ETSI TR 103 637 V1.1.1 (2020-02)

Digital Enhanced Cordless Telecommunications (DECT);
DECT-2020 New Radio (NR) interface;
Study on Security Architecture

Reference
DTR/DECT-00340

Keywords
5G, DECT, IMT-2020, OFDM, security

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

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

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
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2020.
All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.
3GPP™ and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.
oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.
GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables 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 (https://ipr.etsi.org/).

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.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Digital Enhanced Cordless Telecommunications (DECT).

The present document presents a study of a new radio interface named DECT-2020. DECT-2020 is a state of the art radio interface based on OFDM with options for MIMO and is intended as long-term evolution of DECT technology.

The present document is focused on the study of the Security Architecture for the initial release of DECT-2020.

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

The current DECT radio interface was designed in the early 1990's and is based on TDMA/TDD with Gaussian Frequency Shift Keying (GFSK) modulation. Although this interface is able to provide a cost-effective solution for cordless telephony applications with an appropriate reuse of the spectrum, it cannot provide the high data rates and bandwidth efficiency required by most modern evolution scenarios. In addition, promising applications such as Audio-Streaming and Wireless Industrial Automation in Internet of Things (IoT) domain introduces Ultra Reliability and Low Latency requirements that have to be taken into account in any technology evolution.

IMT-2000 is the term used by the International Telecommunications Union (ITU) for a set of globally harmonised standards for third generation (3G) mobile telecoms services and equipment. 3G services are designed to offer broadband cellular access at speeds of 2 Mbps, which will allow mobile multimedia services to become possible.

DECT is, and will continue to be, one of the IMT-2000 technologies. However, the ITU work continued, first with IMT-Advanced, and it is now going further with IMT-2020. The term IMT-2020 was coined in 2012 by the ITU and means International Mobile Telecommunication system with a target date set for 2020, with the intention of addressing fifth generation (5G) mobile telecoms services and equipment.

The ETSI DECT Technical Committee and the industry body DECT Forum are currently supporting activities to develop DECT to meet the IMT-2020 requirements. This will require major changes to the existing DECT standards, and specifically to the MAC and PHL layers.

For the purpose of the present document the terms "DECT-2020", "DECT-2020 New Radio", "DECT-2020 NR" or "PHL-2020" have all the same meaning and all of them refer to the new radio interface based on OFDM outlined in the ETSI TR 103 514 [i.14] (PHY layer) and in the ETSI TR 103 635 [i.15] (MAC and higher layers). This new radio interface is targeted to meet the IMT-2020 requirements.

The terms FP-2020 or PP-2020 refer to FP and PP (respectively) devices supporting DECT-2020.

The present document is motivated by recent efforts to identify new ways of utilizing efficiently DECT frequency bands and potentially additional bands. New modes of operation are defined to target a more diverse set of use cases, while addressing 5G requirements for low latency, high spectral efficiency and large numbers of client nodes.

The present document is focused on the Security Architecture.

1 Scope

The present document aims on studying "DECT-2020: New Radio", a new radio interface based on state of the art paradigms able to offer the required data rates, propagation characteristics and spectrum efficiency, while maintaining compatibility with the carrier and time structure of the DECT band.

The scope of the present document is the definition of the initial overall Security Architecture to be used in the first release of DECT-2020 to be published in 2020. It covers all the necessary aspects: mutual authentication, confidentiality and integrity.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI EN 300 175-1: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 1: Overview".

[i.2] ETSI EN 300 175-2: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 2: Physical Layer (PHL)".

[i.3] ETSI EN 300 175-3: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 3: Medium Access Control (MAC) layer".

[i.4] ETSI EN 300 175-4: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 4: Data Link Control (DLC) layer".

[i.5] ETSI EN 300 175-5: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 5: Network (NWK) layer".

[i.6] ETSI EN 300 175-6: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 6: Identities and addressing".

[i.7] ETSI EN 300 175-7: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 7: Security features".

[i.8] ETSI EN 300 175-8: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 8: Speech and audio coding and transmission".

[i.9] ETSI TS 102 939-1: "Digital Enhanced Cordless Telecommunications (DECT); Ultra Low Energy (ULE); Machine to Machine Communications; Part 1: Home Automation Network (phase 1)".

[i.10] ETSI TS 102 939-2: "Digital Enhanced Cordless Telecommunications (DECT); Ultra Low Energy (ULE); Machine to Machine Communications; Part 2: Home Automation Network (phase 2)".

[i.11] ETSI TR 103 515: "Digital Enhanced Cordless Telecommunications (DECT); Study on URLLC use cases of vertical industries for DECT evolution and DECT-2020".

[i.12] IEEE 802.11™ family of standards.

[i.13] Bluetooth Core Specification, Version 5.2.

NOTE: Available at https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=478726.

[i.14] ETSI TR 103 514: "Digital Enhanced Cordless Telecommunications (DECT); DECT-2020 New Radio (NR) interface; Study on Physical (PHY) layer".

[i.15] ETSI TR 103 635: "Digital Enhanced Cordless Telecommunications (DECT); DECT-2020 New Radio (NR) interface; Study on MAC and Higher layers".

[i.16] FIPS Publication 197 (2001): "Advanced Encryption Standard (AES)", National Institute of Standards and Technology (NIST).

[i.17] NIST Special Publication 800-38A (2001): "Recommendation for Block Cipher Modes of Operation".

[i.18] NIST Special Publication 800-38B (2001): "Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication".

[i.19] IETF RFC 3610: "Counter with CBC-MAC (CCM)".

[i.20] ETSI TS 133 401 (V15.7.0): "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 3GPP System Architecture Evolution (SAE); Security architecture (3GPP TS 33.401 version 15.7.0 Release 15)".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI EN 300 175-1 [i.1] and the following apply:

beacon bearer packet types: packet formats intended for use in beacon bearers and C/L downlink bearers

NOTE: They include synchronization fields and do not need to support MIMO.

DFT bandwidth (MHz): maximum theoretical bandwidth that can be handled by the DFT in a given configuration

"HE" packet types: packet formats intended for continuous data transmission over several frames

NOTE: They may support circuit-mode traffic, URLLC traffic as well as packet mode traffic, and may implement MIMO.

"Legacy" DECT: current DECT technology as defined by ETSI EN 300 175 parts 1 [i.1] to 8 [i.8]

RAC packet types: packet types formats intended for use in Random Access Channels (RAC)

NOTE: They may be used for initially accessing a channel, carry only C-plane traffic, and do not need to support MIMO.

"Standard" packet types: packets intended for IP data packet-mode transmissions

NOTE: They are self-detectable packets usable in either synchronous or asynchronous way and may implement MIMO. The design of these packets is closer to the designs used in other WLAN technologies.

ULE packet types: packet formats intended for use in ULE (Ultra Low Energy) packet data transmissions

NOTE: They may be used for initially accessing a channel, are able to carry both U-plane and C-plane traffic, and do not need to support MIMO.

Ultra-Low Energy (ULE): ultra-low power consumption packet data technology based on DECT intended for M2M communications and defined by ETSI TS 102 939 parts 1 [i.9] and 2 [i.10]

3.2 Symbols

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

N_{BPSC}	Number of Bits Per SubCarrier
N_{CBPS}	Number of Coded Bits Per Symbol
N_{CTF}	Number of channel training symbols
N_{DBPS}	Number of data bits per symbol
N_{DC}	Number of null subcarriers at or surrounding DC
N_{DFT}	Discrete Fourier transform size
N_{SD}	Number of data subcarriers per OFDM symbol
$N_{SERVICE}$	Number of bits in the SERVICE subfield of the Data field
N_{SN}	Number of null subcarriers
N_{SP}	Number of pilot subcarriers per OFDM symbol
N_{SR}	Highest data subcarrier index per OFDM symbol
N_{SS}	Number of Spatial Streams
N_{ST}	Total number of used subcarriers per OFDM symbol,
N_{SYM}	Number of data SYMbols
N_{TAIL}	Number of TAIL bits for BCC encoder
T_{CTF}	Channel Training Field Time
T_{DFT}	DFT period
T_{FRAME}	Frame Time
T_{GT}	Guard field Time
T_{HF}	Header Field Time
T_{HFS}	Short Header Field Time
T_{SLOT}	Slot Time
T_{STF}	Synchronization Training Field Time
T_{STFS}	Short Synchronization Training Field Time
T_{SYM}	Symbol Time
W_{BC}	Basic Channel Bandwidth/Spacing

3.3 Abbreviations

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

AC	Authentication Code
AE	Authentication Entity
AES	Advanced Encryption Standard
AKA	Authentication and Key Agreement
AP	Access Point
ARQ	Automatic Retransmission Query
BCC	Binary Convolutional Codes
CCM	Counter with CBC MAC
DC	Direct Current
DCK	Derived Cipher Key
DECT	Digital Enhanced Cordless Telecommunications
DECT-2020	Physical Layer for DECT-2020
DFT	Discrete Fourier Transform
DLC	Data Link Control
DSAA2	DECT Standard Authentication Algorithm #2
DSC2	DECT Standard Cipher #2
ECDH	Elliptic Curve Diffie Hellman
EEA	EPS Encryption Algorithms
EIA	EPS Integrity Algorithms
EPS	Evolved Packet System
FP	Fixed Part

NOTE: Equivalent to the e-nodeB in 3GPP and to the AP in IEEE 802.11 [i.12].

FP-2020	PP implementing DECT-2020
FT	Fixed part radio Termination

GFSK	Gaussian Frequency Shift Keying
HE	High Efficiency
IP	Internet Protocol
ITU	International Telecommunication Union
ITU-R	International Telecommunication Union - Radiocommunication sector
KS	PT authentication Session Key
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
MIC	Message Integrity Code
MIMO	Multiple Input/Multiple Output
MITM	Man-In-The-Middle
mMTC	massive Machine Type Communications
NFC	Near Field Communication
NR	New Radio

NOTE: Refers to DECT-2020 New Radio radio interface as described in the present document.

NWK	NetWorK
OFDM	Orthogonal Frequency-Division Multiplexing
OOB	Out Of Band
PC	Personal Computer
PDF	Probability Density Function
PER	Packet Error Rate
PHL	PHysical Layer
PHL-2020	PHysical Layer for DECT-2020
PHY	PHYsical
PP	Portable Part

NOTE: Equivalent to the UE in 3GPP.

PP-2020	PP implementing DECT-2020
PT	Portable part radio Termination
R	code Rate
RAC	Random Access Channel
RFP	Radio Fixed Part
RIT	Radio Interface Technology
TDMA	Time Division Multiple Access
U	Uplink
UAK	User Authentication Key
UE	User Equipment

NOTE: Equivalent to the DECT PP.

ULE	Ultra-Low Energy
UPI	User Personal Identification
URLLC	Ultra-Reliable and Low Latency Communications
USIM	Universal Subscriber Identity Module
WLAN	Wireless LAN

4 Overview of security requirements

4.1 Background documents

A separate study on DECT evolution and DECT-2020 use cases and requirements has been conducted and published as ETSI TR 103 515 [i.11].

4.2 Requirements used as inputs to the present document

The following requirements have been used as principles for the design of the security architecture.

Basic features and algorithms:

- The basic security algorithms should be of comparable strength (or better) compared to the algorithms used by other 5G developments.
- The architecture should support at least two options of key sizes:
  - A practical value ensuring protection against all expected attacks mechanisms, except quantum computing. This value is assumed to be 128 bits.
  - An optional extended value ensuring protection against attacks mechanisms using quantum computing. This value is assumed to be 256 bits.
- The basic security algorithms specifications should be publicly available.
- The architecture should support mutual authentication.
- The architecture should provide confidentiality by means of encryption.
- The architecture should provide integrity protection to selected traffics by means of authenticated encryption.
- The architecture should provide selective application of encryption to some channels only depending on product application.
- An option of stream ciphering will be provided for some types of products.

Network topology:

- The architecture should provide a self-contained solution for standalone simple products consisting on independently deployed single cells.
- The architecture should support multi-cell deployments.
- The architecture should support repeaters.
- The architecture should support mesh network topologies, at least for mMTC scenarios.
- The architecture should support multi-cell radio networks with complex fixed part scenarios where FP security functions are not placed necessarily in the RFP.

Radio link services:

- The architecture should support and provide protection for both unicast and multicast traffics.
- The architecture should support and provide protection for both scheduled and random access traffics.
- The architecture should support MIMO.
- The architecture should support single-carrier and multicarrier radio operation.

Integration in 3GPP architectures:

- The architecture should support the integration of the DECT-2020 RIT as a node part of 3GPP 5G network architecture.
- Trusted an untrusted access should be supported.
- When integrated as part of a 3GPP 5G network, it should be possible to use the authentication and key agreement provided by the 3GPP network and deriving from it the necessary keys for use in the DECT-2020 NR component.

Temporary and collocation features:

- The architecture should support implementation of mixed or collocated devices (FPs and PPs) providing also legacy DECT service:
  - In such a case, when using legacy DECT service, it may be assumed that the most updated algorithms DSAA2 and DSC2 are used.
- For DECT-2020 release 1, it should be assumed that the DECT-2020 NWK layer concept may not be ready and that such release should be a PHY and MAC layer design (as in IEEE 802.11 [i.12]). The architecture should provide a temporary solution for this case. Such solution may be based or may require collocated implementation of legacy DECT in the devices.

5 Procedures for the establishment of security credentials

5.1 Overview and scope

This clause describes the initial procedures and possible strategies for the establishment of security credentials.

5.2 Discussion

The aim of DECT-2020 is providing state-of-the-art security level. Security credential (keys) can be established in several ways, for example security modules, SIM-cards, e-SIM, pre-shared at equipment production and by over-the-air pairing. Most existing DECT and ULE devices use over-the-air pairing.

An important proposal was to consider the "*Bluetooth secure simple pairing*" [i.13] methodology. DECT over-the-air pairing using an 4-8 digit AC and especially the widely used easy pairing is vulnerable to eavesdropping. If capturing the DECT pairing communication, the master security key (UAK) can easily or directly be calculated. This should be improved in DECT-2020.

5.3 Analysis

5.3.1 General

The establishment of security credentials in legacy DECT (as defined by EN 300 175-7 [i.7]) is based on user entry of the AC (a PIN code), which usually are just 4 digits and often default values are used. An intruder who has recorded the over-the-air communication during DECT key-allocation, would easily be able to guess most likely used AC codes or try all combinations of 4 digits to calculate and find the assigned security key (UAK). The AC code entry can be extended up to 8 digits, but this represents only up 10^8 ($\sim 2^{26}$) combinations and it is less convenient for the user.

In Bluetooth the primary goal of Secure Simple Pairing is to simplify the pairing procedure for the user. Secondary goals are to maintain or improve the security in Bluetooth wireless technology.

Secure Simple Pairing has two security goals: protection against passive eavesdropping and protection against man-in-the-middle attacks (active eavesdropping). It is a goal of Secure Simple Pairing to exceed the maximum security level provided by the use of a 16 alphanumeric PIN, which were used earlier.

5.3.2 Passive Eavesdropping Protection

A strong link key coupled with a strong encryption algorithm is necessary to give the user protection against passive eavesdropping. The strength of the link key is based on the amount of entropy (or randomness) in its generation, which is not known by an attacker. Using legacy pairing, the only source of entropy is the PIN which, in many use cases, is typically four digits either selected by the user or fixed for a given product. Therefore, if the pairing procedure and one authentication exchange is recorded one can run an exhaustive search to find the PIN in a very short amount of time on commonly available computing hardware. With Secure Simple Pairing, the recording attack becomes much harder as the attacker had to solve a hard problem in public key cryptography in order to derive the link key from the recorded information. This protection is independent of the length of the passkey or other numeric values that the user can handle. Secure Simple Pairing gives the same resistance against the recording and passive eavesdropping attacks even when the user is not required to do anything. Secure Simple Pairing uses Elliptic Curve Diffie Hellman (ECDH) public key cryptography as a means to thwart passive eavesdropping attacks. ECDH provides a very high degree of strength against passive eavesdropping attacks but it may be subject to man-in-the-middle attacks, which however, are much harder to perform in practice than the passive eavesdropping attack.

Using the security protocols with a 16 character alphanumeric, case sensitive PIN yields about 95 bits of entropy when the entire 62 character set is used ([0, … 9, 'A', … 'Z', 'a', …'z']). Secure Simple Pairing has approximately 95 bits of entropy using the FIPS approved P192 elliptic curve. ECDH cryptography was selected over standard Diffie Hellman (often referred to as DH76) since it is computationally less complex and less likely to exceed the low computational capacity in common Bluetooth Controllers.

5.3.3 Man-In-The-Middle Protection

A Man-In-The-Middle (MITM) attack occurs when a user wants to connect two devices but instead of connecting directly with each other they unknowingly connect to a third (attacking) device that plays the role of the device they are attempting to pair with. The third device then relays information between the two devices giving the illusion that they are directly connected. The attacking device may even eavesdrop on communication between the two devices (known as active eavesdropping) and is able to insert and modify information on the connection. In this type of attack, all of the information exchanged between the two devices are compromised and the attacker may inject commands and information into each of the devices thus potentially damaging the function of the devices. Devices falling victim to the attack are capable of communicating only when the attacker is present. If the attacker is not active or out range, the two victim devices will not be able to communicate directly with each other and the user will notice it. To prevent MITM attacks, Secure Simple Pairing offers two user assisted numeric methods: numerical comparison or passkey entry. The chance for a MITM to succeed inserting its own link keys in this case is a 1 in $10^{16} = 2^{53}$ pairing instances, which is an unnecessarily low probability. The strength of the MITM protections was selected to minimize the user impact by using a six-digit number for numerical comparison and Passkey entry. This level of MITM protection was selected since, in most cases, users can be alerted to the potential presence of a MITM attacker when the connection process fails as a result of a failed MITM attack. While most users feel that provided that they have not compromised their passkey, a 4-digit key is sufficient for authentication, the use of six digits allows Secure Simple Pairing to be FIPS compliant and this was deemed to have little perceivable usability impact.

5.3.4 Bluetooth Association Models

5.3.4.1 Introduction

Secure Simple Pairing uses four association models referred to as Numeric Comparison, Just Works, Out Of Band, and Passkey Entry. The association model used is deterministic based on the I/O capabilities of the two devices.

5.3.4.2 Numeric Comparison

The Numeric Comparison association model is designed for scenarios where both devices are capable of displaying a six-digit number and both are capable of having the user enter "yes" or "no". A good example of this model is the cell phone/PC scenario. The user is shown a six-digit number (from "000000" to "999999") on both displays and then asked whether the numbers are the same on both devices. If "yes" is entered on both devices, the pairing is successful. The numeric comparison serves two purposes. First, since many devices do not have unique names, it provides confirmation to the user that the correct devices are connected with each other. Second, the numeric comparison provides protection against MITM attacks. Note that there is a significant difference from a cryptographic point of view between Numeric Comparison and a PIN entry model. In the Numeric Comparison association model, the six digit number is an artefact of the security algorithm and not an input to it, as is the case in the Bluetooth security model. Knowing the displayed number is of no benefit in decrypting the encoded data exchanged between the two devices.

5.3.4.3 Just Works

The Just Works association model is primarily designed for scenarios where at least one of the devices does not have a display capable of displaying a six-digit number nor does it have a keyboard capable of entering six decimal digits. A good example of this model is the cell phone/mono headset scenario where most headsets do not have a display. The Just Works association model uses the Numeric Comparison protocol, but the user is never shown a number and the application may simply ask the user to accept the connection. The Just Works association model provides the same protection as the Numeric Comparison association model against passive eavesdropping but offers no protection against the MITM attack. When compared against today's experience of a headset with a fixed PIN, the security level of the Just Works association model is considerably higher since a high degree of protection against passive eavesdropping is realized.

5.3.4.4 Out of Band

The Out Of Band (OOB) association model is primarily designed for scenarios where an Out of Band mechanism is used to both discover the devices as well as to exchange or transfer cryptographic numbers used in the pairing process. In order to be effective from a security point of view, the Out of Band channel should provide different properties in terms of security compared to the Bluetooth radio channel. The Out of Band channel should be resistant to MITM attacks. If it is not, security may be compromised during authentication. The user's experience differs a bit depending on the Out of Band mechanism. As an example, with a Near Field Communication (NFC) solution, the user(s) will initially touch the two devices together and is given the option to pair the first device with the other device. If "yes" is entered, the pairing is successful. This is a single touch experience where the exchanged information is used in both devices. The information exchanged includes discovery information (such as the Bluetooth Device Address) as well as cryptographic information.

5.3.4.5 Passkey Entry

The Passkey Entry association model is primarily designed for the scenario where one device has input capability but does not have the capability to display six digits and the other device has output capabilities. A good example of this model is the PC and keyboard scenario. The user is shown a six-digit number (from "000000" to "999999") on the device with a display, and is then asked to enter the number on the other device. If the value entered on the second device is correct, the pairing is successful. Note that there is a significant difference from a cryptographic point of view between Passkey Entry and the PIN entry model. In the Passkey Entry association model, the six digit number is independent of the security algorithm and not an input to it. Knowing the entered number is of no benefit in decrypting the encoded data exchanged between the two devices.

5.4 Potential inclusion in the DECT security model

This clause gives an overview on how the methods presented in the previous clauses could be used to improve the DECT security level.

The security level of current DECT pairing and associated key allocation is limited and subject for passive eavesdropping. Using public/private key algorithms similar to those used in the new Bluetooth standard [i.13] could definitely be used to increase the level of security related to DECT key-allocation. The Elliptic Curve Diffie Hellman algorithm seems as state of the art solution for establishing a shared secret over an insecure channel. In DECT key allocation the shared secret of interest is the UAK. The authentication and encryption technologies used in the latest revisions of the DECT security standard (ETSI EN 300 175-7 [i.7]) seem adequate.

To understand possible usage of ECDH in the DECT key-allocation, a message scenario is suggested in the example below. The public keys could also be sent in other dedicated messages instead. Any other implication to overall DECT security model has not been analysed at this moment.

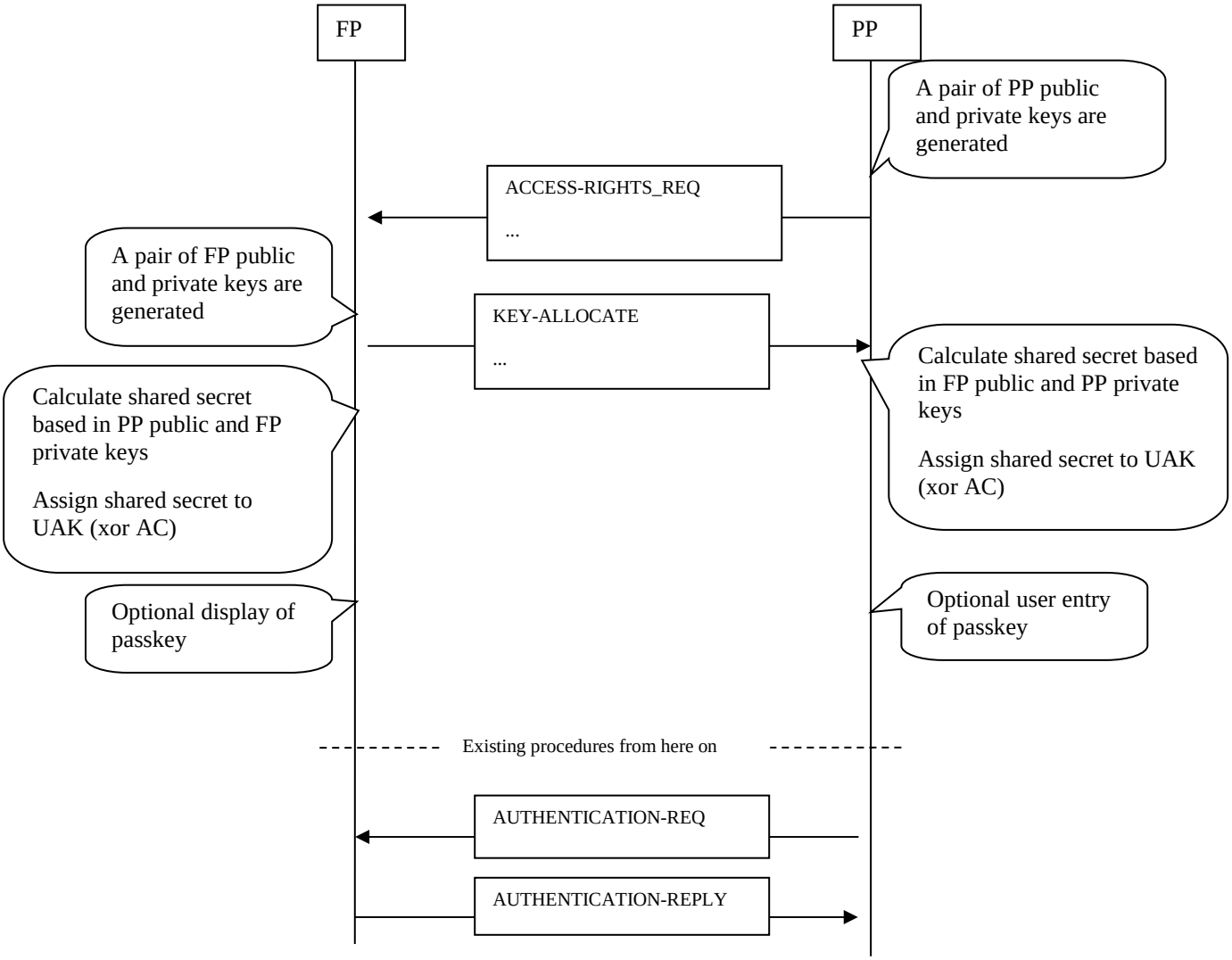


Figure 1: Illustrative example of ECDH usage in DECT key-allocation

The UAK can be derived from the shared secret, optionally xor'ed with the AC. All the following authentication and encryption can from then on be used unchanged.

The shown application of ECDH in key allocation scheme provides an effective guard against passive eavesdropping. To prevent man-in-the-middle attack, a display or keyboard on the FP is needed, assuming the PP has both display and keyboard. The passkey to verified or entered is a code generated as part of the shared secret. Out-of-band communication could also be used to prevent man-in-the-middle attach.

5.5 Final recommendation

DECT-2020 will include improved features for over-the-air pairing, and this will be taken into account when developing DECT-2020 security architecture.

6 Proposed security architecture

6.1 Overview of the solution

The fundamental DECT-2020 NR security architecture will be based in AES (Advanced Encryption Standard) with basic key size 128 bits, extendable to 256 and will provide the following features:

- Mutual authentication between FP and PP by means of Authentication Entity (AE) with signalling exchange for the Authentication and Key Agreement (AKA). This procedure is also able to generate the keys that will be used by the several encryption and integrity protection mechanisms.
- Encryption of U-plane and C-plane communications by means of AES-128 (or optionally 256) either using CCM [i.19] at DLC or convergence layers, CCM at MAC layer, other stream ciphering at MAC layer or a combination of them.
- Integrity check of C-plane and U-plane communications by means of the addition of a MIC (Message Integrity Code) using CCM.

For systems implementing only the DECT-2020 RIT, the authentication exchange will be done using AES-128 (or 256) between PP and an AE (located in the FP for single cell systems) or between PP and a central AE entity serving a network of FPs (multi-cell systems).

For systems that also implement the other component RIT (3GPP-NR), the option of using 3GPP algorithms for the AKA (potentially integrating USIM in the PP) is foreseen. The AKA will also generate the keys to be further used by DECT-NR encryption and integrity mechanisms.

Legal interception, when required, will be a feature external to the RIT.

6.2 Provided Protection

The protection against the several potential security attacks is provided by the security architecture. I.e.: protection against passive tapping is provided by the confidentiality algorithm, man in the middle attacks are prevented by the primary authentication exchange (that requires an input key not known by the attacker and generates derived keys not radio exposed) and the subsequent integrity protection. Attempts to run man in the middle attacks by emulating repeaters (supported by the technology) are prevented by the authentication of the repeater and confidentiality algorithms.

6.3 Basic security algorithms, key sizes and processes

6.3.1 Proposed basic algorithms

The fundamental security algorithm will be Advanced Encryption Standard AES [i.16].

6.3.2 Key size

The basic key size for all security processes to be used in DECT-2020 NR release 1 will be 128 bits. The architecture should allow the extension of this value to 256 bits in further releases.

6.3.3 Security processes

The structure of security processes given in ETSI EN 300 175-7 [i.7], figure 0.2, will be reused for A11, A12, A21 and A22 processes and for derivation of the ciphering keys for stream and CCM ciphering.

The processes B1 and B2 can be reused for Key generation. When used, all input keys (UAK, UPI and AC will be of 128 bits).

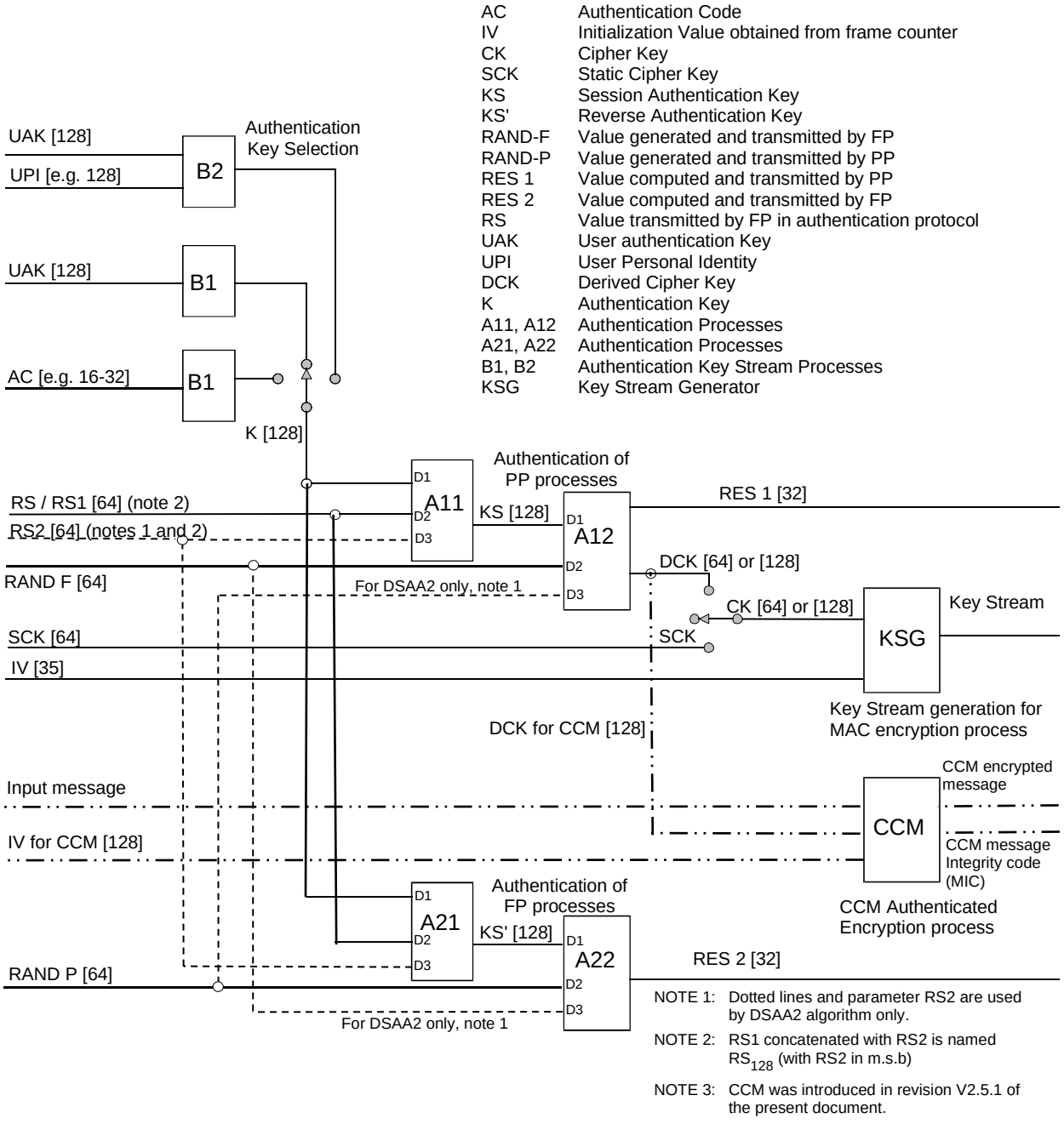


Figure 2: Overview of DECT security processes that will be reused in DECT-2020 release 1 (figure 0.2 from ETSI EN 300 175-7 [i.7])

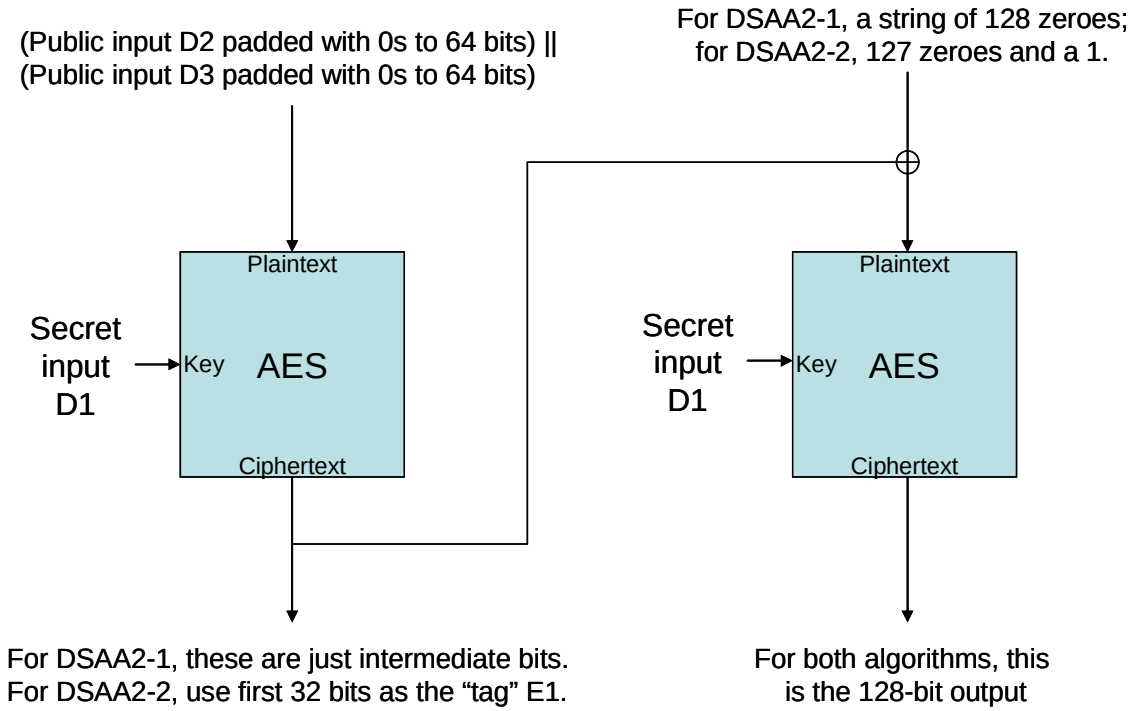
6.4 Mutual Authentication procedures

6.4.1 Algorithms

The processes A11, A12, A21, and A22 will be used.

The algorithm DSAA2, which is an implementation of AES with key size 128, will be used as authentication algorithm. The exact description of this algorithm for key size 128 bits is provided in ETSI EN 300 175-7 [i.7], annex L.

The basic structure of DSAA2 is shown in figure 3.



NOTE 1: The "tag" E1 is used to produce the RES1 or RES 2 parameters (for PT or FT authentications).
NOTE 2: For DECT-2020 it is proposed to increase the size of this "tag" to 64 bits (see clause 6.4.3).

Figure 3: The DSAA2-1 and DSAA2-2 algorithms

6.4.2 Signalling procedures

6.4.2.1 General

The procedure "Authentication of a PT type 2" as defined in ETSI EN 300 175-7 [i.7], clause 6.3.3 will be used.

The procedure "Authentication of a FT type 2" as defined in ETSI EN 300 175-7 [i.7], clause 6.3.4 will be used.

These procedures are based on processes A11, A12, A21 and A22 and use DECT Standard Authentication Algorithm #2 (DSAA2).

Tables 1 and 2 describe the operations to be executed by both authentications.

The tables 1 and 2 also describe two DECT NWK layer messages {AUTHENTICATION-REQUEST} and {AUTHENTICATION-REPLY} to be used in the procedures. Since NWK layer design of DECT-2020 is not yet ready, the tables should be understood as providing the semantic to be transported in future NWK layer messages (to be defined in the future) .

6.4.2.2 Authentication of an PT type 2 procedure

This procedure is used when the DECT Standard Authentication Algorithm #2 (DSAA2) is used.

Table 1: Authentication of a PT type 2 procedure

Sequence	Description
1	FT obtains a fresh random number RS_{128}
2	FT generates KS by means of the authentication process A11
3	FT generates a random value, RAND_F, used only once for this authentication process
4	FT constructs {AUTHENTICATION-REQUEST} including the parameters RAND_F and RS_{128},and sends it to PT
5	PT generates a random value, RAND_P, used only once for this authentication process
6	PT checks if it has internally stored a KS associated to the received value of RS_{128}. Otherwise, the PT executes authentication process A11 using K and the received RS_{128} as inputs
7	The PT executes the authentication process A12 using as inputs KS, RAND_F and RAND_P. It obtains the parameter RES1 (64 bits) and a new Derived Ciphering Key of 128 bits
8	PT constructs {AUTHENTICATION-REPLY} including RAND_P and the computed RES1
9	On receipt of RES1 and RAND_P, the FT uses the authentication process A12 to compute XRES1 (inputs: KS, RAND_F, RAND_P) and compares this value with the RES1 sent by the PT. If the two values are identical, the FT accepts the authenticity of the PT
10	As result of the execution of process A12, the FT obtains also a new value of the DCK of 128 bits

If the values compared in sequence 9 are not equal, then the authentication procedure has failed.

6.4.2.3 Authentication of an FT type 2 procedure

This procedure is used when the DECT Standard Authentication Algorithm #2 (DSAA2) is used.

Table 2: Authentication of an FT type 2 procedure

Sequence	Description
1	PT generates a "fresh" value RAND_P
2	PT constructs {AUTHENTICATION-REQUEST} including the parameter RAND_P and sends it to FT
3a OR	FT obtains (computes or retrieves) the last value of KS generated by the process A21 (that will be named KS') and the associated value of RS_{128} that was used for its generation (that will be named RS_{128}') OR
3b	FT generates a new value of RS_{128} (that will be named RS_{128}'), executes process A21 using RS_{128}' and K as inputs and generates KS (that will be named KS')
4	FT generates a "fresh" value of random parameter RAND_F
5	FT computes the value of RES2 (64 bits) running authentication process A22 and using as inputs: RAND_F, RAND_P and KS'
6	FT sends to the PT an {AUTHENTICATION-REPLY} message including the calculated value RES2 together with the parameters RAND_F and RS_{128}' used in its generation
7	PT independently calculates the value of XRES2 running process A22 and using KS', RS_{128}', RAND_P and RAND_F as inputs In order to get KS', the PT may check if the received RS_{128}' matches with the value of RS_{128}' it has stored from last execution of process A21. In such a case, the PT may use the last stored value of KS'. Otherwise, it should run process A21 with the received RS_{128}' to produce a new KS'
8	If XRES2 is equal to the value RES2 received from the FT, the PT accepts the authenticity of the FT

6.4.3 Proposed improvements to the authentication procedures

6.4.3.1 General

These clauses propose improvements to the authentication procedures to be used in DECT-2020 compared to the equivalent procedures used in DECT [i.1] to [i.8].

6.4.3.2 Immediate improvements

To be introduced in the first release of DECT-2020.

The following immediate improvement has been identified:

- It is proposed to increase the size of the authentication responses (RES1/XRES1 and RES2/XRES2) to 64 bits.

Implementation: the generation of 64 bits RES1/RES2 is already supported by the DSAA2 architecture (see clause 6.4.1). Current 32 bit RES1/RES2 are produced by truncation of a much longer cryptographic stream. Therefore there is no fundamental issue to extend the size from 32 to 64 bits.

6.4.3.3 Further improvements

To be introduced in release two and subsequent releases of DECT-2020.

The following additional improvements have been identified:

- It has been proposed to introduce the use of the 3GPP AKA procedure as additional optional procedure.

6.5 Confidentiality and integrity

6.5.1 Overview

Confidentiality and integrity will be provided by the encryption or authenticated encryption feature. Three approaches are provided:

1) Authenticated encryption based on CCM [i.19] with MIC insertion operating at DLC or convergence layer.

2) Stream ciphering at lower MAC layer based on CCM or other schema, preferably with MIC insertion.

3) Dual encryption approach (combination of 1 and 2).

The authenticated encryption based on CCM is the primary approach. It fundamentally solves all cases of radio network and fixed network topology, including complex distributed multi-cell topologies, operation with repeaters and mesh topologies. It is therefore the recommended approach. It also provides solid integrity protection due to the MIC insertion.

The stream ciphering at lower MAC layer is provided as alternative solution for simple systems. It may have some implementation advantages when implemented by hardware and is also very similar to legacy DECT DCS2 ciphering.

6.5.2 Analysis

Before entering in details of the two different ciphering alternatives, a brief analysis of the data flow through the protocol stack should be done. Figure 4 (extracted from ETSI TR 103 635 [i.15]) shows the expected construction of the MAC and PHY layer payload from the different flows from higher layers.

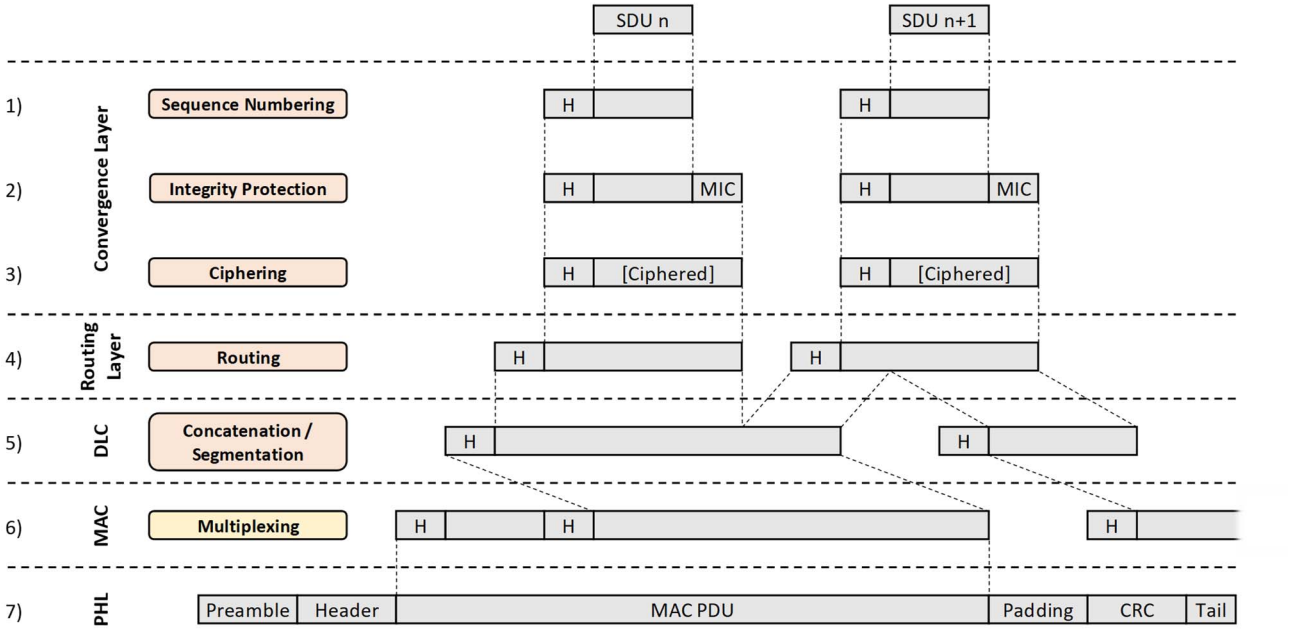


Figure 4: Data-flow through the protocol stack
(figure 35 from ETSI TR 103 635 [i.15])

6.5.3 CCM end-to-end approach

6.5.3.1 General

In the CCM approach, each logical channel is individually encrypted using CCM with MIC end-to-end. End-to-end means that the CCM ciphering/deciphering is applied at the DLC or convergence layer by the entity origin or destination of the packet. For instance, in a mesh radio network topology packets assembled by the convergence layer in the originating node will be CCM encrypted and will cross encrypted all intermediate nodes.

In a similar way, signalling packets will be encrypted by the protocol entity that produces them and will cross encrypted all lower layers. E.g. a NWK CC message will be encrypted by the NWK layer CC entity. In some cases the entity that processes the ciphering may be located in different places depending on the type of signalling (i.e. NWK vs. MAC).

It is assumed that some lower MAC logical channels will need to be unencrypted. Other MAC channels and most NWK signalling will be CCM encrypted.

6.5.3.2 Algorithm

The algorithm to be used with CCM encryption will be AES-128 with option to upgrade to AES-256 in further releases. This is the same algorithm proposed for the primary authentication.

6.5.3.3 CCM encryption process

The CCM encryption process will be similar to the one used in DECT ULE [i.9] and [i.10]. The CCM process will be as described in ETSI EN 300 175-7 [i.7], annex N. The initialization vector will be different for each logical channel. Exact definition requires completion of the MAC architecture phase.

IV should not be reused with the same Key in any case.

Keys may be the same or different. For release 1, the same key for all unicast traffic between two given nodes may be assumed, but more elaborated schemas may be designed in further releases.

The CCM encryption process is summarized in figures 5 and 6.

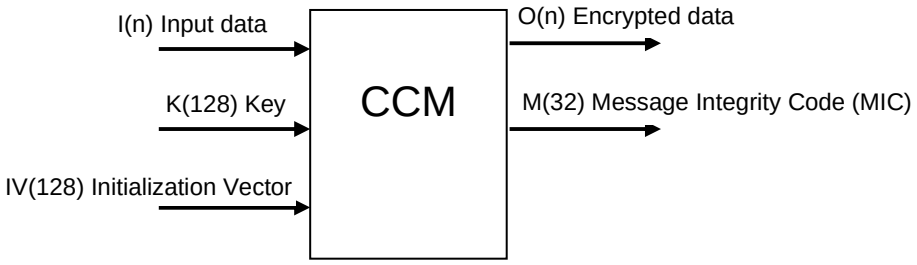
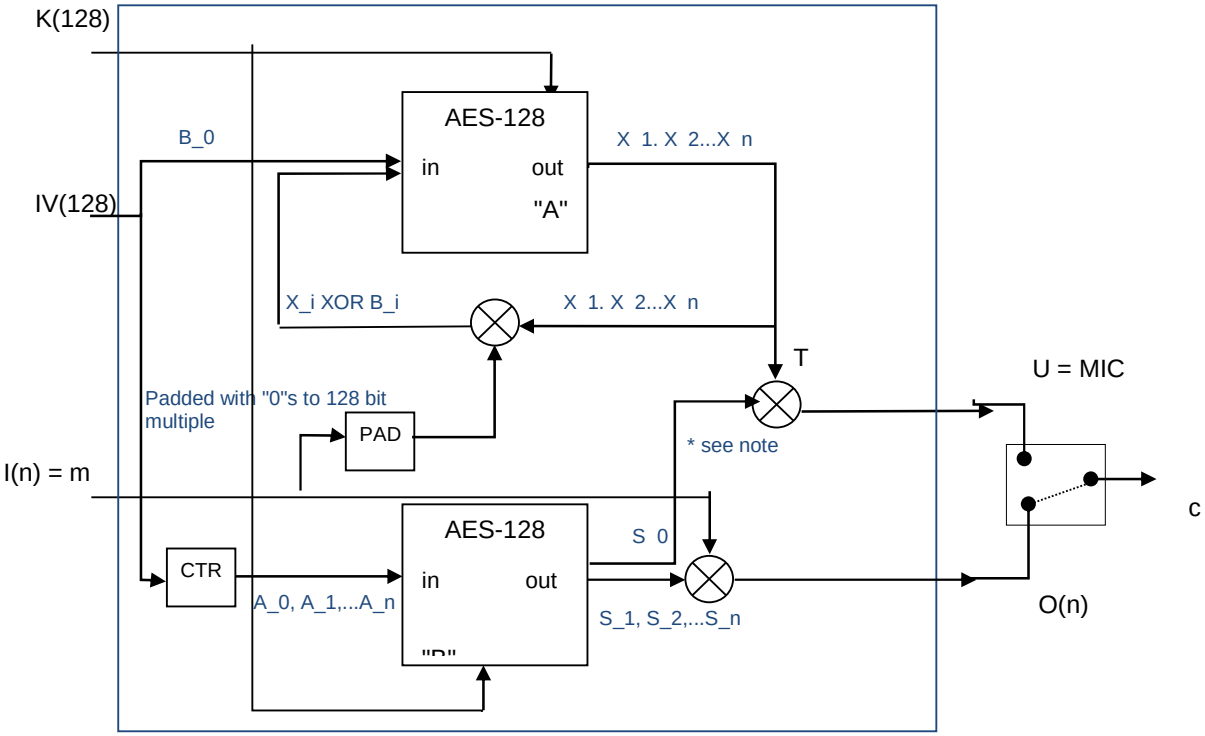


Figure 5: CCM security process overview



NOTE: The segment S_0, if generated by the first execution of "B", should be stored in order to be multiplied by "T" that is generated later. Another option is generating S_0 after the other S_x segments, what is possible due to the nature of the CTR function.

Figure 6: CCM security processes

6.5.3.4 Pros/cons

As indicated, CCM may be applied to all types of unicast and multicast traffics and may solve all cases of complex network topologies, including operation with repeaters and mesh networking scenarios. The insertion of MIC provides an integrity protection at a bandwidth cost not larger than the equivalent insertion of a CRC. It is therefore the ideal approach.

As drawbacks, it can be mentioned the complexity derived of the use of multiple ciphering processes and the potential different end-points for the different logical channels. This may make the implementation in hardware difficult.

An additional drawback is that the schema of ciphering end-to-end has may open the door to certain DoS attacks in mesh network topologies. This is consequence of the fact that CCM is not processed in intermediate nodes. Therefore span traffic inserted by an attacker may only be detected after being relayed by intermediate nodes.

6.5.4 Stream ciphering at lower MAC layer

6.5.4.1 General

The stream ciphering operated at lower MAC layer may be a simpler alternative from implementation point of view in simpler networks. It does not provide the flexibility of the end-to-end CCM when operating in distributed network (i.e. mesh topologies).

The approach is similar to MAC ciphering in legacy DECT. A ciphering stream is generated and applied at lower MAC layer. In order to ensure the operation of the system, some headers in the protocol structure and some logical channels should be excluded from the ciphering process. This is formalized in the "ciphering mask" described in the next clause.

6.5.4.2 Operation

Basically a ciphering stream is generated for each packet at lower MAC layer based on a ciphering key and an initialization vector that should not be repeated. Considering the different packet formats and the variations in MCS and MIMO settings (see ETSI TR 103 635 [i.15]) the length of the ciphering stream may be considerable.

The ciphering process will be:

input data $\oplus$ (ciphering mask $\times$ ciphering stream) = output data

6.5.4.3 Generation of the ciphering stream

Since the intention of the alternative is the construction of simple implementation reusing elements from legacy DECT design, the DECT Standard Cipher #2 (DSC2) is proposed as ciphering algorithm and stream generator with the only difference of the insertion of the ciphering mask to be logically multiplied (AND) by the stream.

The DSC2 is defined in ETSI EN 300 175-7 [i.7], annex M.

6.5.4.4 Insertion of a Message Integrity Code (MIC)

To get the optimal results regarding DoS attack protection and early detection of malicious traffic, the insertion of a Message Integrity Code (MIC) is recommended. The generation, exact insertion position and size of the MIC are for further study.

6.5.4.5 Ciphering mask

The correct operation of the system requires protecting some protocol elements from the effect of the cipher. Also the activation and deactivation of the encryption should be done by a MAC procedure similar to legacy DECT.

The ciphering mask is defined as a stream with "0" for the elements not to be ciphered and "1" in all other traffics.

The following elements seem to require being masked from the cipher ("0" in the mask).

NOTE 1: See figure 4 in clause 6.5.2 for reference:

- The PHY layer header.
- The MAC control and CRC fields (see ETSI TR 103 635 [i.15], figure 38).
- The MAC sequence number (if inserted in the B field).
- The MAC MUX header, including the length indicator (see ETSI TR 103 635 [i.15], figure 23).

NOTE 2: This header is of variable length.

NOTE 3: Since the MAC channels are of variable length, the mask will be different for each packet.

The following elements may require further discussion:

- Identities.
- Power control channels.
- Broadcast information.
- Some elements of the DLC structure: i.e. the DLC sequence number The DLC structure.
- Routing information may also require protection. However a different solution to the one proposed in ETSI TR 103 635 [i.15], clause 8.2.3.2 may be needed to make the solution implementable. For instance moving the routing information to "tags" inserted after the MAC header and protected from the cipher.

6.5.5 Dual encryption approach (CCM plus MAC ciphering)

6.5.5.1 General

In the Dual encryption approach both ciphering mechanisms are used: CCM encryption at DLC or convergence layer plus MAC stream ciphering.

Dual ciphering approach provides all benefits of CCM and MAC stream solutions. Compared to CCM, it provides better protection against DoS attacks.

Due to the overpower of the schema, it is possible to relax requirements for the MAC ciphering. For instance using shared keys.

A specific proposal is provided in clause 6.5.4.2.

6.5.5.2 Specific proposal including mesh topologies

6.5.5.2.1 Discussion

ETSI TR 103 635 [i.15], clause 8.2, discusses some security requirements for a mesh network operation. Both ciphering and integrity protection functions are considered. The considered protocol architecture will be as shown in figure 7. The expected headers of the different layers will be as shown in figure 4 of clause 6.5.2. In the presented protocol architecture the MAC and link layer (or DLC) layer is terminated in each hop allowing link specific ARQ, segmentation, concatenation functions, and above those layers there is a routing layer that is able to decide how data is routed forward, whereas convergence layer is end-to-end between applications.

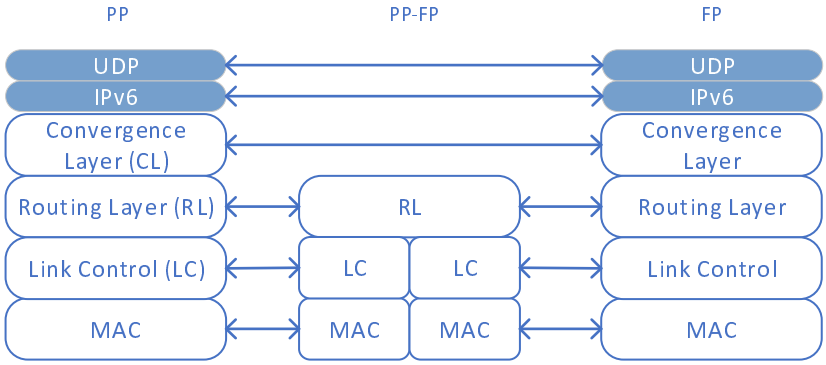


Figure 7: Protocol layer for relay, multi-hop and Mesh communication

The summary of the findings in ETSI TR 103 635 [i.15], clause 8.2, were:

- Security done in convergence layer:
  - Application layer PDU and security procedures can be performed independent from actual transmission procedures, i.e. "offline".

- It introduces end to end security between node and FP in a mesh network and any node between these nodes cannot manipulate or read the actual data, adding extra security to the system.
- The drawback of this schema is that all link control and MAC control headers are send in plain text and without integrity protection.

- Security done in MAC layer (including a MIC insertion):
  - The number fields sent in plain text can be minimized and only fields that are needed in receiver for deciphering and integrity protection are sent in plain text.
  - Any node receiving traffic can check whether sender is valid and can relay only valid data forward.
  - It makes DoS type of attack more difficult in mesh network topologies since packets that fail integrity will not be distributed any further in the system.
  - The final destination and original source address may be ciphered and not visible in the radio interface as plain text.

These options are presented in figure 8.

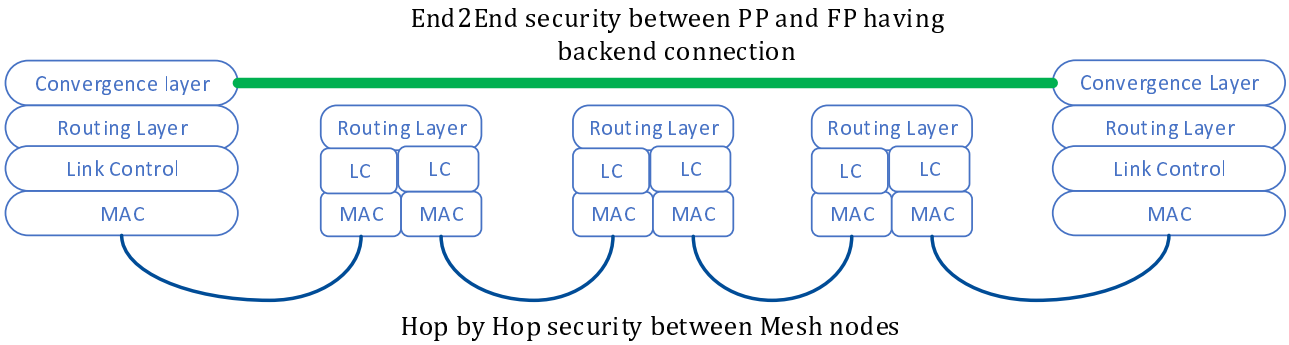


Figure 8: End to end vs. Hop by Hop security in Mesh system operation

Thus there seems to be also good reasons to perform security functions also in MAC layer when operating in mesh architecture. Similarly to convergence layer security it is believed that MAC level security for mesh network operation would operate in CCM mode providing encryption and authentication with MIC. MAC security would be:

- AES-128, CCM mode, so that actual ciphering & integrity process could be identical to convergence layer security.
- Both ciphering and integrity protection.
- Initialization vector would be sender and receiver specific.
- Security keys could be network wide allowing changing next hop device without key negotiation.
- As security is terminated between two nodes the hop-by-hop security is not limiting the number of supported hops.

6.5.5.2.2 Proposal

The following dual encryption schema is proposed:

- DLC or convergence layer security:
  - AES-128, CCM mode.
  - Both ciphering and integrity protection.
  - Initialization vector would be sender and receiver specific.
  - Security keys would be sender and receiver specific (in unicast traffic).

  - Security is terminated between end-nodes at DLC or convergence layer.

- MAC security:
  - AES-128, CCM mode, so that actual ciphering & integrity process could be identical to convergence layer security.
  - Both ciphering and integrity protection.
  - Initialization vector would be sender and receiver specific.
  - Security keys could be network wide allowing changing next hop device without key negotiation.
  - As security is terminated between two nodes the hop-by-hop security is not limiting the number of supported hops.

6.5.5.3 For further study

The problem of shared keys should be further studied, particularly when they are used in large groups where the attacker may be part of the group.

7 Items for further study

The following security ideas are identified as candidate for further study and possible introduction if further releases.

Security credentials distribution:

- The problem of distribution of security credentials in general:
  - There will be different types of devices - IoT devices without display and keyboard.
- Use of Security modules, including e-SIM.
- Other mechanisms for security key distribution.

Integration with 3GPP:

- Integration with 3GPP UEs - use of 3GPP security keys - how to derive keys:
  - Use of 3GPP AKA.
  - Use of 3GPP security keys.
  - Derivation of keys to be used in DECT-2020 from 3GPP processes and keys.

Fundamental security considerations:

- Refreshment rates for keys:
  - How often a PP should be re-authenticated due to security considerations (assuming AES-128).
- Key is further exposed due to the use of the same key with multiple IVs. How this may impact security.
- How to build the several Initialization Vectors.

Key derivation:

- Algorithms for Key Derivation, in general.
- Key Derivation for PP to PP communication options:
  - E.g. Direct PP to PP CCM ciphering (if used).

Options for encryption and integrity:

- Final confirmation of the need of a MIC at MAC layer:
  - Assumed as highly convenient by this study, but final cost/benefit analysis needed.
- Generation of a MIC for MAC layer encryption:
  - Possible solution: using CCM.
- Insertion of the MIC at MAC layer (if finally used):
  - A proposed idea is replacing the MAC CRC by a MIC, however it has radio considerations and collides with other uses of the CRC.
  - MAC CRC can be better masked by an identity (as in LTE).

Annex A:
Comparison with other technologies

A.1 3GPP

A.1.1 3GPP 5G Cryptographic principles and algorithms

A short overview of current 3GPP 4G and 5G cryptographic architecture is summarized below.

LTE introduced a new set of cryptographic algorithms and a significantly different key structure than that of GSM and UMTS. There are 3 sets of 4G cryptographic algorithms for both confidentiality and integrity termed EPS Encryption Algorithms (EEA) and EPS Integrity Algorithms (EIA):

- EEA1 and EIA1 are based **on SNOW 3G**, very similar to algorithms used in UMTS.
- EEA2 and EIA2 are based on the **Advanced Encryption Standard (AES)** with EEA2 defined by AES in CTR mode (e.g. stream cipher) and EIA2 defined by AES-CMAC (Cipher-based MAC). Default key size will be 128.
- EEA3 and EIA3 are both based on a **Chinese cipher ZUC.**

NOTE: 128-EEA1 and 128-EIA1 are also called UEA2 and UIA2.

Snow 3G and AES seem to be mandatory in all 5G devices, ZUC will be an option.

A.1.2 Authentication procedures (Authentication and Key Agreement -AKA)

Authentication procedures in 4G and 5G are not very different to previous 3GPP release and not very different to what DECT uses (as implemented in latest releases). The fundamental Authentication and Key Agreement (AKA) signalling exchange is shown below:

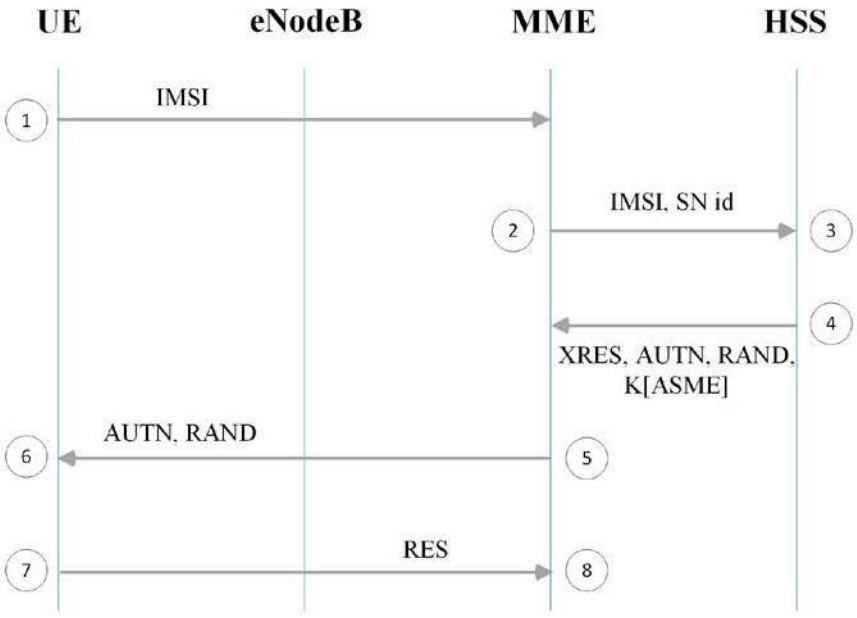


Figure A.1: 3GPP Authentication and Key Agreement Protocol

Security algorithms for the AKA may be different (typically weaker) than the new 5G algorithms and are operator dependent. Cryptographically, AKA algorithms are run in the HSS/AuC and in the USIM, both under control of the network operator.

The complete initial attach procedure is shown below.

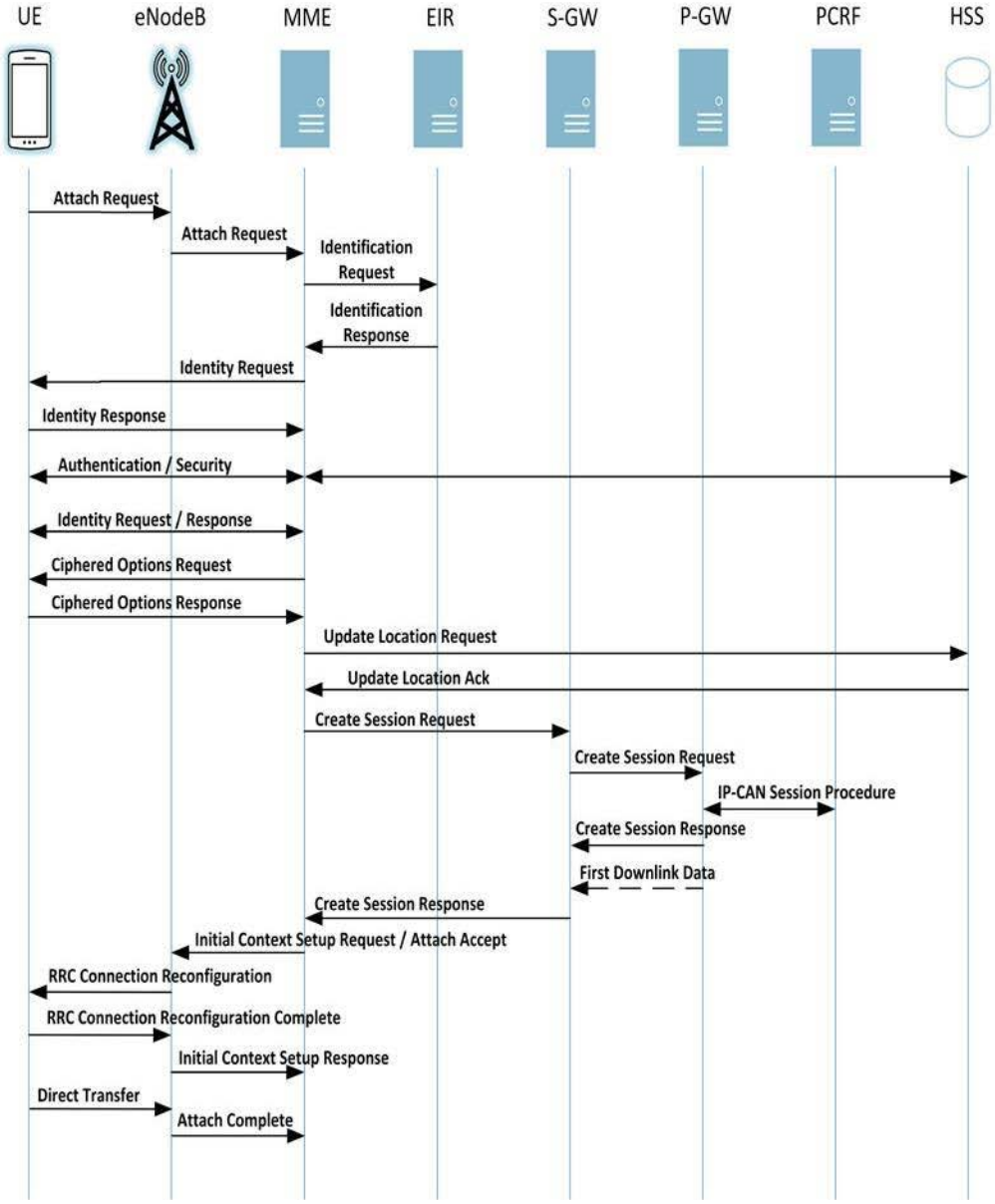


Figure A.2: 3GPP Complete initial attach procedure

A.1.3 5G Cryptography: operation of the ciphering

A.1.3.1 General

According to ETSI TR 133 401 [i.20], the operation of the encryption and the construction of the initialization vectors is summarized in figure A.3 and in the text below. Text is shown for the AES128 (named EEA2) case.

A.1.3.2 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 should be 0 for uplink and 1 for downlink.

Figure A.3 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.

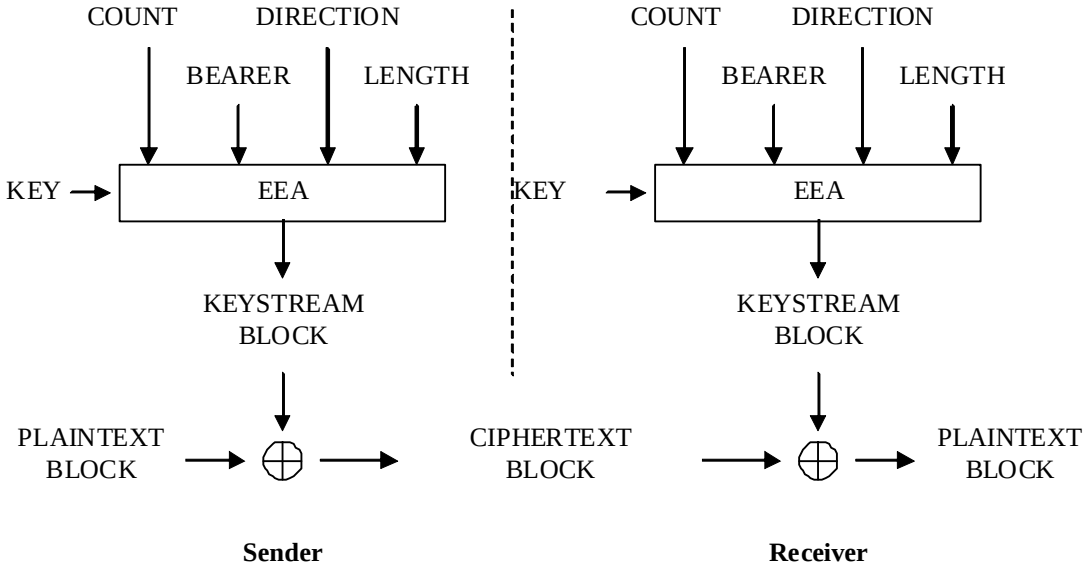


Figure A.3: Ciphering of data

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 should affect only the length of the KEYSTREAM BLOCK, not the actual bits in it.

A.1.3.3 128-EEA2

128-EEA2 is based on 128-bit AES [i.16] in CTR mode [i.17].

The sequence of 128-bit counter blocks needed for CTR mode T_1, T_2, …, T_i, … should be constructed as follows:

The most significant 64 bits of T_1 consist of COUNT[0] .. COUNT[31] ‖ BEARER[0] .. BEARER[4] ‖ DIRECTION ‖ 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 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 [i.17] mod 2^{64} to the least significant 64 bits of the previous counter block.

A.1.4 5G Cryptography: operation of the integrity algorithm

A.1.4.1 General

According to ETSI TR 133 401 [i.20], the operation of the integrity algorithm and the construction of the initialization vectors is summarized in figure A.4 and in the text below. Text is shown for the AES128 (named EIA2) case.

A.1.4.2 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 should be 0 for uplink and 1 for downlink. The bit length of the MESSAGE is LENGTH.

Figure A.4 illustrates the use of the integrity algorithm EIA to authenticate the integrity of messages.

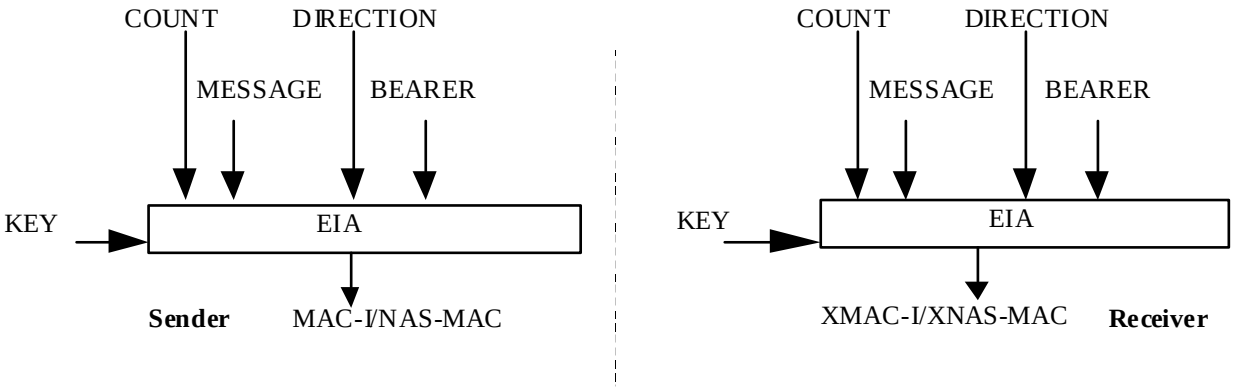


Figure A.4: Derivation of MAC-I/NAS-MAC (or XMAC-I/XNAS-MAC)

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. For integrity protection algorithms other than EIA0 the receiver computes the expected message authentication code (XMAC-I/XNAS-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.

A.1.4.3 128-EIA2

128-EIA2 is based on 128-bit AES [i.20] in CMAC mode [i.18].

The bit length of MESSAGE is BLENGTH.

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

M_0 .. M_{31} = COUNT[0] .. COUNT[31]

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

M_{37} = DIRECTION

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

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

and so Mlen = 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.

Annex B: Bibliography

Draft new Report ITU-R M [IMT-2020.TECH PERF REQ].

ITU Radiocommunication Study Groups; Working Party 5D; draft new Report ITU-R M.[IMT-2020.EVAL]: "Guidelines for evaluation of radio interface technologies for IMT-2020".

ITU Radiocommunication Study Groups; Working Party 5D; Attachment 7.4 to Document 5D/758; Liaison Statement to External Organizations; Further information related to draft new Report for IMT-2020 evaluation.

Guidelines for evaluation of radio interface technologies for IMT-2020, ITU, Revision 2 to Document 5D/TEMP/347-E, 20 June 2017.

History

Document history		
V1.1.1	February 2020	Publication