTRAN and GERAN security capabilities to SGSN. SGSN shall select the encryption algorithm to use in GERAN after handover. The selected algorithm shall be indicated to UE in the handover command message.

10.2.2 From GERAN to E-UTRAN

10.2.2.1 Procedures

If UP ciphering was activated in GERAN, it shall remain activated in E-UTRAN after handover from GERAN to E-UTRAN as well.

SGSN shall be informed by the UE about its EPS security capabilities. This happens via the MS Network Capability IE, that is extended to include also E-UTRAN security capabilities, in Attach Request and RAU Request.

SGSN transfers UE's EPC/E-UTRAN security capabilities to MME in the relocation request message. MME shall select the NAS security algorithms and include the K_{eNB}in the relocation request to the target eNB. The target eNB shall select the RRC and UP algorithms and indicate them in the relocation request acknowledgement. MME shall include the selected NAS, UP, and RRC algorithms in the relocation response sent to SGSN. SGSN shall indicate them in the appropriate message to BSS and BSS shall indicate them to UE in the handover command. MME shall send a $NONCE_{MME}$ and the $KSI_{SGSN}$to the target eNB to be put in the transparent container created by the eNB and to be sent towards the UE via the source BSS.

MME shall send KSI_{ASME} to the target eNB for inclusion into the transparent container if it has cached security context, otherwise the KSI_{SGSN} of the mapped security context. UE shall select cached security context if it has it, otherwise mapped security context and indicate the selected context by including the corresponding KSI (KSI_{ASME} or KSI_{SGSN}) into the HO Confirm message to the target eNB. For this purpose the target MME will provide both mapped and cached context based K_{eNB} for the target eNB along with KSI_{ASME} and KSI_{SGSN} if cached security context exists in the MME. The same algorithms, indicated to the UE inside E-UTRAN to UTRAN the transparent container, are used regardless of whether cached or mapped K_{eNB} is used.

If the UE has cached security context and the network indicated that it has cached security context the

TAU Request message shall be integrity protected and created as described in section 10.1.2, except that the UE does not include $NONCE_{UE}$ into the TAU Request. Otherwise the UE shall use the mapped security context and protects the TAU Request with the same algorithms as selected for the RRC.

If EPS does not have the cached EPS security context for the UE indicated in the TAU Request, or if EPS does not activate the cached EPS security context indicated in the TAU Request, The newly generated keys or the cached EPS security context shall be taken into operation as soon as possible.

10.2.2.2 Key Derivation during Handover

The key derivation as the same as described in subclause 9.2.2.2

10.3 Recommendations on AKA at IRAT-mobility to E-UTRAN

See recommendation provided by subclause 9.3

11 Network Domain Control Plane protection

The protection of IP based control plane signalling for EPS and E-UTRAN shall be done according to TS 33.210 [5].

NOTE 1: In case control plane interfaces are trusted (e.g. physically protected), there is no need to use protection according to TS 33.210 [5].

In order to protect the S1 and X2 control plane, it is required to implement IPsec ESP according to RFC 4303 [7] as specified by TS 33.210 [5]. For both S1-MME and X2-C, IKEv2 certificates based authentication according to TS 33.310 [6] shall be implemented. For S1-MME and X2-C, tunnel mode IPsec is mandatory to implement on the eNB. On the core network side a SEG may be used to terminate the IPsec tunnel.

Transport mode IPsec is optional for implementation on the X2-C and S1-MME.

NOTE 2: Transport mode can be used for reducing the protocol overhead added by IPsec.

12 Backhaul link user plane protection

The protection of user plane data between the eNB and the UE by user specific security associations is covered by clause 5.1.3 and 5.1.4.

In order to protect the S1 and X2 user plane as required by clause 5.3.3, it is required to implement IPsec ESP according to RFC 4303 [7] as profiled by TS 33.210 [5], with confidentiality, integrity and replay protection.

On the X2-U and S1-U, transport mode IPsec is optional for implementation.

NOTE 1: Transport mode can be used for reducing the protocol overhead added by IPsec.

Tunnel mode IPsec is mandatory to implement on the eNB for X2-U and S1-U. On the core network side a SEG may be used to terminate the IPsec tunnel..

For both S1 and X2 user plane, IKEv2 with certificates based authentication shall be implemented. The certificates shall be implemented according to the profile described by TS 33.310 [6]. IKEv2 shall be implemented conforming to the IKEv2 profile described in TS 33.310 [6]

NOTE 2: In case S1 and X2 user plane interfaces are trusted (e.g. physically protected), the use of IPsec/IKEv2 based protection is not needed.

13 Management plane protection over the S1 interface

Clause 5.3.1 requires that eNB setup and configuration traffic, i.e. the management plane, is protected between the EPS core and the eNB. This traffic is carried over the same backhaul link as the S1 interface, so the protection mechanism defined for S1-C and S1-U may be re-used.

To achieve this protection, it is required to implement IPsec ESP according to RFC 4303 [7] as profiled by TS 33.210 [5], with confidentiality, integrity and replay protection.

Tunnel mode IPsec is mandatory to implement on the eNB for the S1 management plane. On the core network side a SEG may be used to terminate the IPsec tunnel. If no SEG is used, the IPsec tunnel may be terminated in the element manager.

For the S1 management plane, IKEv2 with certificates based authentication shall be implemented. The certificates shall be implemented according to the profile described by TS 33.310 [6]. IKEv2 shall be implemented conforming to the IKEv2 profile described in TS 33.310 [6]

NOTE 1: X2 does not carry management plane traffic.

NOTE 2: In case the S1 management plane interfaces are trusted (e.g. physically protected), the use of IPsec/IKEv2 based protection is not needed

14. SRVCC between E-UTRAN and Circuit Switched UTRAN/GERAN

14.1 From E-UTRAN to Circuit Switched UTRAN/GERAN

MME shall derive a confidentiality key CK_{SRVCC}, and an integrity key IK_{SRVCC} from K_{ASME} and the NAS downlink COUNT with the help of a one-way key derivation function KDF in the following way:

$CK_{SRVCC}||IK_{SRVCC} = KDF(K_{ASME,} S_{15})$. The KDF returns a 256-bit output, where the 128 most significant bits are identified with CK_{SRVCC} and the 128 least significant bits are identified with IK_{SRVCC}. S_{15} is the string where the input parameters are defined.

MME will also provide the 4 LSB of the current NAS downlink COUNT value to the source eNB, which then includes the bits to the HO Command to the UE.

If the SRVCC is from E-UTRAN to GERAN, Kc shall be derived from CK_{SRVCC} and IK_{SRVCC} in the SGSN with the help of the key conversion function c3 as specified in TS 33.102 [4].

Annex A (normative): Key derivation functions

A.1 KDF interface and input parameter construction

The input parameters and their lengths shall be concatenated into a string S as follows:

1. The length of each input parameter in octets shall be encoded into two-octet string:

   a) express the number of octets in input parameter Pi as a number k in the range [0, 65535];

   b) Li is then a two-octet representation of the number k, with the most significant bit of the first octet of Li equal to the most significant bit of k, and the least significant bit of the second octet of Li equal to the least significant bit of k.

   EXAMPLE: If Pi contains 258 octets then Li will be the two-octet string 0x01 0x02.

2. String S shall be constructed from n input parameters as follows:

   S = FC || P0 || L0 || P1 || L1 || P2 || L2 || P3 || L3 ||... || Pn || Ln

   where:

   FC is single octet used to distinguish between different instances of the algorithm,

   P0 ... Pn are the n input parameters, and

   L0 ... Ln are the two-octet representations of the corresponding input parameters.

3. The final output, i.e. the derived key is equal to the KDF computed on the string S using the key Key. The present document defines the following KDF:

   derived key = HMAC-SHA-256 (Key, S),

   as specified in [10] and [11], which has the KDF identity 1.

All key derivations shall be performed using the negotiated key derivation function (KDF). This clause specifies the set of input strings, S_i, to the KDF (which are input together with the relevant key). For each of the distinct usages of the KDF, the input parameters S_i are specified below.

If another KDF is negotiated between the UE and the MME, the interface to that KDF shall be the same, i.e., the input to that KDF shall be the relevant 256-bit key and a string S, which shall have the same formats as described in this annex, and the output shall be a 256-bit long key. It shall be possible to negotiate at most 4 KDFs.

NOTE: The value 0x01 for parameter FC is used by TS 33.220 [8], so the numbering starts at 0x02 in the present document to ensure that no input collisions will accidentally occur.

A.2 KASME derivation function (S_2)

When deriving a K_{ASME} from Ks and serving network ID when producing authentication vectors, the following parameters shall be used to form the input S to the KDF.

- FC = 0x02,
- P0 = serving network ID,
- L0 = length of serving network ID (i.e. 0x00 0x03),
- P1 = RAND

- L1 = length of RAND (i.e. 0x00 0x10)
- P2 = IMSI
- L2 = length of IMSI (variable)
- P3 = SQN ⊕ AK
- L3 = length of SQN ⊕ AK (i.e. 0x00 0x06)

Editor's NOTE 1: The input parameters except (P0, L0) are to be confirmed by SAGE.

NOTE: The string S indexes start from 2 to align with the FC and Appendix section values.

If AK is not used, AK shall be treated in accordance with TS 33.102, i.e. as 000…0.

In case the serving network is E-UTRAN, the network ID shall be a PLMN ID. The PLMN ID consists of MCC and MNC, and shall be encoded as an octet string according to Figure A.2-1.

8	7	6	5	4	3	2	1	
MCC digit 2				MCC digit 1				octet 1
MNC digit 3				MCC digit 3				octet 2
MNC digit 2				MNC digit 1				octet 3

Figure A.2-1 Encoding of PLMN ID as an octet string

The coding of the digits of MCC and MNC shall be done according to TS 24.301 [9].

Editor's NOTE 2: The coding is not yet specified in TS 24.301, but it is expected that the coding specified in TS 24.008 will be used also in TS 24.301.

The encoding of the IMSI shall be done as in TS 31.102 Clause 4.2.2, e.g. an 8 octet array of the BCD representation of the IMSI, where (for shorter IMSIs) unused nibbles are set to 0xf.

The input key shall be the 256-bit Ks key.

A.3 K_{eNB} derivation function used at ECM-IDLE to ECM-CONNECTED transition, ECM-IDLE mode mobility, transition away from EMM-DEREGISTERED to EMM-REGISTERED/ECM-CONNECTED and key change on-the-fly (S_3)

When deriving a K_{eNB} from K_{ASME} and the NAS COUNT in the UE and the MME the following parameters shall be used to form the input S to the KDF.

- FC = 0x03,
- P0 = Uplink NAS COUNT,
- L0 = length of uplink NAS COUNT (i.e. 0x00 0x04)

The input key shall be the 256-bit K_{ASME}.

A.4 NH* derivation function (S_4)

When deriving a NH* from K_{ASME} the following parameters shall be used to form the input S to the KDF.

- FC = 0x04
- P0 = NH
- L0 = length of NH (i.e. 0x00 0x10)

The input key shall be the 256-bit K_{ASME}.

A.5 K_{eNB}* derivation function (S_5)

When deriving a K_{eNB}* from current K_{eNB} or from fresh NH and the target physical cell ID in the UE and the source eNB for handover purposes the following parameters shall be used to form the input S to the KDF.

- FC = 0x05
- P0 = PCI (target physical cell id)
- L0 = length of PCI (i.e. 0x00 0x02)

The input key shall be the 256-bit NH when the index in the handover increases, otherwise the current 256-bit K_{eNB}.

A.6 New K_{eNB}** derivation function used at handover when index increases from the previous handover (S_6)

When deriving a K_{eNB}** from a K_{eNB}* and target cell C-RNTI in the UE and the target eNB when the key index does not increase from the previous handover, the following parameters, S, shall be used. K_{eNB}** is the new K_{eNB} derived from K_{eNB}* during handover.

- FC = 0x06,
- P0 = C-RNTI,
- L0 = length of C-RNTI (i.e. 0x00 0x02)

The input key shall be the 256-bit K_{eNB}*, K_{eNB}*+, K_{eNB}***.

A.8 Algorithm key derivation functions (S_8)

When deriving keys for NAS integrity and NAS encryption algorithms from K_{ASME} and algorithm types and algorithm IDs, and keys for RRC integrity and RRC/UP encryption algorithms from K_{eNB}, in the UE, MME and eNB the following parameters shall be used to form the string S.

- FC = 0x08
- P0 = algorithm type distinguisher
- L0 = length of algorithm type distinguisher (i.e. 0x00 0x01)
- P1 = algorithm identity
- L1 = length of algorithm identity (i.e. 0x00 0x01)

The algorithm type distinguisher shall be NAS-enc-alg for NAS encryption algorithms and NAS-int-alg for NAS integrity protection algorithms. The algorithm type distinguisher shall be RRC-enc-alg for RRC encryption algorithms, RRC-int-alg for RRC integrity protection algorithms and UP-enc-alg for UP encryption algorithms (see table A.6-1). The values 0x06 to 0xf0 are reserved for future use, and the values 0xf1 to 0xff are reserved for private use.

Table A.8-1: Algorithm type distinguishers

Algorithm distinguisher	Value
NAS-enc-alg	0x01
NAS-int-alg	0x02
RRC-enc-alg	0x03
RRC-int-alg	0x04
UP-enc-alg	0x05

The algorithm identity (as specified in clause 5) shall be put in the four least significant bits of the octet. The two least significant bits of the four most significant bits are reserved for future use, and the two most significant bits of the most significant nibble are reserved for private use. The entire four most significant bits shall be set to all zeros.

For NAS algorithm key derivations, the input key shall be the 256-bit K_{ASME}, and for UP and RRC algorithm key derivations, the input key shall be the 256-bit K_{eNB}.

A.11 K_{ASME} to CK, IK derivation (S_{11})

This input string is used when there is a need to derive CK || IK from K_{ASME} during mapping of security contexts from E-UTRAN to GERAN/UTRAN. K_{ASME} is a 256-bit entity, and so is the concatenation of CK and IK (which are 128 bits each). The following input parameters shall be used.

- FC = 0x11
- P0 = NAS downlink COUNT value
- L0 = length of NAS downlink COUNT value (i.e. 0x00 0x04)

The input key shall be K_{ASME}.

A.10 NAS token derivation for inter-RAT mobility (S_{10})

The NAS-token used to ensure that a RAU is originating from the correct UE during IDLE mode mobility from E-UTRAN to UTRAN and GERAN, shall use the following input parameters.

- FC = 0x10
- P0 = Downlink NAS COUNT
- L0 = length of downlink NAS COUNT (i.e. 0x00 0x04)

The input key shall be the 256-bit K_{ASME}.

A.13 K"$_{ASME}$ from CK, IK derivation during handover (S_{13})

This input string is used when there is a need to derive a K'$_{ASME}$ from concatenation of CK and IK and a $NONCE_{MME}$ during mapping of security contexts between GERAN/UTRAN and E-UTRAN during handover. K'$_{ASME}$ is a 256-bit value. $NONCE_{MME}$ is a 32-bit value. The following input parameters shall be used.

- FC = 0x13
- P0 = $NONCE_{MME}$

- L0 = length of $NONCE_{MME}$ (i.e. 0x00 0x04)

The input key shall be the concatenation of CK || IK.

A.14 K"$_{ASME}$ from CK, IK derivation during idle mode mobility (S_{14})

This input string is used when there is a need to derive a K_{ASME} from CK/IK, $NONCE_{UE}$, and $NONCE_{MME}$ during mapping of security contexts from GERAN/UTRAN to E-UTRAN. K_{ASME} is a 256-bit entity, and so is the concatenation of CK and IK (which are 128 bits each). The following input parameters shall be used, where NONCEs are 32 bits long.

- FC = 0x14,
- P0 = $NONCE_{UE}$
- L0 = length of the $NONCE_{UE}$ (i.e. 0x00 0x04)
- P1 = $NONCE_{MME}$
- L1 = length of the $NONCE_{MME}$ (i.e. 0x00 0x04)

The input key shall be the concatenation of CK || IK.

A.15 K_{ASME} to CK_{SRVCC}, IK_{SRVCC} derivation (S_{15})

This input string is used when there is a need to derive CK_{SRVCC}|| IK_{SRVCC} used in CS domain from K_{ASME} during mapping of security contexts between E-UTRAN and GERAN/UTRAN. K_{ASME} is a 256-bit element, and so is the concatenation of CK_{SRVCC} and IK_{SRVCC} (which are 128 bits each).

- FC = 0x15
- P0 = NAS downlink COUNT value
- L0 = length of NAS downlink COUNT value (i.e. 0x00 0x04)

The input key shall be K_{ASME}.

Annex B (normative): Algorithm input and output parameters

B.1 128-bit ciphering algorithm

The input parameters to the ciphering algorithm are a 128-bit cipher key named KEY, a 32-bit COUNT, an 8-bit bearer identity BEARER, the 1-bit direction of the transmission i.e. DIRECTION, and the length of the keystream required i.e. LENGTH. The DIRECTION bit shall be 1 for uplink and 0 for downlink.

Figure B.1-1 illustrates the use of the ciphering algorithm EEA to encrypt plaintext by applying a keystream using a bit per bit binary addition of the plaintext and the keystream. The plaintext may be recovered by generating the same keystream using the same input parameters and applying a bit per bit binary addition with the ciphertext.

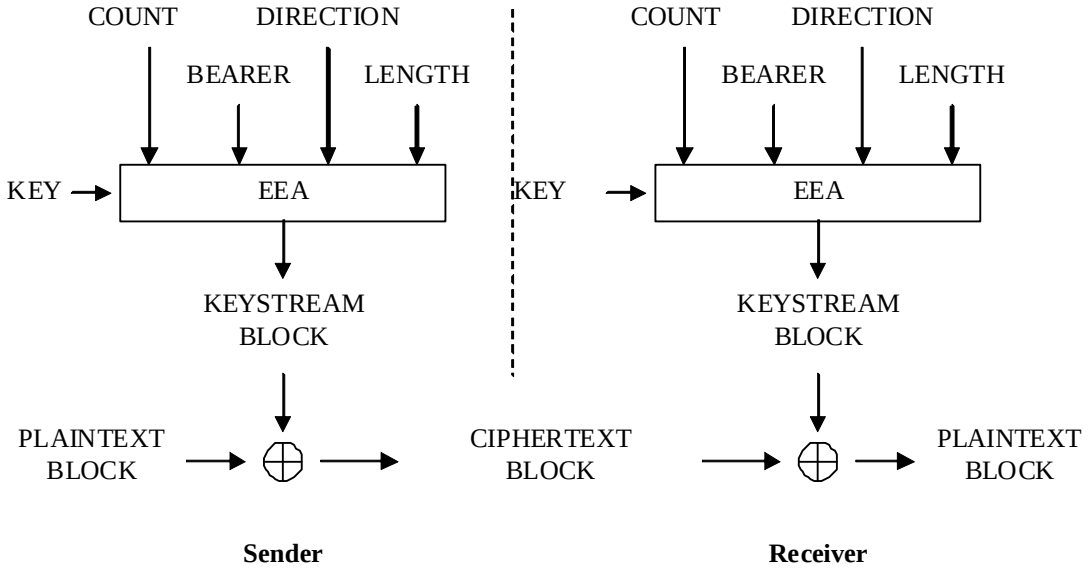


Figure B.1-1: Ciphering of data

Based on the 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.

B.2 128-Bit integrity algorithm

The input parameters to the integrity algorithm are a 128-bit integrity key named KEY, a 32-bit COUNT, an 8-bit bearer identity called BEARER, the 1-bit direction of the transmission i.e. DIRECTION, and the message itself i.e MESSAGE. The DIRECTION bit shall be 1 for uplink and 0 for downlink.

Figure B.2-1 illustrates the use of the integrity algorithm EIA to authenticate the integrity of messages.

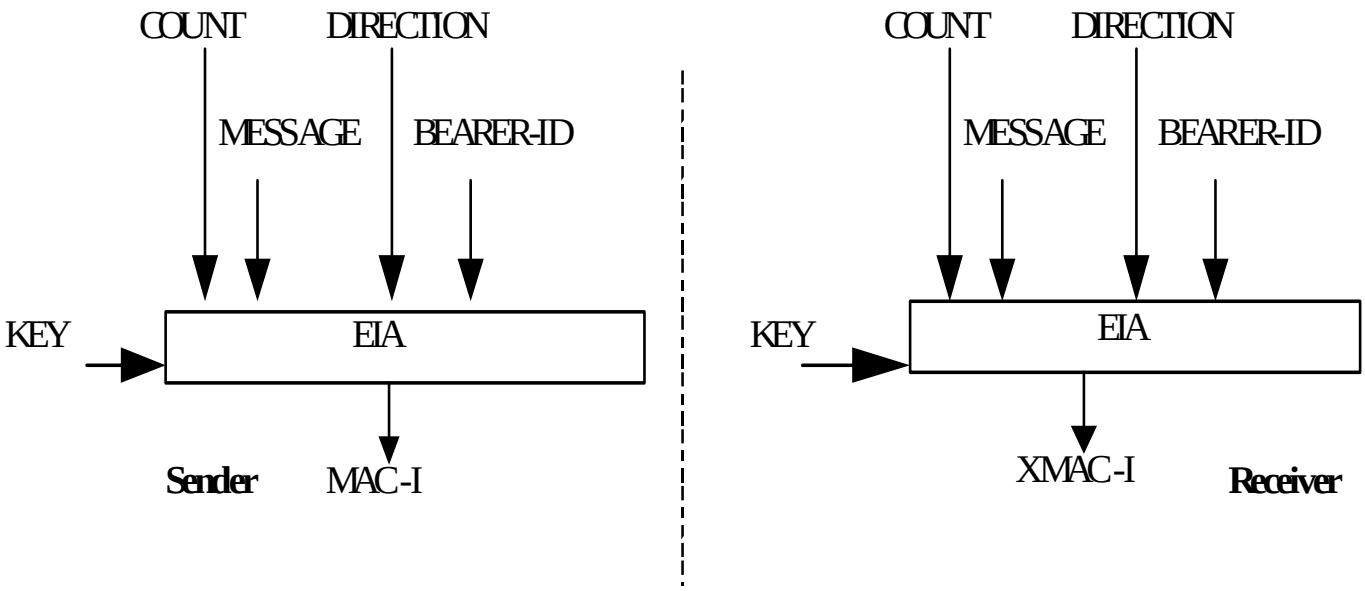


Figure B.2-1: Derivation of MAC-I (or XMAC-I)

Based on these input parameters the sender computes a 32-bit message authentication code MAC-I using the integrity algorithm EIA. The MAC-I is then appended to the message when sent. The receiver computes XMAC-I on the message received in the same way as the sender computed MAC-I on the message sent and verifies the data integrity of the message by comparing it to the received MAC-I.

Annex C (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2007-05	SA3#47				Initial version contains commented Table of Contents with references to TS 33.102 and TR 33.821	-	0.0.0
2007-07	SA3#48				Inclusion of content based on S3-070524, S3-070525 and S3-070526	0.0.0	0.1.0
2007-10	SA3#49				Inclusion of content based on S3-070770, S3-070776, S3-070801, S3-070743, S3-070766	0.1.0	0.2.0
2007-12	SA3#49bis				Inclusion of content based on S3-071021, S3-070963, S3-070951, S3-071015, S3-070968, S3-070923, S3-070922, S3-070919, S3-070960, S3-070953, S3-070958	0.2.0	0.3.0
2008-02	SA3#50				Inclusion of content based on S3-080193; S3-080047; S3-080044; S3-080139; S3-080059; S3-080082; S3-080085; S3-080155; S3-080205; S3-080053; S3-080057; S3-080170; S3-080144; S3-080165; S3-080079; S3-080068;S3-080135; S3-080168; S3-080083	0.3.0	0.4.0
2008-03	SA#39				Presented for information at SA	0.4.0	1.0.0
2008-04	SA3#51				Additions based on S3-080418, 457, 367, 316, 390, 391,325, 490 (414), 466, 380, 364, 402,318, 314, 407, 504	1.0.0	1.1.0
2008-05					MCC preparation for approval	1.1.0	2.0.0
2008-06	SA#40	SP-080257			SA#40 Approval	2.0.0	8.0.0
2008-09	SA#41	SP-080487	0010	1	KeNB forward security simplification	8.0.0	8.1.0
2008-09	SA#41	SP-080653	0040	-	IRAT related changes (merge of CRs 14,24,27)	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0016	1	CR-33401: Security Context Selection on IRAT handover to E-UTRAN	8.0.0	8.1.0
2008-09	SA#41	SP-080653	0041	-	KeNB derivation when activating cached security context and AS SMC clarifications (merge of CR0011 and CR0015)	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0006	1	CR 33.401: Correction of text on security context and authentication data	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0009	1	CR 33.401 :Ensuring security context freshness on handover from an SGSN towards MME	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0012	1	CR-33401: NAS SMC handling clarifications	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0023	1	CR: Algorithm selection cleanup	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0022	1	CR: Editorial cleanup of TS 33.401	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0021	1	CR: Removal of editor's note related to key change on-the-fly	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0013	1	CR-33401: Replay UE capabilities in the NAS SMC Command message	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0007	1	CR 33.401: Completing the specification on air interface ciphering and integrity algorithms inputs	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0005	2	CS key derivation in SRVCC	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0039	-	Note on SNID IP binding	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0038	-	CR on KSI security context desynchronization	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0035	1	NAS integrity tag term consistent	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0017	1	Key freshness during mobility from E-UTRAN to UTRAN/GERAN	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0018	-	Distribution of authentication data from HSS to serving network	8.0.0	8.1.0

2008-09	SA#41	SP-080487	0025	1	CR: Additional inputs to EPS Key Derivation Function (KDF)	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0032	2	Clarification on MSIN_IMEI confidentiality protected limitation	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0029	1	NAS key re-keying	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0030	-	EPS key hierarchy	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0001	1	SAE: backhaul link management plane protection	8.0.0	8.1.0
2008-09	SA#41	SP-080487	0026	1	CR: Add requirement to have replay protection for NAS and RRC	8.0.0	8.1.0
2008-10	--	--	--	--	MCC editorial corrections	8.1.0	8.1.1

History

Document history		
V8.1.1	January 2009	Publication