

# ETSI TS 133 401 V8.3.1 (2009-04)

---

*Technical Specification*

**Digital cellular telecommunications system (Phase 2+);  
Universal Mobile Telecommunications System (UMTS);  
LTE;  
3GPP System Architecture Evolution (SAE);  
Security architecture  
(3GPP TS 33.401 version 8.3.1 Release 8)**

---



---

**Reference**

RTS/TSGS-0333401v831

---

**Keywords**

GSM, LTE, 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

---

**Important notice**

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**Copyright Notification**

No part may be reproduced except as authorized by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2009.  
All rights reserved.

**DECT**<sup>TM</sup>, **PLUGTESTS**<sup>TM</sup>, **UMTS**<sup>TM</sup>, **TIPHON**<sup>TM</sup>, the TIPHON logo and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

**3GPP**<sup>TM</sup> is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

**LTE**<sup>TM</sup> is a Trade Mark of ETSI currently being registered

for the benefit of its Members and of the 3GPP Organizational Partners.

**GSM**<sup>®</sup> and the GSM logo are Trade Marks registered and owned by the GSM Association.

---

## 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

|   |    |
|---|----|
| Intellectual Property Rights .....  | 2  |
| Foreword.....   | 2  |
| Foreword.....   | 6  |
| 1 Scope .....   | 7  |
| 2 References .....  | 7  |
| 3 Definitions, symbols and abbreviations .....  | 8  |
| 3.1 Definitions .....   | 8  |
| 3.2 Symbols.....  | 9  |
| 3.3 Abbreviations .....   | 9  |
| 3.4 Conventions.....  | 10 |
| 4 Overview of Security Architecture.....  | 10 |
| 5 Security Features .....   | 11 |
| 5.1 User-to-Network security .....  | 11 |
| 5.1.1 User identity and device confidentiality .....  | 11 |
| 5.1.2 Entity authentication .....   | 11 |
| 5.1.3 User data and signalling data confidentiality .....   | 12 |
| 5.1.3.1 Ciphering requirements .....  | 12 |
| 5.1.3.2 Algorithm Identifier Values .....   | 12 |
| 5.1.4 User data and signalling data integrity.....  | 13 |
| 5.1.4.1 Integrity requirements .....  | 13 |
| 5.1.4.2 Algorithm Identifier Values .....   | 13 |
| 5.2 Security visibility and configurability .....   | 14 |
| 5.3 Security requirements on eNodeB.....  | 14 |
| 5.3.1 General.....  | 14 |
| 5.3.2 Requirements for eNB setup and configuration.....   | 14 |
| 5.3.3 Requirements for key management inside eNB.....   | 14 |
| 5.3.4 Requirements for handling User plane data within the eNB.....                                 | 15 |
| 5.3.5 Requirements for secure environment of the eNB .....  | 15 |
| 5.4 Void.....   | 15 |
| 6 Security Procedures between UE and EPC Network Elements .....                                     | 16 |
| 6.1 Authentication and key agreement .....  | 16 |
| 6.1.1 AKA procedure.....  | 16 |
| 6.1.2 Distribution of authentication data from HSS to serving network.....                          | 17 |
| 6.1.3 User identification by a permanent identity .....   | 18 |
| 6.1.4 Distribution of IMSI and authentication data within one serving network domain .....          | 18 |
| 6.1.5 Distribution and use of authentication vectors between different serving network domains..... | 19 |
| 6.2 EPS key hierarchy .....   | 19 |
| 6.3 EPS key identification .....  | 22 |
| 6.4 Handling of EPS security contexts .....   | 23 |
| 7 Security Procedures between UE and EPC Access Network Elements .....                              | 24 |
| 7.1 Mechanism for user identity confidentiality.....  | 24 |
| 7.2 Handling of user-related keys in E-UTRAN .....  | 24 |
| 7.2.1 E-UTRAN key setting during AKA .....  | 24 |
| 7.2.2 E-UTRAN key identification.....   | 24 |
| 7.2.3 E-UTRAN key lifetimes .....   | 24 |
| 7.2.4 Security mode command procedure and algorithm negotiation.....                                | 25 |
| 7.2.4.1 Requirements for algorithm selection .....  | 25 |
| 7.2.4.2 Procedures for AS algorithm selection.....  | 25 |
| 7.2.4.2.1 Initial AS security context establishment .....   | 25 |
| 7.2.4.2.2 X2-handover .....   | 26 |
| 7.2.4.2.3 S1-handover.....  | 26 |

|           |   |    |
|-----------|---|----|
| 7.2.4.3   | Procedures for NAS algorithm selection.....   | 26 |
| 7.2.4.3.1 | Initial NAS security context establishment .....  | 26 |
| 7.2.4.3.2 | MME change .....  | 26 |
| 7.2.4.4   | NAS security mode command procedure.....  | 26 |
| 7.2.4.5   | AS security mode command procedure.....   | 27 |
| 7.2.5     | Key handling at state transitions to and away from EMM-DEREGISTERED.....  | 28 |
| 7.2.5.1   | Transition to EMM-DEREGISTERED.....   | 28 |
| 7.2.5.2   | Transition away from EMM-DEREGISTERED.....  | 29 |
| 7.2.5.2.1 | General .....   | 29 |
| 7.2.5.2.2 | With existing EPS NAS security context .....  | 29 |
| 7.2.5.2.3 | With run of EPS AKA.....  | 29 |
| 7.2.6     | Key handling in ECM-IDLE to ECM-CONNECTED and ECM-CONNECTED to ECM-IDLE transitions when in EMM-REGISTERED state..... | 30 |
| 7.2.6.1   | General .....   | 30 |
| 7.2.6.2   | ECM-IDLE to ECM-CONNECTED transition.....   | 30 |
| 7.2.6.3   | ECM-CONNECTED to ECM-IDLE transition.....   | 30 |
| 7.2.7     | Key handling in ECM-IDLE mode mobility .....  | 31 |
| 7.2.8     | Key handling in handover.....   | 31 |
| 7.2.8.1   | General .....   | 31 |
| 7.2.8.1.1 | Access stratum.....   | 31 |
| 7.2.8.1.2 | Non access stratum .....  | 32 |
| 7.2.8.2   | Void.....   | 32 |
| 7.2.8.3   | Key derivations for context modification procedure.....   | 32 |
| 7.2.8.4   | Key derivations during handovers.....   | 32 |
| 7.2.8.4.1 | Intra-eNB Handover .....  | 32 |
| 7.2.8.4.2 | X2-handover .....   | 33 |
| 7.2.8.4.3 | S1-Handover.....  | 33 |
| 7.2.8.4.4 | UE handling.....  | 33 |
| 7.2.9     | Key-change-on-the fly .....   | 34 |
| 7.2.9.1   | General .....   | 34 |
| 7.2.9.2   | K <sub>eNB</sub> re-keying.....   | 34 |
| 7.2.9.3   | KeNB refresh .....  | 35 |
| 7.2.9.4   | NAS key re-keying.....  | 35 |
| 7.3       | UP security mechanisms .....  | 35 |
| 7.3.1     | UP confidentiality mechanisms .....   | 35 |
| 7.4       | RRC security mechanisms.....  | 35 |
| 7.4.1     | RRC integrity mechanisms .....  | 35 |
| 7.4.2     | RRC confidentiality mechanisms .....  | 36 |
| 7.4.3     | K <sub>eNB</sub> * and token preparation for the RRCConnectionRe-establishment Procedure.....                         | 36 |
| 7.5       | Signalling procedure for periodic local authentication.....   | 37 |
| 8         | Security mechanisms for non-access stratum signalling .....   | 38 |
| 8.1       | NAS integrity mechanisms.....   | 38 |
| 8.1.1     | NAS input parameters.....   | 38 |
| 8.1.2     | NAS integrity activation .....  | 38 |
| 8.2       | NAS confidentiality mechanisms .....  | 38 |
| 9         | Security interworking between E-UTRAN and UTRAN.....  | 39 |
| 9.1       | Idle mode procedures .....  | 39 |
| 9.1.1     | Idle mode procedures in UTRAN .....   | 39 |
| 9.1.2     | Idle mode procedures in E-UTRAN .....   | 40 |
| 9.2       | Handover .....  | 41 |
| 9.2.1     | From E-UTRAN to UTRAN .....   | 41 |
| 9.2.2     | From UTRAN to E-UTRAN .....   | 42 |
| 9.2.2.1   | Procedure .....   | 42 |
| 9.2.2.2   | Derivation of NAS keys and K <sub>eNB</sub> during Handover from UTRAN to E-UTRAN.....                                | 45 |
| 9.3       | Recommendations on AKA at IRAT-mobility to E-UTRAN .....  | 46 |
| 10        | Security interworking between E-UTRAN and GERAN.....  | 47 |
| 10.1      | General .....   | 47 |
| 10.2      | Idle mode procedures .....  | 47 |
| 10.2.1    | Idle mode procedures in GERAN .....   | 47 |
| 10.2.2    | Idle mode procedures in E-UTRAN .....   | 47 |

|   |   |           |
|---|---|-----------|
| 10.3  | Handover .....  | 47        |
| 10.3.1  | From E-UTRAN to GERAN .....   | 47        |
| 10.3.2  | From GERAN to E-UTRAN .....   | 47        |
| 10.3.2.1  | Procedures.....   | 47        |
| 10.4  | Recommendations on AKA at IRAT-mobility to E-UTRAN .....  | 47        |
| 11  | Network Domain Control Plane protection.....  | 48        |
| 12  | Backhaul link user plane protection .....   | 48        |
| 13  | Management plane protection over the S1 interface .....   | 49        |
| 14  | SRVCC between E-UTRAN and Circuit Switched UTRAN/GERAN.....   | 50        |
| 14.1  | From E-UTRAN to Circuit Switched UTRAN/GERAN .....  | 50        |
| <b>Annex A (normative): Key derivation functions .....</b>                          |   | <b>51</b> |
| A.1   | KDF interface and input parameter construction .....  | 51        |
| A.1.1   | General .....   | 51        |
| A.1.2   | FC value allocations .....  | 51        |
| A.2   | KASME derivation function ( $S_{10}$ ) .....  | 52        |
| 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, key change on-the-fly and TAU and handover from UTRAN/GERAN to EUTRAN ( $S_{11}$ ) ..... | 53        |
| A.4   | NH derivation function ( $S_{12}$ ) .....   | 53        |
| A.5   | $K_{eNB}^*$ derivation function ( $S_{13}$ ) .....  | 53        |
| A.6   | Void.....   | 54        |
| A.7   | Algorithm key derivation functions ( $S_{15}$ ) .....   | 54        |
| A.8   | $K_{ASME}$ to CK', IK' derivation ( $S_{16}$ ) .....  | 54        |
| A.9   | NAS token derivation for inter-RAT mobility ( $S_{17}$ ).....   | 55        |
| A.10  | $K''_{ASME}$ from CK, IK derivation during handover ( $S_{18}$ ) .....  | 55        |
| A.11  | $K''_{ASME}$ from CK, IK derivation during idle mode mobility ( $S_{19}$ ).....   | 55        |
| A.12  | $K_{ASME}$ to $CK_{SRVCC}$ , $IK_{SRVCC}$ derivation ( $S_{1A}$ ) .....   | 56        |
| <b>Annex B (normative): Algorithms for ciphering and integrity protection .....</b> |   | <b>57</b> |
| B.0   | EAA0 Null ciphering algorithm .....   | 57        |
| B.1   | 128-bit ciphering algorithm.....  | 57        |
| B.1.1   | Inputs and outputs .....  | 57        |
| B.1.2   | 128-EAA1 .....  | 58        |
| B.1.3   | 128-EAA2.....   | 58        |
| B.2   | 128-Bit integrity algorithm.....  | 58        |
| B.2.1   | Inputs and outputs .....  | 58        |
| B.2.2   | 128-EIA1 .....  | 59        |
| B.2.3   | 128-EIA2.....   | 59        |
| <b>Annex C (informative): Change history .....</b>                                  |   | <b>60</b> |
| History .....   |   | 62        |

---

# Foreword

This Technical Specification has been produced by the 3<sup>rd</sup> 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".
- [14] 3GPP TS 35.215: "Confidentiality and Integrity Algorithms UEA2 & UIA2; Document 1: UEA2 and UIA2 specifications".
- [15] NIST: "Advanced Encryption Standard (AES) (FIPS PUB 197) "
- [16] NIST Special Publication 800-38A (2001): "Recommendation for Block Cipher Modes of Operation".
- [17] NIST Special Publication 800-38B (2001): "Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication".



- [18] 3GPP TS 36.331:"Evolved Universal Terrestrial Radio Access (E-UTRA) Radio Resource Control (RRC); Protocol specification".
- [19] Void.
- [20] Void.
- [21] 3GPP TS 36.331:"Evolved Universal Terrestrial Radio Access (E-UTRA) Radio Resource Control (RRC); Protocol specification".
- [22] 3GPP TS 23.216: "Single Radio Voice Call Continuity (SRVCC); Stage 2".
- [23] 3GPP TS 22.101: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Service aspects; Service principles".

---

## 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

**Activation of security context:** the process of taking into use a security context.

**Authentication data:** Data that is part of a security context or of authentication vectors.

**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 in a given system (E-UTRAN or UTRAN/GERAN respectively) with a registered UE has been taken into use by the given system most recently. Note that, in E-UTRAN, a current security context originating from a mapped context may exist simultaneously with a native context, while, in UTRAN/GERAN, only one security context can exist at a time. A current security context exists in MME and SGSN respectively whenever a UE is registered. When the UE is simultaneously registered with both, MME and SGSN, then current EPS and UMTS security contexts exist simultaneously.

**ECM-CONNECTED state:** This is as defined in TS 23.401 [2]. The term ECM-CONNECTED state corresponds to the term EMM-CONNECTED mode used in TS 24.301 [9].

**ECM-IDLE state:** As defined in TS 23.401 [2]. The term ECM-IDLE state corresponds to the term EMM-IDLE mode used in TS 24.301 [9].

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

**EPS security context:** A state that is established between a user and a serving network domain. At both ends "EPS security context data" is stored, that consists of the EPS NAS security context, and the EPS AS security context.

**EPS AS security context:** the cryptographic keys at AS level with their identifiers, the identifiers of the selected AS level cryptographic algorithms and counters used for replay protection. Note that the EPS AS security context only exists when the UE is in ECM-CONNECTED state and is otherwise void.

**EPS NAS security context:** This context consists of  $K_{ASME}$  with the associated key set identifier, the keys  $K_{NASint}$  and  $K_{NASenc}$ , the UE security capabilities, the identifiers of the selected NAS integrity and encryption algorithms, and the uplink and downlink NAS COUNT values. In particular, separate pairs of NAS COUNT values are used for cached and mapped EPS NAS security contexts, respectively. The distinction between native and mapped EPS security contexts also applies to EPS NAS security contexts. For UEs in ECM-CONNECTED state after NAS SMC complete, the EPS NAS security context shall also include the Next Hop parameter NH, and the Next Hop Chaining Counter parameter NCC.

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

**Mapped security context:** Security context created by converting the current security context in the source system to a security context for the target system in inter-system mobility, e.g., UMTS keys created from EPS keys.

**Native EPS security context:** An EPS security context whose  $K_{ASME}$  was created by a run of EPS AKA.

**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  $K_{ASME}$

**UE security capabilities:** The set of identifiers corresponding to the ciphering and integrity algorithms implemented in the UE. This includes capabilities for EPS AS and NAS, and includes capabilities for UTRAN and GERAN if these access types are supported by the UE.

**UE EPS security capabilities:** The UE security capabilities for EPS AS and NAS.

## 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].

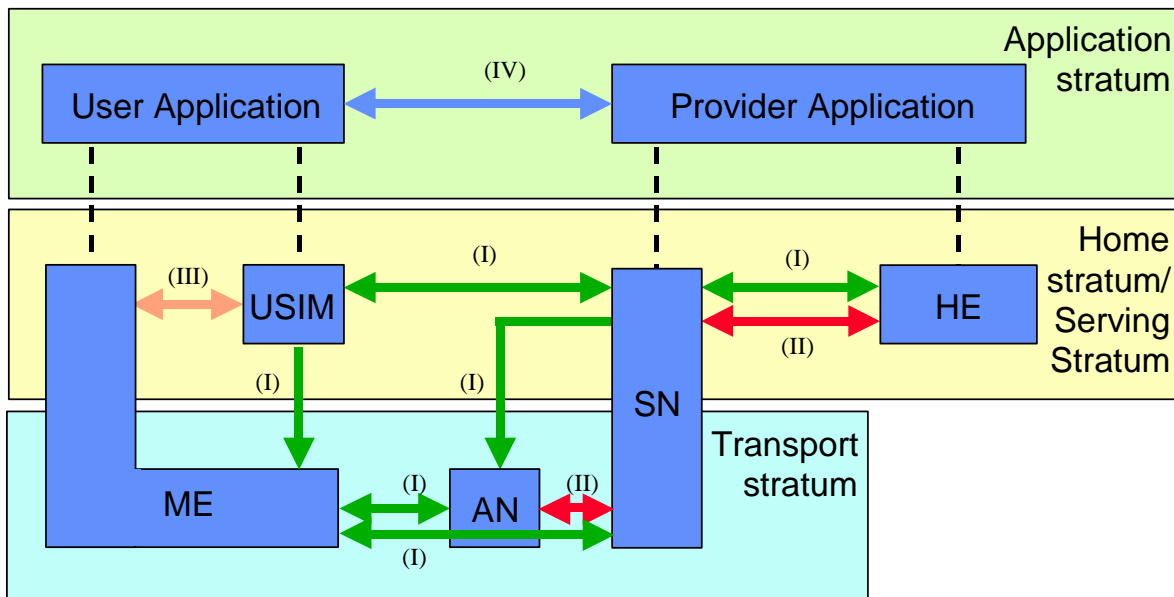
|         |  |
|---------|--|
| AS      | Access Stratum   |
| ASME    | Access Security Management Entity  |
| Cell-ID | CellIdentity as used in TS 36.331 [21]   |
| CP      | Control Plane  |
| C-RNTI  | Cell RNTI as used in TS 36.331 [21]  |
| eKSI    | Key Set Identifier in E-UTRAN  |
| eNB     | 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   |
| KDF     | Key Derivation Function  |
| MAC     | Medium Access Control  |
| MAC-I   | Message Authentication Code for Integrity (terminology of TS36.323 [12])       |
| MACT    | Message Authentication Code T used in AES CMAC calculation                     |
| MME     | Mobility Management Entity   |
| NAS     | Non Access Stratum   |
| NAS-MAC | Message Authentication Code for NAS for Integrity (called MAC in TS24.301 [9]) |
| PCI     | PhysicalCellIdentity as used in TS 36.331 [21]                                 |
| 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, user data (between AN and SN and within AN), 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.

Synchronization of the input parameters for ciphering shall be ensured for the protocols involved in the ciphering.

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 <sub>2</sub> " | EEA0     | Null ciphering algorithm |
| "0001 <sub>2</sub> " | 128-EEA1 | SNOW 3G based algorithm  |
| "0010 <sub>2</sub> " | 128-EEA2 | AES based algorithm      |

The remaining values have been reserved for future use.

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

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

## 5.1.4 User data and signalling data integrity

### 5.1.4.1 Integrity requirements

Synchronization of the input parameters for integrity protection shall be ensured for the protocols involved in the integrity protection.

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

All NAS signaling messages except those explicitly listed in TS 24.301 [9] as exceptions shall be integrity-protected. All RRC signaling messages except those explicitly listed in TS 36.331 [21] as exceptions shall be integrity-protected.

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

### 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<sub>2</sub>" 128-EIA1 SNOW 3G

"0010<sub>2</sub>" 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.

## 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;

The ciphering indicator feature is specified in 3GPP TS 22.101 [23].

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

### 5.3.1 General

The security requirements given in this section apply to all types of eNodeBs. More stringent requirements for specific types of eNodeBs may be defined in other documents.

### 5.3.2 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. The security associations shall be realized according to clause 11 and 12 of this specification.
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.
7. Integrity protection of software transfer towards the eNB shall be ensured.

### 5.3.3 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.

### 5.3.4 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 point and 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, confidentiality 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.

### 5.3.5 Requirements for secure environment of the eNB

The secure environment is logically defined within the eNB and is a composition of functions for the support of sensitive operations.

1. The secure environment shall support secure storage of sensitive data, e.g. long term cryptographic secrets and vital configuration data.
2. The secure environment shall support the execution of sensitive functions, e.g. en-/decryption of user data and the basic steps within protocols which use long term secrets (e.g. in authentication protocols).
3. Sensitive data used within the secure environment shall not be exposed to external entities.
4. The secure environment shall support the execution of sensitive parts of the boot process.
5. The secure environment's integrity shall be assured.
6. Only authorised access shall be granted to the secure environment, i.e. to data stored and used within, and to functions executed within.

## 5.4 Void



## 6 Security Procedures between UE and EPC Network Elements

### 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 [13]

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 in non-volatile memory the EPS NAS security context including the  $K_{ASME}$  resulting from a run of EPS AKA. This EPS NAS security context shall be part of a native EPS security context created by running EPS AKA and it shall not be the mapped EPS NAS security context. If the USIM supports EMM (EPS Mobility Management) parameters storage then the ME shall store the EPS NAS security context (including  $K_{ASME}$  and  $KSI_{ASME}$ ) on the USIM and not keep them in non-volatile ME memory, else the ME shall store the EPS NAS security context in a non-volatile part of its memory.

NOTE 4: If the keys CK, IK resulting from an EPS AKA run were stored in the fields already available on the USIM for storing keys CK and IK 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 stored 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 if the USIM does not support EMM parameters storage.

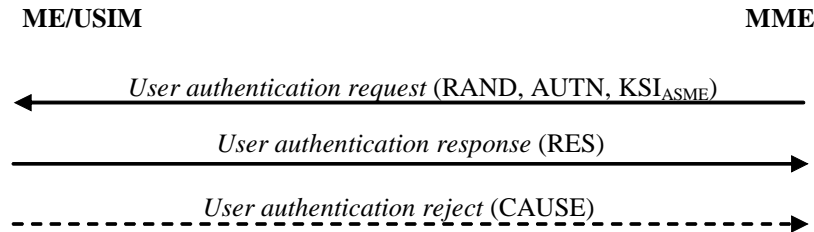
UE shall respond with User authentication response message including RES in case of successful AUTN verification as described in TS 33.102[4] and successful AMF verification as described above. Otherwise UE shall send User authentication reject message with a proper CAUSE value.

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  $K_{ASME}$  shall be generated which is shared between UE and ASME i.e. the MME cfr Figure 6.1.1-1. How this is done is described in subclause 6.1.2.

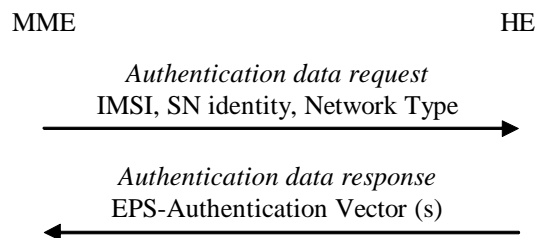


**Figure 6.1.1-1: EPS user authentication (EPS AKA)**

## 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 1: 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.

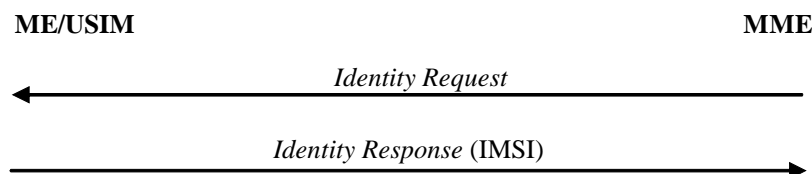
NOTE 2: For  $K_{ASME}$  the possibilities for pre-computation are restricted due to the PLMN-binding.

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).



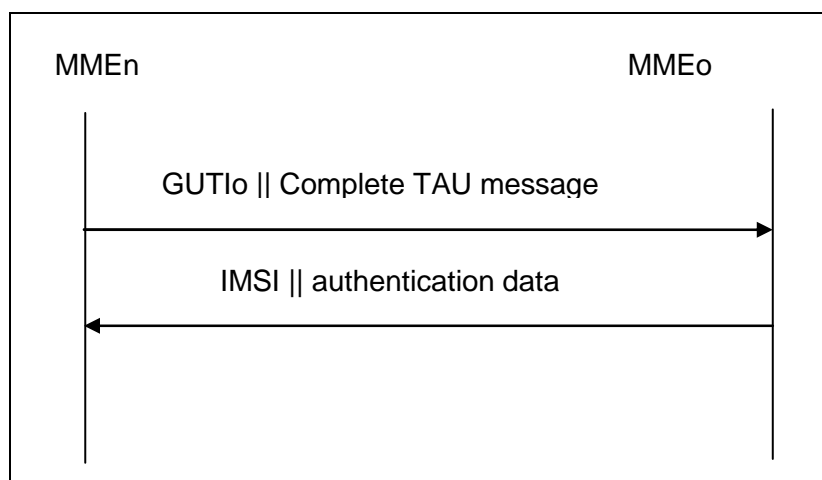
**Figure 6.1.3-1: User identity query**

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 message to the MMEo, this message contains GUTIo and the received TAU message.
- b) The MMEo searches the user data in the database and checks the integrity protection on the TAU message.

If the user is found and the integrity check succeeds, the MMEo shall send a 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 or the integrity check fails, then the MMEo shall send a response indicating that the user identity cannot be retrieved.

- c) If the MMEn receives a 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 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

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

UMTS authentication vectors that were previously received from an SGSN shall not be forwarded between MME's.

- b) SGSN to MME

An SGSN may forward unused UMTS authentication vectors to an MME.

An MME shall not use unused UMTS authentication vectors forwarded from an SGSN in E-UTRAN procedures.

- c) MME to SGSN

UMTS AVs which were previously stored in the MME may be forwarded back towards the same SGSN.

UMTS AVs which were previously stored in the MME shall not be forwarded towards other SGSNs.

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

NOTE: 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 the UE security capabilities to the MME.

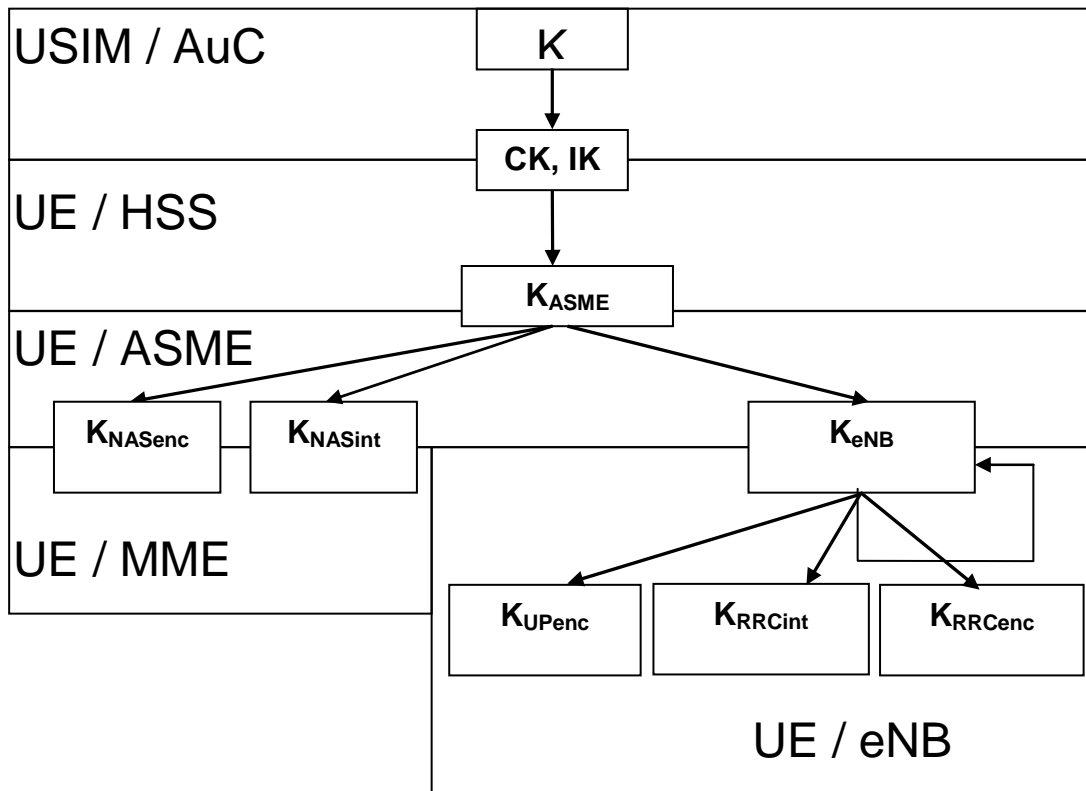


Figure 6.2-1: Key hierarchy in E-UTRAN

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

- $K_{eNB}$  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:

- $K_{NASint}$  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.
- $K_{NASenc}$  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:

- $K_{UPenc}$  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:

- $K_{RRCint}$  is a key, which shall only be used for the protection of RRC traffic with a particular integrity algorithm.  $K_{RRCint}$  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.

Intermediate keys:

- **NH** is a key derived by UE and MME to provide forward security as described in clause 7.2.8. The NH is sent by the MME to the eNB using S1 signalling.
- **$K_{eNB}^*$**  is a key derived by UE and eNB when performing an horizontal or vertical key derivation as specified in clause 7.2.8 using a 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.

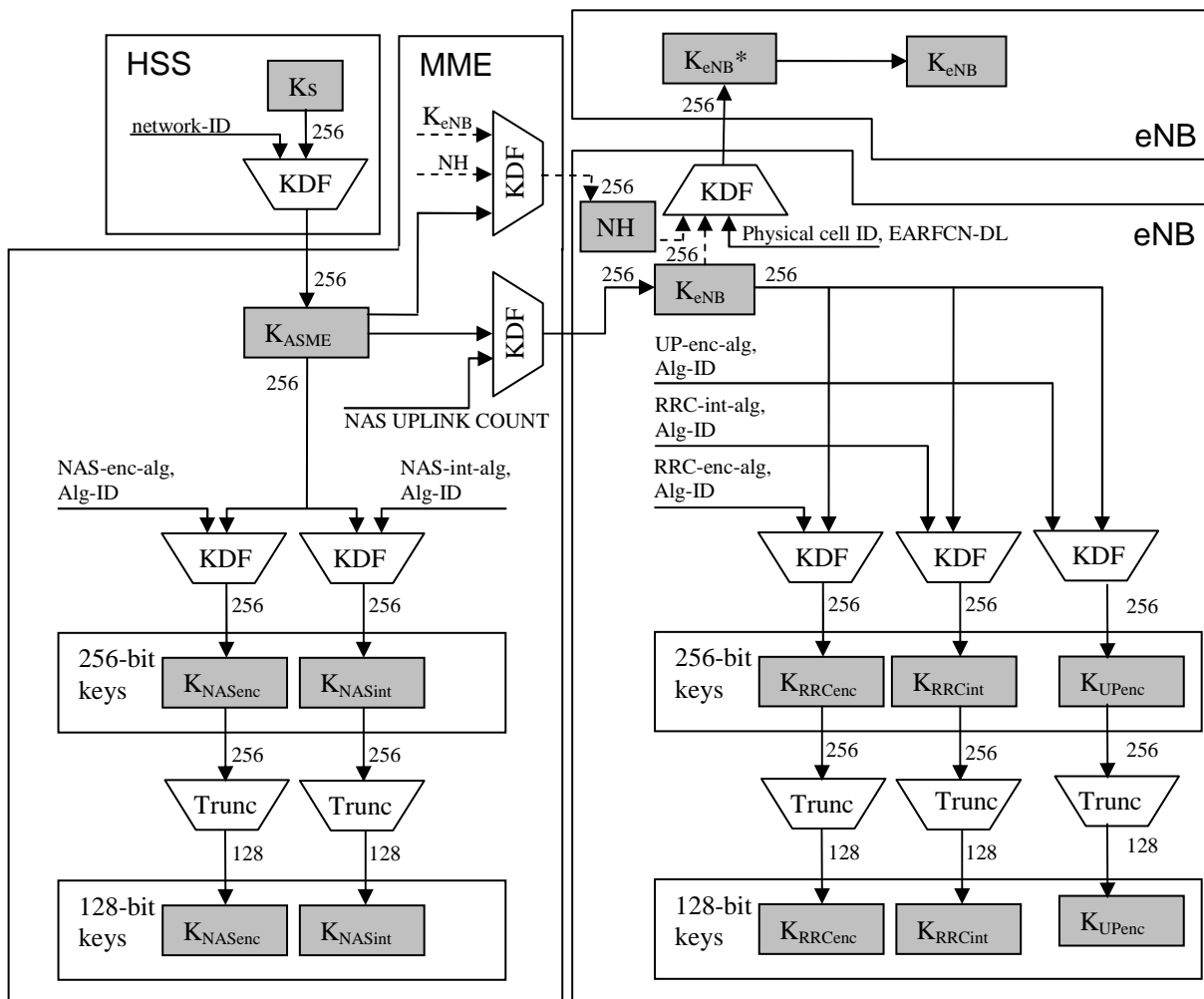
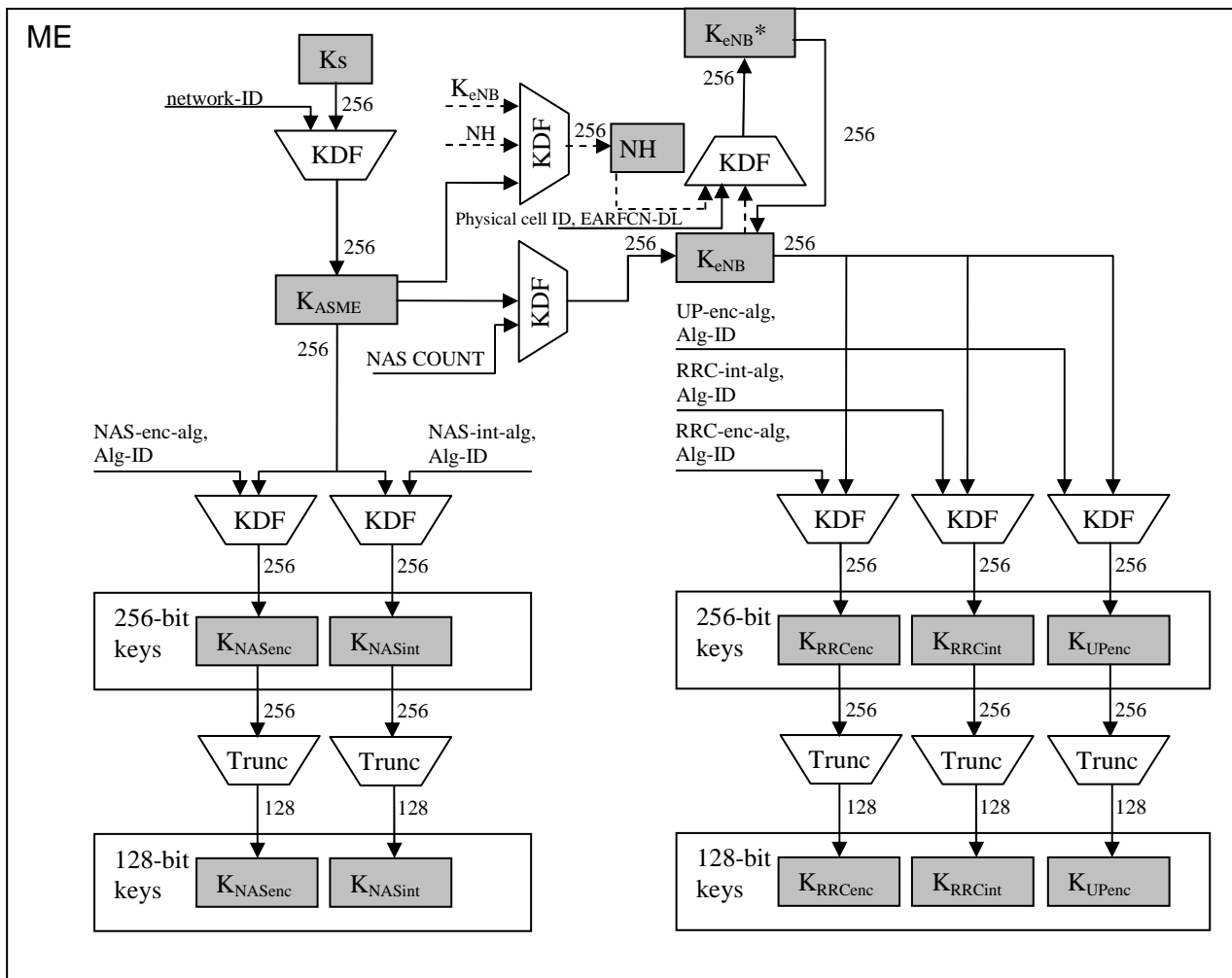


Figure 6.2-2: Key distribution and key derivation scheme for EPS (in particular E-UTRAN) for network nodes. The basic derivations are covered in the figure, but derivations performed at, e.g. inter-RAT mobility is not shown. Two dashed input to a KDF means one of the input is taken depending on condition met.



**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. Two dashed input to a KDF means one of the input is taken depending on condition met.

As the figures 6.2-2 and 6.2-3 show, the length of  $K_{ASME}$  and  $K_{eNB}$  is 256 bits, the length of NH is 256bits, 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.

NOTE: Figures 6.2-2 and 6.2-3 do not include the key handling branches for forward security. This is described in clause 7.2.8 and Figure 7.2.8.1-1.

The function Trunc takes as input a 256-bit string, and returns a truncated output as defined in Annex A.7.

### 6.3 EPS key identification

The key  $K_{ASME}$  shall be identified by the key set identifier eKSI. eKSI may be either of type  $KSI_{ASME}$  or of type  $KSI_{SGSN}$ . An eKSI shall be stored in the UE and the MME together with  $K_{ASME}$  and the temporary identifier GUTI, if available.

NOTE 1: the GUTI points to the MME where the  $K_{ASME}$  is stored.

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

The key set identifier  $KSI_{SGSN}$  is a parameter which is associated with the mapped  $K_{ASME}$  derived from UMTS keys during inter-RAT mobility, cf. clauses 9 and 10 of the present specification. The key set identifier  $KSI_{SGSN}$  is generated in both the UE and the MME respectively when deriving the mapped  $K_{ASME}$  during idle procedures in E-UTRAN and during handover from GERAN/UTRAN to E-UTRAN. The  $KSI_{SGSN}$  is stored together with the mapped  $K_{ASME}$ .

The purpose of the  $KSI_{ASME}$  is to make it possible for the UE and the MME to identify a native  $K_{ASME}$  without invoking the authentication procedure. This is used to allow re-use of the  $K_{ASME}$  during subsequent connection set-ups.

The purpose of the  $KSI_{SGSN}$  is to make it possible for the UE and the MME to indicate the use of the mapped  $K_{ASME}$  in inter-RAT mobility procedures. For details cf. clauses 9 and 10.

$KSI_{ASME}$  and  $KSI_{SGSN}$  have the same format. The format of eKSI shall allow a recipient of such a parameter to distinguish whether the parameter is of type ' $KSI_{ASME}$ ' or of type ' $KSI_{SGSN}$ '. The format shall further contain a value field. The value fields of  $KSI_{ASME}$  and  $KSI_{SGSN}$  are three bits each. 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.

NOTE 2: In addition to EPS security contexts, the UE may also cache UMTS security contexts. These UMTS security contexts are identified by the KSI, as defined in TS 33.102 [4].

## 6.4 Handling of EPS security contexts

Any EPS security context shall be deleted from the ME if:

- a) the UICC is removed from the ME when the ME is in power on state;
- b) the ME is powered up and the ME discovers that a UICC different from the one which was used to create the EPS security context has been inserted to the ME;
- c) the ME is powered up and the ME discovers that no UICC has been inserted to the ME.

$K_{ASME}$  shall never be transferred from the EPC to an entity outside the EPC.

Storage of the EPS NAS security context in the UE during power-off:

- a) The EPS NAS security context to be stored during the power-off shall be part of a native EPS security context, and hence not be part of a mapped EPS security context.
- b) If the USIM supports EMM parameters storage, then the ME shall store the EPS NAS security context parameters on the USIM, mark the EPS NAS security context on the USIM as valid, and not keep the EPS NAS security context in non-volatile ME memory.
- c) If the USIM does not support EMM parameters storage, then the ME shall store the EPS NAS security context in a non-volatile part of its memory, and mark the EPS NAS security context in its non-volatile memory as valid.

After power-on of the ME, the ME shall retrieve EPS NAS security context stored on the USIM if the USIM supports EMM parameters storage and if the stored EPS NAS security context on the USIM is marked as valid. Otherwise the ME shall retrieve the stored EPS NAS security context from its non-volatile memory if the EPS NAS security context is marked as valid.



## 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 UE in order to support the subscriber identity confidentiality. The GUTI is defined in TS 23.003 [3].

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). A new GUTI shall be sent to the UE only after a successful activation of NAS security.

### 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 UE 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 the MME, the eNB and the UE as part of the security mode set-up procedure (see subclause 7.2.4).

#### 7.2.2 E-UTRAN key identification

Clause 6.3 of this specification states how the key  $K_{ASME}$  is identified, namely by the key set identifier eKSI. Keys  $K_{NASenc}$  and  $K_{NASint}$  in the E-UTRAN key hierarchy specified in clause 6.2, which are derived from  $K_{ASME}$ , can be uniquely identified by eKSI together with those parameters from the set {uplink NAS COUNT, algorithm distinguisher, algorithm identifier}, which are used to derive these keys from  $K_{ASME}$  according to Annex A.

The intermediate key NH as defined in clause 7 uniquely can be determined by the key set identifier, eKSI, together with next hop chaining count, NCC.

Intermediate key  $K_{eNB}^*$ , defined in clause 7, as well as keys  $K_{eNB}$ ,  $K_{RRCint}$ ,  $K_{RRCenc}$ , and  $K_{UPenc}$  in the E-UTRAN key hierarchy specified in clause 6.2, which are derived from  $K_{ASME}$ , can be uniquely identified by eKSI together with those parameters from the set {uplink NAS COUNT, algorithm distinguisher, algorithm identifier, NCC, and sequence of PCIs and EARFCN-DLs used in horizontal key derivations with this NCC}, which are used to derive these keys from  $K_{ASME}$  according to clause 7 and Annex A.

It is specified in the remainder of clause 7, as well as in clause 9 and 10, which of the above parameters need to be included in a security-relevant message to allow the entity receiving the message to uniquely identify a certain key.

#### 7.2.3 E-UTRAN key lifetimes

All E-UTRAN keys are derived based on a  $K_{ASME}$ . The key hierarchy which is described in clause 6.2 does not allow direct update to RRC and UP keys, 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 handling (creation, modification and update) of the E-UTRAN keys in the various state transitions is described in clauses 7.2.5, 7.2.6, 7.2.7 and 7.2.8.

$K_{ASME}$  shall be created only by running a successful AKA or by the inter-RAT procedures towards E-UTRAN (cf clauses 9 and 10). In case  $K_{ASME}$  is invalidated by the UE, a  $KSI_{ASME}$  with value "111" shall be sent by the UE to the network, which 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)
- b) The serving network shall select the algorithms to use dependent on
  - the UE security capabilities of the UE,
  - the configured allowed list of security capabilities of the currently serving network entity
- c) 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, i.e. 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 EPS security capabilities to the eNB. The eNB shall choose the ciphering algorithm which has the highest priority from its configured list and is also present in the UE EPS security capabilities. The eNB shall choose the integrity algorithm which has the highest priority from its configured list and is also present in the UE 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 algorithm 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). The chosen algorithms shall be indicated to the UE in the handover command. 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 (possibly 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). The chosen algorithms shall be indicated to the UE in the handover command. 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

Each MME shall be configured via network management with lists of algorithms which are allowed for usage. There shall be one list for NAS integrity algorithms, and one for NAS ciphering algorithms. These lists shall be ordered according to a priority decided by the operator.

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 and one NAS integrity protection algorithm, and indicate them in the corresponding integrity protected message to UE and shall also include the UE security capabilities into that message. The UE shall verify that the message from the MME contains the correct UE security capabilities. If so, the UE shall reply with an integrity protected SMC Complete message, protected by the integrity algorithm selected by the MME. 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 the ordered lists.

#### 7.2.4.3.2 MME change

In case there is change of MMEs and algorithms to be used for NAS, the target MME shall send an integrity protected Security Mode Command message which includes integrity and ciphering algorithms selected for NAS protection and EPS security capabilities. The UE verifies that the message from the MME contains the correct UE security capabilities. If so, the UE shall reply with an integrity protected SMC Complete message. The MME shall select the NAS algorithms which have the highest priority according to the ordered lists (see 7.2.4.3.1).

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.

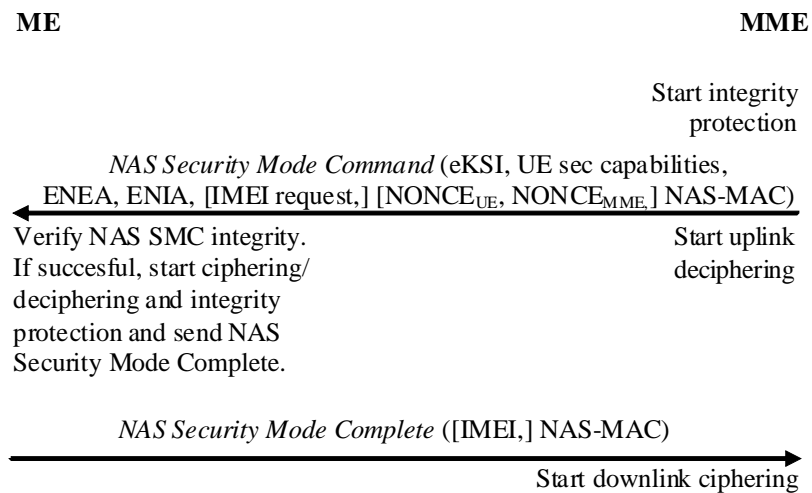
The NAS security mode command message from MME to UE shall contain the replayed UE security capabilities, the selected NAS algorithms, the eKSI for identifying  $K_{ASME}$ , and both NONCE<sub>ue</sub> and NONCE<sub>mme</sub> in case as specified in section 9.1.2. This message shall be integrity protected with NAS integrity key based on  $K_{ASME}$  indicated by the eKSI in the message. See figure 7.2.4.4-1.

UE shall verify the integrity of the NAS security mode command message. If successfully verified, UE shall start 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 eKSI in the NAS security mode command message.

NAS downlink ciphering at the MME shall start after receiving the NAS security mode complete message. NAS uplink deciphering at the MME starts after sending the NAS security mode command message. NAS uplink ciphering and downlink deciphering at the UE shall start after receiving and successfully verifying the NAS security mode command message. The NAS security mode complete message shall include IMEI in case MME requested it in the NAS SMC Command message.

If any verification of the NAS security mode command is not successful in the ME, the ME shall reply with an unprotected NAS security mode reject message (see TS 24.301 [9]).

Only after EPS AKA the NAS security mode command message shall reset 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 eKSI or due to the algorithms change).



**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 7.2.4.5-1.

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

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 the current  $K_{ASME}$ .

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 in the ME, the ME shall reply with an unprotected security mode failure message (see TS 36.331[21]).

AS security mode command always changes the AS keys.

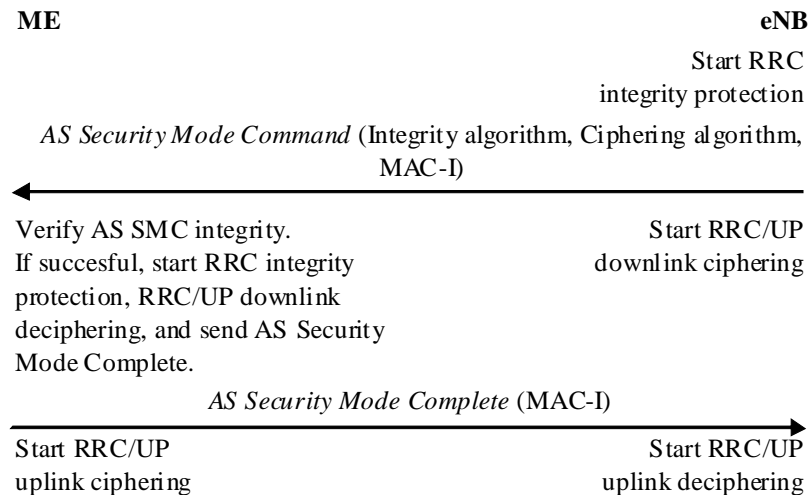


Figure 7.2.4.5-1: AS security setup

## 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 shall be removed from the UE and MME with the exception of:
      - native EPS NAS security context (as in clause 6.1.1), which should remain stored in the MME of the UE, and
      - any unused authentication vectors, which may remain stored in the MME.
    - ii. If the reason is not switch off then MME and UE shall keep all authentication data with the following exception: the UE and the MME shall delete the mapped security context resulting from inter-RAT handover or idle mode mobility as defined in clauses 9 and 10.
  - b. MME-initiated
    - i. Explicit: authentication data shall be kept in the UE and MME if the detach type is re-attach with the following exception: the UE and the MME shall delete the mapped security context resulting from inter-RAT handover or idle mode mobility as defined in clauses 9 and 10.
    - ii. Implicit: authentication data shall be kept in the MME with the following exception: the UE and the MME shall delete the mapped security context resulting from inter-RAT handover or idle mode mobility as defined in clauses 9 and 10.
  - c. HSS-initiated: If the message is "subscription withdrawn" then all authentication data shall be removed from the UE and MME.

If the USIM supports EMM parameters storage then the ME shall update the EPS NAS security context parameters on the USIM with the current values of the native EPS security context and mark the EPS NAS security context on the USIM as valid. Otherwise, the ME shall update the EPS NAS security context in its non-

volatile memory with its current values of the native EPS security context and mark the EPS NAS security context in its non-volatile memory as valid.

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.

For the case that the MME or the UE enter EMM-DEREGISTERED state without using any of the above procedures, the handling of authentication data shall be as specified in TS 24.301 [9].

## 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 EPS NAS security context

If there is an EPS 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 eKSI indicating the current  $K_{ASME}$ .

When the UE receives the AS SMC, it uses the NAS COUNT of 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.

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.

If the USIM supports EMM parameters storage, the ME shall mark the stored EPS NAS security context on the USIM as invalid at the end of the transition away from EMM-DEREGISTERED. Otherwise, the ME shall mark the stored EPS NAS security context on its non-volatile memory as invalid at the end of the transition.

#### 7.2.5.2.3 With run of EPS AKA

If there is no EPS NAS security context available an EPS AKA run is required. If there is an EPS NAS security context available the MME may decide to run an EPS 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 COUNT shall 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 eKSI 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 COUNT in this case cannot lead to the same combination of  $K_{ASME}$  and NAS COUNT being 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 COUNT that 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.

If the USIM supports EMM parameters storage, the ME shall mark the stored EPS NAS security context on the USIM as invalid. Otherwise, the ME shall mark the stored EPS NAS security context on its non-volatile memory as invalid.

## 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, an 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 next-in-sequence uplink NAS sequence number.

Upon receipt of the NAS Service Request message, the MME shall derive key  $K_{eNB}$  as specified in Annex A.3 using the NAS COUNT [9] corresponding to the NAS Service Request and initialize the value of the Next hop Chaining Counter (NCC) to one. The MME shall further derive a next hop parameter NH as specified in Annex A.4 using the newly derived  $K_{eNB}$  as basis for the derivation. This fresh {NH, NCC=1} pair shall be stored in the MME and shall be used for the next forward security key derivation. The MME shall communicate the  $K_{eNB}$  pair and  $KSI_{ASME}$  to the serving eNB in the S1-AP procedure INITIAL CONTEXT SETUP.

As a result of the NAS Service Request, radio bearers are established, and the eNB sends an AS SMC to the UE. When the UE receives the AS SMC including the current eKSI, 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. The UE shall further derive the NH parameter from the newly derived  $K_{eNB}$  in the same way as the MME. From the  $K_{eNB}$  the RRC protection keys and the UP protection keys are derived by the UE and the eNB as described in subclause 6.2.

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

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.

If the USIM supports EMM parameters storage, the ME shall mark the stored EPS NAS security context on the USIM as invalid. Otherwise, the ME shall mark the stored EPS NAS security context in its non-volatile memory as invalid.

### 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 NH and NCC if available, as well as the AS security context.
- MME and the UE shall keep the EPS NAS security context stored. MME shall delete NH and NCC.

If the USIM supports EMM parameters storage, then the ME shall update the EPS NAS security context parameters on the USIM with its current values of the native EPS NAS security context and mark the EPS NAS security context on the USIM as valid. Otherwise, the ME shall update the EPS NAS security context in its non-

volatile memory with the current values of the native EPS NAS security context and mark the EPS NAS security context in its non-volatile memory as valid.

### 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 COUNT of 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 followed by a NAS SMC from the MME as part of the TAU procedure, the uplink and downlink NAS COUNT shall be set to the start values.

In the case source and target MME use different NAS algorithms, the target MME re-derives the NAS keys from  $K_{ASME}$  with the new algorithm identities 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 identity specified in the NAS SMC.

### 7.2.8 Key handling in handover

#### 7.2.8.1 General

##### 7.2.8.1.1 Access stratum

The general principle of key handling at handovers is depicted in Figure 7.2.8.1-1.

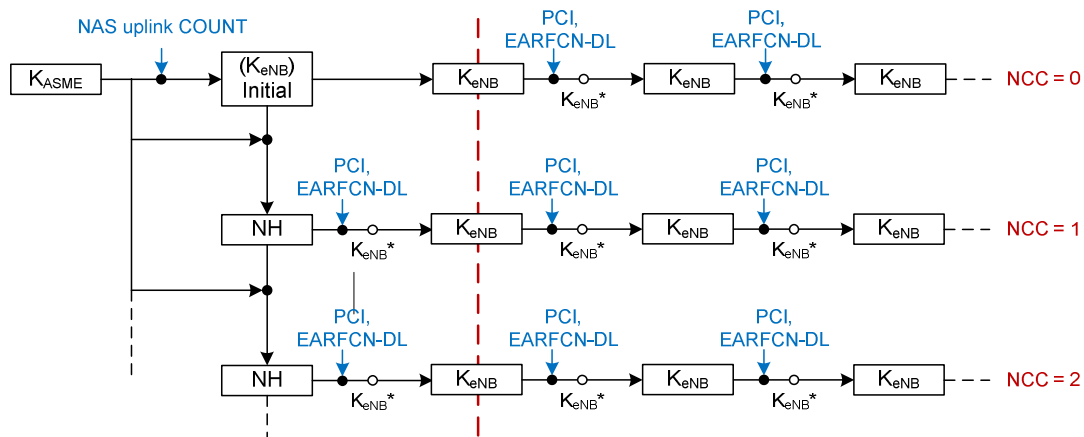


Figure 7.2.8.1-1 Model for the handover key chaining

The following is an outline of the key handling model to clarify the intended structure of the key derivations. The detailed specification is provided in subclauses 7.2.8.3 and 7.2.8.4.

Whenever an initial AS security context needs to be established between UE and eNB, MME and the UE shall derive a  $K_{eNB}$  and a Next Hop parameter (NH). The  $K_{eNB}$  and the NH are derived from the  $K_{ASME}$ . A NH Chaining Counter (NCC) is associated with each  $K_{eNB}$  and NH parameter. Every  $K_{eNB}$  is associated with the NCC corresponding to the NH value from which it was derived. At initial setup, the  $K_{eNB}$  is derived directly from  $K_{ASME}$ , and is then considered to be associated with a virtual NH parameter with NCC value equal to zero. At initial setup, the derived NH value is associated with the NCC value one.

Whether the MME sends the  $K_{eNB}$  key or the {NH, NCC} pair to the serving eNB is described in detail in subclauses 7.2.8.3 and 7.2.8.4. The MME shall not send the NH value to eNB at the initial connection setup.

The UE and the eNB use the  $K_{eNB}$  to secure the communication between each other. On handovers, the basis for the  $K_{eNB}$  that will be used between the UE and the target eNB, called  $K_{eNB}^*$ , is derived from either the currently active  $K_{eNB}$  or from the NH parameter. If  $K_{eNB}^*$  is derived from the currently active  $K_{eNB}$  this is referred to as a horizontal key



derivation (see Figure 7.2.8.1-1) and if the  $K_{eNB}^*$  is derived from the NH parameter the derivation is referred to as a vertical key derivation (see Figure 7.2.8.1-1). On handovers with vertical key derivation the NH is further bound to the target PCI and its frequency EARFCN-DL before it is taken into use as the  $K_{eNB}$  in the target eNB.

As NH parameters are only computable by the UE and the MME, it is arranged so that NH parameters are provided to eNBs from the MME in such a way that forward security can be achieved.

### 7.2.8.1.2 Non access stratum

A NAS aspect that needs to be considered is possible NAS algorithm change at MME change that could occur at a handover. At an eNB handover with MME relocation, 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

The NAS COUNTs shall only be reset to the start value in the following cases:

- for an EPS NAS security context created by an AKA run successfully followed by a NAS SMC,
- or for an EPS NAS security context created through a context mapping during a handover from UTRAN/GERAN to E-UTRAN,
- or for an EPS NAS security context created through a context mapping during idle mode mobility from UTRAN/GERAN to E-UTRAN.

The NAS COUNT shall not be reset during idle mode mobility or handover for an already existing native EPS NAS security context.

In case the target MME decides to use NAS algorithms different from the ones used by the source MME, a NAS SMC including eKSI (new or current value depending on whether AKA was run or not) shall be sent from the MME to the UE.

The start value of NAS COUNT shall be zero (0).

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

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 X2 or S1 handovers for a particular eNB according to the security characteristics of a particular eNB.

### 7.2.8.2 Void

### 7.2.8.3 Key derivations for context modification procedure

As outlined in subclause 7.2.8.1, whenever a fresh  $K_{eNB}$  is calculated from the  $K_{ASME}$  (as described in Annex A.3), the MME and the UE shall also compute the NH parameter from the  $K_{ASME}$  and the fresh  $K_{eNB}$  as described in Annex A.4. An NCC value 1 is associated with the NH parameter and NCC value 0 with the  $K_{eNB}$ . The MME shall transfer the  $K_{eNB}$  to the serving eNB in the message establishing the security context in the eNB. The UE shall compute  $K_{eNB}$  and NH in the same way as the MME.

### 7.2.8.4 Key derivations during handovers

#### 7.2.8.4.1 Intra-eNB Handover

When the eNB decides to perform an intra-eNB handover it shall derive  $K_{eNB}^*$  as in Annex A.5 using target PCI and either NH or the current  $K_{eNB}$  depending on the following criteria: the eNB shall use the NH for deriving  $K_{eNB}^*$  if an unused {NH, NCC} pair is available in the eNB (this is referred to as a vertical key derivation), otherwise if no unused

{NH, NCC} pair is available in the eNB, the eNB shall derive  $K_{eNB}^*$  from the current  $K_{eNB}$  (this is referred to as a horizontal key derivation).

The eNB shall use the  $K_{eNB}^*$  as the  $K_{eNB}$  after handover. The eNB shall send the NCC used for  $K_{eNB}^*$  derivation to UE in HO Command message.

#### 7.2.8.4.2 X2-handover

As in intra-eNB handovers, for X2 handovers the source eNB shall perform a vertical key derivation in case it has a fresh {NH,NCC} pair. The source eNB shall first compute  $K_{eNB}^*$  from target PCI, its frequency EARFCN-DL, and either from currently active  $K_{eNB}$  in case of horizontal key derivation or from the NH in case of vertical key derivation as described in Annex A.5. The target eNB shall associate the NCC value received from source eNB with the  $K_{eNB}$ .

Next the source eNB shall forward the { $K_{eNB}^*$ , NCC} pair to the target eNB. The target eNB shall include the received NCC into the prepared HO Command message, which is sent back to the source eNB in a transparent container and forwarded to the UE by source eNB.

The target eNB shall use the received  $K_{eNB}^*$  directly as  $K_{eNB}$  to be used with the UE.

When the target eNB has completed the handover signaling with the UE, it shall send a S1 PATH SWITCH REQUEST to the MME. Upon reception of the S1 PATH SWITCH REQUEST, the MME shall increase its locally kept NCC value by one and compute a new fresh NH by using the  $K_{ASME}$  and its locally kept NH value as input to the function defined in Annex A.4. The MME shall then send the newly computed {NH, NCC} pair to the target eNB in the S1 PATH SWITCH REQUEST ACKNOWLEDGE message. The target eNB shall store the received {NH, NCC} pair for further handovers and remove other existing unused stored {NH, NCC} pairs if any.

NOTE 1: In case the S1 PATH SWITCH REQUEST ACKNOWLEDGE was not received by the target eNB before the next handover or the PATH SWITCH REQUEST ACKNOWLEDGE was lost, the target eNB does not hold a {NH, NCC} pair, and can hence only perform a horizontal key derivation at the next handover.

NOTE 2: 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, for X2-handovers key separation happens only after two hops because the source eNB knows the target eNB keys. The target eNB can immediately initiate an intra-cell handover to take the new NH into use once the new NH has arrived in the S1 PATH SWITCH ACKNOWLEDGE.

#### 7.2.8.4.3 S1-Handover

When an S1-handover is performed, the source eNB shall not send any keys to the MME in the S1 HANDOVER REQUIRED message.

Upon reception of the HANDOVER REQUIRED message the source MME shall send the {NH, NCC} pair stored within the EPS NAS security context to the target MME in the S10 FORWARD RELOCATION REQUEST message if source MME is different from target MME. The target MME shall store locally the {NH, NCC} pair received from the source MME.

The target MME shall achieve forward security by computing a fresh {NH, NCC} pair from the EPS NAS security context {NH, NCC} pair received from the source MME using the function defined in Annex A.4. The target MME shall then send the fresh {NH NCC} pair to the target eNB within the S1 HANDOVER REQUEST.

Upon receipt of the S1 HANDOVER REQUEST from the target MME, the target eNB shall compute the  $K_{eNB}$  to be used with the UE by performing the key derivation defined in Annex A.5 with the fresh {NH, NCC} pair in the S1 HANDOVER REQUEST and the target PCI and its frequency EARFCN-DL. The target eNB shall include the NCC value from the fresh {NH,NCC} pair into the HO Command to the UE and remove any existing unused stored {NH, NCC} pairs.

NOTE: The source eNB may be the same as the target eNB and the source MME may be the same as the target MME in the description in this subclause.

#### 7.2.8.4.4 UE handling

The UE behaviour is the same regardless if the handover is S1, X2 or intra-eNB.

If the NCC value the UE received in the HO Command message from target eNB via source eNB is equal to the NCC value associated with the currently active  $K_{eNB}$ , the UE shall derive the  $K_{eNB}^*$  from the currently active  $K_{eNB}$  and the target PCI and its frequency EARFCN-DL using the function defined in Annex A.5. The UE shall use the  $K_{eNB}^*$  as the  $K_{eNB}$  when communicating with the target eNB.

If the UE received an NCC value that was different from the NCC associated with the currently active  $K_{eNB}$ , the UE shall first synchronize the locally kept NH parameter by computing the function defined in Annex A.4 iteratively (and increasing the NCC value until it matches the NCC value received from the source eNB via the HO command message. When the NCC values match, the UE shall compute the  $K_{eNB}^*$  from the synchronized NH parameter and the target PCI and its frequency EARFCN-DL using the function defined in Annex A.5.

## 7.2.9 Key-change-on-the fly

### 7.2.9.1 General

Key-change-on-the fly consists of re-keying or key-refresh.

Key refresh shall be possible for  $K_{eNB}$ ,  $K_{RRC-enc}$ ,  $K_{RRC-int}$ , and  $K_{UP-enc}$  and shall be initiated by the eNB when the PDCP COUNTs are about to wrap around. The procedure is described in clause 7.2.9.3.

Re-keying shall be possible for the  $K_{eNB}$ ,  $K_{RRC-enc}$ ,  $K_{RRC-int}$ , and  $K_{UP-enc}$ . This re-keying shall be initiated by the MME when an EPS AS security context different from the currently active one shall be activated. The procedures for doing this are described in clause 7.2.9.2.

Re-keying shall be possible for  $K_{NAS-enc}$  and  $K_{NAS-int}$ . Re-keying of  $K_{NAS-enc}$  and  $K_{NAS-int}$  shall be initiated by the MME when a NAS EPS security context different from the currently active one shall be activated. The procedures for doing this are described in clause 7.2.9.4.

Re-keying of the entire EPS key hierarchy including  $K_{ASME}$  shall be achieved by first re-keying  $K_{ASME}$ , then  $K_{NAS-enc}$  and  $K_{NAS-int}$ , followed by re-keying of the  $K_{eNB}$  and derived keys. For NAS key change-on-the fly, activation of NAS keys is accomplished by a NAS SMC procedure.

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, activation of a native context after handover from UTRAN or GERAN.

### 7.2.9.2 $K_{eNB}$ re-keying

The re-keying procedure is initiated by the MME after a successful AKA run with the UE to activate a new EPS security context, or to re-activate a native EPS security context after handover from GERAN or UTRAN according to clauses 9.2.2.1 and 10.3.2.

In case 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 eKSI 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 has determined that the eKSI has changed, 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 re-keying procedure is initiated by the MME to re-activate a native EPS security context after handover from GERAN or UTRAN the same procedure as above applies with the following differences:

- The UE and the MME shall derive the new  $K_{eNB}$  from the native  $K_{ASME}$  and the native EPS security context NAS uplink COUNT of the NAS Security Mode Complete message.

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 an intra-cell handover with the sole purpose of a  $K_{eNB}$  re-keying.

NOTE: It is left to stage 3 specifications whether re-keying and inter-cell handover can be combined, and, if not, in which order these two procedures are executed.

### 7.2.9.3 KeNB refresh

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.

### 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, according to Annex A.7.

To re-activate a native EPS security context after handover from GERAN or UTRAN, cf. clause 9.2.2 B step 7, the UE and the MME take the NAS keys into use by running a NAS SMC procedure according to clause 7.2.4.5.

MME shall activate fresh NAS keys from an EPS AKA run or activate native security context with sufficiently low NAS COUNT values before the NAS uplink or downlink COUNT wraps around with the current security context.

## 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 use and mode of operation of the 128-EEA algorithms are specified in Annex B.

The input parameters to the 128-bit EEA algorithms as described in Annex B are an 128-bit cipher key  $K_{UPenc}$  as KEY, a 5-bit bearer identity BEARER which value is assigned as specified by TS 36.323 [12], 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 shall be provided by the PDCP layer between UE and eNB and no layers below PDCP shall be integrity protected.

The use and mode of operation of the 128-EIA algorithms are specified in Annex B.

The input parameters to the 128-bit EIA algorithms as described in Annex B are an 128-bit integrity key  $K_{RRCint}$  as KEY, a 5-bit bearer identity BEARER which value is assigned as specified by TS 36.323 [12], 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.

The supervision of failed RRC integrity checks shall be performed both in the ME and the eNB. In case of failed integrity check (i.e. faulty or missing MAC-I) is detected after the start of integrity protection, the concerned message shall be discarded. This can happen on the eNB side or on the ME side.

NOTE: This text does not imply that the concerned message is silently discarded. In fact, TS 36.331 [18] specifies that the UE shall trigger a recovery procedure upon detection of a failed RRC integrity check. When the cause for integrity protection failure is not a context mismatch, such as a key or HFN mismatch, the run of a recovery procedure unnecessarily adds load to the system. However, in the absence of a means for the UE to reliably detect the cause of an integrity protection failure and the fact that the only identified consequence of an active attack is limited to non-persistent DoS effects, priority was given to a procedure allowing recovery from the deadlock caused by a context mismatch.

## 7.4.2 RRC confidentiality mechanisms

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

The use and mode of operation of the 128-EEA algorithms are specified in Annex B.

The input parameters to the 128-bit EEA algorithms as described in Annex B are an 128-bit cipher Key  $K_{\text{RRCenc}}$  as KEY, a 5-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.

## 7.4.3 $K_{\text{eNB}}^*$ and token preparation for the RRCConnectionRe-establishment Procedure

The  $K_{\text{eNB}}^*$  and token calculation at handover preparation are cell specific instead of eNB specific. At potential RRC Connection re-establishment (e.g. in handover failure case), the UE may select a cell different from the target cell to initiate the re-establishment procedure. To ensure that the UE RRCConnectionRe-establishment attempt is successful when the UE selects another cell under the control of the target eNB at handover preparation, the serving eNB should prepare multiple  $K_{\text{eNB}}^*$ s and tokens for multiple cells which are under the control of the target eNB.

The preparation of these cells includes sending security context containing a  $K_{\text{eNB}}^*$  and a token for each cell to be prepared, as well as the corresponding NCC and UE supported security algorithms to the target eNB. The source eNB shall derive the  $K_{\text{eNB}}^*$ s as described in Annex A.5 based on the corresponding target cell's physical cell ID and frequency. Additionally, the source eNB shall derive and the tokens as described below.

When an RRCConnectionReestablishmentRequest is initiated by the UE to access one of the cells which is under in the target eNB's control, the UE shall use the token and the  $K_{\text{eNB}}^*$  that correspond to the selected cell. The UE shall derive  $K_{\text{eNB}}^*$  as described in Annex A.5 based on the target cell's physical cell ID and frequency. Additionally, the UE shall derive the token as described below.

This message is transmitted over SRB0 and hence not integrity protected. The RRCConnectionReestablishmentRequest shall contain the token corresponding to the cell the UE tries to reconnect to.

In order to calculate the token, the negotiated EIA-algorithm from the AS Security context from the source eNB shall be used with the following inputs: source C-RNTI, source PCI and target Cell-ID as defined by *VarShortMAC-Input* in TS 36.331 [18], where source PCI and source C-RNTI are associated with the cell the UE last had an active RRC connection with and target cell ID is the identity of the target cell where the RRCConnectionReestablishmentRequest is sent to. PCI is defined in TS 36.211 [19], C-RNTI in TS 36.331 [18] and Cell-ID in TS 36.401 [20].

- KEY shall be set to  $K_{\text{RRCint}}$  of the source cell;
- all BEARER bits shall be set to 1;
- DIRECTION bit shall be set to 1;
- all COUNT bits shall be set to 1.

The token shall be the 16 least significant bits of the output of the used integrity algorithm.

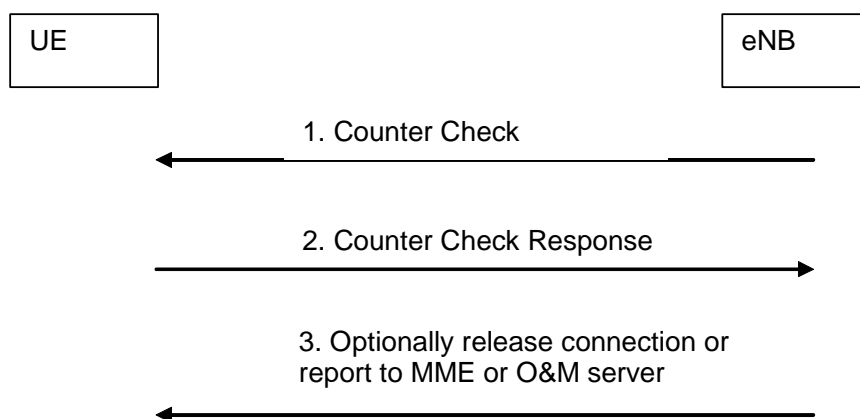
The target eNB receiving the RRCConnectionReestablishmentRequest shall respond with an RRCConnectionReestablishment message containing the NCC received during the preparation phase if the token is valid, otherwise the target eNB shall reply with an RRCConnectionReestablishmentReject message. The RRCConnectionReestablishment and RRCConnectionReestablishmentReject messages are also sent on SRB0 and hence not integrity protected. Next the target eNB and UE shall derive  $K_{\text{eNB}}$ . The UE shall respond with an integrity

protected and ciphered RRCReestablishmentComplete on SRB1. The RRCConnectionReconfiguration procedure used to re-establish the remaining radio bearers shall only include integrity protected and ciphered messages.

## 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

Integrity protection for NAS signalling messages shall be provided as part of the NAS protocol.

#### 8.1.1 NAS input parameters

Input parameters to the NAS 128-bit integrity algorithms as described in Annex B are an 128-bit integrity key  $K_{NASint}$  as KEY, an 5-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 zeros.
- 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.

The use and mode of operation of the 128-bit integrity algorithms are specified in Annex B.

The supervision of failed NAS integrity checks shall be performed both in the ME and the MME. In case of failed integrity check (i.e. faulty or missing NAS-MAC) is detected after the start of NAS integrity protection, the concerned message shall be discarded except for some NAS messages specified in TS 24.301 [9]. For those exceptions the MME shall take the actions specified in TS 24.301 [9] when receiving a NAS message with faulty or missing NAS-MAC. Discarding NAS messages can happen on the MME side or on the ME side.

#### 8.1.2 NAS integrity activation

NAS integrity shall be activated with the help of the NAS SMC procedure immediately after successful 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.

The use and mode of operation of the 128-EIA algorithms are specified in Annex B.

### 8.2 NAS confidentiality mechanisms

The input parameters for the NAS 128-bit ciphering algorithms shall be the same as the ones used for NAS integrity protection as described in clause 8.1, with the exception that a different key,  $K_{NASenc}$ , is used as KEY, and there is an additional input parameter, namely the length of the key stream to be generated by the encryption algorithms.

The use and mode of operation of the 128-bit ciphering algorithms are specified in Annex B.

## 9 Security interworking between E-UTRAN and UTRAN

### 9.1 Idle mode procedures

#### 9.1.1 Idle mode procedures in UTRAN

This subclause covers both the cases of idle mode mobility from E-UTRAN to UTRAN and of Idle Mode Signaling Reduction (ISR), as defined in TS 23.401 [2].

NOTE 1: TS 23.401 states conditions under which a valid P-TMSI or a P-TMSI that is mapped from a valid GUTI ('mapped GUTI') is inserted in the Information Element 'old P-TMSI' in the Routing Area Update Request. It depends on the old P-TMSI which security context can be taken into use after completion of the Routing Area Update procedure.

##### Use of the current UMTS securitycontext

If the UE sends the RAU Request with the "old P-TMSI" Information Element including a valid P-TMSI it shall also include the KSI relating to this P-TMSI. This KSI is associated with the current UMTS security context stored on the UE, and it indicates this fact to the SGSN. In this case the UE shall include P-TMSI signature into the RAU Request if a P-TMSI signature was assigned by the old SGSN. If the network does not have a valid security context for this KSI it shall run AKA. In case of an SGSN change keys from the old SGSN shall overwrite keys in the new SGSN if any.

NOTE 2: if the UE has a valid UMTS security context then this context is stored on the USIM according to TS 33.102 [4].

##### Mapping of EPS security context to UMTS security context

If the UE sends the RAU Request with the "old P-TMSI" Information Element including mapped GUTI it shall also include the KSI equal to the value of the eKSI associated with the current EPS security context (cf. clause 3). The UE shall include a truncated NAS-token, as defined in this clause further below, into the P-TMSI signature IE. The MME shall transfer UE's UTRAN and GERAN security capabilities and CK' || IK' with KSI equal to the value of the eKSI associated with the current EPS security context to SGSN with Context Response/SGSN Context Response message. The MME and UE shall derive CK' and IK' from the  $K_{ASME}$  and the NAS downlink COUNT value corresponding to the truncated NAS-token received by the MME from SGSN as specified in Annex A. Keys CK' and IK' and KSI sent from the MME shall replace the keys and KSI in the target SGSN if any. Keys CK' and IK' and the KSI shall replace the currently stored values on the USIM. START shall be reset to 0 on USIM.

NOTE 3: The new derived security context (including CK", IK" and START value) replacing the old stored values in the USIM is for allowing to reuse the derived security context without invoking the authentication procedure in the subsequent connection set-ups , and also for avoiding that one KSI indicates to two different key sets and consequently leads to security context desynchronization.

NOTE 4: An operator concerned about the security of keys received from another operator may want to enforce a policy in SGSN to run a UMTS AKA as soon as possible after the run of an idle mode mobility procedure. An example of ensuring this is the deletion of the mapped UMTS security context in the SGSN after the completion of the idle mode mobility procedure.

SGSN shall include the allowed security algorithm and transfer them to RNC. An SMC shall be sent to the UE containing the selected algorithms.

The x bits available in the P-TMSI signature field (at minimum 16 bits) shall be filled with the truncated NAS-token, which is defined as the x least significant bits of the NAS-token.

The NAS-token is derived as specified in Annex A.9.

SGSN shall forward the P-TMSI signature including the truncated NAS token to the old MME, which compares the received bits of the truncated NAS-token with the corresponding bits of a NAS-token generated in the MME, for the UE identified within the context request. If they match, the context request message is authenticated and authorized and MME shall provide the needed information for the SGSN. Old MME shall respond 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 compare the received truncated NAS-token with the  $x$  least significant bits of NAS-tokens generated from the current NAS downlink COUNT value down to current NAS COUNT-L downlink values, i.e. the interval [current NAS downlink COUNT - L, current NAS downlink COUNT]. A suitable value for the parameter L can be configured by the network operator. 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 Idle mode procedures in E-UTRAN

This subclause covers both the cases of idle mode mobility from UTRAN to E-UTRAN and of Idle Mode Signaling Reduction, as defined in TS 23.401 [2].

The TAU Request and ATTACH Request message shall include the UE security capabilities. The MME shall store these UE security capabilities for future use.

### Case 1: UE currently registered in E-UTRAN

When the UE is currently registered in E-UTRAN, the UE shall use the current EPS security context to protect the TAU Request and include the corresponding temporary identity and eKSI value. The TAU Request shall be integrity-protected, but not confidentiality-protected. UE shall use the current EPS security context algorithms to protect the TAU Request message.

NOTE 1: Activating a native EPS security context, which is not the current EPS security context, requires a separate procedure, which is defined in clause 7.2.9.2.

### Case 2: UE not currently registered in E-UTRAN

If the UE sends a TAU Request to the MME, and was previously connected to UTRAN where the SGSN assigned a P-TMSI signature to the UE, the UE shall include that P-TMSI signature in the TAU Request.

If the MME received a P-TMSI signature from the UE, the MME shall include that P-TMSI signature in the Context Request message sent to the SGSN. The 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 Annex A. In case the MM context in the Context Response/SGSN Context Response indicates GSM security mode, the MME shall abort the procedure.

MME shall select security algorithms, based on the UE EPS security capabilities received in the TAU request, and indicate them to the UE by e.g. with NAS SMC. In any case capability information received from the UE overwrites any capabilities received with the context transfer as specified in TS 23.401 [2]. The MME shall not make use of any UE security capabilities received from the SGSN.

If the UE is not currently registered in E-UTRAN no current EPS security context is available. Then the UE shall use the E-UTRAN native security context if available in the UE to protect the TAU Request. If not the UE shall send the TAU Request unprotected.

In case 2, the UE shall include in the TAU Request:

- the  $KSI_{SGSN}$  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 native security context is not available in the network. The value field of the  $KSI_{SGSN}$  shall be derived by the assigning the KSI corresponding to the set of keys most recently generated (either by a successful AKA run or mapping from an EPS security context).
- a 32bit  $NONCE_{UE}$ .

#### Case 2.1: Native EPS security context

The UE shall include the corresponding temporary identity and  $KSI_{ASME}$  value in the TAU Request. The TAU Request shall be integrity-protected, but not confidentiality-protected. The UE shall use the native security context algorithms to protect the TAU Request message.

NOTE 2: Case 2.1 relates to the following scenario: a UE established a native EPS security context during a previous visit to EPS, then moves to UTRAN/GERAN while de-registering from E-UTRAN and storing the EPS security context. When the UE moves back to E-UTRAN there is a native EPS security context, but no current EPS security context.

In case MME has the native security context it shall verify the TAU Request message. Otherwise it tries to fetch the security context from a previous MME as described in clause 6.1.4. If either of these is successful, the MME replies

with TAU Accept message protected with the native security context. After successful completion of the TAU procedure, the native security context shall become the current security context in the UE and the MME. 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 shall be used to derive the  $K_{eNB}$  as specified in Annex A. MME shall deliver 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 NAS SMC, which shall also include the UE security capabilities. This message shall not be ciphered. UE shall reply with integrity protected NAS SMC COMPLETE protected with based on the new selected algorithms.

If the USIM supports EMM parameters storage then the new native EPS NAS security context shall be stored on the USIM.

### Case 2.2: Mapped EPS security context

If neither a current nor a native EPS security 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 or can not fetch the native 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, where CK, IK were identified by the  $KSI_{SGSN}$  in the TAU Request. See Annex A.11 for more information on how to derive the fresh  $K'_{ASME}$ . In case the TAU Request had the active flag set or there is pending downlink UP data, the uplink NAS Count which is set to zero shall be used to derive the  $K_{eNB}$  in MME and UE as specified in Annex A. MME shall deliver the  $K_{eNB}$  to the target eNB on the S1 interface. The uplink and downlink NAS COUNT for mapped security context shall be set to start value (i.e., 0) when new mapped security context is created in UE and MME.

The selected algorithms and the  $KSI_{SGSN}$  shall be indicated to the UE in an integrity protected NAS SMC COMMAND message protected with NAS keys based on  $K'_{ASME}$ , which shall also include the UE security capabilities,  $NONCE_{UE}$ , and  $NONCE_{MME}$ . The UE shall generate a mapped  $K''_{ASME}$  from CK and IK in the same way as the MME. The UE shall compare the received  $NONCE_{UE}$  with the  $NONCE_{UE}$  sent, and if the  $NONCE_{UE}$  was modified in the TAU request, then the UE shall return NAS security mode reject message. If the received  $NONCE_{UE}$  is the same as the  $NONCE_{UE}$  sent and the integrity check succeeds, then UE shall reply with integrity protected and ciphered NAS SMC COMPLETE message based on the selected algorithms and NAS keys based on  $K'_{ASME}$  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  $K'_{ASME}$ . After successful completion of the TAU procedure, the mapped security context shall become the current security context in the UE and the MME.

## 9.2 Handover

### 9.2.1 From E-UTRAN to UTRAN

NAS and AS security shall always be activated before handover from E-UTRAN to UTRAN can take place. Consequently the source system in the handover shall always send a key set to the target system during handover. The security policy of the target PLMN determines the selected algorithms to be used within the UTRAN HO command. UE and 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 as specified in Annex A:

UE and MME shall assign the value of eKSI to KSI. MME shall transfer  $CK' || IK'$  with KSI to SGSN. The target SGSN shall replace the stored parameters CK, IK, KSI, if any, with  $CK'$ ,  $IK'$ , KSI received from the MME. The UE shall replace the stored parameters CK, IK, KSI, if any, with  $CK'$ ,  $IK'$ , KSI in both ME and USIM. START shall be reset to 0. For the definition of the Key Derivation Function see Annex A.

NOTE 1: The new derived security context (including  $CK'$ ,  $IK'$  and START value) replacing the stored values in the USIM is for allowing to reuse the derived security context without invoking the authentication procedure in subsequent connection set-ups, and also for avoiding that one KSI value indicates to two different key sets and consequently leads to security context desynchronization.

NOTE 2: An operator concerned about the security of keys received from an E-UTRAN of another operator may want to enforce a policy in SGSN to run a UMTS AKA as soon as possible after the handover. One example of ensuring this is the deletion of the mapped UMTS security context in the SGSN after the UE has left active state.

MME shall also provide at least the 4 LSB of the current NAS downlink COUNT value to the source eNB, which then shall include the bits to the MobilityFromE-UTRANCommand to the UE.

MME shall transfer the UE security capabilities to the SGSN. The selection of the algorithms in the target system proceeds as described in TS 33.102 [4] for UTRAN.

## 9.2.2 From UTRAN to E-UTRAN

### 9.2.2.1 Procedure

The procedure for handover from UTRAN to E-UTRAN, as far as relevant for security, proceeds in the following two consecutive steps:

- A) Handover signalling using the mapped security context (cf. also Figure 9.2.2.1-1);
- B) Subsequent NAS signalling to determine whether a native context is taken in use (not shown in Figure).

The activation of NAS and AS security in E-UTRAN, and selection of the key set from the source system for the handover shall be according to following principles:

- i) As described for inter-SGSN PS handover cases in TS 33.102 [4], the source SGSN shall select the key set from the latest AKA and transfer this key set to the MME in the Forward Relocation Request.

NOTE x: The MME is considered as a target SGSN in case of Gn/Gp interface.

- ii) Activation of AS security (for details cf. TS 36.331 [18]):

The E-UTRAN HO command received at the UE shall activate AS security.

The HO Complete received at the eNB shall activate AS security.

- iii) Activation of NAS security (for details cf. TS 24.301 [9]):

The E-UTRAN HO command received at the UE shall activate NAS security.

The HO Notify received at the MME shall activate NAS security. In case the MME does not have the UE security capabilities stored from a previous visit, then no NAS message shall be sent or accepted by the MME other than a TAU request before a successful check of the UE security capabilities in the TAU request was performed by the MME.

- iv) Both AS and NAS ciphering and integrity protection algorithms shall be selected according to the policy of the target PLMN.

The above four principles consequentially always activate ciphering (potentially NULL ciphering) in E-UTRAN even if it was not active in the source system.

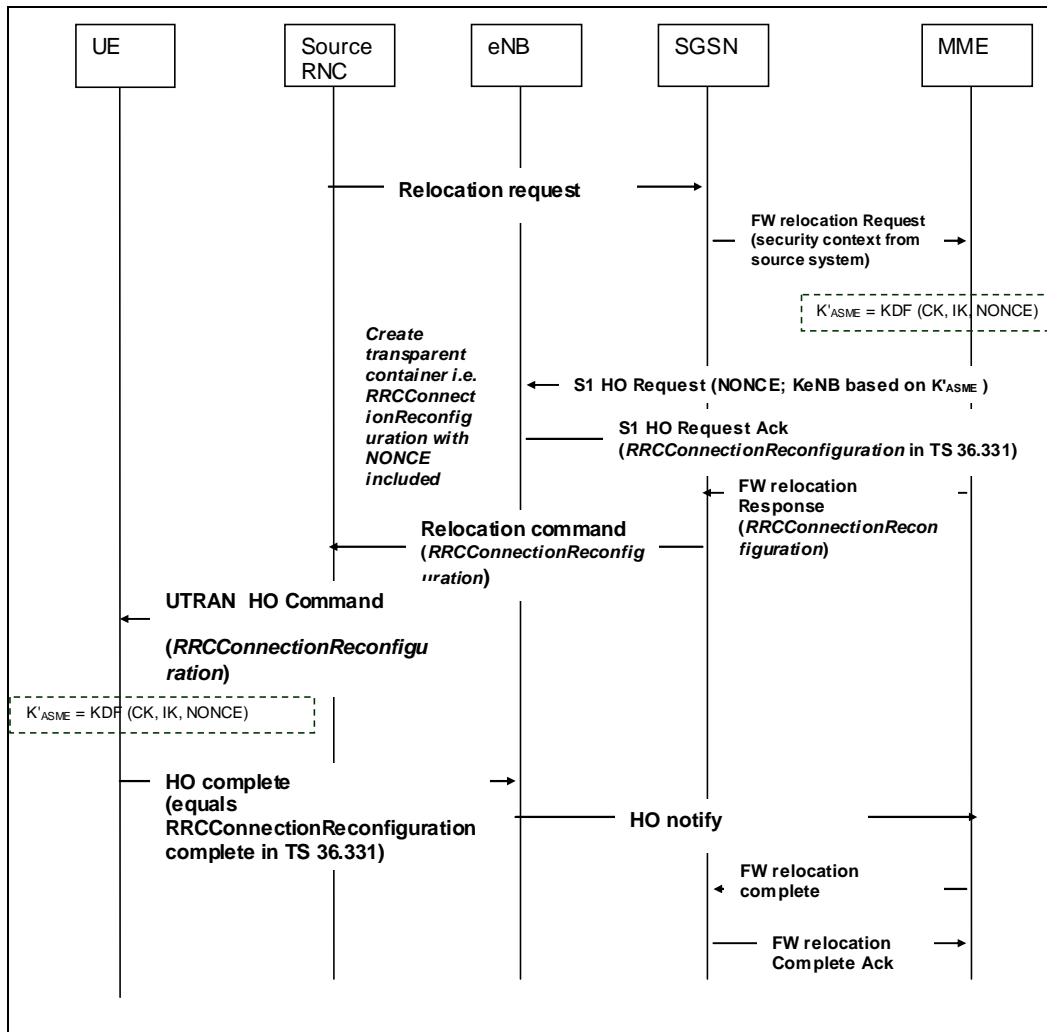


Figure 9.2.2.1-1: Handover from UTRAN to E-UTRAN

A) Handover signalling in case of successful handover

The RNC shall send a Relocation Request message to the SGSN. This message does not contain any security-relevant parameters.

1. The SGSN shall transfer CK and IK, KSI and the UE security capabilities to MME in the relocation request message. In case the MM context in the Context Response/SGSN Context Response indicates GSM security mode, the MME shall abort the procedure. The UE security capabilities, including the UE EPS security capabilities, were sent by the UE to the SGSN via the MS Network Capability IE, that is extended to include also UE EPS security capabilities, in Attach Request and RAU Request. It is possible that an SGSN does not forward the UE EPS security capabilities to the MME. When the MME does not receive UE EPS security capabilities from the SGSN, the MME shall assume that the following default set of EPS security algorithms is supported by the UE:
  - 128-EEA0, 128-EEA1 and 128-EEA2 for NAS signalling ciphering, RRC signalling ciphering and UP ciphering;
  - 128-EIA1 and 128-EIA2 for NAS signalling integrity protection and RRC signalling integrity protection.

NOTE 0: Clauses 5.1.3.2 and 5.1.4.2 of this specification mandate the UE to support the default set of EPS security algorithms, so, for the Rel-8 version of this specification, the default set of EPS security algorithms includes all security algorithms standardised for EPS. The notion of default set of EPS security algorithms is introduced here in order to make this specification future-proof as more security algorithms may be standardised for EPS in future releases.

2. The MME shall create a  $NONCE_{MME}$  to be used in the  $K'_{ASME}$  derivation, MME shall derive  $K'_{ASME}$  from CK and IK with the help of a one-way key derivation function as defined in Annex A and associate it with a Key

Set Identifier  $KSI_{SGSN}$ . The value field of the  $KSI_{SGSN}$  shall be derived by the assigning the KSI corresponding to the set of keys most recently generated (either by a successful AKA run or mapping from an EPS security context). MME shall derive the NAS keys and  $K_{eNB}$  from  $K'_{ASME}$ . The uplink and downlink NAS COUNT values for the mapped security context shall be reset to start value (i.e. 0) in the UE and the MME.

3. MME shall select the NAS security algorithms, MME shall include  $KSI_{SGSN}$ ,  $NONCE_{MME}$ , the selected NAS security algorithms in the NAS Security Transparent Container IE of S1 HO Request message to the target eNB. MME further shall include  $K_{eNB}$  and the UE EPS security capabilities, either the capabilities received from the SGSN or, in the absence of these, the default set of EPS security algorithms, in the S1 HO Request message to the target eNB.
4. The target eNB shall select the RRC and UP algorithms based on the UE EPS Security Capabilities. The target eNB shall create a transparent container (RRCConnectionReconfiguration) including the selected RRC, UP algorithms and the NAS Security Transparent Container IE, and send it in the S1 HO Request Ack message towards the MME.

NOTE 1: This transparent container is not protected by the target eNB.

5. MME shall include the transparent container received from the target eNB in the FW Relocation Response message sent to SGSN.
6. SGSN shall include the transparent container in the relocation command sent to the RNC.
7. The RNC shall include the transparent container in the UTRAN HO command sent to the UE.

NOTE 2: The UTRAN HO command is integrity protected and optionally ciphered as specified by TS 33.102 [4].

8. The UE shall derive  $K'_{ASME}$  in the same way the MME did in step 2, associate it with  $KSI_{SGSN}$  and derive NAS, RRC and UP keys accordingly. The UE shall send a RRCConnectionReconfiguration Complete messages to the eNB. The uplink and downlink NAS COUNT values for the mapped context shall be set to start value (i.e. 0) in the UE and the MME.
9. The mapped EPS security context shall become the current (cf. clause 3.1) EPS security context at AS and NAS level and overwrite any existing current mapped EPS security context. The HO Complete messages and all following AS messages in E-UTRAN shall be ciphered and integrity protected according to the policy of the target PLMN.

## B) Subsequent NAS signalling

In order to prevent that successful bidding down on the UE security capabilities in a previous RAT have an effect on the selection of EPS security algorithm for NAS and AS, the UE security capabilities shall be included in the TAU request after IRAT-HO and be verified by the MME.

NOTE 3: Any TAU request following the handover will be integrity protected. Details are described in clause 9.2.2.1

In any case UE security capability information received from the UE overwrites any capabilities received with the context transfer as specified in TS 23.401 [2].

In case there is a mismatch on the UE security capabilities stored in the MME and those received in the TAU request, the connection shall be aborted.

1. If the MME has native security context for the UE and does not receive a TAU request within a certain period after the HO it shall assume that UE and MME share a native security context.

NOTE 4: A TAU procedure following handover from UTRAN to E-UTRAN is mandatory if the Tracking Area has changed, but optional otherwise, cf. TS 23.401[2].

2. When the UE sends a TAU request it shall protect the request using the mapped EPS security context identified by  $KSI_{SGSN}$ . The UE shall also include  $KSI_{ASME}$  in the TAU request if and only if it has native EPS security context. The  $KSI_{ASME}$  shall be accompanied by a GUTI. When the MME receives a TAU request with a  $KSI_{ASME}$  and GUTI corresponding to the native EPS security context stored on that MME it knows that UE and MME share a native security context.

3. When the MME receives a TAU request with a  $KSI_{ASME}$  and GUTI pointing to a previously visited MME it may fetch the native EPS security context from this other MME. It fetches this context by sending the previous MME a message that indicates it has authenticated the UE and includes the IMSI and GUTI. The previous MME returns the security context if it can find it based on the received IMSI and GUTI. If the MME fetches the native EPS security context from the other MME it shall do so before sending the TAU accept message. If this fetching is successful the MME knows that UE and MME share a native EPS security context. If this fetching is unsuccessful the MME shall delete any native EPS security context for any GUTI it may have for the user who sent the TAU request.

NOTE 5: The indication of the authentication of the UE in the message between MMEs is not meant to imply a particular information element, but to indicate that unlike in other cases of fetching the context the previous MME is not required to check the integrity of the TAU Request. The indication could be based on message type or content of the message.

4. When the MME receives a TAU request without a  $KSI_{ASME}$  it shall delete any native EPS security context for any GUTI it may have for the user who sent the TAU request.
5. The MME shall include  $KSI_{ASME}$  and a GUTI in the TAU Accept message, that shall be ciphered using the mapped security context, if and only if it knows from step 2 or step 3 that it now shares a native EPS security context indexed by this  $KSI_{ASME}$  and this GUTI with the UE. In case the GUTI received in the TAU Request message pointed to a different MME, the allocation of a new GUTI, replacing the received GUTI, and the association of this new GUTI with  $KSI_{ASME}$  is required.
6. When the UE receives a TAU Accept message without  $KSI_{ASME}$  it shall delete the native EPS security context, if any.

NOTE 6: the TAU procedure is integrity protected with the mapped EPS security context even if the UE and the MME share a native EPS security context.

7. When the MME knows, after having completed the preceding steps, that it shares a native EPS security context with the UE, the MME may (depending on configured policy) activate this native EPS security context. This activation may occur in two ways:
  - a. During ECM-CONNECTED state: the MME shall initiate a key change on the fly procedure according to clause 7.2.9 for the entire EPS key hierarchy.
  - b. After the next transition to ECM-IDLE state following the handover from UTRAN: Upon receiving the first message from the UE after the UE has gone to ECM-IDLE state the MME shall use the procedures defined in clauses 7.2.4.4 and 7.2.4.5 to activate the native EPS security context.
8. The UE and the MME shall delete the mapped context and set the native context to the current context after a successful completion of the NAS SMC procedure according to 7a. or 7b.

NOTE 7: The run of an NAS SMC procedures ensures that the uplink NAS COUNT has increased since the last time a  $K_{eNB}$  was derived from the  $K_{ASME}$ .

NOTE 8: For the handling of native and mapped contexts after a state transition to EMM-DEREGISTERED cf. clause 7.2.5.1.

### 9.2.2.2 Derivation of NAS keys and $K_{eNB}$ during Handover from UTRAN to E-UTRAN

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

MME and UE shall derive  $K_{eNB}$  from  $K'_{ASME}$  as follows:

The MME sets NAS COUNT equal to zero and uses it with the mapped key  $K'_{ASME}$  to derive  $K_{eNB}$  by applying the KDF defined in Annex A for IDLE to CONNECTED transition.

If the USIM supports EMM parameters storage then the new native EPS NAS security context shall be stored on the USIM.

### 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 native 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 native 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

### 10.1 General

An SGSN supporting interworking between E-UTRAN and GERAN is capable of handling UMTS security contexts and supports the key conversion function c3 specified in TS 33.102 [4]. Furthermore, as a consequence of the UE being able to access EPS, the user has a USIM, and the ME and the HSS are UMTS-capable. Hence, UMTS AKA is used when the UE is authenticated even when attached to GERAN, and UMTS security contexts are available. The security procedures for interworking between E-UTRAN and GERAN are therefore quite similar to those between E-UTRAN and UTRAN.

### 10.2 Idle mode procedures

#### 10.2.1 Idle mode procedures in GERAN

This subclause covers both the cases of idle mode mobility from E-UTRAN to GERAN and of Idle Mode Signaling Reduction, as defined in TS 23.401 [2].

As the SGSN is capable of handling UMTS security contexts clause 9.1.1 applies here with the following changes

- the SGSN and UE shall derive Kc from CK' and IK' with the help of the key conversion function c3 of TS 33.102;
- SGSN shall select the encryption algorithm to use in GERAN.

#### 10.2.2 Idle mode procedures in E-UTRAN

This subclause covers both the cases of idle mode mobility from GERAN to E-UTRAN and of Idle Mode Signaling Reduction, as defined in TS 23.401 [2].

As the SGSN shares a UMTS security context with the UE clause 9.1.2 applies here without changes.

### 10.3 Handover

#### 10.3.1 From E-UTRAN to GERAN

As the SGSN is capable of handling UMTS security contexts clause 9.2.1 applies here with the following changes:

- SGSN and UE shall derive Kc from CK' and IK' with the help of the key conversion function c3 of TS 33.102.
- SGSN shall select the encryption algorithm to use in GERAN after handover.

#### 10.3.2 From GERAN to E-UTRAN

##### 10.3.2.1 Procedures

As the SGSN shares a UMTS security context with the UE clause 9.2.2 applies here without changes.

### 10.4 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

Single Radio Voice Call Continuity (SRVCC) is specified in 3GPP TS 23.216 [22].

The MME and the UE 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 as specified in Annex A.

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  $IK_{SRVCC}$ .

The MME shall 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.

UE and MME shall assign the value of eKSI to KSI. MME shall transfer  $CK_{SRVCC}$ ,  $IK_{SRVCC}$  with KSI and the UE security capability to the enhanced MSC server. The enhanced MSC server shall replace the stored parameters CK, IK, KSI, if any, with  $CK_{SRVCC}$ ,  $IK_{SRVCC}$ , KSI received from the MME. The UE shall replace the stored parameters CK, IK, KSI, if any, with  $CK_{SRVCC}$ ,  $IK_{SRVCC}$ , KSI in both ME and USIM. START shall be reset to 0.

NOTE 1: The new derived security context (including  $CK_{SRVCC}$ ,  $IK_{SRVCC}$ , KSI and START value) replacing the stored values in the USIM is for allowing to reuse the derived security context without invoking the authentication procedure in subsequent connection set-ups, and also for avoiding that one KSI value indicates to two different key sets and consequently leads to security context desynchronization.

NOTE 2: An operator concerned about the security of keys received from an E-UTRAN of another operator may want to enforce a policy in SGSN to run a UMTS AKA as soon as possible after the handover. One example of ensuring this is the deletion of the mapped UMTS security context in the SGSN after the UE has left active state.

If the SRVCC is from E-UTRAN to GERAN, the enhanced MSC server and the UE shall derive Kc from  $CK_{SRVCC}$  and  $IK_{SRVCC}$  with the help of the key conversion function c3 as specified in TS 33.102 [4]. The UE and the enhanced MSC Server shall assign the value of eKSI to CKSN.

NOTE: Non-voice bearers may be handed over during the SRVCC handover operation. Key derivation for non-voice bearers is specified in clause 9 of the present specification.

---

## Annex A (normative): Key derivation functions

### A.1 KDF interface and input parameter construction

#### A.1.1 General

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  $P_i$  as a number  $k$  in the range [0, 65535];
  - b)  $L_i$  is then a two-octet representation of the number  $k$ , with the most significant bit of the first octet of  $L_i$  equal to the most significant bit of  $k$ , and the least significant bit of the second octet of  $L_i$  equal to the least significant bit of  $k$ .

EXAMPLE: If  $P_i$  contains 258 octets then  $L_i$  will be the two-octet string 0x01 0x02.

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

$$S = FC \parallel P_0 \parallel L_0 \parallel P_1 \parallel L_1 \parallel P_2 \parallel L_2 \parallel P_3 \parallel L_3 \parallel \dots \parallel P_n \parallel L_n$$

where:

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

$P_0 \dots P_n$  are the  $n$  input parameters, and

$L_0 \dots L_n$  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:

$$\text{derived key} = \text{HMAC-SHA-256}(\text{Key}, S),$$

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

All key derivations for EPS shall be performed using the key derivation function (KDF) specified in this Annex. 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.

#### A.1.2 FC value allocations

The FC number space is used controlled by TS 33.220 [8], FC values allocated for this specification are in range of 0x10 – 0x1F.

## A.2 KASME derivation function ( $S_{10}$ )

When deriving a  $K_{ASME}$  from CK, IK and serving network ID when producing authentication vectors, and when the UE computes  $K_{ASME}$  during AKA, the following parameters shall be used to form the input S to the KDF.

- FC = 0x10,
- P0 = PLMN ID,
- L0 = length of PLMN ID (i.e. 0x00 0x03),
- P1 = SQN  $\oplus$  AK
- L1 = length of SQN  $\oplus$  AK (i.e. 0x00 0x06)

NOTE: The string S indexes start from 10 to align with the FC values and Annex subclause numbering.

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

The PLMN ID consists of MCC and MNC, and shall be encoded as an octet string according to Figure A.2-1.

|             |          |          |          |             |          |          |          |         |
|-------------|----------|----------|----------|-------------|----------|----------|----------|---------|
| <b>8</b>    | <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b>    | <b>3</b> | <b>2</b> | <b>1</b> |         |
| 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: 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 Key shall be equal to the concatenation CK || IK of CK and IK.

---

### 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, key change on-the-fly and TAU and handover from UTRAN/GERAN to EUTRAN ( $S_{11}$ )

When deriving a  $K_{eNB}$  from  $K_{ASME}$  and the uplink NAS COUNT in the UE and the MME the following parameters shall be used to form the input  $S$  to the KDF. During TAU and handover from UTRAN/GERAN to EUTRAN and when mapped context is used, the uplink NAS COUNT is set to 0 by UE and M

- FC = 0x11,
- 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 ( $S_{12}$ )

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

- FC = 0x12
- P0 = SYNC-input
- L0 = length of SYNC-input (i.e. 0x00 0x20)

The SYNC-input parameter shall be the newly derived  $K_{eNB}$  for the initial NH derivation, and the previous NH for all subsequent derivations. This results in a NH chain, where the next NH is always fresh and derived from the previous NH.

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

---

### A.5 $K_{eNB}^*$ derivation function ( $S_{13}$ )

When deriving a  $K_{eNB}^*$  from current  $K_{eNB}$  or from fresh NH and the target physical cell ID in the UE and eNB as specified in clause 7.2.8 for handover purposes the following parameters shall be used to form the input  $S$  to the KDF.

- FC = 0x13
- P0 = PCI (target physical cell id)
- L0 = length of PCI (i.e. 0x00 0x02)
- P1 = EARFCN-DL (target physical cell downlink frequency)
- L1 length of EARFCN-DL (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 Void

---

## A.7 Algorithm key derivation functions ( $S_{15}$ )

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  $K_{cNB}$ , in the UE, MME and eNB the following parameters shall be used to form the string S.

- FC = 0x15
- 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  $K_{cNB}$ .

For an algorithm key of length n bits, where n is less or equal to 256, the n least significant bits of the 256 bits of the KDF output shall be used as the algorithm key.

---

## A.8 $K_{ASME}$ to CK', IK' derivation ( $S_{16}$ )

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 = 0x16
- P0 = NAS downlink COUNT value
- L0 = length of NAS downlink COUNT value (i.e. 0x00 0x04)

The input key shall be  $K_{ASME}$ .

---

## A.9 NAS token derivation for inter-RAT mobility ( $S_{17}$ )

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 = 0x17
- 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.10 $K'_{ASME}$ from CK, IK derivation during handover ( $S_{18}$ )

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 to E-UTRAN.

$K'_{ASME}$  is a 256-bit value. The  $NONCE_{MME}$  is a 32-bit value. The following input parameters shall be used.

- FC = 0x18
- P0 =  $NONCE_{MME}$
- L0 = length of  $NONCE_{MME}$  (i.e. 0x00 0x04)

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

---

## A.11 $K'_{ASME}$ from CK, IK derivation during idle mode mobility ( $S_{19}$ )

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 = 0x19,
- 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.12 $K_{ASME}$ to $CK_{SRVCC}$ , $IK_{SRVCC}$ derivation ( $S_{1A}$ )

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 = 0x1A
- 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): Algorithms for ciphering and integrity protection

### B.0 EEA0 Null ciphering algorithm

The EEA0 algorithm shall be implemented such that it has the same effect as if it generates a KEYSTREAM of all zeroes (see subclause B.1.1). The length of the KEYSTREAM generated shall be equal to the LENGTH input parameter. The generated KEYSTREAM requires no other input parameters but the LENGTH. Apart from this, all processing performed in association with ciphering shall be exactly the same as with any of the ciphering algorithms specified in this annex.

NOTE: that EEA0 provides no security.

### B.1 128-bit ciphering algorithm

#### B.1.1 Inputs and outputs

The input parameters to the ciphering algorithm are a 128-bit cipher key named KEY, a 32-bit COUNT, a 5-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 0 for uplink and 1 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.

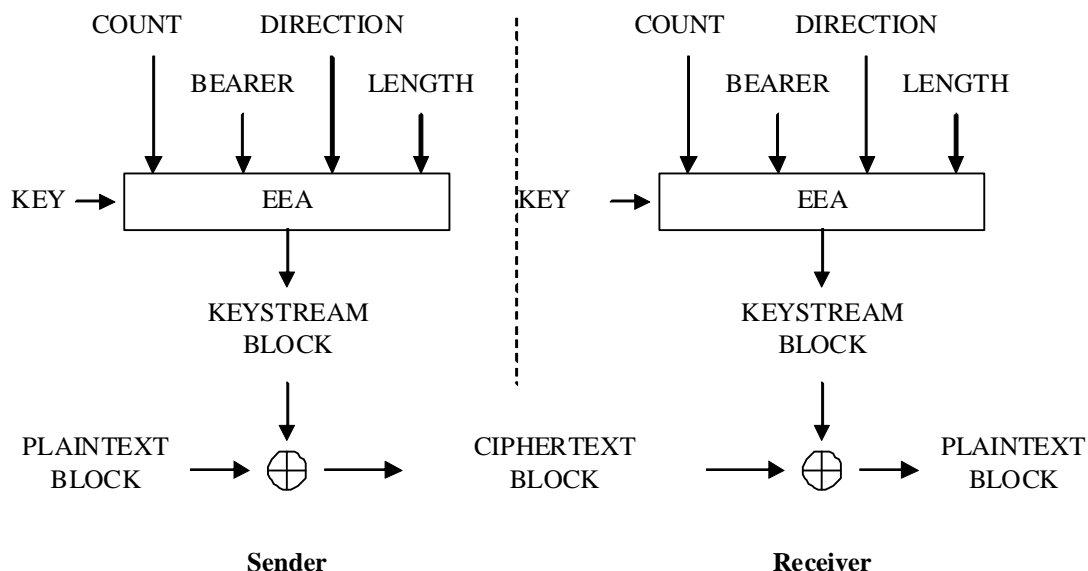


Figure B.1-1: Ciphering of data

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.1.2 128-EEA1

128-EEA1 is based on SNOW 3G and is identical to UEA2 as specified in [14]. The used IV is constructed the same way as in subclause 3.4 of that TS.

## B.1.3 128-EEA2

128-EEA2 is based on 128-bit AES [15] in CTR mode [16]

The sequence of 128-bit counter blocks needed for CTR mode  $T_1, T_2, \dots, T_i, \dots$  shall be constructed as follows:

The most significant 64 bits of  $T_1$  consist of  $\text{COUNT}[0] \dots \text{COUNT}[31] \mid \text{BEARER}[0] \dots \text{BEARER}[4] \mid \text{DIRECTION} \mid 0^{26}$  (i.e. 26 zero bits). These are written from most significant on the left to least significant on the right, so for example  $\text{COUNT}[0]$  is the most significant bit of  $T_1$ .

The least significant 64 bits of  $T_1$  are all 0.

Subsequent counter blocks are then obtained by applying the standard integer incrementing function (according to Appendix B1 in [16])  $\text{mod } 2^{64}$  to the least significant 64 bits of the previous counter block.

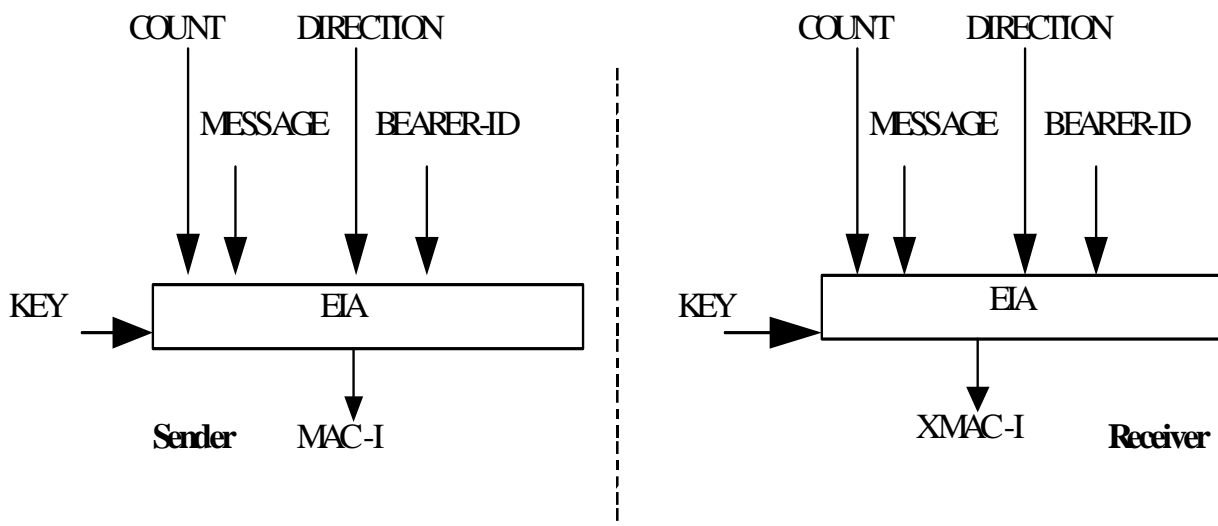
---

## B.2 128-Bit integrity algorithm

### B.2.1 Inputs and outputs

The input parameters to the integrity algorithm are a 128-bit integrity key named KEY, a 32-bit COUNT, a 5-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 0 for uplink and 1 for downlink. The bit length of the MESSAGE is LENGTH.

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/NAS-MAC (or XMAC-I/XNAS-MAC)**

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

## B.2.2 128-EIA1

128-EIA1 is based on SNOW 3G and is implemented in the same way as UIA2 as specified in [14]. The used IV is constructed the same way as in subclause 4.4 of that TS, with the only difference being that FRESH [0], ... FRESH [31] shall be replaced by BEARER[0] ... BEARER[4] | 0<sup>27</sup> (i.e. 27 zero bits)

## B.2.3 128-EIA2

128-EIA2 is based on 128-bit AES [15] in CMAC mode [17].

The bit length of MESSAGE is BLENGTH.

The input to CMAC mode is a bit string M of length Mlen (see [18, section 5.5]). M is constructed as follows:

$M_0 .. M_{31} = \text{COUNT}[0] .. \text{COUNT}[31]$

$M_{32} .. M_{36} = \text{BEARER}[0] .. \text{BEARER}[4]$

$M_{37} = \text{DIRECTION}$

$M_{38} .. M_{63} = 0^{26}$  (i.e. 26 zero bits)

$M_{64} .. M_{\text{BLENGTH}+63} = \text{MESSAGE}[0] .. \text{MESSAGE}[\text{BLENGTH}-1]$

and so  $Mlen = \text{BLENGTH} + 64$ .

AES in CMAC mode is used with these inputs to produce a Message Authentication Code T (MACT) of length Tlen = 32. T is used directly as the 128-EIA2 output MACT[0] .. MACT[31], with MACT[0] being the most significant bit of T.



|         |       |           |     |   |  |       |       |
|---------|-------|-----------|-----|---|--|-------|-------|
| 2009-03 | SA#43 | SP-090129 | 131 | - | Clarification on Token Calculation for RCR procedure   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 138 | - | Removal of Editor's note about PDCP COUNT threshold value  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 93  | 1 | Correction of EPS NAS Security Context definition  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 166 | - | NAS COUNT for mapped security context  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 117 | 1 | Storage of the EPS NAS security context in non-volatile memory of the UE   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 116 | 1 | Handling of security contexts in transition to EMM-DEREGISTERED state  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 97  | 1 | Security context for transition to EMM DEREGISTERED  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 114 | 1 | Definition of UE security capabilities   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 125 | 1 | Clarification on ciphering for GUTI confidentiality  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 136 | 1 | AKA when NAS COUNT about to wrap around  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 95  | 1 | Key derivation figures   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 102 | 1 | Rekeying of the entire EPS key hierarchy   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 121 | 1 | Various corrections related to algorithm definitions, NAS token and P-TMSI signature   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 145 | 1 | Adding EARFCN-DL to KeNB* derivation   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 128 | 1 | Forward security corrections related to R3-083569 (S3-09xxxx)  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 130 | 1 | KeNB re-keying corrections   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 135 | - | Removal of editor's note on key-change-on-the-fly for possible RLF   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 108 | 1 | Multi-KeNB* forwarding for RRCConnectionRe-establishment   | 8.2.2 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 143 | 1 | Definition of NULL ciphering   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 164 | 1 | Correction of definition of direction bit DIR  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 103 | 1 | NAS COUNT start value  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 154 | 1 | Correction of NAS COUNTs reset to the start value  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 96  | 1 | Algorithm selection clarification  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 129 | 1 | Verification of NONCEU   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 133 | 1 | Removal of 4LSB and eKSI in the AS SMC   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 134 | - | Removal of editor's note related 4LSB NAS downlink COUNT in the HO COMMAND   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 156 | 1 | Correcting the EPS security context fetching between MMEs  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 153 | 1 | Correction of description  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 153 | 1 | Correction of MAC-I  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 105 | 1 | Transparent container in UTRAN to E-UTRAN handover   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 106 | 1 | UE keys for E-UTRAN and GERAN  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 115 | 1 | Handling of UE capabilities in Idle Mode Mobility and Handover to E-UTRAN  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 155 | 1 | Adding definition of native contexts and clarifying the mapped and current definitions   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 104 | 1 | I-RAT handover E-UTRAN and UTRAN   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 158 | 1 | Some corrections to the UTRAN to E-UTRAN handover procedure  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 132 | 1 | Selection of key set and security activation on Handover from GERAN/UTRAN to E-UTRAN   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 147 | 1 | Correction of security requirements on eNodeB  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 119 | - | Replacing the cached security context with the mapped security context and setting START value in SRVCC  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 162 | - | Handling of integrity protection failures  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 120 | 1 | Clarification on security contexts and correction of protection of TAU requests in E-UTRAN , Handling of UE capabilities in Idle Mode Mobility | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 113 | 1 | Clarification for KSIsgsn  | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090129 | 99  | 2 | Making CK* and IK*   | 8.2.1 | 8.3.0 |
| 2009-03 | SA#43 | SP-090136 | 161 | - | Add reference to ciphering indicator feature specification   | 8.2.1 | 8.3.0 |
| 2009-03 |       |           |     |   | Editorial corrections  | 8.3.0 | 8.3.1 |

---

## History

| <b>Document history</b> |              |             |
|-------------------------|--------------|-------------|
| V8.1.1                  | January 2009 | Publication |
| V8.2.1                  | January 2009 | Publication |
| V8.3.1                  | April 2009   | Publication |
|                         |              |             |
|                         |              |             |