ETSI

EUROPEAN
TELECOMMUNICATION
STANDARD

ETS 300 175-7

September 1996

Second Edition

Source: ETSI TC-RES

Reference: RE/RES-03027-7

ICS: 33.060. 33.060.50

Key words: DECT, radio, security

Radio Equipment and Systems (RES);
Digital Enhanced Cordless Telecommunications (DECT);
Common Interface (CI);
Part 7: Security features

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Contents

Blank page

Foreword

This second edition European Telecommunication Standard (ETS) has been produced by the Radio Equipment and Systems (RES) Technical Committee of the European Telecommunications Standards Institute (ETSI).

Annexes A to K to this ETS are informative.

The following cryptographic algorithms are subject to controlled distribution:

a) DECT standard cryptographic algorithms;

b) DECT standard cipher.

These algorithms are distributed on an individual basis. Further information and details of the current distribution procedures can be obtained from the ETSI Secretariat at the address on the first page of this ETS.

Further details of the DECT system may be found in the ETSI Technical Reports ETR 015, ETR 043 and ETR 056.

This ETS forms part 7 of a series of 9 laying down the arrangements for the Digital Enhanced Cordless Telecommunications (DECT) Common Interface (CI).

Part 1: "Overview".

Part 2 "Physical layer (PHL)".

Part 3 "Medium Access Control (MAC) layer".

Part 4 "Data Link Control (DLC) layer".

Part 5: "Network (NWK) layer".

Part 6: "Identities and addressing".

Part 7: "Security features".

Part 8: "Speech coding and transmission".

Part 9: "Public Access Profile (PAP)".

Transposition dates	
Date of adoption of this ETS:	6 September 1996
Date of latest announcement of this ETS (doa):	31 December 1996
Date of latest publication of new National Standard or endorsement of this ETS (dop/e):	30 June 1997
Date of withdrawal of any conflicting National Standard (dow):	30 June 1997

Introduction

This ETS contains a detailed specification of the security features which may be provided by DECT systems. An overview of the processes required to provide all the features detailed in this ETS is presented in figure 1.

The ETS consists of four main clauses (clauses 4 - 7), together with a number of informative and important annexes (A - J). The purpose of this introduction is to briefly preview the contents of each of the main clauses and the supporting annexes.

Each of the main clauses starts with a description of its objectives and a summary of its contents. Clause 4 is concerned with defining a security architecture for DECT. This architecture is defined in terms of the security services which may be offered (subclause 4.2), the mechanisms which must be used to provide these services (subclause 4.3), the security parameters and keys required by the mechanisms (challenges, keys etc.), and which must be passed across the air interface or held within DECT Portable Parts (PPs), Fixed Parts (FPs) or other network entities (e.g. management centres) (subclause 4.4), the processes which are required to provide the security mechanisms (subclause 4.5), and the recommended combinations of services (subclause 4.6).

Clause 5 is concerned with specifying how certain cryptographic algorithms are to be used for the security processes. Two algorithms are required:

- a key stream generator; and
- an authentication algorithm.

The key stream generator is only used for the encryption process, and this process is specified in subclause 4.4. The authentication algorithm may be used to derive authentication session keys and cipher keys, and is the basis of the authentication process itself. The way in which the authentication algorithm is to be used to derive authentication session keys is specified in subclause 5.2. The way in which the algorithm is to be used to provide the authentication process and derive cipher keys is specified in subclause 5.3.

Neither the key stream generator nor the authentication algorithm are specified in this ETS. Only their input and output parameters are defined. In principle, the security features may be provided by using appropriate proprietary algorithms. The use of proprietary algorithms may, however, limit roaming in the public access service environment, as well as the use of PPs in different environments.

For example, for performance reasons, the key stream generator will need to be implemented in hardware in PPs and FPs. The use of proprietary generators will then limit the interoperability of systems provided by different manufacturers.

Two standard algorithms have been specified. These are the DECT Standard Authentication Algorithm (DSAA, see annex H) and the DECT Standard Cipher (DSC, see annex I).

Because of the confidential nature of the information contained in them, these annexes are not included in this ETS. However, the algorithms will be made available to DECT equipment manufacturers. The DSAA may also need to be made available to public access service operators who, in turn, may need to make it available to manufacturers of authentication modules.

Clause 6 is concerned with integrating the security features into the DECT system. Four aspects of integration are considered. The first aspect is the association of user security parameters (in particular, authentication keys) with DECT identities. This is the subject of subclause 6.2. The second aspect of integration is the definition of the NWK layer protocol elements and message types needed for the exchange of authentication parameters across the air interface. This is dealt with in subclause 6.3. The MAC layer procedures for the encryption of data passed over the air interface are the subject of subclause 6.4. Finally, subclause 6.5 is concerned with security attributes which DECT systems may support, and the NWK layer messages needed to enable PPs and FPs to identify which security algorithms and keys will be used to provide the various security services.

Clause 7 is concerned with key management issues. Careful management of keys is fundamental to the effective operation of a security system, and subclause 7.2 is intended to provide guidance on this subject. The subclause includes an explanation of how the DECT security features may be supported by different key management options.

For example, schemes which allow authentication keys to be held in a central location within a public access service network are described, as are schemes which allow authentication keys to be derived locally in public access service base stations. The subclause is very much less specific than the other subclauses in this ETS. This is because the key management issues discussed are not an integral part of the CI. In the end it is up to network operators and service providers to decide how they are going to manage their cryptographic keys. This ETS can at best provide some suggestions and guidelines.

The main text is supplemented by a set of informative annexes. There are two types of annex. Those of the first type provide background information justifying the inclusion of a particular service, or the use of a particular type of mechanism in the security features. Those of the second type provide guidance on the use and management of certain of the security features. The content of each of the annexes is briefly reviewed below.

Annex A contains the results of a security threats analysis which was undertaken prior to designing the DECT security features.

Annex B is concerned with the impact of the security features on roaming, in particular with the concurrent use of a PP in public access service, wireless Private Branch eXchange (PBX) and residential environments.

Annex C is provided for background information. It contains a justification for some of the decisions taken by EG-1, e.g. why symmetric rather than public key (asymmetric) cryptographic mechanisms were selected.

Annex D provides an overview of the DECT security features specified in this ETS.

No security system is perfect, and annex E discusses the limitations of the DECT security features.

Annex F relates the security features specified in this ETS to the DECT environments identified in ETR 015 (see Bibliography). Each of the local networks identified in the reference model is considered in turn. For each of these networks a security profile is suggested. The networks considered are Public Switched Telephone Network (PSTN), Integrated Services Digital Network (ISDN), X.25, Global System for Mobile communications (GSM), Local Area Networks (LANs) and public access service.

Annex G consists of a brief discussion of the compatibility of DECT and GSM authentication. In particular, the concept of a DECT Authentication Module (DAM) is considered and its functionality compared with the functionality of the GSM Subscriber Interface Module (SIM).

Annex H refers to the DECT standard authentication algorithm.

Annex J refers to the DECT standard cipher.

AC   Authentication Code
IV   Initialisation Value obtained from frame counter
CK   Cipher Key
SCK   Static Cipher Key
KS   Session Authentication Key
KS'   Reverse Authentication Key
RAND F   Value generated and transmitted by FP
RAND P   Value generated and transmitted by PP
RES 1   Value computed and transmitted by PP
RES 2   Value computed and transmitted by FP
RS   Value transmitted by FP in authentication protocol
UAK   User authentication Key
UPI   User Personal Identity
DCK   Derived Cipher Key
K   Authentication Key
A11, A12   Authentication Processes
A21, A22   Authentication Processes
B1, B2   Authentication Key Stream Processes
KSG   Key Stream Generator

UAK [128]

UPI [e.g. 128]   B2   Authentication Key Selection

UAK [128]   B1

AC [e.g. 16-32]   B1

K [128]

RS [64]   A11   Authentication of PP processes

RAND F [64]   KS [128]   A12   RES1 [32]

                                                DCK

SCK [64]   SCK   CK [64]   KSG   Key Stream

IV [35]

                                A21   Authentication of FP processes

RAND P [64]   KS' [128]   A22   RES2 [32]

Key Stream generation for encryption process

Figure 1: Overview of DECT security processes

1 Scope

This second edition ETS is part of the Digital Enhanced Cordless Telecommunications (DECT) Common Interface (CI) and specifies the security architecture, the types of cryptographic algorithms required, the way in which they are to be used, and the requirements for integrating the security features provided by the architecture into the DECT CI . It also describes how the features can be managed and how they relate to certain DECT fixed systems and local network configurations.

The security architecture is defined in terms of the security services which are to be supported at the CI, the mechanisms which are to be used to provide the services, and the cryptographic parameters, keys and processes which are associated with these mechanisms.

The security processes specified in this ETS are each based on one of two cryptographic algorithms:

- an authentication algorithm; and
- a key stream generator.

The architecture is, however, algorithm independent, and either the DECT standard algorithms, or appropriate proprietary algorithms, or indeed a combination of both can, in principle, be employed. The use of the employed algorithm is specified in this ETS.

Integration of the security features is specified in terms of the protocol elements and processes required at the Network (NWK) and Medium Access Control (MAC) layers of the CI.

The relationship between the security features and various network elements is described in terms of where the security processes and management functions may be provided.

This ETS does not address implementation issues. For instance, no attempt is made to specify whether the DSAA should be implemented in the PP at manufacture, or whether the DSAA or a proprietary authentication algorithm should be implemented in a detachable module. Similarly, this ETS does not specify whether the DSC should be implemented in hardware in all PPs at manufacture, or whether special PPs should be manufactured with the DSC or proprietary ciphers built into them. The security architecture supports all these options, although the use of proprietary algorithms may limit roaming and the concurrent use of PPs in different environments.

2 Normative references

This European Telecommunication Standard (ETS) incorporates, by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this ETS only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1]   ETS 300 175-1 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 1: Overview".

[2]   ETS 300 175-2 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 2: Physical Layer (PHL)".

[3]   ETS 300 175-3 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 3: Medium Access Control (MAC) layer".

[4]   ETS 300 175-4 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI) Part 4: Data Link Control (DLC) layer".

[5]   ETS 300 175-5 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 5: Network (NWK) layer".

[6]   ETS 300 175-6 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 6: Identities and addressing".

[8]   ETS 300 175-8 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 8: Speech coding and transmission".

[9]   ETS 300 175-9 (1996): "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 9: Public Access Profile (PAP)".

3   Definitions and abbreviations

3.1   Definitions

For the purposes of this ETS, the following definitions apply:

algorithm: A mathematical process or function that transforms an input into an output. In cryptographic applications an algorithm is a process used for encipherment, decipherment or non-reversible encipherment under control of a key.

algorithm identifier: A designator to show which algorithm is in use, so that the correct one may be chosen.

asymmetric algorithm: See public key algorithm.

authentication: The corroboration that an entity is the one that is claimed.

Cipher Key (CK): A value that is used to determine the transformation of plaintext to ciphertext in a cryptographic algorithm.

Cipher Key (CK) generation: A process for generating cryptographic keys.

ciphertext: The output of a cryptographic algorithm. Ciphertext is not intelligible unless (in a reversible algorithm) the reverse transformation is performed.

confidentiality: Rendering information secret as ciphertext unless the capability is possessed to recover the plaintext from ciphertext.

countermeasure: A device, instrument or procedure used to counteract or defend against a threat.

cryptography: Literally secret writing. Used to describe the hiding of information.

Data Encryption Standard (DES): United States Federal data encryption standard.

decipherment: The rendering of ciphertext into plaintext.

DECT Standard Authentication Algorithm (DSAA): An algorithm used for authentication in DECT.

DECT Standard Cipher (DSC): An algorithm used for data encryption in DECT.

encipherment: the rendering of plaintext into ciphertext.

FEAL algorithm: Fast Encryption Algorithm; a particular encryption algorithm in the public domain.

GSM: A pan-European standard for digital mobile telephones in the 900 - 1 000 MHz band.

impersonation: Where one identity claims the part of another identity.

Integrated Services Digital Network (ISDN): A digital telecommunications infrastructure to the Consultative Committee on International Telegraphy and Telephony (CCITT) standards.

key management: The way in which cryptographic keys are generated, distributed and used.

Key Stream Generator (KSG): A cryptographic algorithm which produces a stream of binary digits which can be used for encipherment and decipherment.

Local Area Network (LAN): Electronic systems which are interconnected and in physical proximity to each other.

masquerading: Where one identity plays the part of, or acts as, another identity.

mutual authentication: Where two entities corroborate the identity of each other.

Personal Identity Number (PIN): A short sequence of numbers (usually 4 - 8 digits) which may be used in an authentication process to prove identity. The term User Personal Identity (UPI) may also be used.

plaintext: Information or data which is intelligible to everyone.

proprietary algorithm: An algorithm which is the intellectual property of a legal entity.

public access service: A service that provides access to a public network for the general public.

public key algorithm: A cryptographic algorithm in which a different key is used for encipherment and for decipherment. Also known as an asymmetric algorithm.

random number: A number generated by a non-deterministic process.

RS: A value used to establish authentication session keys, as defined in subclause 4.4.3.

RSA (Rivest, Shamir & Adleman) algorithm: A public key algorithm.

security attribute: A protocol element indicating security services, mechanisms, processes or algorithms that are supported.

Session Key (KS): A key which is used only for a single session; a session may be a single connection or call, or it may be a number of calls made by a particular user through a particular system (e.g. the calls made by a roaming portable with a particular visited network).

Subscriber Interface Module (SIM): A smart card used for authentication in GSM.

stream cipher: An algorithm in which the output is combined bit by bit with plaintext to produce the ciphertext.

symmetric algorithm: A cryptographic algorithm in which the same key is used for both encipherment and decipherment.

synchronisation: Methods used to ensure that time correspondence exists between processes to ensure that data is not repeated or lost.

threat: An indication of coming evil.

User Authentication Key (UAK): A cryptographic key held by a user to prove identity. May also be combined with a PIN ("Something you have and something you know").

XRES1: An expected response calculated by an Fixed radio Termination (FT), as defined in subclause 4.4.2.

XRES2: An expected response calculated by a Portable radio Termination (PT), as defined in subclause 4.4.2.

X.25: A packet switched network based on CCITT Recommendation X25.

3.2   Abbreviations

For the purposes of this ETS, the following abbreviations apply.

A	Algorithm
AC	Authentication Code
BCT	Business Cordless Telephone
CK	Cipher Key
C-plane	Control plane
CRFP	Cordless Radio Fixed Part
DAM	DECT Authentication Module
DCK	Derived Cipher Key
DECT	Digital Enhanced Cordless Telecommunications
DLC	Data Link Control
DSAA	DECT Standard Authentication Algorithm
DSC	DECT Standard Cipher
ECN	Exchanged Connection Number
FP	DECT Fixed Part
FT	Fixed radio Termination
HDB	Home Data Base
IPUI	International Portable User Identity
IV	Initial Vector
K	authentication Key
KS	PT authentication Session Key
KS'	FT authentication Session Key
KSG	Key Stream Generator
KSS	Key Stream Segment
LAPC	a DLC protocol
MAC	Medium Access Control layer
MSC	Mobile Switching Centre
PAS	Public Access Service
PBX (PABX)	Private (Automatic) Branch eXchange
PP	DECT Portable Part
PSTN	Public Switched Telephone Network
PT	Portable radio Termination
RAND_F	a RANDom challenge issued by an FT
RAND_P	a RANDom challenge issued by a PT
RES1	a RESponse calculated by a PT
RES2	a RESponse calculated by an FT
RFP	Radio Fixed Part
RU	Residential Unit
SCK	Static Cipher Key
TPUI	Temporary Portable User Identity
UAK	User Authentication Key
UPI	User Personal Identification
U-plane	User plane
VDB	Visitors Data Base

4 Security architecture

4.1 Background

Subclause 4.2 contains a description of each of the security services provided in the DECT system. Five services are provided: authentication of a PT, authentication of a FT, mutual authentication, data confidentiality and user authentication. For a discussion of the way in which these security services may be applied in different DECT environments, the reader should refer to annex F.

A description of the mechanisms which are used to provide the security services is given in subclause 4.3. Throughout subclause 4.3 a number of security parameters and processes are referred to. A description of these parameters is given in subclause 4.4, and the processes are defined in subclause 4.5.

Subclause 4.6 describes how the various security services may be combined.

4.2 Security services

4.2.1 Authentication of a PT

This is an FT initiated service which enables an FT to authenticate a PT making or receiving a call through it.

The service is invoked at the beginning of a call. It may be re-invoked at any time during a call.

Authentication of a PT is a NWK layer service.

4.2.2 Authentication of an FT

This is a PT initiated service which enables a PT to authenticate an FT through which it is making or receiving a call.

The service is invoked at the beginning of a call, and may be re-invoked at any time during a call.

Authentication of an FT is a NWK layer service.

4.2.3 Mutual authentication

This service enables a PT and an FT, through which a call is connected, to authenticate each other.

This service may be provided by combining a number of other security services as described in subclause 4.6.

4.2.4 Data confidentiality

This service provides for the confidentiality of user data and certain control data transmitted between a PT and an FT.

Data confidentiality is requested at the NWK layer, although the service is provided at the MAC layer.

The service is provided only over the CI. It does not provide any cryptographic protection for data passed through the fixed networks.

4.2.5 User authentication

The user authentication service allows an FT to authenticate a user of a PT by checking a User Personal Identity (UPI) value associated with that user. This service is similar to on-line PIN verification provided by banking systems.

The user authentication service is initiated by the FT. It is invoked at the beginning of a call. It may be re-invoked at any time during a call.

4.3 Security mechanisms

4.3.1 Authentication of a PT

The purpose of this subclause is to define the mechanism which is used to provide the authentication of a PT service defined in subclause 4.2.1.

The service is provided using a cryptographic challenge-response mechanism. The FT issues a challenge to the PT, which responds by returning the result of a computation performed using the challenge and an authentication key associated with the PT. The FT compares the response from the PT with the value it expects to receive, and deems the authentication to be successful if the two values agree. In this way the PT is authenticated by demonstrating knowledge of the authentication key associated with it.

The authentication exchange, which includes a key management feature, is illustrated in figure 2 and proceeds as follows:

1)   the FT obtains a value RS, a value RAND_F, and a value XRES1.

     The value XRES1 is the expected result of a computation applied to RS, RAND_F and the authentication key K associated with the PT.
     The computation is performed in two stages using the authentication processes A11 and A12 defined in subclause 4.5.3. The first stage uses A11 to produce a value KS from RS and K. The second stage uses A12 to produce XRES1 from RAND_F and KS. These two computations may be performed by different entities within the fixed network and, provided the value RS is not changed, the computation of KS need not be repeated for every instance of authentication. All values may be computed in advance of the instance of authentication.

     The FT sends the values RS and RAND_F to the PT;

2)   on receipt of RS and RAND_F, the PT uses the authentication process A11 to compute KS from RS and the authentication key K, and then uses the authentication process A12 to compute RES1 from KS and RAND_F. It then sends RES1 to the FT;

3)   on receipt of RES1, the FT compares this value with XRES1. If the two values are identical, the FT accepts the authenticity of the PT.

```
        PT                                              FT

                                            Obtain: RS, RAND_F, XRES1
                        RS, RAND_F
          <─────────────────────────────────────────
Compute: RES1

                            RES1
          ─────────────────────────────────────────>
                                            Compare: RES1 with XRES1
```
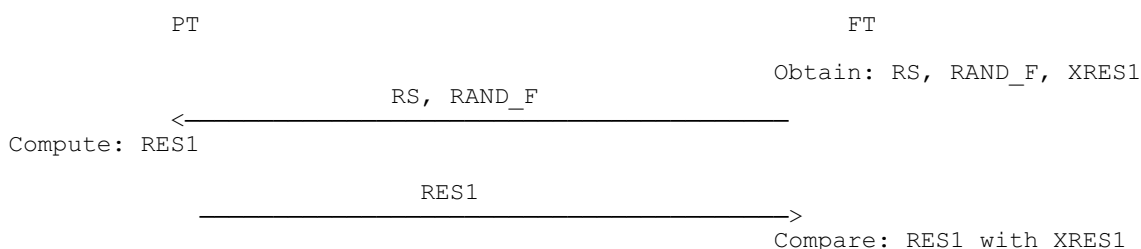
Figure 2: Authentication of PT

The value KS is called an authentication session key. It may be used in a roaming environment to provide a visited system with an authentication key for a particular visiting PT, without the PT's authentication key K being divulged to the visited system. The PT's home system, or a management centre associated with that system, selects an RS for the visited system and generates a KS from RS and the PT's authentication key K using the process A11. The visited system is provided with the pair (RS, KS) for the particular PT. Whenever the visited system authenticates the PT it sends the given value of RS (along with its challenge Rand_F) to the PT so that the PT may compute KS.

In general, the way in which the FT obtains the values RS, RAND_F and XRES1 depends upon the particular DECT application and the fixed network management. Thus a detailed specification of all options is outside the scope of this ETS. Some options are however outlined in clause 7. Clause 7 also contains a more detailed description of the way in which the session authentication keys may be used in a roaming environment as outlined above.

A detailed description of the parameters RAND_F and RS is given in subclause 4.4.2. The description specifies requirements for the generation of these parameters.

The authentication key K is described in subclause 4.4.3. The authentication process A11 used to compute KS from RS and K is described in subclause 4.5.3.1, and the authentication process A12 used to compute XRES1 (and RES1) from RAND_F and KS is described in subclause 4.5.3.2.

4.3.2 Authentication of an FT

The purpose of this subclause is to define the mechanism which is used to provide the authentication of an FT service defined in subclause 4.2.2.

The service is provided using a cryptographic challenge-response mechanism. The PT sends a challenge to the FT, which responds by returning the result of a computation performed using the challenge and an authentication key associated with the PT. The PT computes the expected value for the computation, and deems the authentication to be successful if this value agrees with the one received from the FT. In this way, the FT is authenticated by demonstrating that it can provide the result of a computation that depends upon knowledge of an authentication key associated with the challenging PT.

> NOTE: Throughout the text the authentication key associated with the PT, and used to authenticate the FT, is denoted K. The reader should, however, note that this key may be different from that associated with the PT and used to authenticate the PT as described in subclause 4.3.1. Similarly, the values RS used in the two mechanisms are not necessarily the same.

The authentication exchange, which includes a key management feature, is illustrated in figure 3 and proceeds as follows:

1) the PT generates a value RAND_P and sends it to the FT;

2) the FT obtains a value RS and a value RES2, and sends them both to the PT.

   The value RES2 is the result of a computation applied to RS, RAND_P and the authentication key K associated with the PT. The computation is performed in two stages using the authentication processes A21 and A22 defined in subclause 4.5.3. The first stage uses A21 to produce a value KS' from RS and K. The second stage uses A22 to produce RES2 from RAND_P and KS'. These two computations may be performed by different entities within the fixed network. Moreover, the computation of KS' may be performed in advance of the instance of authentication and, provided the value of RS is not changed, the computation does not need to be repeated for subsequent instances of authentication;

3) on receipt of RS and RES2, the PT applies the authentication process A21 to compute KS' from RS and the authentication key K, and then uses the authentication process A22 to compute an expected result XRES2 from KS' and RAND_P. If XRES2 is equal to the value RES2 received from the FT, the PT accepts the authenticity of the FT.

```
                       PT                                            FT
        Generate: RAND_P
                                        RAND_P
                         ------------------------------------------>
                                                        Obtain: RS, RES2
                                        RS, RES2
                         <------------------------------------------
        Compute: XRES2
        Compare: RES2 with XRES2
```
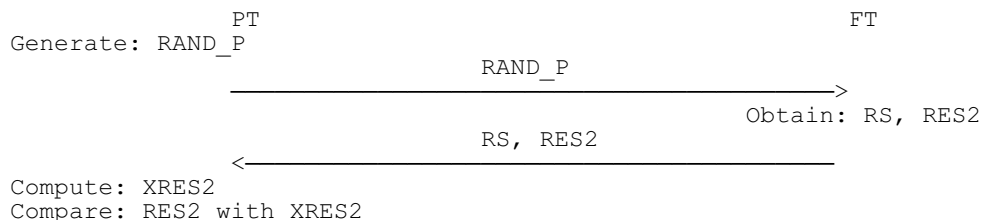
Figure 3: Authentication of FT

The value KS' is called an authentication session key. It may be used in a roaming environment to provide a visited system with a key by which it may authenticate itself to a particular visiting PT, without the PT's authentication key K being divulged to the visited system. The PT's home system, or a management centre associated with that system, selects an RS for the visited system and generates a KS' from RS and the PT's authentication key K. The visited system is provided with the pair (RS, KS') for the particular PT. The visited FT, or a management centre associated with it, always uses the given value of KS' to compute RES2 from RAND_P, and always sends the given RS in the response to the PT's challenge.

In general, the precise way in which the FT obtains the values RS and RES2 will depend upon the particular DECT application and the fixed network management. Thus a detailed specification of all options is outside the scope of this ETS. Some options are, however, outlined in clause 7. A detailed description of the parameters RAND_P and RS is given in subclause 4.4.2. The description specifies requirements for the generation of these parameters.

The authentication key, K, is described in subclause 4.4.3.

The authentication process A21, used to compute KS' from RS and K, is described in subclause 4.5.3.1, and the authentication process A22 used to compute XRES2 (and RES2) from RAND_P and KS', is described in subclause 4.5.3.2.

4.3.3 Mutual authentication

Mutual authentication is achieved by combining certain of the other services as described in subclause 4.6. No explicit mutual authentication mechanism is required.

4.3.4 Data confidentiality

In order to provide the data confidentiality service defined in subclause 4.2.4, both the PT and the FT must share a cipher key CK. This key is then used in conjunction with a key stream generator to generate a key stream for encrypting data at the MAC layer. The key stream generation process is specified in subclause 4.5.4. The detailed specification of how encryption is performed at the MAC layer is the subject of subclause 6.4. The purpose of this subclause is to specify how a CK may be established. Two ways are provided, thereby providing two types of key:

- a Derived Cipher Key (DCK); and
- a Static Cipher Key (SCK).

4.3.4.1 Derived Cipher Key (DCK)

The DCK is established as part of the procedure used to authenticate the PT specified in subclause 4.3.1. More precisely, the authentication process A12, when applied to KS and RAND_F, has two output values:

- the value RES1 (or XRES1); and
- the value DCK.

The PT computes DCK as part of the procedure described in 2), of subclause 4.3.1. The FT obtains DCK as part of the procedure for obtaining the values RS, RAND_F and XRES1 as described in 1) of subclause 4.3.1. The DCK shall be made available to the appropriate radio fixed part of the FT.

This technique allows a new cipher key to be established for each call, and used for the duration of that call, provided the PT is authenticated at the beginning of the call.

Under certain circumstances (see subclause 4.6), the DCK may be retained and used by the PT and FT for providing confidentiality for all calls until the next authentication of the PT takes place. At that point, it is replaced by the new DCK.

It should be noted that authentication of an FT cannot be used to establish a cipher key.

4.3.4.2 Static Cipher Key (SCK)

With the cipher key derivation process defined in subclause 4.3.4.1, the cipher key is obtained as part of the PT authentication process, with PT authentication being a necessary pre-requisite for its derivation. However, in certain applications, it may be desirable to be able to provide confidentiality without having to first apply authentication. In this case the parties may use a cipher key which has been established by other means. Such a key is called a Static Cipher Key (SCK).

> NOTE: Although this mechanism allows for confidentiality of user and control data without the need for authentication, no service is included within the DECT security features to provide for management of SCKs.

4.3.5 User authentication

User authentication is achieved by applying the authentication of the PT service using an authentication key K which is derived from a UPI value.

The UPI is entered manually into the PT by the user whenever the user authentication service is invoked. Within the PT the UPI is used to generate the authentication key K, possibly in combination with other key material already present in the PT. The key K is also generated by the system responsible for the user's subscription data (using the same key material) for use by the fixed system.

A mechanism is specified in subclause 4.5.2 to combine a UPI with a User Authentication Key (UAK) to produce an authentication key K for the user authentication service. A user authentication key is part of a user's subscription data, established at the time of subscription registration, and is described in subclause 4.4.3.

It should be stressed that user authentication requires a user to enter his UPI each time user authentication is required.

4.4 Cryptographic parameters and keys

4.4.1 Overview

In this subclause the cryptographic parameters and types of keys used in the security architecture are described.

A length in bits is given for each parameter or key. This is the length of the parameter or key for use with the DECT standard algorithms.

The expression "non-repeating" means that it should be highly unlikely that the value should repeat itself within the lifetime of the authentication key. A "non-repeating" value could be, for example, the output from a counter which is unlikely to repeat during the lifetime of the authentication key, or a date/time stamp.

The expression "randomly generated" means that it should not be possible to predict the value with a chance that is significantly larger than 0 (e.g. greater than $(0,5)^{64}$ for the values defined in this ETS).

4.4.2 Cryptographic parameters

For the authentication of a PT the following three parameters are used and transmitted over the air interface.

RAND_F:

This parameter is sent from the FT to the PT as part of the authentication of a PT exchange defined in subclause 4.3.1.

It has a length of 64 bits.

It is generated by the FT or within a local network. In a roaming scenario, it may be generated within the subscriber's home network (see subclause 7.2 for a discussion of the options for the generation of RAND_F in a roaming environment). A new value of RAND_F shall be randomly generated for each instance of authentication of a particular PT.

RS:

This parameter is used to enable roaming between network. It is an element of the mechanism which allows authentication session keys to be made available to visited networks (see also the discussion of KS and KS' in subclause 4.4.3).

It is sent from the FT to the PT as part of the authentication of a PT or authentication of an FT exchange defined in subclauses 4.3.1 and 4.3.2 respectively.

It has a length of 64 bits.

It is generated by the FP, within a local network or, in a roaming scenario, within the subscriber's home network.

Different values of RS may be used for authentication of the PT and authentication of the FT even though the two services may be invoked during a single call. It is, however, envisaged that, in this situation, the same value will be used.

The same value may be used a number of times and for a number of different subscribers. For example, in the roaming environment, a single constant value of RS may be selected by a home network for use by a particular visited network (see subclause 7.2 for more details).

RES1:

This value is sent from the PT to the FT, as part of the authentication of a PT exchange defined in subclause 4.3.1, where it is compared with XRES1.

It has a length of 32 bits.

It is computed by the PT from RAND_F and KS using the authentication process A12.

XRES1:

This value is compared with RES1 at the FT as part of the authentication of a PT mechanism. Authentication succeeds if the two values are equal.

It has a length of 32 bits.

It is computed from RAND_F and KS using the authentication process A12. Computation of XRES1 may be performed by the FT or within a local network. In a roaming scenario, it may be computed within the subscriber's home network (see subclause 7.2 for more details concerning the possible locations for computing XRES1 in a roaming environment).

For the authentication of an FT by a PT, the parameter RS and the following three parameters are used:

RAND_P:

This parameter is sent from the PT to the FT as part of the authentication of an FT exchange defined in subclause 4.3.2.

It has a length of 64 bits.

It is generated by the PT.

A new value of RAND_P shall be generated for each instance of authentication. The generation mechanism shall produce random or non-repeating values.

RES2:

This value is sent from the FT to the PT, as part of the authentication of an FT exchange defined in subclause 4.3.2, where it is compared with XRES2.

It has a length of 32 bits.

It is computed from RAND_P and KS' using the authentication process A22. Computation of RES2 may be performed by the FT or within a local network.

XRES2:

This value is compared with RES2 at the PT as part of the authentication of an FT mechanism. Authentication succeeds if the two values are equal.

It has a length of 32 bits.

It is computed by the PT from RAND_P and KS' using the authentication process A22.

4.4.3 Cryptographic keys

There are three types of cryptographic key used to provide the DECT security features: the authentication key K, the authentication session keys KS and KS', and the cipher key CK.

4.4.3.1 Authentication key K

This is the key value used in the authentication mechanisms defined in subclauses 4.3.1 and 4.3.2.

The authentication key, K, has a length of 128 bits.

The authentication key may be derived from other keys associated with the user. Such keys may be specific to a particular DECT application or a particular authentication service. Three such keys are identified below. The processes for deriving K from these keys are given in subclauses 4.5.2.

User Authentication Key (UAK):

This is secret authentication data contained within the subscriber's (user's) registration data. It is uniquely associated with the particular subscriber (user) and the subscription.

It may be used to derive K when authentication of the PT or authentication of an FT service is applied. It may be used in combination with the UPI to derive K when the user authentication service is applied.

The UAK is held in non-volatile memory within the PP (or within a detachable DECT Authentication Module (DAM)).

User Personal Identity (UPI):

This is usually a short value (e.g. 16 - 32 bits).

It is used in combination with the UAK to derive K when the user authentication service is applied.

A single UAK may have at most one UPI associated with it.

The UPI is not stored in the PT. It is entered manually into the PT by the user each time the user authentication service is required.

Authentication Code (AC):

This is usually a short value (e.g. 16 - 32 bits).

It may be used to derive K when authentication of the PT or authentication of an FT service is applied.

The AC may be held in non-volatile memory within the PP or it may be manually entered by the user when required for an authentication service. This depends upon the application.

> NOTE: There is no conceptual difference between a UAK and an AC. It is intended that an AC should be used only for those applications which require a temporary or short term key for PT or FT authentication. For all other applications a UAK is more appropriate.

For a discussion of the ways in which K (UAK, UPI and AC) may be distributed to the PT and managed within fixed networks the reader is referred to subclause 7.2.

4.4.3.2   Authentication session keys KS and KS'

Authentication session key, KS:

In a roaming environment it may be necessary for a PT's home network to provide visited networks with information sufficient for them to authenticate the PT, without necessarily having to divulge the authentication key K and without the need for a visited network to contact the home network at every instance of authentication. An authentication session key KS may be used for this purpose. The visited network is provided (by the home network) with a KS for use with a particular PT and may use it for an indefinite number of calls.

The key KS is computed from K and RS using the authentication process A11 as specified in subclause 4.5.3.1.

Computation of KS within the PT occurs (in real-time) as part of the authentication of a PT exchange.

Computation of KS on the fixed side of the air interface may be performed within the home network, within a local network or at the FT. The location depends upon the application and the key management procedures in force. Although computation of KS is part of the authentication of a PT mechanism, it does not have to be performed at an instance of authentication or repeated for each instance of authentication. It only needs to be repeated if the value of RS (or K) is changed.

KS has a length of 128 bits.

Authentication session key, KS':

In a roaming environment it may be necessary for a PT's home network to provide visited networks with information sufficient for them to be authenticated by the PT, without necessarily having to divulge the authentication key K and without the need for a visited network to contact the home network at every instance of authentication. An authentication session key KS' may be used for this purpose. The visited network is provided (by the home network) with a KS' for use with a particular PT, and may use it for an indefinite number of calls.

The key KS' is computed from K and RS using the authentication process A21 as specified in subclause 4.5.3.1.

Computation of KS' within the PT occurs (in real-time) as part of the authentication of the FT exchange.

Computation of KS' on the fixed side of the air interface may be performed within the home network, within a local network or at the FT. The location depends upon the application and the key management procedures in force. Although computation of KS' is part of the authentication of the FT mechanism, it does not have to be performed at an instance of authentication or repeated for each instance of authentication. It only needs to be repeated if the value of RS (or K) is changed.

KS' has a length of 128 bits.

> NOTE 1:   The session keys KS and KS' may be derived from different values for K and RS, although it is envisaged that the values will usually be the same. One scenario where the values for K may well be different is when both user authentication and authentication of the FT are separately applied. In this case, the value for K used to apply user authentication may be derived from a UAK and a UPI (see subclause 4.4.3.1), whilst the value used for authentication of the FT may be derived simply from the UAK.

> NOTE 2:   If the same values for K and RS are used to derive KS and KS', the values of KS and KS' will be different; authentication processes A11 and A21 produce different outputs for identical inputs (see subclause 4.5.3.1).

4.4.3.3   Cipher key CK

This is a value which is used as the key for the encryption process. It may be a DCK. In this case it is computed using the authentication process A12 (as specified in subclause 4.5.3.2) during authentication of the PT.

Alternatively it may be a SCK shared by the FT and the PT. CK has a length of 64 bits.

4.5   Security processes

4.5.1   Overview

This subclause is divided into three main parts. In the first part, subclause 4.5.2, the processes which are supported for the derivation of the authentication key K are defined. The second part, subclause 4.5.3, is concerned with the authentication processes. In the third part, subclause 4.5.4, the basic encryption process is defined.

For an overview of the DECT security processes the reader is referred to figure 1. This illustrates the relationship between the processes defined in this subclause and the parameters and keys defined in subclause 4.4. The lengths of values are those for use with the DECT standard algorithms.

4.5.2 Derivation of authentication key, K

Three processes have been defined for the derivation of the authentication key K. These are specified in subclauses 4.5.2.1, 4.5.2.2 and 4.5.2.3.

Throughout this subclause the following notation is used:

- LEN_X   denotes the bit-length of value X;
- X[i]   denotes the i'th bit of X for $0 \le i \le$ LEN_X-1;
- X[0]   denotes the least significant bit of X.

4.5.2.1 K is derived from UAK

In this case the authentication key is derived from UAK using a process B1 as illustrated in figure 4.

```
UAK ——>———┌──────────────┐————>——— K
           │  B1 Process  │
           └──────────────┘
```
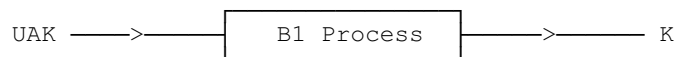
Figure 4: Derivation of K from UAK

The process B1 is defined as follows:

- K[i] = UAK [i MODULO LEN_UAK], $0 \le i \le$ LEN_K-1.

4.5.2.2 K is derived from AC

In this case K is derived from AC using the process B1 as illustrated below in figure 5.

```
AC ——>——┌──────────────┐————>——— K
         │  B1 Process  │
         └──────────────┘
```
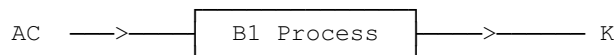
Figure 5: Derivation of K from AC

The process B1 is defined as follows:

- K[i] = AC [i MODULO LEN_AC], $0 \le i \le$ LEN_K-1.

4.5.2.3 K is derived from UAK and UPI

In this case, K is derived from UAK and UPI using a process B2 as illustrated below in figure 6.

```
UAK ——>———┌──────────────┐
           │  B2 Process  │————>———K
UPI ——>———└──────────────┘
```
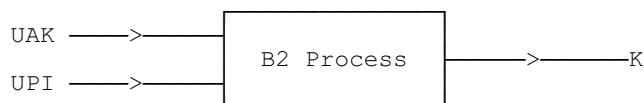
Figure 6: Derivation of K from UAK and UPI

The process B2 is defined as follows:

- K[i] = (UAK [i MODULO LEN_UAK] + UPI [i MODULO LEN_UPI]) MODULO 2, $0 \le i \le$ LEN_K-1.

4.5.3 Authentication processes

Authentication of a PT and authentication of an FT each require the use of an authentication process. In each case the authentication process is subdivided into two processes. The first process is used for the derivation of the authentication session key KS or KS' respectively. The second process is used for the derivation of DCK and RES1 (XRES1) or of RES2 (XRES2) respectively.

4.5.3.1 Processes for the derivation of KS and KS'

KS is derived from K and RS using process A11. KS' is derived from K and RS using process A21. This is illustrated below in figure 7.

```
K————————>
              ┌─────────────────┐
              │                 │
              │   A11 Process   │————>————— KS
              │                 │
RS———————>    └─────────────────┘


K————————>
              ┌─────────────────┐
              │                 │
              │   A21 Process   │————>————— KS'
              │                 │
RS———————>    └─────────────────┘
```
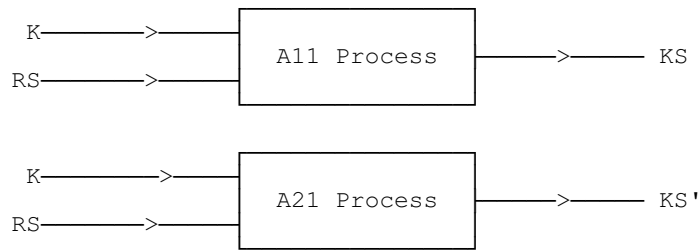
Figure 7: Derivation of KS and KS'

The processes A11 and A21 may be based on the use of an authentication algorithm as specified in subclause 5.2. Either the DSAA, or an appropriate proprietary authentication algorithm, may be used. Requirements for the authentication algorithm needed for these processes are discussed in more detail in subclause 5.2.

Alternatively, if the roaming features provided by the use of KS and KS' are not required, then a proprietary implementation may obtain KS and KS' directly from K (see subclauses 5.2.1 and 5.2.2). In this case, the protocol element RS may be input directly to a proprietary A12 (or A22) process along with RAND_F (respectively RAND_P). For an example see annex G, clause G.3.

> NOTE: The values for K and RS input to the A11 process may be different from the values input to the A21 process.

4.5.3.2 Processes for the derivation of DCK, RES1 and RES2

The values DCK and RES1 (XRES1) are derived from KS and RAND_F using a process A12. The value RES2 (XRES2) is derived from KS' and RAND_P using a process A22. This is illustrated below in figure 8.

```
KS ————————>
                ┌─────────────────┐
                │                 │————>————— RES1
                │   A12 Process   │
                │                 │————>————— DCK
RAND_F ————————>└─────────────────┘


KS' ————————>
                ┌─────────────────┐
                │                 │
                │   A22 Process   │————>————— RES2
                │                 │
RAND_P ————————>└─────────────────┘
```
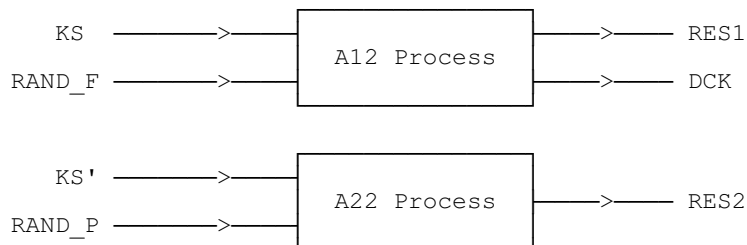
Figure 8: Derivation of RES1, DCK and RES2

The processes A12 and A22 are based on the use of an authentication algorithm as specified in subclause 5.3. Either the DSAA, or an appropriate proprietary authentication algorithm, may be used. Requirements for the authentication algorithm needed for these processes are discussed in more detail in subclause 5.3.

As an alternative, if the roaming features provided by the use of session authentication keys is not required, and if a proprietary process is used to derive KS (and KS') directly from K, then the parameter RS may also be input directly to a proprietary A12 (respectively A22) process. For an example see annex G, clause G.3.

4.5.4 Key stream generation

As part of the encryption process, a Key Stream Generator is used to generate a key stream from a CK (of type DCK or SCK) and an initialisation value IV, as depicted below in figure 9.

```
IV  ———————>—————┌──────────────┐
                 │              │
                 │     KSG      │————>——— Key stream
                 │              │
CK  ———————>—————└──────────────┘
```

Figure 9: Key stream generation

The way in which the key stream is used to encrypt data is described in subclause 6.4.4. Either the DSC, or an appropriate proprietary algorithm, may be used for the KSG.

4.6   Combinations of security services

The purpose of this subclause is to specify the envisaged combinations of security services. These combinations of services are as follows:

(S1)   authentication of a PT;

(S2)   authentication of a PT followed by authentication of an FT;

(S3)   authentication of a PT followed by data confidentiality using the DCK;

(S4)   authentication of a PT followed by authentication of an FT followed by data confidentiality using the DCK;

(S5)   data confidentiality using a SCK;

(S6)   data confidentiality using the DCK established at the last instance of S3 or S4.

The following points should be noted concerning these combinations of services:

1)   whenever S2 or S4 are applied, it is envisaged that the same authentication key K will be used for authentication of the PT and authentication of the FT. This is however not necessary, and an exception may be where user authentication is applied;

2)   S6 can only be used in those networks which have a management facility which can maintain the current DCK for a PT, and ensure that it is made available to an FT whenever needed;

3)   the above list of combinations of services does not distinguish between user authentication and authentication of the PT. If the application requires user authentication, then this is achieved by applying S1, S2, S3 or S4 with the authentication key (for authentication of the PT) derived using the UPI as explained in subclause 4.3.5;

4)   combinations S2 and S4 provide direct mutual authentication. Indirect mutual authentication is provided by S3, S5 or S6. In the case of S3, the principle underlying authentication of the FT is that if the FT does not know the authentication key K then it is unable to derive the DCK, so is unable to encrypt or decrypt data sent to or received from the PT. In the case of S5 or S6, the principle underlying the mutual authentication is that both parties must know the cipher key in order to successfully encrypt and decrypt data;

5)   it is not foreseen that S4 will be used in practice as the confidentiality service using the DCK from PT authentication, is sufficient to achieve FT authentication;

6)   although other combinations of services are possible, for some combinations, certain security weaknesses may result (see annex E);

7)   data confidentiality using the DCK (S6) provides a continuous form of authentication because only that PT which knows the DCK can correctly encrypt and decrypt the call.

5 Algorithms for security processes

5.1 Background

In this subclause, the security processes defined in subclause 4.5 are specified. Two sorts of processes will be described: the derivation of authentication session keys (subclause 5.2) and the authentication and derived cipher key generation processes (subclause 5.3).

These processes will be described in terms of a common algorithm A, the input and output parameters of which are specified in subclause 5.1.1. Throughout this subclause the following notation is used:

- LEN_X   denotes the bit-length of value X;
- X[i]   denotes the i'th bit of X for $0 \leq i \leq$ LEN_X-1; where
- X[0]   denotes the least significant bit of X.

5.1.1 A algorithm

The A algorithm has two inputs D1 and D2 and one output E. A block diagram of the A algorithm is given in figure 10.
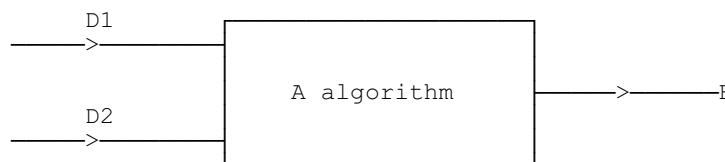
Figure 10: A algorithm

The A algorithm may be either DSAA, or a suitable proprietary algorithm. Requirements for the DSAA algorithm are summarised in annex D. Annex G illustrates a proprietary implementation by describing how the GSM SIM may be used to provide authentication in DECT.

5.2 Derivation of session authentication key(s)

5.2.1 A11 process

In the A11 process the session authentication key KS is derived from the authentication key K and the value RS.

For the derivation of KS the A algorithm is used, the input and output of which is specified in subclause 5.1.1. The inputs of the A algorithm are set as:

- D1 [i] = K [i MODULO LEN_K], $0 \leq i \leq$ LEN_D1-1; and

- D2 [i] = RS [i MODULO LEN_RS], $0 \leq i \leq$ LEN_D2-1.

The output KS of the A11 process is now defined in terms of the output E of the A Algorithm by:

- KS [i] = E [i MODULO LEN_E], $0 \leq i \leq$ LEN_KS-1.

If a proprietary algorithm A is used and authentication session keys are not required, then process A11 may be replaced by:

- KS [i] = K [i MODULO LEN_K], $0 \leq i \leq$ LEN_KS-1.

    NOTE: For a discussion of how roaming may be managed in this case see subclause 7.2.

5.2.2 A21 process

In the A21 process the session authentication key KS' is derived from the authentication key K and the value RS.

For the derivation of KS' the A Algorithm is used, the input and output of which is specified in subclause 5.1.1. The inputs of the A Algorithm are set as:

- D1 [i] = K [i MODULO LEN_K], $0 \leq i \leq$ LEN_D1-1; and

- D2 [i] = RS [i MODULO LEN_RS], $0 \leq i \leq$ LEN_D2-1.

The output KS' of the A21 process is now defined in terms of the output E of the A algorithm by:

- KS' [i] = (E [i MODULO LEN_E] + i) MODULO 2, $0 \leq i \leq$ LEN_KS'-1.

If a proprietary algorithm A is used and authentication session keys are not required, then process A21 may be replaced by:

- KS' [i] = (K [i MODULO LEN_K] + i) MODULO 2, $0 \leq i \leq$ LEN_KS'-1.

> NOTE: It is envisaged that for most instances of authentication of the PT and authentication of the FT the same K and RS values will be input to the A11 and A21 processes. In this case the outputs KS and KS' will be different. This is important since, as discussed in annex E, if KS and KS' are the same then it may be possible to launch certain reflection attacks. If different values for K or RS are used, then the system operator is advised to avoid pairs for which KS and KS' are the same.

5.3 Authentication and cipher key generation processes

5.3.1 A12 process

In the A12 process the value RES1 (XRES1) and the derived cipher key DCK are derived from the session key KS and the value RAND_F.

For the derivation of RES1 (XRES1) and DCK the A algorithm is used, the parameters of which are specified in subclause 5.1.1.

The inputs of the A algorithm are set as:

- D1 [i] = KS [i MODULO LEN_KS], $0 \leq i \leq$ LEN_D1-1; and

- D2 [i] = RAND_F [i MODULO LEN_RAND_F], $0 \leq i \leq$ LEN_D2-1

The outputs RES1 and DCK of the A12 process are now defined in terms of the output E of the A algorithm by:

- RES1 [i] = E [i MODULO LEN_E], $0 \leq i \leq$ LEN_RES1-1; and

- DCK [i] = E [(i + LEN_RES1) MODULO LEN_E], $0 \leq i \leq$ LEN_DCK-1.

5.3.2 A22 process

In the A22 process the value RES2 (XRES2) is derived from the session key KS' and the value RAND_P.

For the derivation of RES2 (XRES2) the A algorithm is used, the parameters of which are specified in subclause 5.1.1. The inputs of the A algorithm are set as:

- D1 [i] = KS' [i MODULO LEN_KS'], $0 \leq i \leq$ LEN_D1-1; and

- D2 [i] = RAND_P [i MODULO LEN_RAND_P], $0 \leq i \leq$ LEN_D2-1.

The output RES2 of the A22 process is now defined in terms of the output E of the A algorithm by:

- RES2 [i] = E [i MODULO LEN_E], $0 \leq i \leq$ LEN_RES2-1.

6 Integration of security

6.1 Background

The purpose of this subclause is to specify how the security features defined in previous subclauses are to be integrated into the DECT air interface. Four aspects of integration are considered. Subclause 6.2 is concerned with the first aspect: the way in which the cryptographic keys are associated with the DECT identities defined in ETS 300 175-6 [6]. Subclause 6.3 deals with the NWK layer messages and procedures needed for the authentication exchanges. Subclause 6.4 is concerned with the MAC layer procedures and messages needed for encryption of data over the air interface. The contents of subclauses 6.3 and 6.4 are previewed in more detail in subclauses 6.3.1 and 6.4.1 respectively. Finally, subclause 6.5 is concerned with the NWK layer messages needed to enable a PT and an FT to identify the security algorithms and keys which will be used. Subclause 6.5 also contains a description of the NWK layer control of ciphering and a method for over-the-air allocation of a UAK from a manually distributed AC.

6.2 Association of keys and identities

6.2.1 Authentication key

The purpose of this subclause is to specify how an authentication key K is associated with the DECT identities defined in ETS 300 175-6 [6]. The association is specified for each of the three ways of deriving K identified in subclause 4.5.2.

> NOTE: Throughout it is assumed that a PT has one or more IPUIs, each of which relates it to a subscription. Such an IPUI is the unique identity of the PT for within a particular system or network of FPs. This system may, for example, consist of a simple residential FP, the collection of all FPs belonging to a public access service system or the set of FPs for a single PBX system.

6.2.1.1 K is derived from UAK

If K is derived from a UAK, then in the PT the UAK is associated with either an IPUI or an IPUI/PARK pair. All authentication information required by fixed system elements (e.g. FT, local network, management centre for subscriber's home network) is associated with either the IPUI or the IPUI/PARK pair.

Included in this authentication information is data which may be provided to an FT or a visited network in a roaming scenario. For example, the subscriber's home network may provide a visited network with pairs (RS,KS) and (RS,KS') for the particular PT, in which case these pairs would be associated with either the IPUI or the IPUI/PARK pair.

> NOTE 1: For a discussion of the authentication data which may be transferred in a roaming environment, see subclause 7.2.

There may be more than one UAK associated with a single IPUI or IPUI/PARK pair. In this case the different keys are numbered as described in subclause 6.5.5.

NOTE 2: The <<AUTH-TYPE>> information element identifies if the key is related to the IPUI or IPUI/PARK pair, see also subclause 6.5.5.

6.2.1.2 K derived from AC

If K is derived from an AC, then in the PT the AC is associated with either an IPUI or an IPUI/PARK pair.

All authentication information required by fixed system elements (e.g. FT, local network) is associated with either the IPUI or the IPUI/PARK pair.

There may be more than one AC associated with a single IPUI or IPUI/PARK pair. In this case the different keys are numbered as described in subclause 6.5.5.

NOTE: The <<AUTH-TYPE>> information element identifies if the key is related to the IPUI or IPUI/PARK pair, see also subclause 6.5.5.

6.2.1.3 K derived from UAK and UPI

If K is derived from an UAK and an UPI, then in a PT the UAK and the manually entered UPI are both associated with either an IPUI or an IPUI/PARK pair.

All authentication information required by fixed system elements (e.g. FT, local network, management centre for the subscriber's home network) is associated with either the IPUI or the IPUI/PARK pair.

A single UAK may have at most one UPI associated with it.

NOTE: The <<AUTH-TYPE>> information element identifies if the key is related to the IPUI or IPUI/PARK pair, see also subclause 6.5.5.

6.2.2 Cipher keys

If the cipher key DCK is derived as part of authentication of the PT as described in subclause 4.3.4.1, then it is always associated with the same identity as the authentication key from which it is derived.

There may be more than one DCK associated with a single IPUI or IPUI/PARK pair. In this case the keys are numbered as described in subclause 6.5.5. (see note 1).

If the cipher key SCK is a static key, as described in subclause 4.3.4.2, then it is associated with either an IPUI or an IPUI/PARK pair.

There may be more than one SCK associated with a single IPUI/PARK pair. In this case the keys are numbered as described in subclause 6.5.5.

NOTE 1: Allowing a number of different DCKs, which will be updated at different times, could result in management problems for the fixed network. In most applications it is envisaged that one DCK per IPUI is sufficient.

NOTE 2: The <<CIPHER-INFO>> information element identifies if the key is related to the IPUI or IPUI/PARK pair, see also subclause 6.5.5.

6.3 NWK layer procedures

6.3.1 Background

The purpose of this subclause is to specify the NWK layer procedures for the security mechanisms defined in subclause 4.3.

The structure of the NWK layer messages is illustrated in figure 11, where the formats for the transfer of elements are depicted.

The NWK layer messages required for the authentication exchanges are specified in subclause 6.3.2. Subclause 6.3.3 contains a list of the sequences of messages which are sent for the specific authentication mechanisms defined in subclauses 4.3.1 and 4.3.2 as well as a description of the processes carried out on each of the messages. Subclause 6.3.4 identifies the need for primitives to establish the cipher key using the mechanisms defined in subclause 4.3.4.

NOTE: This subclause is concerned specifically with the NWK layer messages needed for the mechanisms defined in subclause 4.3. Other NWK messages or information elements which are needed to identify the keys and algorithms to be used by these mechanisms are defined in subclause 6.5.

```
        8 bits        8 bits
  |<——————————>|<——————————>|
  ┌──────┬──────┬───┬──────────┐ ──────────────────────────────
  │Trans.│Protoc.│ 0 │ Message  │
  │Ident.│Discr. │   │  Type    │
  └──────┴──────┴───┴──────────┘ ──────────────────────────────


        8 bits        8 bits        0-255 bytes
  |<——————————>|<——————————>|<——————————>|
 ──────────┬───┬──────────┬──────────┬──────────┐ ──────────
           │ 0 │ Element  │ Length of│ Contents │
           │   │Identifier│ element  │of element│
 ──────────┴───┴──────────┴──────────┴──────────┘ ──────────
```
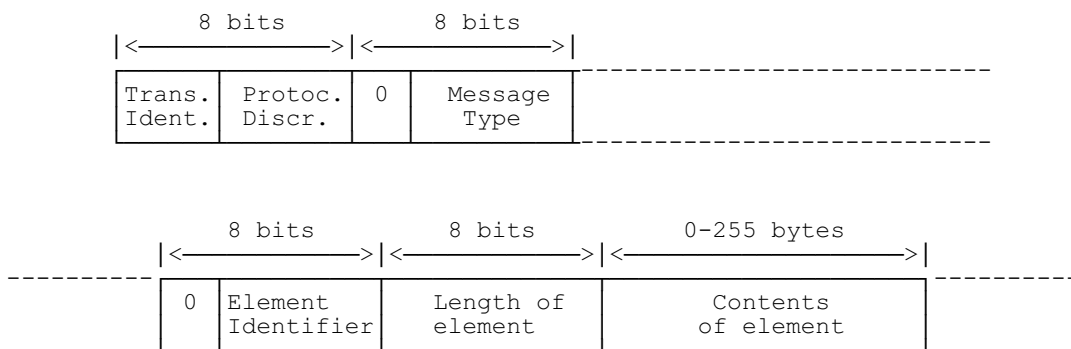
Figure 11: NWK Layer Message Structure (NLMS)

6.3.2 Authentication exchanges

The messages required for the authentication exchanges described in subclauses 4.3.1 and 4.3.2 are defined in figure 12. The sequence of messages for each of these exchanges is given in subclause 6.3.3.

The contents of the elements in an AUTH_MESSAGE depend upon whether the message is originated by a PT or an FT. The elements and their contents are specified for the originator of an AUTH_MESSAGE in the column of figure 12 headed AUTH_ELEMENTS. Thus, for example, if {AUTHENTICATION-REQUEST} is originated by a PT it contains the elements <<AUTH-TYPE>> and <<RAND>>, where the content of <<RAND>> is the parameter RAND_P. If it is originated by an FT, then it contains <<AUTH-TYPE>>, <<RS>> and <<RAND>>, where the content of <<RAND>> is the parameter RAND_F.

An eight bit code for the message type is given for each of the AUTH_MESSAGES. The most significant bit is always 0. This is the bit depicted in figure 11. The coding given in figure 11 is taken from subclause 7.4.5 of ETS 300 175-5 [5]. In the event of any conflict between figure 11 and ETS 300 175-5 [5], the coding in the latter shall be the prime source.

Only those messages which are needed for the authentication exchanges defined in subclause 4.3 are listed. Moreover, the messages listed may contain additional elements which are not relevant for the security features specified in this ETS. For full details of the messages and their contents the reader is referred to ETS 300 175-5 [5].

Concerning the inclusion of <<RES>> (RES1) in the PT originated {AUTHENTICATION-REQUEST} see note.

AUTH_MESSAGE	Message Type	AUTH_ELEMENTS	
		PT	FT
{AUTHENTICATION -REQUEST}	01000000	<<RAND>> (RAND_P) <<AUTH-TYPE>> <<RES>>(RES1)	<<RS>>,<<RAND>> (RAND_F) <<AUTH-TYPE>>
{AUTHENTICATION -REPLY}	01000001	<<RES>>(RES1)	<<RS>>,<<RES>> (RES2)

NOTE: The element <<RES>> (RES1) is only used in the PT originated {AUTHENTICATION-REQUEST} when this request is sent as part of the key allocation protocol defined in subclause 6.5.6. It is therefore not relevant to the discussions in this subclause.

Figure 12: Authentication Messages (AM)

The AUTH_ELEMENTS are listed in figure 13. The elements <<RAND>>, <<RS>> and <<RES>> correspond to the parameters described in subclause 4.4.2.

The element <<AUTH-TYPE>> is used to identify which keys and algorithms are to be used. This is described in subclause 6.5.

The length of <<RAND>>, <<RS>> and <<RES>> may depend upon the A algorithm (see subclause 5.1.1) which is used for the authentication processes. figure 13 specifies the exact length of the elements when the DSAA is used.

When the DSAA is used, the following convention is adopted for the coding of the <<RAND>>, <<RS>> and <<RES>> information elements (the numbering of the octets in an information element and of the bits within an octet is as defined in ETS 300 175-5 [5]).

The most significant bit RAND[63] (respectively RES[31] and RS[63]) is represented by bit 8 of octet 3 (the first octet after the length indicator) of the information element.

The least significant bit RAND[0] (respectively RES[0] and RS[0]) is represented by bit 1 of octet 10 (respectively octet 6 and octet 10) of the information element.

An eight bit code for the Element Identifier (EI) is given for each of the AUTH_ELEMENTS. The coding given in figure 13 is taken from subclause 7.7.1 of ETS 300 175-5 [5]. In the event of any conflict between figure 13 and ETS 300 175-5 [5], the coding in the latter shall be the prime source.

AUTH_ELEMENT	Element Identifier	Length in Bytes when DSAA is used
<<RAND>>	00001100	10
<<RES>>	00001101	6
<<RS>>	00001110	10
<<AUTH-TYPE>>	00001010	5

Figure 13: Authentication Elements (AE)

6.3.3 Authentication procedures

6.3.3.1 Authentication of a PT

SEQUENCE	DESCRIPTION
1	FT obtains RS, RAND_F and XRES1
2	FT constructs {AUTHENTICATION-REQUEST} and sends it to PT
3	PT calculates RES1
4	PT constructs {AUTHENTICATION-REPLY} and sends it to FT
5	FT compares RES1 with XRES1

If the values compared in sequence 5 are not equal, then the FT drops the call.

6.3.3.2 Authentication of an FT

SEQUENCE	DESCRIPTION
1	PT generates RAND_P
2	PT constructs {AUTHENTICATION-REQUEST} and sends it to FT
3	FT obtains RS, and value RES2
4	FT constructs {AUTHENTICATION-REPLY} and sends it to PT
5	PT calculates XRES2 and compares it with RES2

If the values compared in sequence 5 are not equal, then the PT drops the call.

6.3.4 Transfer of Cipher Key, CK

NWK layer primitives required to establish the cipher key at the MAC layer are defined in subclause 6.5.3.

6.4 MAC layer procedures

6.4.1 Background

The purpose of this subclause is to specify all the MAC layer processes needed for the provision of data confidentiality over the air interface.

Subclause 6.4.2 provides a brief overview of the MAC layer field structure to a level adequate for a specification of the encryption procedures. For more detail on the MAC layer, and for an explanation of terminology and notation, the reader is referred to ETS 300 175-3 [3].

Subclause 6.4.3 contains details of which data will be encrypted when operating in encryption mode.

Subclause 6.4.4 details the encryption process itself and explains where this is placed within the MAC layer processes.

Subclause 6.4.5 is concerned with initialisation and synchronisation of the encryption process, whilst subclause 6.4.6 is concerned with the procedures for switching from clear to encrypted mode. Subclause 6.4.7 contains details of how encryption is managed during handover.

Throughout subclauses 6.4.2 to 6.4.7 the simple case of a full slot is assumed.

Subclause 6.4.8 details the differences for a half slot specification. Subclause 6.4.9 details the differences for a double slot specification. Subclause 6.4.10 details the differences for multiple connections between a single FT and PT.

6.4.2 MAC layer field structure

The MAC layer field structure for a full slot physical channel, in as far as it is relevant for the encryption process, is illustrated below in figure 14.

```
     8       40        16              320                    4
  ┌─────┬──────────┬──────┬────────────────────────────┬──────────┐
  │  H  │    T     │  RA  │           data             │    X     │
  ├─────┴──────────┴──────┼────────────────────────────┴──────────┤
  │           A           │                  B                     │
  ├───────────────────────┴────────────────────────────────────────┤
  │                            D32                                 │
  └────────────────────────────────────────────────────────────────┘
  | a0              a63| b0                                    b323|
```

Figure 14: Field structures

At the lowest level, the D32-field is the interface to the physical channel. It contains a total of 388 bits, and consists of a 64-bit A-field and a 324-bit B-field.

The A-field consists of an 8-bit header H, a 40-bit tail T and a 16-bit redundancy field RA. The content of RA is a cyclic redundancy check computed on the contents of H and T.

The tail T can be one of five types: P_T, N_T, Q_T, M_T or C_T. The type of T depends on the information it carries, and is indicated by a code in H.

The B-field consists of a 320-bit data field and a 4-bit redundancy field X. The content of X is a cyclic redundancy check computed on the contents of the data field.

Two types of B-field are defined: E-type and U-type. A B-field of E-type is used to carry error protected MAC layer control data, C_F or G_F channel data. A B-field of U-type is used to carry I_N or I_P channel data.

In a B-field of U-type which is carrying IN channel data all 320 data bits in the B-field are channel data. This is the unprotected format for the B-field. The B-field mapping is the U32a mapping described in ETS 300 175-3 [3].

In a B-field of E-type, and in a B-field of U-type which is carrying I_p channel data, 64 bits of the 320 data bits are a redundancy code added at the MAC layer. This is the protected format for the B-field. It is illustrated below in figure 15. The 16-bit field RBj is a cyclic redundancy check computed on the data content of Bj. The protected format is used with the B-field mappings U32b and E32 defined in ETS 300 175-3 [3].

The B-field format (protected or unprotected) is identified by a code in the header H.

```
  ┌───────────┬─────────────┬─────────────┬─────────────┬─────┐
  │    B0     │     B1      │     B2      │     B3      │  X  │
  ├──────┬────┼────────┬────┼────────┬────┼────────┬────┤     │
  │ data │RB0 │  data  │RB1 │  data  │RB2 │  data  │RB3 │     │
  └──────┴────┴────────┴────┴────────┴────┴────────┴────┴─────┘
    64     16     64     16     64     16     64     16    4
```

Figure 15: B-field protected format

6.4.3 Data to be encrypted

When operating in encrypt mode, the following rules apply to the encryption of data in the various fields.

A-field:

H field	-	Never encrypted;
T field	-	Encrypted precisely when it is of type C_T;
RA field	-	Never encrypted.

B-field:

X-field	-	Never encrypted.

Unprotected format:

All data bits encrypted (see figure 14).

Protected format:

All data bits encrypted (see figure 15).

> NOTE: For protected format B-fields, the redundancy checks RB0, RB1, RB2 and RB3 are not encrypted.

6.4.4 Encryption process

For each pair of D32-fields which carries encrypted data the key stream generator, defined in subclause 4.5.4, outputs two Key Stream Segments (KSS) each of 360 consecutive bits. The bits of a KSS are labelled KSS(0),...,KSS(359), where KSS(0) is the first bit output from the generator and KSS(359) is the last.

The bits in a KSS are used to encrypt the contents of the D32-field.

The encryption process is as follows:

If T is of type C_T, the bits KSS(0),...,KSS(39) are XORed with the bits a8,...,a47 from T, so that is KSS(0) is XORed with a_8, etc. The labelling of the bits in T is that specified in ETS 300 175-3 [3].

If T is not of type C_T, then bits KSS(0),...,KSS(39) are discarded.

If the B-field has the unprotected format, the bits KSS(40),...,KSS(359) are XORed with the bits $b_0,...,b_{319}$ from B, so that KSS(40) is XORed with b_0, etc.

If the B-field has the protected format, the bits KSS(40),..., KSS(103) are XORed with the bits $b_0,...,b_{63}$ from B0, so that KSS(40) is XORed with b_0, etc.

Bits KSS(104),...,KSS(119) are not used.

Bits KSS(120),...,KSS(183) are XORed with the bits $b_{80},...,b_{143}$ from B1, so that KSS(120) is XORed with b_{80}, etc.

Bits KSS(184),...,KSS(199) are not used.

Bits KSS(200),...,KSS(263) are XORed with the bits $b_{160},...,b_{223}$ from B2, so that KSS(200) is XORed with b_{160}, etc.

Bits KSS(264),...,KSS(279) are not used.

Bits $KSS(280),...,KSS(343)$ are XORed with the bits $b_{240},...,b_{303}$ from B3, so that $KSS(280)$ is XORed with b_{240}, etc.

Bits $KSS(344),...,KSS(359)$ are not used.

The decryption operation is identical to the encryption operation. For this reason, the term encryption is used where decryption would be more precise. However, it will be apparent from the context whether the process is being used to encrypt clear data prior to transmission, or to recover clear data from received encrypted data.

The placement of the encryption process within the MAC layer is chosen so that when encryption is applied the redundancy checks RA, RB0, RB1, RB2 and RB3 are always computed on the encrypted data, as depicted in figure 16.
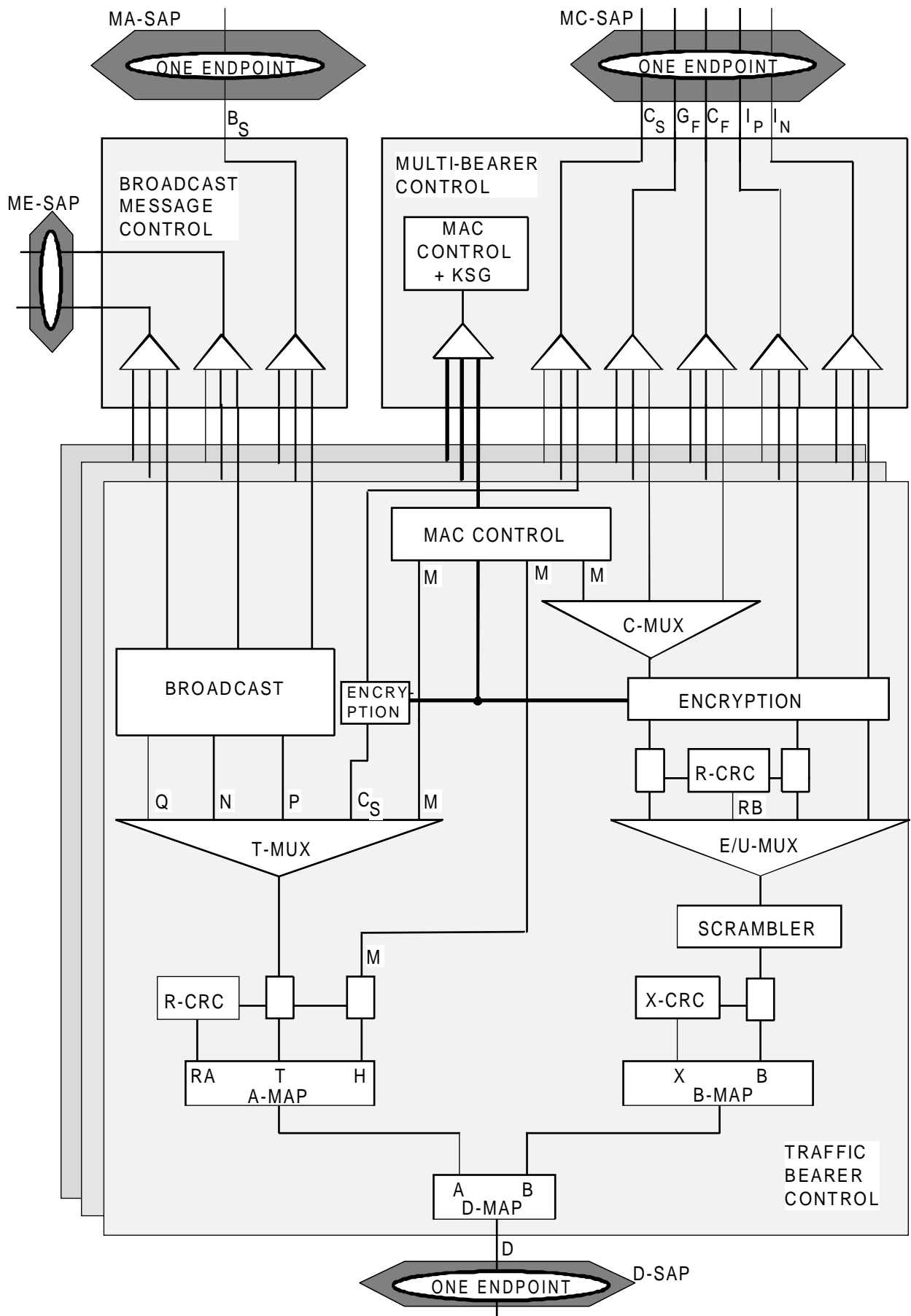
Figure 16: Placement of encryption

6.4.5 Initialisation and synchronisation of the encryption process

For a MAC connection, both the PT and the FT are initially in clear (non-encrypt) mode. Both switch to encrypt mode for a specific frame number and remain in this mode until switched back to clear mode for another specific frame number. The way in which the mode is switched is specified in subclause 6.4.6.

The respective KSGs are initialised by the PT and the FT loading them with the same CK and initialisation value IV (see annex J).

The construction of CK is described in subclause 4.3.4. The cipher key will always be represented by the binary vector CK[0],...,CK[63], where CK[0] is the least significant bit.

The IV is a 35-bit value which is represented in the form IV[0],...,IV[34] where IV[0] is the least significant bit. The IV value depends upon the bearer which is to be encrypted. For basic connections it is defined as follows:

Bits IV[0],...,IV[3] correspond to the (interpolated) frame number, with IV[0] the least significant bit of the frame number; bits IV[4],...,IV[27] correspond to the multiframe number, with IV[4] the least significant bit of the multiframe number; bits IV[28],..,IV[34] are set to O.

For advanced connections (see subclause 6.4.9), bits IV[0],...,IV[27] are defined as above; bits IV[28],...,IV[31] correspond to the Logical Bearer Number (LBN), with IV[28] the least significant bit of the LBN; bits IV[32],..,IV[34] correspond to the three least significant bits of the Extended Connection Number (ECN), with IV[32] the least significant bit of the ECN.

The KSG is re-initialised for every frame which is to be encrypted. Thus, for a basic connection, if frames with numbers M, M+1,...,L are encrypted, then the cipher is initialised with M as IV for the first frame, M+1 as IV for the second frame,..., L as IV for the last frame in this sequence.

When a frame (with frame number N) is to be encrypted, a pair of consecutive key stream segments is generated. Thus if $S_F(N)$ denotes the first KSS in the pair and $S_P(N)$ denotes the second, then the concatenated sequence $S_F(N)$, $S_P(N)$ is 720 consecutive output bits of the KSG.

In duplex mode there are two D32-fields per frame, one of which is transmitted and one of which is received. When in encrypt mode, both D32-fields are encrypted. The order in which the key stream segments in a pair are used to encrypt and decrypt data in duplex mode is as follows.

The segment $S_F(N)$ is used by the FT to encrypt the D32-field transmitted to the PT in frame number N. The segment $S_P(N)$ is used by the FT to decrypt the D32-field received from the PT in frame number N.

The segment $S_F(N)$ is used by the PT to decrypt the D32-field received from the FT in frame number N. The segment $S_P(N)$ is used by the PT to encrypt the D32-field transmitted to the FT in frame number N.

The encryption process is specified in subclause 6.4.4.

The PT and the FT rely upon synchronised frame numbering to maintain synchronisation of their key stream segments.

> NOTE: The PT always chooses the same LBN for the first bearer (LBN = 15). This means that connection handover of single bearer connections is possible with a single IV.

6.4.6 Encryption mode control

6.4.6.1 Background

Switching a MAC connection from clear to encrypt mode and from encrypt to clear mode is achieved by an exchange of MAC layer T-field messages of type M_T (see subclause 6.4.2).

The transmission of MAC layer T-field messages of type M_T is restricted. A PT may transmit such messages only in (the second half of) even numbered frames. An FT may transmit such messages only in (the first half of) odd numbered frames. Messages of type M_T are always unencrypted (see subclause 6.4.3).

At the MAC layer switching from clear to encrypt mode and from encrypt to clear mode is always initiated by the PT.

6.4.6.2 MAC layer messages

A MAC layer T-field message of type MT is defined to enable switching from clear to encrypt mode.

START.REQ: this is sent by the PT. It is transmitted in every even numbered frame until the receipt of a START.CONF or a timeout.

START.CONF: this is sent by the FT. It is transmitted in every odd numbered frame after receipt of a START.REQ and before receipt of a START.GRANT or a timeout.

START.GRANT: this is sent by the PT. It is transmitted in the even numbered frame immediately following the receipt of a START.CONF.

A MAC layer T-field message of type MT is defined to enable switching from encrypt to clear mode.

STOP.REQ: this is sent by the PT. It is transmitted in every even numbered frame until the receipt of a STOP.CONF or a timeout.

STOP.CONF: this is sent by the FT. It is transmitted in every odd numbered frame after receipt of a STOP.REQ and before receipt of a STOP.GRANT or a timeout.

STOP.GRANT: this is sent by the PT. It is transmitted in the even numbered frame immediately following the receipt of a STOP.CONF.

6.4.6.3 Procedures for switching to encrypt mode

The procedure followed by a PT to switch from clear to encrypt mode is described below, it is illustrated in figure 17 for a good link, and in figure 18 for a bad link.

PT procedure for switching from clear to encrypt mode

The PT receives a MAC_ENC_EKS-req primitive. This initiates a switching process.

The PT starts to transmit in the next even numbered frame (numbered 2S in figure 17) an MT message with the start ciphering request command START.REQ. This is repeated in five successive even numbered frames or until the PT receives a start ciphering confirm command START.CONF. The START.CONF event is reported with a MAC_ENC_EKS-cfm primitive. If the START.CONF is not observed, the connection is released and the DLC layer is informed using the MAC_DIS-ind primitive.

The PT starts decryption of messages immediately the START.CONF message is received, i.e. the D32-field containing the START.CONF message is treated as being encrypted. All B-field and CT channel data received by the PT after sending the first request and before receiving the first START.CONF is ignored.

> NOTE 1: The PT may begin decryption from frame 2S+1 but all resultant output is ignored until the START.CONF is observed.

The PT acknowledges receipt of the START.CONF by the transmission of a START.GRANT in the next even numbered frame. Encryption of transmitted data begins with this START.GRANT frame and continues in all subsequent frames.

FT procedure for switching from clear to encrypt mode

The FT receives a START.REQ from the PT in frame 2S for some S. The FT switches immediately into encrypt mode and starts transmission of START.CONF commands in frame 2S+1. This command is repeated in all successive odd numbered frames until 2S+11 or until the FT receives a START.GRANT. If a START.GRANT is not received the connection is released and the DLC layer is informed using the MAC_DIS-ind primitive.

All B-field and CT channel data received by the FT after (but not including) frame 2S+1 is ignored until the START.GRANT is observed. Frame 2S+1 is received in clear. Decryption of data begins immediately on receipt of the frame including the START.GRANT. The START.GRANT is reported with a MAC_ENC_EKS-ind primitive.

> NOTE 2: Decryption may begin at the FT from frame 2S+2 but the resultant output is ignored until the START.GRANT is observed.
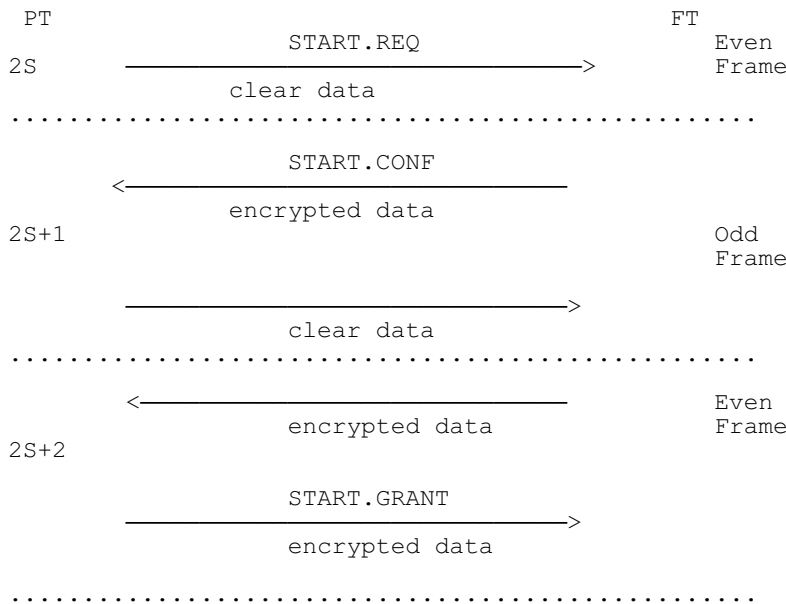
Figure 17: Encryption start - good link example

```
PT                                                  FT
                          START.REQ               Even
            ───────────────────────────────>     Frame
                         clear data      not received
..................................................
ignore  <───────────────────────────────
                        clear data
                                                  Odd
                                                  Frame
            ───────────────────────────────>  clear
                        clear data
..................................................
ignore  <───────────────────────────────         Even
                        clear data                Frame
2S
                         START.REQ
            ───────────────────────────────>  clear
                        clear data
..................................................
                         START.CONF
not received  <───────────────────
                       encrypted data
2S+1                                              Odd
                                                  Frame
            ───────────────────────────────>  clear
                        clear data
..................................................
ignore  <───────────────────────────────         Even
                       encrypted data             Frame
2S+2
                         START.REQ
            ───────────────────────────────>  ignore/assume clear
                        clear data
..................................................
                         START.CONF
decrypt <───────────────────────────────         Odd
                       encrypted data             Frame
2S+3
            ───────────────────────────────>  ignore/assume clear
                        clear data
..................................................
decrypt <───────────────────────────────         Even
                       encrypted data             Frame
2S+4
                         START.GRANT
            ───────────────────────>  not received
                       encrypted data
..................................................
                         START.CONF
decrypt <───────────────────────────────         Odd
                       encrypted data             Frame
2S+5
            ───────────────────────────────>  ignore
                       encrypted data
..................................................
decrypt <───────────────────────────────         Even
                       encrypted data             Frame
2S+6
                         START.GRANT
            ───────────────────────────────>  decrypt
                       encrypted data
..................................................

decrypt <───────────────────────────────         Odd
                       encrypted data             Frame
2S+7
            ───────────────────────────────>  decrypt
                       encrypted data
..................................................
decrypt <───────────────────────────────         Even
                       encrypted data             Frame
2S+8
            ───────────────────────────────>  decrypt
                       encrypted data
..................................................
```

Figure 18: Encryption start - poor link example

6.4.6.4 Procedures for switching to clear mode

This feature is provided to facilitate handover with interrupted ciphering (see subclause 6.4.7). Encryption should be resumed once the handover is complete. The procedures for switching to clear mode are illustrated in figure 19.

PT procedure for switching from encrypt to clear mode

The PT receives a MAC_ENC_EKS-req primitive. This initiates a switching process.

The PT starts to transmit in the next even numbered frame an M_T message with the stop ciphering request command STOP.REQ. This is repeated in five successive even numbered frames or until the PT receives a stop ciphering confirm command STOP.CONF. The STOP.CONF event is reported with a MAC_ENC_EKS-cfm primitive. If the STOP.CONF is not observed, the connection is released and the DLC layer is informed using the MAC_DIS-ind primitive.

The PT stops decryption of messages immediately the STOP.CONF message is received, i.e. the D-field containing the STOP.CONF message is treated as being clear. All B-field and C_T channel data received by the PT after sending the first request and before receiving the first STOP.CONF is ignored.

> NOTE 1: Decryption may stop from frame 2S+1 (in figure 19) but all resultant output is ignored until the STOP.CONF is observed.

The PT acknowledges receipt of the STOP.CONF by the transmission of a STOP.GRANT in the next even numbered frame. Transmission of clear data begins with this STOP.GRANT frame and continues in all subsequent frames.

FT procedure for switching from encrypt to clear mode

The FT receives a STOP.REQ from the PT in frame 2S for some S. The FT switches immediately into clear mode and starts transmission of STOP.CONF commands in frame 2S+1. This command is repeated in all successive odd numbered frames until 2S+11 or until the FT receives a STOP.GRANT. If a STOP.GRANT is not received the connection is released and the DLC layer is informed using the MAC_DIS-ind primitive.

All B-field and C_T channel data received by the FT after (but not including) frame 2S+1 is ignored until the STOP.GRANT is observed. Frame 2S+1 is received encrypted. Decryption of data stops immediately on receipt of the frame including the STOP.GRANT. The STOP.GRANT is reported with a MAC_CYP_EKS-ind primitive.

> NOTE 2: Decryption may be stopped at the FT from frame 2S+2 but the resultant output is ignored until the STOP.GRANT is observed.

```
PT                                                          FT

                              STOP.REQ                    Even
        2S        ────────────────────────────────>      Frame
                             encrypted data
        ...............................................................

                              STOP.CONF
                  <────────────────────────────────
                              clear data
        2S+1                                             Odd
                                                         Frame

                  ────────────────────────────────>
                             encrypted data
        ...............................................................

                  <────────────────────────────────     Even
                              clear data                 Frame
        2S+2
                              STOP.GRANT
                  ────────────────────────────────>
                              clear data
        ...............................................................
```
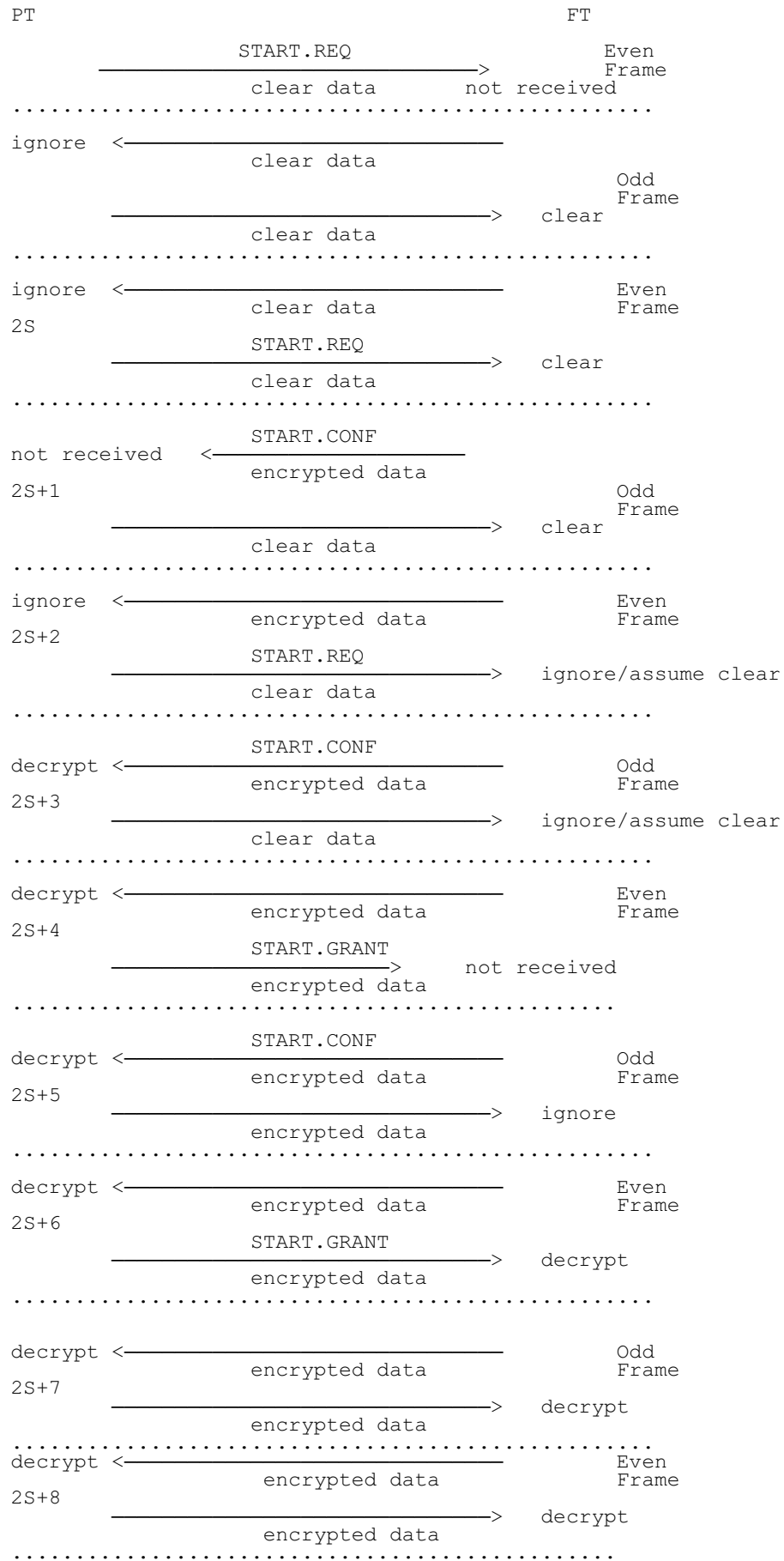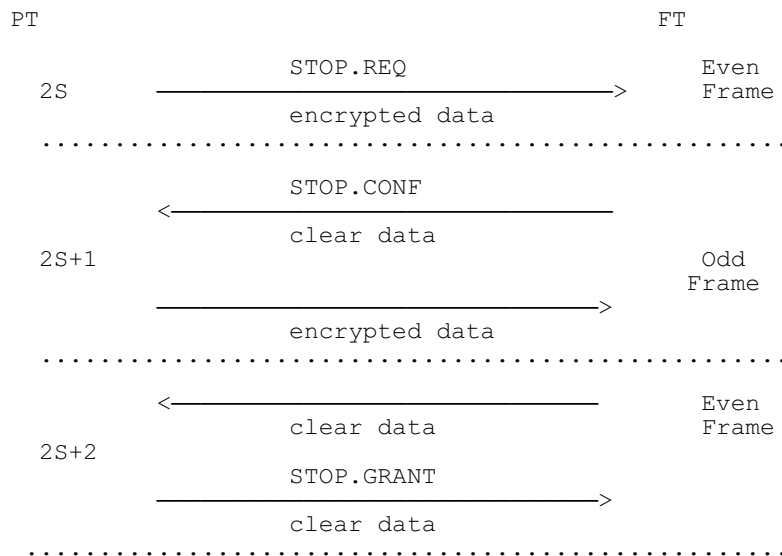
Figure 19: Encryption stop - good link example

6.4.7   Handover of the encryption process

Two types of handover are considered:

1)   internal handover:

        -       bearer handover.
        -       connection handover;

2)   external handover.

In principle, the techniques associated with connection handover can be used in place of bearer handover and the techniques associated with external handover can be used in place of connection handover or bearer handover.

External handover caters for the situation where the cipher key cannot be transferred between FTs in a secure manner. A method by which the cipher key could be securely transferred between FTs would allow this to be removed and the same procedure as used for connection handover to be applied.

6.4.7.1   Bearer handover, uninterrupted ciphering

Bearer handover is a MAC layer operation. It is transparent to the operation of the encryption processes.

The new bearer switches to the appropriate encryption mode during bearer establishment (see ETS 300 175-3 [3]). If encryption is enabled the new bearer uses the same CK and IV as the original bearer.

6.4.7.2   Connection handover, uninterrupted ciphering

Connection handover is a DLC layer operation. During handover of a PT from one connection to another connection on the same FT, the PT will maintain both connections for a certain period. During this period the same sequence of key stream segments will be used for each connection.

        NOTE: Connection 1 and connection 2 may belong to the same or different RFPs.

The notation S(N) is used to denote the pair of key stream segments ($S_F(N)$, $S_P(N)$) for frame number N.

During the first phase of handover, communication on connection 2 is in clear mode. Encryption is enabled on connection 2 by the PT using the procedures of 6.4.6. Once encryption has been established on connection 2, connection 1 may be released.

The handover is illustrated in figure 20.

```
┌──────────────────────────────────────────────────────────────────────┐
│ Frame number          Connection 1              Connection 2           │
├──────────────────────────────────────────────────────────────────────┤
│    N                  S(N)                       clear                  │
│    N+1                S(N+1)                      clear/start           │
│                                                   set-up                │
│    .                     .                           .                  │
│    .                     .                           .                  │
│    N+k-1              S(N+k-1)                     clear/start           │
│                                                   encryption            │
│    N+k                S(N+k)                      S(N+k)                 │
│    N+k+1              S(N+k+1)                    S(N+k+1)               │
│    .                     .                           .                  │
│    .                     .                           .                  │
│    N+m-1              S(N+m-1)                    S(N+m-1)               │
│    N+m                release                     S(N+m)                │
│    N+m+1                 -                         S(N+m+1)             │
└──────────────────────────────────────────────────────────────────────┘
```

Figure 20: Connection handover, uninterrupted ciphering

6.4.7.3 External handover - handover with ciphering

The new connection is set up in clear mode. It is unnecessary to cease ciphering on the old connection prior to set-up of the new connection. It may not be possible to initiate ciphering on the new leg until the old connection is either released or switched to clear mode. Switching to encrypted mode is described in subclause 6.4.6.3. Switching to clear mode is described in subclause 6.4.6.4.

If the PP is to resume encryption using the same cipher key as it used with the original FP, then a copy of the cipher key shall be made available to the new FP during handover. The transfer of the cipher key shall be made entirely within the fixed network and shall not involve transmission over the air interface. If this is possible, consideration should be given to the use of a seamless handover as described in subclause 6.4.7.2.

If such a transfer of key is not possible, then the data confidentiality service shall be re-invoked and a new cipher key established. To establish a new derived cipher key, re-authentication of the PP will be necessary.

6.4.8 Modifications for half slot specifications

6.4.8.1 Background

The purpose of this subclause is to specify how the encryption processes given in the previous subclauses are modified for the half slot formats.

6.4.8.2 MAC layer field structure

The MAC layer field structure for a half slot physical channel is depicted in figure 21.

```
   8      40     16         80+j           4
┌──────┬──────┬──────┬──────────────────┬──────┐
│  H   │  T   │  RA  │ data             │  X   │
├──────┴──────┴──────┼──────────────────┴──────┤
│  A                 │  B                       │
├────────────────────┴──────────────────────────┤
│              D08                               │
└────────────────────────────────────────────────┘
 a0               a63│ b0               b83+j│
```

Figure 21: Half slot field structures

As with the full slot structure, two types of B-field are defined:

E-type and U-type. These carry the same types of channel data as in the full slot case.

In a B-field of U-type which is carrying I_N channel data all 80+j data bits are channel data. This is the unprotected format for the B-field and corresponds to the U08a mapping in ETS 300 175-3 [3].

The protected format for the B-field is illustrated below in figure 22. The protected format is used with the B-field mappings U08b and E08, as defined in ETS 300 175-3 [3]. The B0-field carries 64 bits of control or I_p channel data and a 16-bit redundancy check R_{B0} on this data.

B0		B1	X
data	RB0		
64	16	j	4
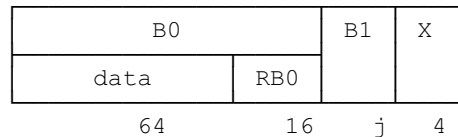
Figure 22: Half slot B-field protected format

6.4.8.3   Data to be encrypted

When operating in encrypt mode the same rules apply to the encryption of data as given in subclause 6.4.3. However, for the protected format B-field there are two portions excluded from encryption, namely the redundancy check R_{B0} and the X-field.

6.4.8.4   Encryption process

For each pair of D08-fields which carries encrypted data the KSG outputs two key stream segments each of 120+j consecutive bits. The bits of a KSS are labelled as KSS(0)...KSS(119+j).

The encryption process is as follows:

-        if T is of type C_T, the bits KSS(0),...,KSS(39) are XORed with the bits a_8,...,a_{47} from T. Thus KSS(0) is XORed with a_8 etc. If T is not of type C_T, then bits KSS(0),...,KSS(39) are discarded;

-        if the B-field has the unprotected format, the bits KSS(40),..., KSS(119+j) are XORed with the bits b_0,...,b_{79+j} from B, so that KSS(40) is XORed with b_0, etc;

-        if the B-field has the protected format, the bits KSS(40),...,KSS(103) are XORed with the bits b_0,...,b_{63} from B0, so that KSS(40) is XORed with b_0, etc;

         Bits KSS(104),...,KSS(119) are not used. If j>0, bits KSS(120),...,KSS(119+j) are XORed with the bits b_{80}, ...,b_{79+j}, so that KSS(120) is XORed with b_{80}, etc.

6.4.8.5   Initialisation and synchronisation of the encryption process

This is the same as described in subclause 6.4.5, except that the concatenated key stream segments are 240+2j bits long.

6.4.8.6   Encryption mode control

This is exactly the same as described in subclause 6.4.6.

6.4.8.7   Handover of the encryption process

This is exactly the same as described in subclause 6.4.7.

6.4.9   Modifications for double slot specifications

6.4.9.1   Background

The purpose of this subclause is to specify how the encryption processes given in the previous subclauses are modified for the double slot formats.

6.4.9.2 MAC layer field structure

The MAC layer field structure for a double slot physical channel is depicted in figure 23a.

```
    8        40       16              800                     4
 ┌──────┬──────────┬──────┬──────────────────────────────┬────────┐
 │  H   │    T     │  RA  │            data              │   X    │
 ├──────┴──────────┴──────┼──────────────────────────────┴────────┤
 │            A           │                  B                     │
 ├────────────────────────┴────────────────────────────────────────┤
 │                          D80                                     │
 └──────────────────────────────────────────────────────────────────┘
 │ a0                 a63│ b0                                  b803│
```
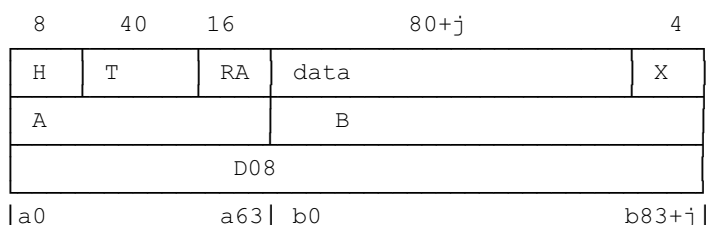
Figure 23a: Double slot field structures

As with the full slot structure, two types of B-field are defined:

E-type and U-type. These carry the same types of channel data as in the full slot case.

In a B-field of U-type which is carrying I_N channel data all 800 data bits are channel data. This is the unprotected format for the B-field and corresponds to the U80a mapping in ETS 300 175-3 [3].

The protected format for the B-field is illustrated below in figure 23b. The protected format is used with the B-field mappings U80b and E80, as defined in ETS 300 175-3 [3]. The Bj-field carries 64 bits of control or I_p channel data and a 16-bit redundancy check R_{Bj} on this data.

```
 ┌──────────┬──────────────┐      ┌──────────┬──────┬───┐
 │    B0    │      B1      │ ─ ─ ─│    B9    │      │ X │
 ├──────┬───┼──────────┬───┤ ─ ─ ─├──────────┬───┤───┤   │
 │ data │RB0│   data   │RB1│ ─ ─ ─│   data   │RB9│   │   │
 └──────┴───┴──────────┴───┘      └──────────┴───┴───┘
   64    16     64      16           64       16   4
```
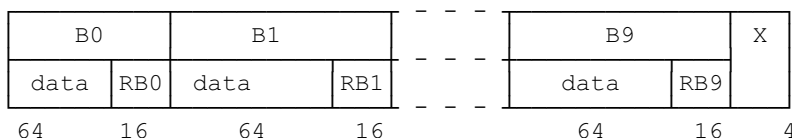
Figure 23b: Double slot B-field protected format

6.4.9.3 Data to be encrypted

When operating in encrypt mode the same rules apply to the encryption of data as given in subclause 6.4.3. However, for the protected format B-field there are two portions excluded from encryption, namely the redundancy check R_{Bj} and the X-field.

6.4.9.4 Encryption process

For each pair of D80-fields which carries encrypted data the key stream generator outputs two key stream segments each of 840 consecutive bits. The bits of a KSS are labelled as KSS(0)...KSS(839).

The encryption process is as follows:

If T is of type C_T, the bits KSS(0),...,KSS(39) are XORed with the bits a_8,...,a_{47} from T. Thus KSS(0) is XORed with a_8 etc. If T is not of type C_T, then bits KSS(0),...,KSS(39) are discarded.

If the B-field has the unprotected format, the bits KSS(40),..., KSS(839) are XORed with the bits b_0,...,b_{799} from B. That is KSS(40) is XORed with b_0, etc.

If the B-field has the protected format, the bits KSS(40),...,KSS(103) are XORed with the bits b_0,...,b_{63} from B0, so that KSS(40) is XORed with b_0, etc.

Bits KSS(104),...,KSS(119) are not used.

Bits KSS(120),...,KSS(183) are XORed with the bits b_{80},...,b_{143} from B1, so that KSS(120) is XORed with b_{80}, etc.

Bits KSS(184),...,KSS(199) are not used.

Bits KSS(200),...,KSS(263) are XORed with the bits b_{160},...,b_{223} from B2, so that KSS(200) is XORed with b_{160}, etc.

Bits KSS(264),...,KSS(279) are not used.

Bits KSS(280),...,KSS(343) are XORed with the bits b_{240},...,b_{303} from B3, so that KSS(280) is XORed with b_{240}, etc.

Bits KSS(344),...,KSS(359) are not used.

Bits KSS(360),...,KSS(423) are XORed with the bits b_{320},...,b_{383} from B4, so that KSS(360) is XORed with b_{320}, etc.

Bits KSS(424),...,KSS(439) are not used.

Bits KSS(440),...,KSS(503) are XORed with the bits b_{400},...,b_{463} from B5, so that KSS(440) is XORed with b_{400}, etc.

Bits KSS(504),...,KSS(519) are not used.

Bits KSS(520),...,KSS(583) are XORed with the bits b_{480},...,b_{543} from B6, so that KSS(520) is XORed with b_{480}, etc.

Bits KSS(584),...,KSS(599) are not used.

Bits KSS(600),...,KSS(663) are XORed with the bits b_{560},...,b_{623} from B7, so that KSS(600) is XORed with b_{560}, etc.

Bits KSS(664),...,KSS(679) are not used.

Bits KSS(680),...,KSS(743) are XORed with the bits b_{640},...,b_{703} from B8, so that KSS(680) is XORed with b_{640}, etc.

Bits KSS(744),...,KSS(759) are not used.

Bits KSS(760),...,KSS(823) are XORed with the bits b_{720},...,b_{783} from B9, so that KSS(760) is XORed with b_{720}, etc.

Bits KSS(824),...,KSS(839) are not used.

6.4.9.5   Initialisation and synchronisation of the encryption process

This is the same as described in subclause 6.4.5, except that the concatenated key stream segments are 1 680 bits long.

6.4.9.6   Encryption mode control

This is exactly the same as described in subclause 6.4.6.

6.4.9.7   Handover of the encryption process
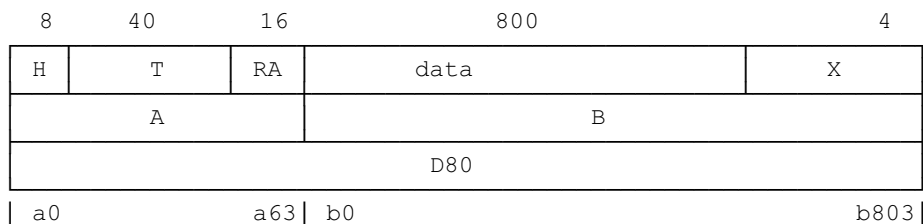
This is exactly the same as described in subclause 6.4.7.

6.4.10 Modifications for multi-bearer specifications

Between one single FT and PT several connections may exist. Each connection may consist of a single or multiple bearers.

For connections with multiple bearers, encryption of each bearer is performed. The IV for each bearer is derived from the frame number, multiframe number, LBN and ECN as described in subclause 6.4.5. Thus each bearer is encrypted with a distinct KSS.

Simplex bearers have two D-fields per frame, but unlike duplex bearers, both are transmitted by one side and received by the other side. The SF(N) is used by the transmitting side to encrypt the first D-field transmitted in a frame and the SP(N) to encrypt the second D-field transmitted. The receiving side consequently uses SF(N) to decrypt the first D-field received in the frame and SP(N) to decrypt the second D-field received.

Encryption may be enabled or disabled for each connection independently using the processes described in subclause 6.4.6 applied to one duplex bearer of the connection. Once enabled/disabled, encryption for all bearers for that connection is enabled/disabled.

> NOTE 1: The MAC_ENC_EKS-ind and the MAC_ENC_EKS-cfm primitives report the encryption mode for the whole connection.

All connections relating to the same U-plane call shall be switched to the same encryption mode.

> NOTE 2: Unless LAPC data is routed to an LC that controls an enciphered connection, C-plane data transferred from this LAPC is not encrypted even if it belongs to a call whose U-plane data is encrypted.

6.5 Security attributes

6.5.1 Background

Security attributes are used to determine which security algorithms and keys are used by a PT and an FT.

Subclause 6.5.2 describes how an FT (respectively a PT) identifies the authentication algorithm and authentication key which it wishes to use for an instance of authentication of the PT (respectively authentication of the FT) by including an <<AUTH-TYPE>> information element in the {AUTHENTICATION-REQUEST} message. The structure of the <<AUTH-TYPE>> information element is defined in ETS 300 175-5 [5]. The authentication algorithm identifier octet in <<AUTH-TYPE>> identifies the authentication algorithm (A algorithm in subclause 5.1.1) and the Auth.Key type/Auth.key nr identifies the authentication key. An {AUTHENTICATION-REJECT} message (see figure 24a) is used to signal that an <<AUTH-TYPE>> is unacceptable to the recipient.

AUTH_MESSAGE	Message Type	AUTH_ELEMENTS
{AUTHENTICATION-REJECT}	01000011	<<REJECT REASON>> (<<AUTH-TYPE>>)

Figure 24a: Authentication reject message

Subclause 6.5.3 describes how an FT (respectively a PT) identifies the cipher algorithm which it wishes to use for an instance of data confidentiality by sending a {CIPHER-REQUEST} (respectively a {CIPHER-SUGGEST}) message which includes a <<CIPHER-INFO>> information element. The structure of the <<CIPHER-INFO>> information element is defined in ETS 300 175-5 [5]. The cipher algorithm identifier octet in <<CIPHER-INFO>> identifies the cipher algorithm (KSG in subclause 4.5.4) and the cipher key type/cipher key nr identifies the cipher key. A {CIPHER-REJECT} message (see figure 24b) is used to signal that the <<CIPHER-INFO>> is unacceptable.

CIPHER_MESSAGE	Message Type	CIPHER_ELEMENTS
{CIPHER-REQUEST}	01001100	<<CIPHER-INFO>>
{CIPHER-REJECT}	01001111	<<REJECT REASON>> (<<CIPHER-INFO>>)
{CIPHER-SUGGEST}	01001110	<<CIPHER-INFO>>

Figure 24b: Cipher messages

Subclause 6.5.4 describes how a PT may send <<AUTH-TYPE>> and <<CIPHER-INFO>> elements in an {ACCESS-RIGHTS-REQUEST} message to indicate the authentication and cipher algorithms which it wishes to use in conjunction with the particular system (subscription). It is the intention that if these elements are accepted by the FT, then the identified algorithms will be associated with the system (subscription) and used whenever a call is made on that particular system (subscription). The FT signals acceptance of the elements by including them in an {ACCESS-RIGHTS-ACCEPT} message. It signals rejection of one or both of them by returning an {ACCESS-RIGHTS-REJECT} message. The access-rights messages are summarised in figure 25.

ACCESS_MESSAGE	Message Type	ELEMENTS
{ACCESS-RIGHTS-REQUEST}	01000100	<<AUTH-TYPE>> <<CIPHER-INFO>>
{ACCESS-RIGHTS-ACCEPT}	01000101	<<AUTH-TYPE>> <<CIPHER-INFO>>
{ACCESS-RIGHTS-REJECT}	01001011	<<REJECT REASON>> (<<AUTH-TYPE>>, <<CIPHER-INFO>>)

Figure 25: Access-rights messages

The key identifier octets in the <<AUTH-TYPE>> and <<CIPHER-INFO>> elements include key type and key number fields. The way in which these fields are used is described in subclause 6.5.5.

In subclause 6.5.6 an over-air key allocation protocol is defined. This protocol may be used to transform a manually distributed AC into a UAK. It is recommended that the procedure should only be used if a secure means to distribute the UAK cannot be implemented.

It should be noted that the NWK layer messages used in this subclause may contain information elements other than those listed in figures 24a, 24b and 25. For full details of the messages the reader is referred to ETS 300 175-5 [5].

6.5.2 Authentication protocols

The protocols for authentication of a PT and authentication of an FT are described in subclauses 6.5.2.1 and 6.5.2.2 respectively.

6.5.2.1 Authentication of a PT

The authentication protocol proceeds as follows:

1) the FT sends an {AUTHENTICATION-REQUEST} to the PT. The <<AUTH-TYPE>> element in this message identifies the authentication algorithm and authentication key which are requested (by the FT) to be used by the PT;

2) upon receipt of the {AUTHENTICATION-REQUEST}, the PT examines the <<AUTH-TYPE>> element to see if it is acceptable. This element is defined to be acceptable if the PT can implement the authentication algorithm and has the authentication key identified in the element (see note 1);

3) if the <<AUTH-TYPE>> element is acceptable, the PT returns an {AUTHENTICATION-REPLY} message with the <<RES>> element computed using the identified authentication algorithm and authentication key;

4) if the <<AUTH-TYPE>> element is unacceptable, the PT returns an {AUTHENTICATION-REJECT} message. The <<REJECT REASON>> element in the message is set to indicate the reason for the rejection. The possible rejection reasons are listed below in figure 26. The {AUTHENTICATION-REJECT} message may optionally include an <<AUTH-TYPE>> element constructed by the PT (see note 2). This element identifies an authentication algorithm and an authentication key which is acceptable to the PT;

5) upon receipt of an {AUTHENTICATION-REJECT}, the FT QS examines the <<REJECT REASON>> and then takes a prescribed action (see note 3). This will be based on the reason for the rejection and the reason for requesting authentication.

> NOTE 1: It is envisaged that in most applications the FT will know what authentication algorithm(s) and key(s) are supported by the PT from its knowledge of the IPUI, so that <<AUTH-TYPE>> should always be acceptable to the PT. (See also subclause 6.5.4).

> NOTE 2: The {AUTHENTICATION-REJECT} message may optionally include a sequence of <<AUTH-TYPE>> elements, whereby the position of the element in the sequence indicates the order of preference.

> NOTE 3: No attempt is made in this ETS to specify the prescribed actions, only the options are presented. A prescribed action may depend upon the <<REJECT REASON>> and the circumstances under which authentication of the PT was requested.

A list of possible actions is given in figure 26.

Code	Reject reason (from PT)	Actions open to F
11	No authentication algorithm	1, 2
12	Identified authentication algorithm not supported	1, 2, 3
13	Identified authentication key not available	1, 2, 3
14	UPI not entered (see note 4)	1, 2, 3

Figure 26: REJECT REASON, authentication of a PT

Codes for Actions open to FT

1)   Release call;

2)   Proceed with call without authentication of the PT (see note 5);

3)   Send new {AUTHENTICATION-REQUEST} with new <<AUTH-TYPE>> element.
                               (This element may have been received in the {AUTHENTICATION-REJECT}).

NOTE 4:   The <<REJECT REASON>> "UPI not entered" is returned when user authentication is
            requested, the <<AUTH-TYPE>> element is acceptable to the PT but the user fails to
            enter his UPI.

NOTE 5:   The option to proceed without authentication is not recommended.

6.5.2.2   Authentication of an FT

The authentication protocol proceeds as follows:

1)   the PT sends an {AUTHENTICATION-REQUEST} to the FT. The <<AUTH-TYPE>> element in the
       message identifies the authentication algorithm and authentication key which are requested (by the
       PT) to be used by the fixed network to derive the value RES in the {AUTHENTICATION-REPLY};

2)   upon receipt of the {AUTHENTICATION-REQUEST}, the FT examines the <<AUTH-TYPE>>
       element to see if it is acceptable. The element is defined to be acceptable if the FT can return a
       <<RES>> generated using the identified authentication algorithm and authentication key (see
       note 1);

3)   if the <<AUTH-TYPE>> element is acceptable, the FT returns an {AUTHENTICATION-REPLY}
       message with the <<RES>> element computed using the identified authentication algorithm and
       authentication key;

4)   if the <<AUTH-TYPE>> element is unacceptable, the FT returns an {AUTHENTICATION-REJECT}
       message. The <<REJECT REASON>> element in the message is set to indicate the reason for the
       rejection.   The   possible   rejection   reasons   are   listed   below   in   figure   27.
       The{AUTHENTICATION-REJECT} message may optionally include an <<AUTH-TYPE>> element
       (see note 2). This element identifies an authentication algorithm and an authentication key which is
       acceptable to the fixed network (see note 3);

5)   upon receipt of an {AUTHENTICATION-REJECT}, the PT examines the <<REJECT REASON>> and
       then takes a prescribed action (see note 4). This will be based on the reason for the rejection and
       the reason for requesting authentication. A list of possible actions is given in figure 27.

NOTE 1:   It is envisaged that in most applications the PT will know what authentication
            algorithm(s) and key(s) are acceptable to the FT from the IPUI with which it uses to
            identify itself and its knowledge of the subscription associated with the FT. Thus
            <<AUTH-TYPE>> should in most instances be acceptable to the FT. (See also
            subclause 6.5.4.)

NOTE 2:   The {AUTHENTICATION-REJECT} message may optionally include a sequence of
            <<AUTH-TYPE>> elements, whereby the position of the element in the sequence
            indicates the order of preference.

NOTE 3:   It is envisaged that, in most applications, the FT will know what authentication
            algorithm(s) and key(s) are acceptable to the PT from the IPUI.

Code	Reject reason (from FT)	Actions open to PT
11	No authentication algorithm	1, 2
12	Identified authentication algorithm not supported	1, 2, 3
13	Identified authentication key not available	1, 2, 3

Figure 27: REJECT REASON, authentication of an FT

Codes for Actions open to PT

1)   Release call;

2)   Proceed with call without authentication of the FT (see note 5);

3)   Send new {AUTHENTICATION-REQUEST} with new <<AUTH-TYPE>> element. (This element may have been received in the {AUTHENTICATION-REJECT}).

> NOTE 4:   No attempt is made in this subclause to specify the prescribed actions, only the options are presented. A prescribed action may depend upon the <<REJECT REASON>> and the circumstances under which authentication of the FT was requested.

> NOTE 5:   The option to proceed without authentication is not recommended.

6.5.3   Confidentiality protocols

The following protocol is used to start (or stop) the confidentiality service, identify the cipher algorithm (KSG) and cipher key used to provide the service, and to initiate the loading of the cipher key to the KSG and the exchange of MAC layer encryption mode control messages described in subclause 6.4.6.

The protocol is as follows:

1)   optionally the PT sends a {CIPHER-SUGGEST} message to the FT. The Y/N bit in the <<CIPHER-INFO>> element indicates whether the PT is suggesting that a ciphering session should be started (Y/N = 1), or whether it is suggesting that the ciphering session should be terminated (Y/N = 0). In the case of a suggestion to start ciphering, the <<CIPHER-INFO>> element also identifies the cipher algorithm (KSG) which is to be used (cipher algorithm identity) and the cipher key which is to be used to drive it (cipher key type and cipher key number). The FT may respond with a {CIPHER-REQUEST} (as described in 2) below) or with a {CIPHER-REJECT} (see figure 28);

2)   the FT sends a {CIPHER-REQUEST} message to the PT (see note 1). The Y/N bit in the <<CIPHER-INFO>> element indicates whether the FT is requesting the start or termination of a ciphering session (see 1) above). In the case of a request for starting ciphering. the <<CIPHER-INFO>> element in the message also identifies the cipher algorithm (KSG) which is to be used and the cipher key which is to be used to drive it. The FT establishes this cipher key for use in the selected KSG (see note 2);

3)   upon receipt of the {CIPHER-REQUEST}, the PT examines the <<CIPHER-INFO>> element to see if it is acceptable. The <<CIPHER-INFO>> element is defined to be acceptable if the Y/N bit is consistent with the current cipher mode, and if the PT can implement the cipher algorithm and has the cipher key identified in the element;

4)   if the <<CIPHER-INFO>> is acceptable (see note 3) and ciphering is requested (Y/N = 1), the PT establishes the cipher key for use in the KSG and initiates the transmission of the MAC layer START.REQ message defined in subclause 6.4.6.2 (see note 4). If the <<CIPHER-INFO>> is acceptable and switching to clear mode is requested (Y/N = 0), the PT initiates the transmission of the MAC layer STOP.REQ message defined in subclause 6.4.6.2);

5)   if the <<CIPHER-INFO>> is unacceptable, the PT returns a {CIPHER-REJECT} message. The <<REJECT REASON>> element in the message is set to indicate the reason for the rejection. The possible reject reasons are listed below in figure 28. The {CIPHER-REJECT} message may optionally include an alternative <<CIPHER-INFO>> element constructed by the PT (see note 5). This element identifies a cipher algorithm and cipher key which can be used by the PT;

6)   upon receipt of a {CIPHER-REJECT}, the FT examines the <<REJECT REASON>> element and then takes a prescribed action (see note 6) based on the reason for the rejection. A list of possible actions is given in figure 28.

NOTE 1:   It is important that the {CIPHER-REQUEST} is sent before the transfer of any C-plane data which is intended to be encrypted (e.g. dialled number).

NOTE 2:   The FT sends DL_ENC_KEY-req and DL_ENCRYPT-req primitives to the DLC layer. The first of these primitives carries the selected cipher key and the second identifies the KSG which is to be used and the relevant MAC connection. Upon receipt of these primitives, the DLC stores the information (this is necessary for connection handover) and sends MAC_ENC_KEY-req and MAC_ENC_EKS-req primitives to the MAC layer. The MAC layer loads the received key for use with the identified KSG.

NOTE 3:   It is envisaged that the FT will know the cipher algorithm and cipher key supported by the PT from the IPUI. (See also subclause 6.5.4).

NOTE 4:   To start/stop the ciphering, the PT sends DL_ENC_KEY-req and DL_ENCRYPT-req primitives to the DLC layer. The first of these primitives carries the selected cipher key and the second identifies the KSG and MAC connection that are to be used. Upon receipt of these primitives, the DLC stores the information (necessary for connection handover) and sends MAC_ENC_KEY-req and MAC_ENC_EKS-req primitives to the MAC layer. Upon receipt of these the MAC layer loads the received key for use with the identified KSG and starts transmission of the START.REQ/STOP.REQ message defined in subclause 6.4.6.2.

Once ciphering has been started/stopped at the PT, a MAC_ENC_EKS-cfm primitive is sent to the DLC which then sends a DL_ENCRYPT-cfm to the NWK.

Once ciphering has been started/stopped at the FT, a MAC_ENC_EKS-ind is sent to the DLC which then sends a DL_ENCRYPT-ind to the NWK.

NOTE 5:   The {CIPHER-REJECT} message may optionally include a sequence of <<CIPHER-INFO>> elements, whereby the position of the element in the sequence indicates the order of preference.

NOTE 6:   No attempt is made in this ETS to specify the prescribed actions, only the options are presented.

Code	Reject reason (from PT)	Actions open to FT
17	No cipher algorithm	1, 2
18	Cipher algorithm not available	1, 2, 3
19	Cipher key not available	1, 2, 3, 4
20	incompatible service	1

Code	Reject reason (from FT)	Actions open to PT
17	No cipher algorithm	1, 2
18	Identified cipher algorithm not supported	1, 2, 5
19	Identified cipher key not available	1, 2, 5
20	incompatible service	1

Figure 28: REJECT REASON, request for ciphering

Codes for Actions open to FT

1) Release call;

2) Proceed with call without encryption;

3) Send new {CIPHER-REQUEST} with new <<CIPHER-INFO>> element (This element may have been received in the {CIPHER-REJECT});

4) Authenticate the PT (and thereby establish a new DCK) and then send new {CIPHER-REQUEST};

5) Send new {CIPHER-SUGGEST} with a new <<CIPHER-INFO>> element (This element may have been received in the {CIPHER-REJECT}).

6.5.4   Access-rights protocols

A PT may establish which authentication algorithm and cipher algorithm will be used for a particular system (subscription) as part of the access-rights exchange. The protocol is as follows:

1) the PT should include an <<AUTH-TYPE>> and a <<CIPHER-INFO>> in the {ACCESS-RIGHTS-REQUEST} message. The <<AUTH-TYPE>> element, if included, identifies the authentication algorithm which the PT wishes to use in conjunction with the particular system (subscription). The <<CIPHER-INFO>> element identifies the cipher algorithm which it wishes to use (see note 1);

2) upon receipt of an {ACCESS-RIGHTS-REQUEST} message, the FT (or a management element associated with the FT) examines the <<AUTH-TYPE>> and <<CIPHER-INFO>> elements to see if they are acceptable. An element of this type is defined to be acceptable if the FT supports use of the identified algorithm;

3) if the <<AUTH-TYPE>> and <<CIPHER-INFO>> elements are acceptable, the FT returns these elements in an {ACCESS-RIGHTS-ACCEPT} message (see note 2);

4) if the <<AUTH-TYPE>> or <<CIPHER-INFO>> element is not acceptable, the FT returns an {ACCESS-RIGHTS-REJECT}. The <<REJECT REASON>> element in the message indicates the reason(s) why the element(s) is (are) not acceptable. The possible reasons are given below in figure 29. The {ACCESS-RIGHTS-REJECT} may optionally include an <<AUTH-TYPE>> or a <<CIPHER-INFO>> element which identifies an algorithm acceptable to the FT (see note 3);

NOTE 1: If the PT does not support an authentication algorithm, respectively a cipher algorithm, then an <<AUTH-TYPE>> element, respectively a <<CIPHER-INFO>> element, is not sent.

NOTE 2: If an <<AUTH-TYPE>>, respectively a <<CIPHER-INFO>>, element is accepted, then the PT and the FT (or a management centre associated with the FT) assigns the identified algorithm to the subscription details.

NOTE 3: The FT may optionally include a preferential sequence of <<AUTH-TYPE>>, respectively <<CIPHER-INFO>>, elements.

Code	Reject reason (from FT)	Actions open to PT
11	No authentication algorithm	1
17	No cipher algorithm	1
12	Authentication algorithm not supported	1, 2
18	Cipher algorithm not supported	1, 3

Figure 29: REJECT REASON, access rights request

Codes for actions open to PT

1) Abort access attempt;

2) Send new {ACCESS-RIGHTS-REQUEST} with new <<AUTH-TYPE>> element. (This element may have been received in the {ACCESS-RIGHTS-REJECT});

3) Send new {ACCESS-RIGHTS-REQUEST} with new <<CIPHER-INFO>> element. (This element may have been received in the {ACCESS-RIGHTS-REJECT}).

6.5.5 Key numbering and storage

Authentication keys may be of three types, UAK, AC or UAK/UPI. Cipher keys may be of two types, DCK and SCK.

6.5.5.1 Authentication keys

Authentication keys are associated with a particular IPUI or IPUI/PARK pair. Separate sets of UAK and AC authentication keys are held for each IPUI or IPUI/PARK pair, and number ranges are associated with each key type. A UPI may be associated with each UAK.

Thus a particular IPUI or IPUI/PARK pair will have associated to it a range of UAKs (e.g. UAK1,......,UAK4) and a range of ACs (e.g. AC1, AC2, AC3). Each range comprises a maximum number of 8 elements. With each UAK a UPI may be associated.

The key allocation procedure, described in subclause 6.5.6, may be used to transform an initial AC into a UAK.

Within a PP UAKs and ACs shall be held in non-volatile memory. ACs are manually entered to non-volatile storage. With each UAK a UPI may be associated. Such a UPI is not stored within the PP but is entered every time it is required for the user authentication service.

Within an <<AUTH-TYPE>> element the 4 bit Auth.key type field identifies the authentication key type (currently, UAK, AC or UAK/UPI) and the 4 bit field Auth.key nr. identifies the number of the key of that particular type. The most significant bit of the Auth. key nr. identifies if the key is associated with the IPUI (MSB = 0) or the IPUI/PARK pair (MSB = 1). If the authentication key is of type UAK/UPI the Auth.key nr. refers to the UAK portion of the key.

Within an allocation-type element (see subclause 6.5.6) the 4 bit AC number field identifies the AC which is to be used to generate a UAK. The most significant bit of the AC number identifies if the key is associated with the IPUI (MSB = 0) or the IPUI/PARK pair (MSB = 1). The newly generated UAK is to be assigned the number indicated in the 4 bit UAK number field.

6.5.5.2 Cipher keys

Cipher keys are associated with a particular IPUI or IPUI/PARK pair. Separate SCKs and DCKs are stored for each IPUI or IPUI/PARK pair. For each IPUI or IPUI/PARK pair at most 8 keys of each type may be stored (see note 4).

Thus a particular IPUI or IPUI/PARK pair will have associate to it a range of SCKs (e.g., SCK1,...SCK4) and a range of DCKs (e.g., DCK1). Each range comprises a maximum number of 8 elements (see note 1).

Within a PP SCKs and DCKs shall be held in non-volatile memory. SCKs are manually entered into this memory. DCKs may be automatically updated as part of the authentication of the PT procedure.

Within a <<CIPHER-INFO>> element the 4-bit cipher key type field identifies the type of a cipher key (currently, SCK or DCK) and the 4-bit cipher key nr. field identifies the number of the key of that particular type. The most significant bit of the cipher key number identifies if the key is associated with the IPUI (MSB = 0) or the IPUI/PARK pair (MSB = 1).

Within an <<AUTH-TYPE>> element the UPC bit indicates whether the DCK derived as part of the authentication of the PT should be stored or not. If UPC is set to 1 the DCK is stored under the number indicated in the 4 bit cipher key nr. field. If UPC is set to 0 the DCK is not stored (see notes 2, 3 and 4). Assignment of a DCK to a particular number causes the DCK currently held under that number to be overwritten.

> NOTE 1: Static keys are not suitable for use on large systems where many PPs may have to access many FPs. They have been introduced for systems which may not support the authentication services. An example of such a system might be a residential system which supports confidentiality but not the authentication mechanism. In this case an SCK would be unique to a particular PT/FT pair forming a residential system.

> NOTE 2: Where authentication of a PT is applied and followed by confidentiality, the normal mode of operation is to derive a new DCK and then use this key for the confidentiality service. Thus in the {AUTHENTICATION-REQUEST} the UPC bit in the <<AUTH-TYPE>> is set to 1 and the cipher key nr. field is set (for example) to 0001. In the {CIPHER-REQUEST} the cipher key type field in the <<CIPHER-INFO>> element is set to indicate DCK and the cipher key nr. field is set to 0001.

> NOTE 3: When the system provides central control of keys, a DCK may be stored for use on a later call (avoiding the need to repeat the authentication of the PT procedure). However, if there is any doubt about the current value of a stored DCK, authentication of the PT is applied and a new DCK generated.

> NOTE 4: For most applications it is envisaged that only one cipher key will be needed for each IPUI. For those systems which implement authentication of the PT this will be of type DCK, for those systems which do not it will be of type SCK.

6.5.6 Key allocation

6.5.6.1 Introduction

The purpose of this subclause is to define a method for the initial allocation of a UAK using an over-the-air protocol. The UAK is to be associated with one IPUI or IPUI/PARK pair and the mechanism requires that an AC is already associated with that IPUI or IPUI/PARK pair.

The mechanism has been introduced for those systems where distribution of a (128 bit) UAK to the user on paper or within a DAM is not considered to be feasible. However, the following points should be noted concerning the security of the protocol:

1) other methods for UAK distribution (e.g. in a DAM or on paper) are to be preferred (see the discussion in clause 7);

2) the AC from which the UAK is derived should be at least 32 bits long;

3) the way in which the AC is initially distributed does not form part of this ETS (see note 1);

4) this mechanism is not to be used for roaming key allocation (refer to note 2).

    NOTE 1:   The AC is manually entered into the PP. Possible means of distribution include on paper or verbally (perhaps over a telephone line).

    NOTE 2:   In a roaming environment visitors are provided with the session key(s) KS (and KS') as described in subclause 4.4.3.2.

6.5.6.2 UAK allocation

The protocol for over-the-air allocation of a UAK is defined below. The {KEY-ALLOCATE} message is summarised in figure 29a. All other messages used in the protocol are the authentication messages defined in subclause 6.3.2 (see note 1).

1) the FT sends a {KEY-ALLOCATE} message to the PT. The Allocation-type information element in the message identifies the relevant authentication algorithm and the number of the AC which is to be used to derive the UAK, as well as the number which is to be assigned to the derived UAK;

2) upon receipt of the {KEY-ALLOCATE} message the PT examines the Allocation-type element to see if it is acceptable. This element is defined to be acceptable if the PT can implement the authentication algorithm and has the AC identified in the element;

3) if the Allocation-type element is unacceptable, the PT returns an{AUTHENTICATION-REJECT} message (see subclause 6.5.2.1, 4), and note that the possible reject reasons are those listed in figure 26 with codes 11, 12 and 13);

4) if the allocation-type element is acceptable, the PT returns an {AUTHENTICATION-REQUEST} to the FT. The <<AUTH-TYPE>> element in this message identifies the authentication algorithm indicated in the accepted allocation-type, the Auth.key type is set to indicate AC and the Auth.key nr. is the number indicated in the accepted allocation-type. The value <<RES>> (RES1) in the message is calculated from the <<RAND>> (RAND_F) and <<RS>> elements received in the {KEY-ALLOCATE} using the authentication algorithm and key identified in the accepted allocation-type;

5) upon receipt of the {AUTHENTICATION-REQUEST}, the FT checks <<RES>> (RES1), to complete the authentication of the PT. If the check is not successful the FT must drop the call. If the check is successful the FT returns an {AUTHENTICATION-REPLY}. In this message the element <<RS>> is the same as that in the original {KEY-ALLOCATE}. The element <<RES>> (RES2) is computed with this value of RS and the RAND (RAND_P) received in the {AUTHENTICATION-REQUEST} using the authentication algorithm and key identified in the received <<AUTH-TYPE>>.

The authentication session key value KS' computed during the process of computing <<RES>> (RES2) (see subclause 5.2.2) is the derived UAK This is assigned to the UAK number identified in the original allocation-type.

Its status shall be marked as unconfirmed (see note 2);

6) upon receipt of the {AUTHENTICATION-REPLY}, the PT checks <<RES>> (RES2), in order to complete the authentication of the FT. If the check fails, the PT must drop the call. If the check succeeds the PT stores the authentication session key KS', computed during the process of computing <<RES>> (RES2) (see subclause 5.2.2), under the UAK number identified in the original allocation-type. The PT erases the used AC.

NOTE 1: It should be observed that the <<RES>> (RES1) element in the PT originated {AUTHENTICATION-REQUEST} is used only in this protocol. It is not required in this message for the authentication protocols defined in subclause 6.5.2.

NOTE 2: The FT (or a key management facility associated to the FT) retains both the AC and the UAK (with unconfirmed status) until the PT has been successfully authenticated using the UAK. Once a successful authentication of the PT has been achieved using the UAK, the AC is erased and the unconfirmed status marking can be removed from the UAK.

Key Allocation Message	Message Type	Elements
{KEY-ALLOCATE}	01000010	<<Allocation-type>> <<RAND>> (RAND_F) <<RS>>

Figure 29a: Key allocation message

7 Use of security features

7.1 Background

In this subclause the options for key management within the network are described.

In subclause 7.2.1 an overview is given of the security parameters relevant for a discussion of key management. For more details the reader is referred to subclause 4.4.3.

Subclause 7.2.2 identifies different options for the generation of keys, and subclause 7.2.3 contains a list of possibilities for the distribution of keys to a PT.

Finally, in subclause 7.2.4, an overview is given of the ways in which the keys and security parameters may be managed within the fixed systems. Particular attention is paid to the options for managing authentication keys and authentication parameters in a roaming environment.

NOTE: The following assumptions are made throughout this subclause:

1) the only authentication algorithm used is the DSAA and the only cipher (KSG) used is the DSC;

2) for any particular PT the same authentication key K is used for both authentication of the PT and authentication of the FT.

These assumptions are made in order to simplify the discussion. In practice, network management will have to explicitly include features for key and algorithm identification.

7.2 Key management options

7.2.1 Overview of security parameters relevant for key management

At the operational level in the DECT security architecture two keys are used:

- an authentication key K;

- and a cipher key CK.

These two operational keys are derived from authentication and encryption parameters which are available in the PT and in the fixed systems.

To date three authentication parameters have been identified as possible parameters from which to derive the authentication key K (see subclause 4.4.3):

1) the UAK. This is secret authentication data contained within the user (subscriber) registration data. The length of a UAK is unspecified. It is however expected to be substantially greater than the length of an AC (see note below);

2) the AC. This is a short value (e.g. 16-32 bits). An AC may be held within the PT or it may be manually entered whenever it is required;

3) the UPI. This is a short value (e.g. 16-32 bits). It is entered manually in a PT by the user each time it is required. The UPI is required for user authentication and is always used in combination with a UAK.

   NOTE: There is no conceptual difference between a UAK and an AC. It is intended that an AC should be used only for those applications which require a temporary or short term authentication key. An example of such an application might be short term registration of a PT with a hotel or restaurant.

Two possible ways to establish the cipher key CK have been identified;

1) the DCK This cipher key is automatically generated as part of the authentication of a PT process;

2) the SCK. This is a cipher key which is entered in both a PT and a fixed system and which can be used for an indefinite period.

The static cipher key is provided for those applications where data confidentiality is required, but where the authentication of a PT mechanism is not implemented. An example of such an application might be a residential DECT system. In this application the SCK would be chosen by the user and manually entered into both fixed and portable parts of his system.

The figure below (figure 30) indicates for which of the three different DECT environments, Public Access Service (PAS), Business Cordless Telecommunications (BCT) and Residential Use (RU), the parameters described above are thought to be applicable. It should be stressed that this figure is only an indication and deviations are possible.

	PAS	BCT	RU
UAK	A	A	A
AC	A**	A	A
UPI/UAK	A *	A *	N
SCK	N	A	A

A = Applicable

N = Not applicable

* = Where user authentication is required

A** = Only for initial subscription.

Figure 30: Application areas and keys

When discussing the management of operational keys, and the way in which they are used to provide the security services, it is not necessary to consider the parameters from which they are actually derived. Therefore, in the following subclauses, only K and CK will be considered.

7.2.2   Generation of authentication keys

Two options are identified for the generation of authentication keys:

- generation independent of other information; and

- generation related to other information.

1)   Generation independent of other information

In this case, the keys are first generated independently of any other information related to the user and then distributed and stored. Typically the keys would be generated in a truly random or pseudo random way.

2)   Generation related to other information

In this case, the keys are generated from other data associated with the user. Such keys can be either stored subsequent to generation, or generated in real time whenever they are required.

A typical application of the second method for generating K is DECT public access service. The authentication key is stored in the handset, but is directly calculated by the fixed system from the user subscription data when access is attempted. The key is generated from the user subscription data using a secret function. The appropriate subscription data must be transmitted by the PT when access is attempted, and the FT must have access to a system element which can generate the authentication key from this data.

The main advantage with this approach is that the network does not have to maintain a database of authentication keys. However, local derivation of keys from subscription data may pose a higher security risk than in the case where keys are generated centrally using random or pseudo-random generators.

Neither of the two methods for generating K is directly supported by an element in the security specification. Thus this ETS contains neither a specification for a pseudo-random generator, nor a discussion of the requirements for a secret algorithm (or class of algorithms) which can be used to generate authentication keys from subscription data.

7.2.3   Initial distribution and installation of keys

The following options for the initial distribution of keys for installation in portable systems have been identified:

1)   Over the air distribution and automatic installation

The keys can be distributed to the user equipment over the air and automatically installed into the equipment along with other subscription data. For example, in a public access service application, keys may be centrally generated by the public access service operator and transmitted over the air to the PT as part of subscription registration.

2)   Remote distribution and manual installation

The keys can be distributed remotely, e.g., on paper or by means of a telephone call, to the user who can enter them manually into the portable equipment.

3)   **Local distribution**

The keys can be locally generated and distributed to the portable and fixed systems. For example, keys used in a residential application might be chosen by the user, manually installed in the PP via key pad entry and then also entered manually into the FP, or sent over the air to the FP for automatic installation into the fixed equipment.

4)   **Installation at manufacture**

Keys might be installed in the equipment at manufacture.

5)   **Use of a DAM**

The keys can be stored in a hardware DAM, which can be distributed and attached to the DECT equipment. For example, in a public access service or a business cordless telecommunications application, all authentication processes may be implemented in a DAM. The system operator would personalise the DAM by entering the authentication key, user identity, etc. and then send it to the user for attachment to his portable equipment.

No comments are made on the level of security provided by these various options since this will depend on operational circumstances. In the case of the first UAK over-the-air allocation it is clear no protection against detection is given unless the process identified in subclause 6.5.6 is employed. This requires the pre-distribution of an AC via one of the options above, e.g. items 2) to 5).

When selecting a method for distribution and installation of keys, the operator should bear in mind that the effectiveness of the security features defined in this ETS depends upon the keys being kept secret (i.e. not divulged to unauthorised third parties). Thus the key should not be vulnerable to third party interception during distribution, and they should be held in portable and fixed equipment in such a way that they cannot readily be read out from the equipment by unauthorised individuals or processes. Stored key material should also be protected against unauthorised modification or deletion.

7.2.4   Use of keys within the fixed network

In order to execute the security processes, keys (e.g. K , KS, CK) or security parameters derived using these keys (e.g. RAND_F, XRES1 pairs) have to be available to the FT. In the following subclauses options for the provision of such material are outlined. Firstly the keys and parameters necessary for the authentication services are considered.

If the actual key K is available at the point in the FT where the authentication process is performed, there is only one straightforward option for the use of K.

a)   **Use of actual authentication key K**

The actual authentication key K is directly used to perform the authentication process. The key management for roaming in this option is depicted in figure 31.

The actual authentication key K might not always be available at the point in the FT where the security functions are performed. This can, for example, be the case with a visiting PT or if a service operator chooses to hold this (sensitive) information centrally. In this case session authentication keys may be used.

b)   **Use of session keys**

The authentication is performed using a fixed RS-KS-KS' triple according to the security mechanisms described in subclauses 4.3.1, 4.3.2 and 4.3.3. If K is the authentication key corresponding to the user (or user's equipment), KS and KS' is derived from K and RS using, respectively, the A11 and A21 processes as specified in subclause 4.5.2.1.

The RS-KS-KS' triple is sent to the point in the FT where authentication is performed. It can be used for an indefinite period. The calculation of RES1, CK and RES2 is done in the normal way (see subclause 4.5.2.2). If the encryption mechanism is used, the DCK is also calculated in the normal way. The key management for roaming in this option is depicted in figure 32.

Common A12 and A22 processes shall be available in the PT and at the point within the FT where the security functions are performed.

This option is supported by the security specification to facilitate the authentication of visitors by a visited network (see subclause 4.3.1). The home network of these visitors has to provide the visited network with information needed to authenticate a visitor, but it does not have to transfer the authentication key K. This option can also be used for authentication of an FT (see subclause 4.3.2) and mutual authentication (see subclause 4.3.3).

The above option can be applied within a single network for its own subscribers.

The security specification does not specify mechanisms for the distribution of RS-KS-KS' triples.

There are two other options for the use of the authentication parameters in the case where the actual authentication key is not available at the point in the FT where the security functions are performed.

c)   **Use of precalculated sets**

The FT can use precalculated sets of authentication information to perform authentication of a PT according to the security mechanism described in subclause 4.3.1. For authentication of a PT (see subclause 4.3.1) the set RS, RAND_F, XRES1 and optionally DCK is transferred to the point in the FT where authentication is performed. The key management for roaming in this option is depicted in figure 33.

This option is only secure if each set is used only once by the FT. Although useful primarily for the authentication of visitors by a visited network, the technique can also be applied within a single network for its own subscribers.

A disadvantage compared with option (5b) is that the sets have to be updated regularly by the home network. An advantage is that the PT and the visited network do not need to have a common A12 process.

The security specification does not specify mechanisms for the distribution of precalculated sets of authentication information.

d)   **Check with home database**

The authentication of a PT (see subclause 4.3.1) can also be realised by sending the RES1 obtained from the PT to a home database for checking.

The security specification does not specify mechanisms for the transfer of RES1.

There is only one practical option for the use of the cipher key CK.

e) **Use of the actual cipher key CK**

The SCK or the DCK shall be available at the point in the FT where the encryption is performed.

A DCK is automatically changed after each instance of authentication of the PT. An SCK is not changed automatically. If the SCK is used for applications needing a high level of security, then it should be changed regularly.

The security specification does not contain features to support the management of SCKs.
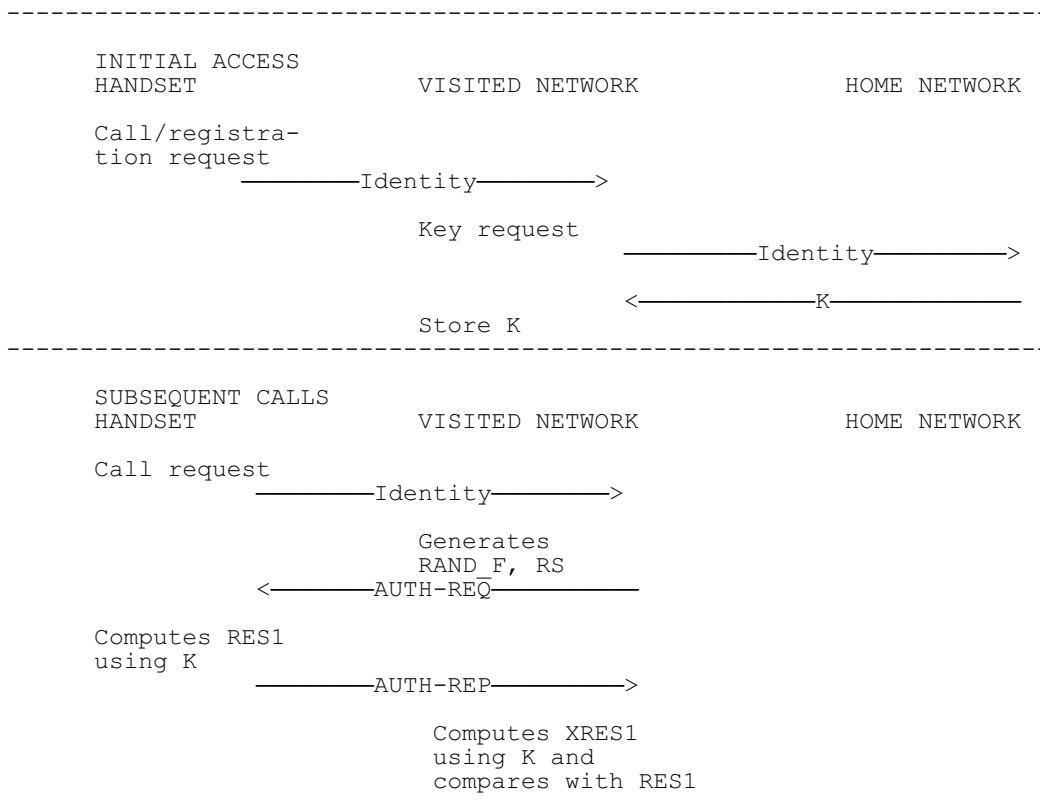
```
  -----------------------------------------------------------------------
        INITIAL ACCESS
        HANDSET               VISITED NETWORK            HOME NETWORK

        Call/registra-
        tion request
                    ─────────Identity─────────>

                              Key request
                                          ─────────Identity─────────>
                                          <───────────K───────────
                              Store K
  -----------------------------------------------------------------------
        SUBSEQUENT CALLS
        HANDSET               VISITED NETWORK            HOME NETWORK

        Call request
                    ─────────Identity─────────>
                              Generates
                              RAND_F, RS
                    <─────────AUTH-REQ─────────
        Computes RES1
        using K
                    ─────────AUTH-REP─────────>
                              Computes XRES1
                              using K and
                              compares with RES1
  -----------------------------------------------------------------------
```

NOTE 1:   Only authentication of the PT is depicted. The cases of authentication of the FT and mutual authentication are essentially the same. Generation of the DCK by the visited network is straightforward and is also not shown in the figure.

NOTE 2:   AUTH-REQ is an abbreviation for {AUTHENTICATION-REQUEST} defined in subclause 6.3.2. Similarly, AUTH-REP is an abbreviation for {AUTHENTICATION-REPLY}.

Figure 31: Roaming using authentication key K
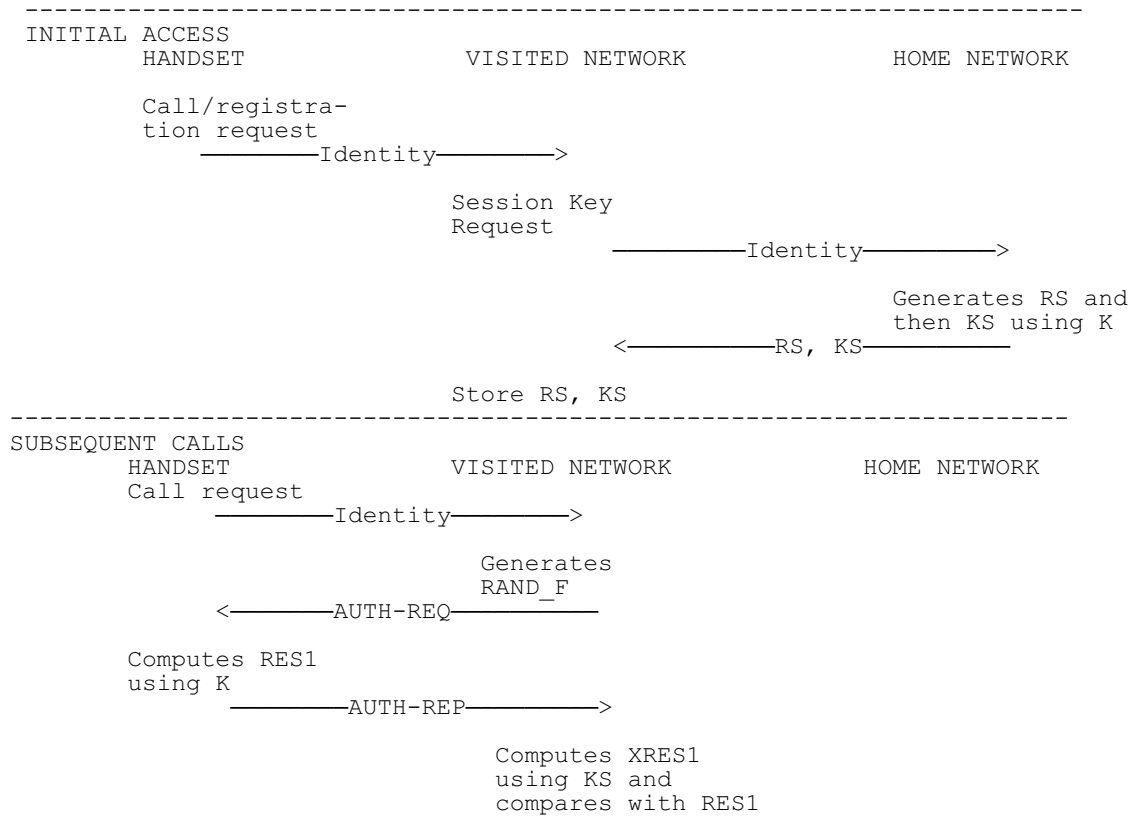
```
        ------------------------------------------------------------------
        INITIAL ACCESS
                HANDSET                VISITED NETWORK            HOME NETWORK

                Call/registra-
                tion request
                    ──────────Identity─────────>

                                    Session Key
                                    Request
                                            ──────────Identity──────────>

                                                            Generates RS and
                                                            then KS using K
                                            <─────────────RS, KS──────────

                                    Store RS, KS
        ------------------------------------------------------------------
        SUBSEQUENT CALLS
                HANDSET                VISITED NETWORK            HOME NETWORK
                Call request
                    ────────────Identity─────────>

                                    Generates
                                    RAND_F
                    <─────────AUTH-REQ─────────────

                Computes RES1
                using K
                    ──────────AUTH-REP──────────>

                                    Computes XRES1
                                    using KS and
                                    compares with RES1
        ------------------------------------------------------------------
```

NOTE 1:   Only authentication of the PT is depicted. The cases of authentication of the FT and mutual
           authentication are essentially the same. Generation of the DCK by the visited network is
           straightforward and is also not shown in the figure.

NOTE 2:   AUTH-REQ is an abbreviation for {AUTHENTICATION-REQUEST} defined in subclause 6.3.2.
           Similarly, AUTH-REP is an abbreviation for {AUTHENTICATION-REPLY}.

Figure 32: Roaming using session keys

```
-----------------------------------------------------------------------
HANDSET                    VISITED NETWORK            HOME NETWORK
Call request
      ──────────Identity──────────>

                            Authentication Data Request
                                    ──────────Identity──────────>

                                                    Generates RS and
                                                    RAND_F and compu-
                                                    tes X̄RES1 using K
                                    <────── RAND_F, RS, XRES1 ────

                            Store RAND_F,
                                    RS, XRES1
              <──────────AUTH-REQ──────────

Computes RES1
using K
      ──────────AUTH-REP──────────>

                            Compares XRES1
                            with RES1
-----------------------------------------------------------------------
```
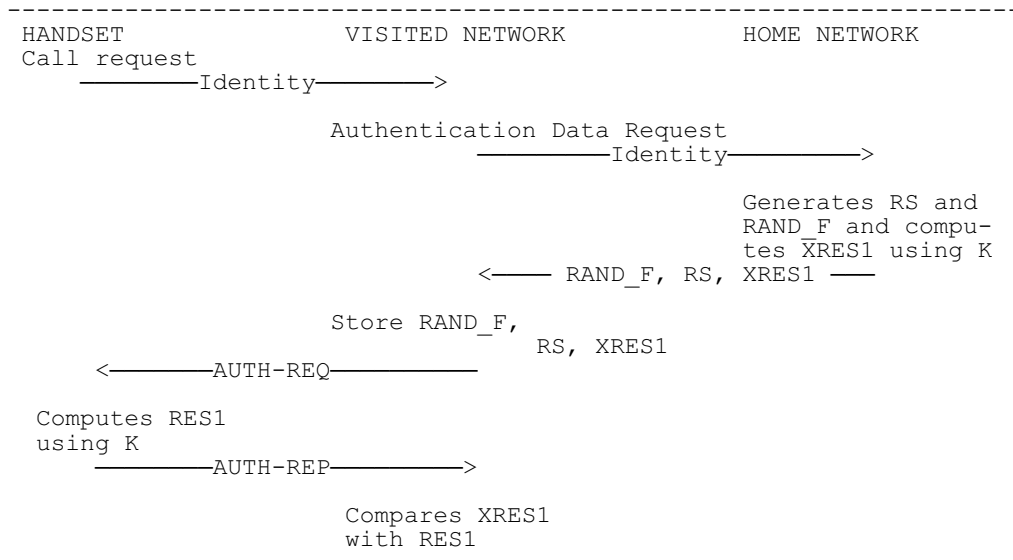
NOTE 1:   Only authentication of the PT is depicted. The cases of authentication of the FT and mutual
            authentication are essentially the same.

NOTE 2:   If encryption is offered the home network also computes the DCK and transmits this to the
            visited network.

NOTE 3:   The home network can generate and transmit more than one set of authentication data at a
            time. In this case the visited network uses the sets one at a time, each set being used exactly
            once.

Figure 33: Roaming using precalculated sets

7.3   Confidentiality service with a Cordless Radio Fixed Part (CRFP)

7.3.1   General

The FT shall initiate the procedure for cipher key transfer of subclause 7.3.2 when the CRFP requires a
cipher key:

-       when FT sends a {CIPHER-REQUEST} to a PT that is relayed via a CRFP,

-       during bearer or connection handover when the FT receives a relayed handover request.

These procedures with the PT shall be temporarily stopped until the cipher key is transferred to the CRFP
or until timeout of the connection.

7.3.2   CRFP initialization of PT cipher key

The FT shall have an active connection with the specific CRFP. If needed the FT shall suspend the
connection with the PT and resume the connection with the CRFP as defined in subclause 10.9.3 of
ETS 300 175-3 [3].

The FT shall initiate ciphering between the FT and CRFP that requires a PT CK using
{CIPHER-REQUEST} message and the CRFP CK.

When the connection between the FT and the destination CRFP is completely ciphered, the FT shall send
the PT CK to the CRFP using the {MM-INFO-SUGGEST} message.

Annex A (informative): Security threats analysis

A.1 Introduction

The purpose of this annex is to provide an overview of possible security threats for the DECT system. For each of these threats it should be considered very carefully whether it is a realistic threat for the DECT system and if a protection mechanism against the threat has to be provided.

For this consideration, it is important to realise that there are two aspects to a threat:

- its likelihood of occurrence; and

- the possible impact on the system.

To provide a guideline, an average assessment has been produced, based on the individual scores of the threats, bearing in mind the likelihood and the security impact on the system. Marketing and commercial aspects (e.g. how should the security of DECT be compared to the security of GSM, or what do potential customers expect) were not considered.

To rate the threats, the following three levels were used:

Level 1:   weak or non-existing threat, not necessary to provide a countermeasure;

Level 2:   medium threat, for which it is desirable to provide a countermeasure;

Level 3:   strong threat, for which it is necessary to provide a countermeasure.

In this ETS, the average score from the DECT security group is given. These scores have associated comments from the DECT security group, which are given in *italic*s where appropriate.

The threats are divided into five categories:

Threat A: impersonating a subscriber identity;

Threat B: illegal use of a handset (PP);

Threat C: illegal use of a base station (FP);

Threat D: impersonation of a base station (FP);

Threat E: illegally obtaining user and user related signalling information.

In general, these threats are considered under the assumption that no special security measures are taken. However, for threat E an attempt has been made to distinguish between the desired level of protection if a protection mechanism (encryption) is implemented.

DECT will be used in three different environments:

- Public Access Service (PAS);

- Business Cordless Telecommunication (BCT);

- Residential Use (RU).

In this ETS a score is, in general, given for each of the threats in each separate environment, although in the end a suitable common denominator may have to be found.

A.2   Threat A - impersonating a subscriber identity

In this case, the identity of another (existing or non-existing) DECT subscriber is used to make a call. In most cases this is done to avoid call charges but it can also be done for reasons of anonymity and untraceability.

	ENVIRONMENT			
THREAT	PAS	BCT	RU	
Impersonate subscriber identity to avoid call charging	3	3	3	(*)
Impersonate subscriber identity for anonymity or untraceability	2	1	1	(**)

(*)   If a successful attack is launched it will highly discredit the DECT system.

(**)   The likelihood of this threat seems low and the threat only exists in public access service, but the impact could be very high (criminals using the system).

There is also a threat that incoming calls are, accidentally or by active misuse, routed to a wrong subscriber. The accidental case should be handled by the normal data communication protocol. The misuse is only a threat if the user data is sensitive and there probably is an automatic protection if encryption of user data is provided (see subclause A.1.6).

COMMENT BY the DECT security group:

Impersonation of subscriber identity for anonymity should be weighted as "2" in BCT and RU environments.

A.3   Threat B - illegal use of a handset (PP)

It might be possible to use PPs which are not allowed to be used in a DECT network, although the costs are billed in the right way. This could be the case with type approved PPs (e.g. stolen) or non-type approved PPs.

	ENVIRONMENT			
THREAT	PAS	BCT	RU	
Illicit use of type approved PP	2	2	2	(*)
Use of non type approved PP	3	2	3	(**)

(*)   Considered to be a medium threat to the reputation of the system if there is no countermeasure, but a weak threat if simple countermeasures, like PIN on the PP or an electronic serial number together with a black list are implemented.

(**)   Strong to medium threat, which goes down if a type approval procedure is applied to DECT.

COMMENT BY the DECT security group:

The use of non-type approved equipment is not considered to be a serious threat, and no marketing need for this protection is considered necessary. It may even be unmanageable. A type approval process will be applied to DECT equipment, and this seems to be covered in the point (**) in threat B.

A.4 Threat C - illegal use of a base station (FP)

In this case a call is made using a certain FP without authorisation by the operator of the FP. This could, for example, be done to overload a FP (denial of service), to avoid call charges (if billing is to the FP) or just to avoid the costs for a FP.

No rating is given because the level of the threat C attack depends on how DECT is operated. If countermeasures against threat A are implemented, then automatic protection against the threat C attack is provided.

A.5 Threat D - impersonation of a base station (FP)

It might be possible to impersonate a FP or part of a FP functionality to attract calls, which are originally meant for another FP, in order to eavesdrop the user data or perhaps even to handle these calls ("steal phone customers").

A second threat is that a FP is impersonated in order to change data (ZAP code, subscription registration data) in PPs. Obviously this is only possible if the system allows FPs to make such changes.

	ENVIRONMENT		
THREAT	PAS	BCT	RU
Impersonate a base station (FP)	3	3	3

(*)
(**)

(*)   Considered to be a strong threat if a FP can change data in handsets.

(**)   Considered to be a strong threat in BCT if an impersonating FP could bypass an existing encryption mechanism without detection.

Protection against this attack can be provided either directly, i.e. by a special authentication protocol, or indirectly by mandatory encryption of data which cannot be switched off by the FP.

COMMENT BY the DECT security group:

Unauthorised use of the ZAP function is of low priority. Impersonation of a FP to bypass message privacy, particularly in BCT, is considered a very serious threat.

A.6 Threat E - illegally obtaining user data and user related signalling information

In cordless systems there is always a threat to the privacy of the user information transmitted over the air interface. The nature of the threat depends on the protection afforded. As a guideline, five threats are considered. For the first threat it is assumed that no countermeasure is implemented.

1)   Passive eavesdropping.

For the following threats it is assumed that a countermeasure (i.e. encryption) is implemented and that this countermeasure is attacked.

2)   Active attack by someone having limited knowledge of the system (e.g. knowledge of signalling procedures but not of the encryption algorithm used).

3)   Active attack with all knowledge but limited resources (e.g. PCs, limited time on mainframes).

4)   Active attack with all knowledge and "unlimited" resources (e.g. mainframes, special hardware).

5)   Academic attacks showing theoretical weaknesses, without being able to practically use them, but thereby discrediting the system.

```
THREAT: Obtaining user information
                                    ENVIRONMENT
```

LEVEL	PAS	BCT	RU	
Passive eavesdropping.	1	3	2	(*)
Active attack, limited knowledge	1	1	1	(**)
Active attack, all knowledge, limited resource.	1	2	2	
Active attack, all knowledge, "unlimited resources".	1	2	2	
Academic attack.	2	2	2	(***)

(*)   Considered to be a small threat in public access service because no privacy exists in this environment and, since the DECT cells are small, there are easier ways than monitoring the radio channel to eavesdrop a call.

(**)   Threat is small if a reasonable encryption mechanism is implemented.

(***)   An academic attack will undermine confidence in the system.

COMMENT BY the DECT security group:

Privacy in residential applications is a desirable marketing option. There will be a market need for privacy in a public access service environment.

Apart from the calling-number and called-number, also other signalling elements (e.g. request for re-routeing calls) might need protection.

```
THREAT: Obtaining signalling information
                                    ENVIRONMENT
```

LEVEL	PAS	BCT	RU	
Obtaining calling-number	1	1	1	(*)
Obtaining called-number	1	2	1	
Obtaining other signalling information	1	2	1	

(*)   Not considered to be a threat because of the small cells in DECT.

A.7 Conclusions and comments

This annex gives an overview of possible security threats for the DECT system. It provides a rating which may be used as a guide to determine how realistic the threat actually is for the DECT system, bearing in mind the different environments in which DECT will be used.

The purpose of the annex is to help establish a consensus view within the DECT security group on the security threats for DECT and the countermeasures to be taken.

During the preparation of the ETS the individual members of the DECT security group rated the threats. The average scoring, representing a consensus view is reflected in this annex. The rating is based on the likelihood and the security impact of the threat. Marketing aspects were not taken into account.

Based on the threats assessment, the following conclusions were drawn

1)   it is necessary that there is a means for the network operator to authenticate the subscriber identity;

2)   it is desirable that there is a mechanism to check that a PP is used in a way approved by the service operator, i.e. that a PP is not stolen and is type approved;

3)   it is necessary that a PP can authenticate a FP if the FP can be used to change information in the PP (e.g. ZAP code or other subscription registration information). However, a security mechanism for this authentication may be difficult to implement and manage, and it is advised not to allow a FP to make any changes in a PP;

4)   encryption of user data is essential in business applications and desirable for residential use. It is however not necessary in the public access service application. Also, in the business application, it is desirable to encrypt signalling elements.

Based on these conclusions, design was started of a security system for:

-       the authentication of subscriber identities and if possible a method to extend this to the authentication of FPs;

-       providing means for the privacy of user data in the business environment with, if feasible within the time scales, the possibility to extend this to residential use.

COMMENT BY the DECT security group:

The DECT security group considers the top priority for the work to be the design of a security system for the authentication of subscriber identities.

The provision of privacy for traffic (speech and data) in a business environment is considered essential.

The provision of privacy in residential applications will be a desirable marketing option.

While the assessment of the threat to privacy in a public access service environment is accepted, it is considered that there will be a market need for this feature. If the "cost" implications of its use are not too serious, application of privacy in a public access service environment can be at the operator's discretion. The priority is considered as low, but this application should not be ignored.

The PT authentication of an FT to prevent unauthorised use of the ZAP function is considered to be of low priority.

Threat D (impersonation of a FP) is considered as a very serious threat, particularly in a business environment. Impersonation of a FP would allow an attack for eavesdropping, because a pirate FP could bypass any message privacy.

Annex B (informative): Security features and operating environments

B.1 Introduction

DECT will be operational in three different environments: Public Access Service (PAS), Business Cordless Telecommunications (BCT) and Residential Use (RU). It should be possible to use the same handset in all three environments.

The purpose of this annex to investigate how a PP can roam from one environment to another or to another network in the same environment. This roaming only refers to enrolment (as defined below in subclause B.2), since in-call roaming between different environments and different networks will not be possible.

In this ETS an overview is given of possible options for enrolment and their possible applications. It should be noted that certain schemes, such as unprotected over-air distribution of secret authentication information, are not considered in this ETS.

Definitions for "subscription registration", "enrolment" and "roaming" will be given first. These will be used in the rest of the document.

The definitions of subscription registration and roaming are taken from ETS 300 175-1 [1].

B.2 Definitions

Subscription registration: The infrequent process whereby a subscriber obtains access rights to one or more FPs.

> NOTE 1: Subscription registration is usually required before a user can make or receive calls.

At subscription registration all the secret parameters required for the security services are made known to the PP and to the operator of a system.

Enrolment (temporary registration): A subset of registration, where a PP registers with an operator of a system (FP) (whether PAS, BCT, or RU), and as a result of the enrolment both entities are made aware of some secret parameters for subsequent service provision. (The secret parameters in this case do not necessarily reveal the original secret parameters established as part of subscription registration).

Roaming: The movement of a PP from one FP coverage area to another FP coverage area, where the capabilities of the FPs enable the PP to make or receive calls in both areas.

> NOTE 2: Roaming requires the relevant FPs and PP to be interoperable.

A PP may roam between networks for which service provision has been established by an enrolment or a subscription registration.

B.3   Enrolment options

There are 9 different cross application possibilities between environments and network, which are shown in the matrix below.

to from	PAS	BCT	RU
PAS	A,B,C	D,E,F	D,E,F
BCT	A	D,E	D,E
RU	A	D,E	D,E

In this matrix the possible options for cross environment enrolment are denoted by means of codes in the corresponding entries. The scenarios corresponding to the codes are described below.

A)   Normal subscription registration as a new subscriber for the PAS service. New key issued by visited PAS operator (on paper or in a detachable module). Insertion of this key in handset. This registration has no connection with registration in other networks or in other environments. If this kind of registration is done for several networks with a different (or possibly the same) key for each, it is multiple registration.

B)   On line or off line, temporary subscription registration as a new subscriber for the PAS service. New (local) key or authentication information issued by home PAS operator on line to visited PAS operator. No insertion of new keys in handset.

C)   Visited PAS operator confirms on line with the home operator that the PP is allowed to make a call. No authentication or authentication information to home PAS operator for checking (this is not possible if the use of authentication is mandatory for all FPs, but the possibility should be offered according to "DECT User and Portable-Part Identification Security" (see Bibliography in annex K).

D)   Authentication key entered off line (manually or by insertion of a detachable module) into PP and, if necessary, into FP/PBX.

E)   Short authentication code entered off line, for temporary use (hotel, restaurant, friends at home). Costs charged to FP.

F)   Authentication information (e.g. local key) transferred only once by the PAS operator to a PBX or residential FP. This assumes an electronic communications contact between the different DECT environments.

Annex C (informative): Reasons for not adopting public key techniques

The purpose of this annex is to outline the reasons why EG-1 decided to use authentication procedures based on a symmetric algorithm, rather than on an asymmetric algorithm.

The expert group believe that asymmetric algorithms could have several advantages over symmetric algorithms within the DECT environment. However, they came to the conclusion that, for the reasons given below, the authentication procedure should be based on symmetric algorithms.

The two main constraints which were taken into account when considering the options were the following:

- the entire authentication process has to be completed within 1 second;

- no specific hardware, such as purpose-built encryption chips or digital signal processors, can be used for reasons of cost, size, power consumption and other more general ones.

The expert group also distinguished between public key (signature) and zero-knowledge schemes. The latter require, by definition, a separate mechanism for establishing a cipher key which is to be used in certain environments. For this reason, and the fact that most zero-knowledge schemes are covered by patents or patents pending, they do not seem to be suitable for the DECT application.

The three public key mechanisms considered were:

- elliptic curves;

- discrete logarithm; and

- RSA.

Elliptic curves have been around for a good hundred years. Their usefulness for cryptographic purposes was only established a few years ago. The expert group was of the opinion that this method has not yet been sufficiently scrutinised by the cryptographic community to recommend the use of elliptic curves for authentication purposes in DECT.

Mechanisms based on the use of discrete logarithms are considered to be secure as long as the size of the field is large enough. As this implies that even larger bit strings need to be handled than with RSA, the expert group ruled out their use.

Although the use of RSA was considered to be an attractive option, it was eventually ruled out on the basis of complexity and speed requirements. The main problem is that an algorithm is required which will execute in 100 - 200 ms and, when implemented in the PP, occupy only about 1 Kbytes of ROM. In comparison, figures quoted for RSA implementations were of the order of 1-2 seconds on an 80286 processor with 20-30 Kbytes of code. It should, however, be pointed out that smart cards incorporating special RSA logic have been announced for the coming year. These were considered to be an attractive option for the detachable module implementations (DAM implementations). Unfortunately their availability and performance are not yet known.

Annex D (informative):   Overview of security features

D.1   Introduction

In the past it has been found necessary to secure mobile telecommunications. For DECT a security threat analysis has been made. The threats considered were:

-        impersonation of a subscriber identity;

-        illegal use of a handset (PP);

-        illegal use of a base station (FP);

-        impersonation of a base station (FP);

-        illegal acquisition of user and user related signalling information.

It was decided that it was necessary to provide some countermeasures against these threats. Annex A contains the full results of this security threat analysis.

To provide an adequate level of protection against the security threats a number of security services have been specified within DECT. These are:

-        authentication of a PT;

-        authentication of an FT;

-        mutual authentication of a PT and an FT;

-        data confidentiality;

-        user authentication.

This annex contains a summary of these security services. Also it describes the key management scenarios in case of roaming.

D.2   Authentication of a PT

Some threats, such as using the identity of another DECT subscriber to avoid call charges or for reasons of anonymity, and using stolen or non-type approved handsets, may be prevented by authentication of a PT.

Authentication of a PT enables an FT to authenticate a PT making or receiving a call through it. The service is initiated by the FT and is invoked at the beginning of a call (and may be re-invoked at any time during a call provided the call is not encrypted).

Authentication of a PT is realised by means of an authentication key which is known both to the PT and the FT. The FT and the PT compute a session authentication key from the authentication key. The FT sends a random value to the PT. Both parties encrypt this value with the session authentication key. The PT returns the encryption result to the FT, showing in this way its knowledge of the authentication key to the FT. The FT compares this value with its own encryption result. If the two values are identical the FT accepts the authenticity of the PT, if not the call is cleared.

The authentication key itself will never be sent over the air. One advantage of using a session authentication key is that the authentication key itself does not need to be revealed to visited networks.

D.3 Authentication of an FT

Authentication of an FT is provided to prevent impersonation of a base station. This service is necessary if the base station can be used to change information in the handset (e.g. by ZAP code). Moreover, when a base station is impersonated, user data may more easily be revealed and calls may be handled in an irregular way.

Authentication of an FT is a PT initiated service which enables a PT to authenticate an FT through which it is making or receiving a call. The service is invoked at the beginning of a call (and may be re-invoked at any time during a call).

Again this service is realised using the authentication key known by the PT and the FT. The mechanism is the reverse of the authentication of a PT. The PT sends a random value to the FT and the FT returns the encrypted value, showing in this way its knowledge of the session authentication key to the PT. The PT accepts the authenticity of the FT if this value is identical to its own encryption result.

D.4 Mutual authentication of a PT and an FT

Mutual authentication enables a PT and an FT, through which a call is connected, to authenticate each other. It may be provided using one of the three mechanisms described in the following subclauses.

D.4.1 Direct method

Direct mutual authentication is provided by combining the two mechanisms described in clauses D.2 and D.3. An FT authenticates a PT making or receiving a call through it, and afterwards the PT authenticates the FT.

D.4.2 Indirect method 1

Mutual authentication is also provided by combining authentication of a PT with data confidentiality. The PT is authenticated using the mechanism described in clause D.2, but the FT is not authenticated directly. From the session authentication key the PT and the FT compute a cipher key (as described in subclause D.5.1). This key is used for data confidentiality. If the FT system does not know the authentication key, then it is unable to derive the cipher key, so it is unable to encrypt or decrypt data sent to and from the PT. The FT shows its authenticity by encrypting all its data sent to the PT. The mechanism requires the PT to enforce the data confidentiality service and drop any unencrypted call.

D.4.3 Indirect method 2

Mutual authentication is provided by using the data confidentiality service with the static cipher key mechanism described in subclause D.5.2. In certain applications, a PT (or a set of PTs) and an FT may share such a fixed cipher key. This key is used for confidentiality of user and control data. The FT and the PT must both know the cipher key if the data is to be successfully encrypted and decrypted. The FT and the PT show their authenticity by encrypting all their data sent to each other. This mechanism requires both the PT and the FT to enforce data confidentiality, and for both parties to drop any unencrypted call.

D.5 Data confidentiality

Data confidentiality provides for the confidentiality of the user data and certain control data over the air interface between a PT and an FT.

It does not provide any cryptographic protection for data passed through the target network.

Both the PT and the FT establish a common cipher key. With this key, a key stream for encrypting data is generated. Two mechanisms for establishing the cipher key are provided.

D.5.1 Cipher key derivation as part of authentication

If the PT is authenticated at the beginning of the call, using the mechanism defined in clause D.2, a new cipher key will be established for each call, and used for the duration of that call. Both the PT and the FT compute the cipher key from the session authentication key.

D.5.2 Static cipher key

In certain applications, a PT (or a set of PTs) and an FT may share a static (fixed) cipher key. This key is used for confidentiality of user and control data. Within the DECT security features no service is included to provide for management of static keys.

D.6 User authentication

The user authentication service allows an FT to authenticate a user in a manner similar to the on-line PIN verification provided by banking systems. The service is initiated by the FT. It is invoked at the beginning of a call, and may be re-invoked at any time during a call provided the call is not encrypted.

This service requires a UPI value to be held by the user. This value is entered manually into the PT by the user. It is combined with the secret user authentication key (which is already in the PT), to determine the authentication key. This key is used in the authentication processes described in clauses D.2, D.3, and D.4.

D.7 Key management in case of roaming

D.7.1 Introduction

Management of keys is an essential aspect of the DECT security. For the effective operation of the security system, sound key management is essential.

In principle, key management is a network, user and operator topic and therefore cannot be standardised within the specification of security features for DECT. However, the specification includes elements which support the key management of the authentication key K in case of roaming.

For roaming it is not necessary that the actual authentication key K is stored at that point in the FT where the security functions are performed. The three options for when the FT has to use the authentication key K are described below as follows:

1)   use of actual authentication key K;

2)   use of session keys;

3)   use of pre-calculated sets.

D.7.2 Use of actual authentication key K

The actual authentication key K is, in this case, directly used to perform the authentication process. It is sent by the home network to the visited network on request. This key, K, is used by the visited network in subsequent calls to authenticate the handset. The key management for roaming in this option is depicted in figure D.1.
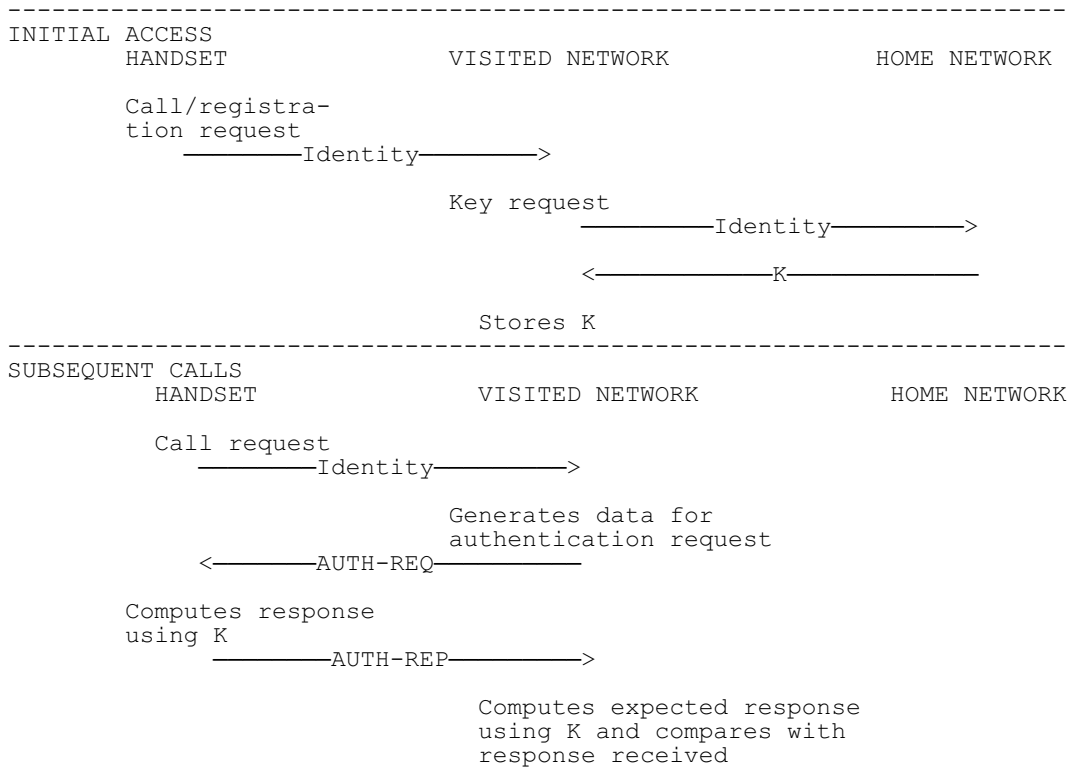
```
--------------------------------------------------------------------------
INITIAL ACCESS
        HANDSET                   VISITED NETWORK              HOME NETWORK

        Call/registra-
        tion request
            ─────────Identity────────>

                                  Key request
                                       ──────────Identity─────────>

                                       <────────────────K───────────────

                                  Stores K
--------------------------------------------------------------------------
SUBSEQUENT CALLS
        HANDSET                   VISITED NETWORK              HOME NETWORK

        Call request
            ─────────Identity────────>

                                  Generates data for
                                  authentication request
                   <────────AUTH-REQ───────────

        Computes response
        using K
            ─────────AUTH-REP─────────>

                                  Computes expected response
                                  using K and compares with
                                  response received
--------------------------------------------------------------------------
```

NOTE: Only PT authentication is depicted. The case of FT and mutual authentication is essentially the same.

Figure D.1: Roaming using authentication key K

D.7.3   Use of session keys

It might be desirable not to send the authentication key K to the visited network. The DECT security architecture specifically supports the use of a session key, instead of the key K, in this case. This session key can be used for an indefinite period by the visited network to authenticate the handset, without the authentication key K being revealed to the visited network. The key management for roaming is depicted in figure D.2.

```
        -------------------------------------------------------------------
        INITIAL ACCESS
                HANDSET                  VISITED NETWORK              HOME NETWORK

                Call/registra-
                tion request
                    -------Identity-------->

                                         Session Key
                                         Request
                                             --------Identity--------->

                                                                    Generates
                                                                    Session Key
                                             <-------Session Key--------

                                         Stores
                                         Session Key
        -------------------------------------------------------------------
        SUBSEQUENT CALLS
                HANDSET                  VISITED NETWORK              HOME NETWORK

                Call request
                    ————————Identity————————>

                                         Generates data for
                                         authentication request
                    <————————AUTH-REQ————————

                Computes response
                using K
                    ————————AUTH-REP————————>

                                         Computes expected response
                                         using Session Key and compares
                                         with response received
        -------------------------------------------------------------------
```
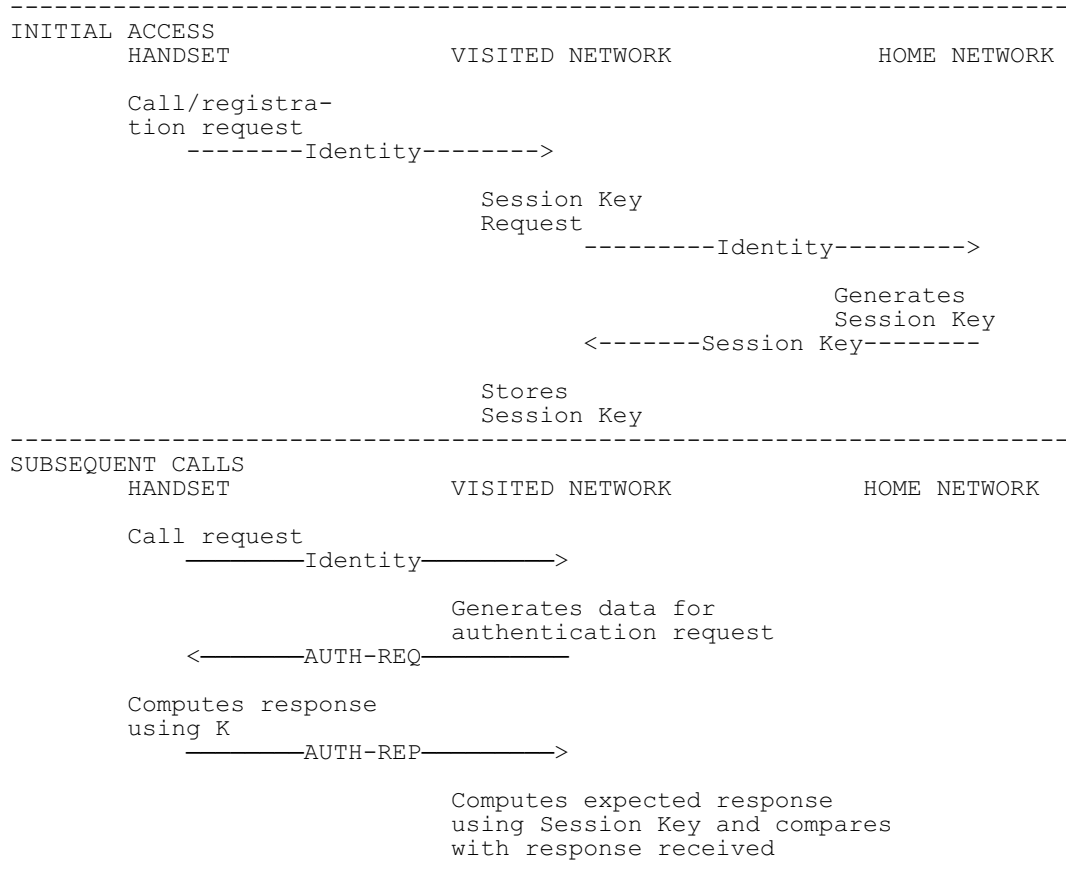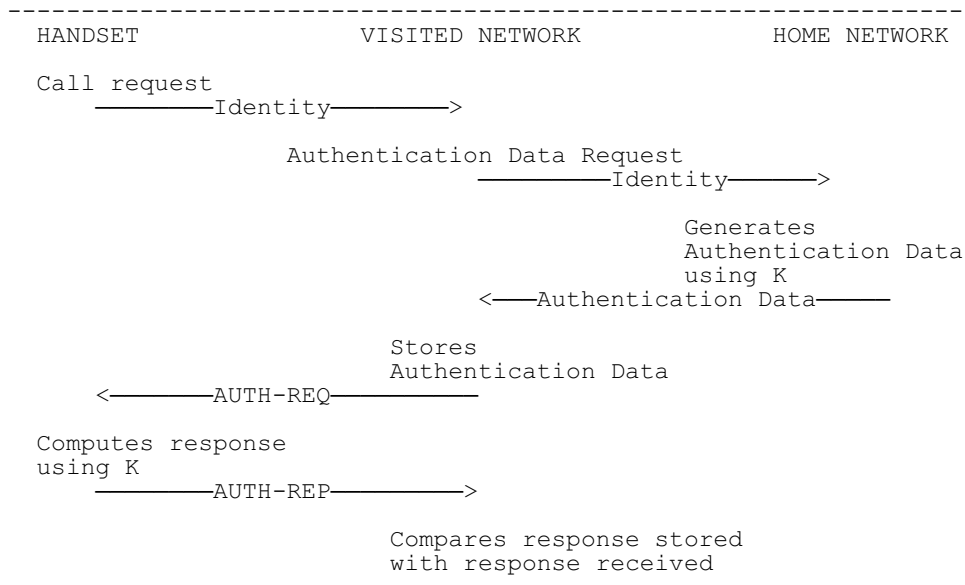
NOTE: Only PT authentication is depicted. The case of FT and mutual authentication is essentially the
           same.

Figure D.2: Roaming using sessions keys

D.7.4   Use of precalculated sets

A third option is that the home network calculates for each call a set of authentication data consisting of an authentication request (AUTH-REQ) and an authentication response (AUTH-RES). This set is transferred to the visited network on request, which can use it to authenticate the handset. The key management for roaming is depicted in figure D.3.

```
     ----------------------------------------------------------------
       HANDSET                 VISITED NETWORK              HOME NETWORK

       Call request
            ————————Identity————————>

                            Authentication Data Request
                                  ————————Identity————————>

                                                 Generates
                                                 Authentication Data
                                                 using K
                                    <————Authentication Data————

                            Stores
                            Authentication Data
              <————————AUTH-REQ————————

       Computes response
       using K
            ————————AUTH-REP————————>

                            Compares response stored
                            with response received
     ----------------------------------------------------------------
```

NOTE 1:   Only PT authentication is depicted. The case of FT and mutual authentication is essentially the same.

NOTE 2:   The home network can generate and transmit more than one (say n) set at a time. If n sets are obtained, the visited network can undertake n instances of authentication before it has to request more authentication data from the home network.

Figure D.3: Roaming using precalculated sets

Annex E (informative):   Limitations of DECT security

E.1   Introduction

The proposed security features for DECT have been carefully selected to counter the threats identified for the DECT systems. To make cost effective solutions possible it is however accepted that the security profiles finally selected for DECT may contain only some of the security features described in this ETS.

Furthermore, some security limitations were identified during the specification of the present security features. These were judged as having a negligible or acceptable impact on the system security. The judgement was therefore that it would not be appropriate or necessary to counter these threats, even if it was technically possible to do so. It has to be recognized that, for technical and economic reasons, security features do not give 100% protection; perfect security is never possible.

This annex aims at describing some of these remaining threats. This is done for completeness and, in some cases, for motivating and recommending suitable administrative procedures.

E.2   Protocol reflection attacks

In this type of attack there is a malicious entity which can masquerade as an FT and as a PT. In the authentication schemes this entity will try to place itself between the genuine PT and FT, in order to attract the call and subsequently handle it in some way, advantageous for the malicious entity. The authentication procedure, be it PT authentication, FT authentication, or both combined, will take place exactly in the normal way. The only difference is that the protocol exchange is relayed via the false entity. After the authentication procedure is finished the false entity may take over the call completely, either as a false PT or as a false FT - the choice resides with the false entity.

This kind of attack is very hard to protect against, as the false entity need not be more intelligent than having to technically follow the protocols and quickly re-transmit the protocol elements. The attack is, in that respect, logically passive and not much more complicated than a successful monitoring of the radio link. To counter the take-over of calls in this way, authentication has to be repeated several times during the call. Even then the protection is limited; only continuous encryption gives total protection.

However, the negative consequences need not be too severe. If encryption is not used, the typical threat is that calls are re-directed to wrong destinations or are terminated at the false entity which may also act as the called party. If encryption is used there might be a threat if one party in the DECT connection could disable the encryption process or prevent it from starting. It is desirable that it is not possible for the encryption process to be terminated in such a way.

E.3   Static cipher key and short Initial Vector (IV)

In the DECT encryption process the frame number is used as the IV for varying the encryption process. If the cipher key is derived from the authentication process, as described in this ETS, this will also guarantee a change in the cipher process. If, however, the SCK option is used, the cipher key will be the same for many calls, and the needed variation is dependent on the IV alone. In this case the size of the IV may be important for the security of the cipher system, as there is a risk that calls being encrypted with the same IV (that is the same frame numbers are used) may be analysed to yield the two underlying plaintexts.

The analysis is not trivial, but may be considered possible if the redundancy in the plaintexts is essential and known to, or guessed by, the attacker. As more and more calls are made with the same cipher key, more and more of the frames are used and may be collected and used for subsequent analysis. This will be possible when the same frame numbers are used for the second time. Furthermore, as the frame numbers tick on in a predictable way, the attacker will know the important moments for monitoring a call. He may even initiate calls at these moments to the PT under attack in order to obtain the needed material.

Clearly the size of the IV is important for thwarting this attack. Currently the size is 28 bits, which gives a frame number repetition cycle of 31 days. Attacks will not be possible if the SCK is changed once a month. Even if the key is not changed that often, attacks may be judged to be improbable, or demanding too much, in the way of resources, to constitute a real threat. That decision will have to be made by the operator/user of the DECT systems.

E.4   General considerations regarding key management

In all security systems involving secret cryptographic keys the overall security will stand or fall with the security of the keys. DECT security is thus limited by the security practices and standards used for generating, distributing and storing the secret keys.

Although not a formal part of the security features for the DECT air interface, some general observations on key management will be found in subclause D.5.2. Over-the-air distribution of key material is one example of a procedure where the balance between risk of exposure and convenient management has to be carefully evaluated (and monitored for attacks).

The physical protection of the secret authentication key, and possibly the SCK, in the handset is another evident limit to system security. The read-out or otherwise unnoticed copying of the key should be considered to be a serious threat. This threat can be limited by a tamper-resistant physical design. Full tamper-proof designs are generally not possible for reasons of cost.

When Integrated Chip Cards (ICCs) are used for key storage (i.e., by using a DAM), a fairly high level of physical protection is offered to the keys. The use of DAMs also considerably simplifies key distribution whilst maintaining physical security.

E.5   Use of a predictable challenge in FT authentication

The use of a non-repeating, but predictable, challenge RAND_P in FT authentication (see subclause 4.4.2) provides an opportunity for FT impersonation. The weakness can however be countered by ensuring that if RAND_P generation is predictable, then FT authentication is always immediately preceded by PT authentication as proposed in subclause 4.6.

Annex F (informative): Security features related to target networks

F.1 Introduction

This annex is intended to indicate where the standard security services are implemented and how they interface with the DECT and attached networks. The attached target networks described in this annex are those identified in ETR 056 (see Bibliography).

It should be stressed that all of the models presented are purely functional. In many cases some of the functional building blocks will be physically co-located.

Each of the target networks in ETR 056 (see Bibliography) is considered in turn. For each of these networks an explanation is given of the significance of each of the security services and a description is given of where the security mechanism may be implemented.

For each of the target environments example security profiles are given. These profiles encompass the services to be provided and the mechanisms and algorithms to be implemented (e.g. mutual authentication, DSAA, etc.).

F.1.1 Notation and DECT reference model

All of the functional models include a Home Data Base (HDB) and a Visitors Data Base (VDB). This subclause gives a summary of the functionality of these data bases.

Home Data Base (HDB): The HDB manages home subscribers or PPs. The HDB stores all security parameters associated with the home subscriber and functionally forms a part of the local or global network.

Visitors Data Base (VDB): The VDB manages visiting subscribers or PPs. The VDB stores all security parameters associated with the visitor and functionally forms a part of the local or global network.

The security parameters held by the VDB and HDB will be described specifically in the applications below.

F.1.2 Significance of security features and intended usage within DECT

Authentication of a PT: The purpose of this service is twofold:

1) for public access systems it is to provide a cryptographically secure method of identifying subscriptions for billing purposes;

2) in private systems it is to prevent illicit access to private base stations in order to avoid charge, or for reasons of anonymity (see "DECT User and Portable-Part Identification Security", Bibliography, and annex A).

Throughout this annex no distinction is made between the ways which may be used to derive the authentication key (e.g. from an AC or a UAK), although in certain cases examples are provided.

Data Confidentiality: This is to provide privacy for user information (speech or data) and signalling information (C_T field) across the air interface.

The cipher key can be generated either as part of the authentication of the PT process or it may be a static cipher key.

Authentication of an FT: This service was introduced to counter the following threats (see annex A):

1) unauthorised loading of information (such as a Zap code) into a PP which may render the PP unusable;

2) FP impersonation in order to bypass privacy.

Mutual authentication: this is achieved by combining other services (see subclause 4.6). The mechanisms required for implementation are those used for the component services.

Direct method: This is achieved by combining the FT and PT authentication services.

Indirect method 1: This is provided by combining data confidentiality using the DCK with authentication of the PT.

Indirect method 2: This is provided by applying the confidentiality mechanism using a SCK or a previous DCK.

User authentication: This is to provide authentication of the user as opposed to just authentication of user subscription data held in the portable equipment. It requires the user to enter a value (e.g. a UPI) whenever the service is invoked.

F.1.3 Mechanism/algorithm and process requirements

This subclause indicates where, and which, algorithms/processes need to be implemented in order to provide the security services. Table F.1 shows which processes/mechanisms are required within the PT. Table F.2 shows which processes/mechanisms are required within the FT.

Table F.1: PT mechanisms

Security Services/ Algorithms Processes	Authentication			Confidentiality
	PT	FT	User	
Sequence/ Random Number Generator		Y		
KSG				Y
A1	Y		Y	Y*
A2		Y		

Y: algorithm/process required;
Y*: algorithm/process required if authentication of the PT is used for the cipher key derivation;
A1: represents the combined A11 and A12 processes;
A2: represents the combined A21 and A22 processes.

NOTE 1: Processes A1 and A2 include the implementation of an authentication algorithm; this may be proprietary or the DSAA.

NOTE 2: If the authentication key is derived from a UAK or an AC (as described in subclauses 4.5.2.1 and 4.5.2.2 respectively), then process B1 is required in the PT. Similarly if the authentication key is derived from a UAK and a UPI (as described in subclause 4.5.2.3) then process B2 is required in the PT.

Table F.2: FT mechanisms

Security Services/ Algorithms Processes	Authentication			Confidentiality
	PT	FT	User	
Random Number Generator	Y		Y	
KSG				Y
A1	Y		Y	Y*
A2		Y		

Y: algorithm/process required;
Y*:   algorithm/process required if authentication of the PT is used for the cipherkey derivation;
A1:   represents the combined A11 and A12 processes;
A2:   represents the combined A21 and A22 processes.

NOTE 1:   Processes A1 and A2 include the implementation of an authentication algorithm; this may be proprietary or the DSAA.

NOTE 2:   If the authentication key is derived from a UAK or an AC (as described in subclauses 4.5.2.1 and 4.5.2.2 respectively), then process B1 is required in the FT or in the fixed network element where the authentication key is derived. Similarly if the authentication key is derived from a UAK and a UPI (as described in subclause 4.5.2.3) then process B2 is required in the FT or in the fixed network element where the authentication key is derived.

F.2   PSTN reference configurations

F.2.1   Domestic telephone

The domestic telephone reference configuration is shown in ETR 056, figure 3.5. For this configuration two example security profiles P1 and P2 are suggested (see below) reflecting two distinct envisaged applications.

P1 profile: The P1 profile provides for the simplest case where visitors or multiple handsets are not catered for. The basic security requirements for this environment (as identified in annex A) are for confidentiality and mutual authentication.

The P1 profile provides the requirements for confidentiality using the DSC algorithm and mutual authentication via indirect method 2. Stored within local storage facilities within the fixed radio termination will be the IPUI, TPUI and the SCK. On receipt of the IPUI the SCK is loaded into the cipher.

By tables F.1 and F.2, the only algorithm/process which needs to be installed is the DSC. This will be installed within the FT and the PT.

Table F.3: Profile P1

Domestic telephone example Profile P1	
DECT Security Service	Profile
Authentication of the PT	
Authentication of the FT	
User Authentication	
Mutual Authentication	Indirect method 2
Confidentiality	Using a static cipher key and the DSC

P2 profile: This profile allows greater flexibility and caters for visitors and multiple handsets through the introduction of authentication of the PT service.

The P2 profile provides authentication of the PT using the DSAA, data confidentiality using the DSC and a DCK, and indirect mutual authentication.

By tables F.1 and F.2, the processes/algorithms required to be implemented are as follows: A1, DSC, random number generator. These processes/algorithms are implemented in the following areas:

Algorithm/processes requiring implementation within the FT:

- DSC;
- random number generator;
- process A1.

Algorithm/processes requiring implementation within the PT:

- DSC;
- Process A1.

Security parameters associated with the home subscriber which are stored within the local storage facilities and the PT:

- IPUI/TPUI;
- authentication key.

Security parameters associated with a visitor which are stored within the local storage facilities:

- IPUI/TPUI;
- authentication key.

Table F.4: Profile P2

Domestic telephone example Profile P2	
DECT Security Service	Profile
Authentication of the PT	Using the DSAA
Authentication of the FT	
User Authentication	
Mutual Authentication	Using indirect method 1
Confidentiality	Cipher key derived using authentication of the PT. Encryption provided using the DSC

F.2.2 PBX

The PBX reference configurations are depicted in figures 3.6 a) and b) of ETR 056. Two example profiles, P1 and P2, are given reflecting the requirements different companies may impose on the system.

> NOTE 1: The profiles below are illustrative.

P1 profile: The P1 profile is to cater for the need for a temporary registration (e.g., in hotels and restaurants).

> NOTE 2: A confidentiality service is not a requirement within restaurants; however, it does appear to be a requirement within hotels.

The profile in this case will be the same as the domestic telephone P2 case. However, the authentication process will take place within the local network as opposed to the FT. This results in the following implementation:

Algorithm/processes requiring implementation within the local network;

- Random number generator;
- Process A1.

Algorithm/processes requiring implementation within the FT:

- DSC.

Algorithm/processes requiring implementation within the PT:

- DSC;
- Process A1.

The DCK is loaded into the DSC within the FT via the use of an IWU between the local network and the FT.

Security parameters stored within the HDB and PT:

The HDB and portable radio termination will contain:

- IPUI/TPUI;
- authentication key.

Security parameters stored by the VDB

The VDB will store:

- IPUI/TPUI;
- authentication key.

(Note the HDB and VDB are a part of the local network).

P2 profile: The PBX P2 profile caters for the case when a private company wishes to use the system for confidentiality as well as to implement a proprietary authentication algorithm (or the DSAA) using, for example, a DAM.

The DAM will be inserted into the PT.

Algorithm/processes requiring implementation within the local network:

- Random number generator;
- process A1.

Algorithm/processes requiring implementation within the FT:

- DSC.

The DCK is loaded into the DSC within the FT via the use of an IWU between the local network and the FT.

Algorithm/processes requiring implementation within the DAM:

- Process A1.

Algorithms/processes requiring implementation within the PT:

- DSC.

Security parameters stored within the HDB and DAM:

- IPUI/TPUI;
- authentication key.

Security parameters stored by the VDB:

- IPUI/TPUI;
- authentication key.

NOTE 3:   The HDB and VDB are a part of the local network.

NOTE 4:   The P2 profile may also be adopted without the use of a DAM.

Table F.5: Profile P2

PBX example Profile P2	
DECT security service	Profile
Authentication of the PT	Using a proprietary authentication algorithm or the DSAA
Authentication of the FT	
User authentication	
Mutual authentication	Using indirect method 1
Confidentiality	Cipher key derived using authentication of the PT. Encryption provided by the DSC

F.2.3 Local loop

The local loop reference configuration is depicted in figure 3.7 of ETR 056. For this configuration the example profile was driven by the following requirements:

- the need for authentication of the PT for billing purposes;

- the need for a confidentiality service to provide privacy over the air interface since within this target configuration the user would have the impression of being able to hold a private conversation.

In this target reference configuration the DECT network is integrated into the PSTN as a local loop replacement. An important point to note about this configuration is that authentication can take place either centrally by implementing the authentication of the PT processes within the global network, or locally within the FT. If central authentication is used an IWU with the FT is needed to load the DCK into the DSC, and all security parameters will be stored in the HDB and VDB within the global network. If local authentication is used, then the authentication of the PT processes will be implemented with the DSAA in the FT. For local authentication, local security parameter storage facilities will be required within the FT.

Central authentication will implement and store the following in the following locations:

Algorithm/processes requiring implementation within the global network:

- Random number generator;
- process A1.

Algorithm/processes requiring implementation within the FT:

- DSC.

Algorithm/processes requiring implementation within the PT:

- DSC;
- process A1.

The DCK is loaded into the DSC within the FT via the use of an IWU between the global network and the FT.

Security parameters stored within the HDB and PT:

- IPUI/TPUI;
- Authentication key.

Security parameters stored within the VDB:

- IPUI/TPUI;
- KS, RS.

It should be noted in this configuration the handset is connected to a socket just like a normal phone, with the PT being located within the socket.

Table F.6: Local Loop Profile

Local loop example profile	
DECT security service	Suggested security profile
Authentication of the PT	Using the DSAA
Authentication of the FT	
User authentication	
Mutual authentication	Using indirect method 1
Confidentiality	With the cipher key derived from authentication of the PT; the DSC being used for encryption

F.3 ISDN reference configurations

F.3.1 Terminal equipment

This subclause contains two example profiles P1 and P2 reflecting the two distinct configurations depicted in figures 3.8 and 3.8b of ETR 056. The profiles are different because of the way the PT is represented; either as part of the handset (profile P1, figure 3.8a), or fixed and hard-wired to the terminal (profile P2, figure 3.8b).

P1 profile

The P1 profile caters for the case where the local network offers the supplementary services of the ISDN network to the TE1 connection on a per handset basis. In this configuration the FT will be located on the customer premises, and the PT and end system are the handset. Since the handset is portable and can be assumed to be associated to a unique user, the services will be offered on a per handset basis as opposed to a per user basis. The configuration will be exactly the same as the domestic telephone P2 profile so as to cater for visitors and multiple handsets. However, it should be noted that the local network and FT are not physically co-located.

P2 profile

Since the PT and end system are fixed, the supplementary services of the ISDN network will be offered on a per user basis via the user authentication service. This means that many terminals may be attached to the same PT and still be billed on an individual basis.

The way this process works is as follows:

- the user wishing to access the ISDN network is first requested to enter his IPUI/TPUI and the personal portion of his authentication key (e.g. UPI);

- these parameters are then transferred to the PT where the authentication key is generated (e.g. from the UPI and a UAK) and the authentication process proceeds as normal.

The authentication will be implemented, and security parameters stored, within the local network and PT. The authentication process will be centrally operated. Such a local network will usually be privately operated with a local, or wide area extent, and will require an IWU with the FT to allow user selection of supplementary services and for the down loading of the DCK into the DSC. The base station in this model will consist of just the FT.

To speed up the authentication process the user authentication processes could be implemented within the FT with authentication taking place locally. However, in this case local storage facilities will have to be provided and it is suggested that the security parameters stored there should be the RS and KS (session authentication key).

Algorithm/processes requiring implementation within the local network:

- Random number generator;
- process A1.

Algorithm/processes requiring implementation within the FT:

- DSC.

Algorithm/processes requiring implementation within the PT:

- DSC;
- process A1.

The DCK is loaded into the DSC within the FT via the use of an IWU between the local network and the FT.

Security parameters stored within the HDB:

- IPUI/TPUI;
- authentication key material (e.g. UAK and UPI).

Security parameters stored within the PT:

- Authentication key material (e.g. UAK).

The IPUI/TPUI will be entered by the user.

Security parameters stored by the VDB:

The VDB will store:

- IPUI/TPUI;
- authentication key.

Table F.7: Profile P2

ISDN terminal equipment example Profile P2	
DECT security service	Profile
Authentication of the PT	
Authentication of the FT	
User authentication	Using (e.g. UPI)
Mutual authentication	Indirect method 1
Confidentiality	Using the cipher key derived from user authentication. Encryption provided by the DSC

F.3.2 Network termination 2

This configuration is depicted in figure 3.9 of ETR 056. Since the PT and end system are again fixed, the example profile will be the same as the ISDN terminal equipment P2.

F.3.3 Local loop

The local loop configuration is shown in figure 3.10 of ETR 056. Since the PT and end system are again fixed, the example profile will be the same as the ISDN terminal equipment P2.

NOTE: Note that in the configuration depicted in figure 3.10 of ETR 056 the HDB and VDB will form part of the global network.

F.4 X.25 reference configuration

F.4.1 Data Terminal Equipment (DTE)

The configuration for this application is depicted in figure 3.11 of ETR 056. The example profile is to cater for the situation where the DTE has been supplied to a user with the aim of supplying all, or a fixed subset of, the supplementary services offered by the global network. In this case the supplementary services will be offered by the local network and the global network will bill the local network connection. Only one user (corporate company) is envisaged in this environment, the local network being under the control of the user, so that the profile provides authentication via the knowledge of a SCK. The profile will be the same as the domestic telephone P1 case.

F.4.2 PAD equipment

This configuration is shown in figure 3.12 of ETR 056. The example profile caters for the situation where the local network offers the supplementary services of the global network by acting as a gateway to the X.25 network. The global network bills the local network connection, with the local network billing the end systems on a user basis. The billing on a per user basis is to allow multiple terminals to be attached to the PT (this also provides extra security by making the user enter a UPI before accessing the network). The PAD provider could be, and will be, in most cases the X.25 provider. The profile will be the same as the ISDN terminal equipment P2 case.

NOTE: Although the local loop and FT are functionally represented as the base station, they need not be physically co-located. This physical separation is important in terms of the storage of the authentication keys since the FT may be located on the customer's premises.

F.5 GSM reference configuration

F.5.1 Base station substation

This configuration is depicted in figure 3.13 of ETR 056. Here the DECT network acts as an interface to the MSC of the GSM network. The service will be provided by the local network provider who bills users on a handset basis, with the MSC billing the local network for the calls made. The requirements for this configuration will be for authentication of the handset for billing purposes and for the privacy of the air interface information. To cater for these requirements the suggested profile will be the PBX P1 profile.

F.5.2 Mobile Station

This configuration is depicted in figure 3.14 of ETR 056. The profile in this case will be the same as the base station substation as the requirements are the same.

F.6 IEEE.802 reference configuration

F.6.1 Bridge

Figure 3.15 of ETR 056 (see Bibliography), depicts this configuration. Since the end system and portable radio termination will be fixed, to allow the local network to control user access when different users are sharing the same terminal, or multiple terminals are attached to the same portable radio termination, the user authentication service will be used. The profile will be the same as the terminal equipment P2 profile.

F.6.2 Gateway

The gateway configuration is depicted in figure 3.16 of ETR 056. The profile will be the same as the terminal equipment P2 profile.

F.7 Public access service reference configurations

F.7.1 Fixed public access service reference configuration

This configuration is represented by figure 3.1 of ETR 056 (see Bibliography). Two profiles, P1 and P2, are described, one with a confidentiality service, the other without. It should be noted that the profiles are illustrative.

P1 profile: The primary requirement in this environment will be subscriber registration data authentication for billing purposes, hence the need for authentication of the PT. In a public access environment there is no requirement for confidentiality; however, there is a requirement for authentication of the FT. The P1 profile just caters for these needs.

Table F.8: Profile P1

Fixed public access service example Profile P1	
DECT security service	Profile
Authentication of the PT	Using the DSAA
Authentication of the FT	Using the DSAA
User authentication	
Mutual authentication	Direct method (automatic)
Confidentiality	

P2 profile: Example profile P2 is based on the use of the confidentiality service to provide the requirement for authentication of the FT. The profile will be the same as the PBX P1 profile.

Annex G (informative): Compatibility of DECT and GSM authentication

G.1 Introduction

DECT and GSM apply a similar but not identical cryptographic procedure for the authentication of a handset. Special modules (like smart cards) are used in GSM (where they are called SIMs) and may be used in DECT (where they are called DAMs) to store authentication information and perform the necessary calculations for authentication.

A desirable feature would be that a SIM could be used for authentication in DECT and a DAM could be used for authentication in GSM.

This annex explains how, and under which conditions, these features can be realised from a security point of view. Since GSM only offers mobile part authentication, only the corresponding PP authentication for DECT is considered.

A SIM contains all user related information and other system specific information and this is also be true for a DAM. How this user and system specific information is transferred to and from the cards is not treated here. It should be clear, however, that the use of a DAM in GSM and a SIM in DECT will only be possible if they have the same functionality.

G.2 SIM and DAM functionality

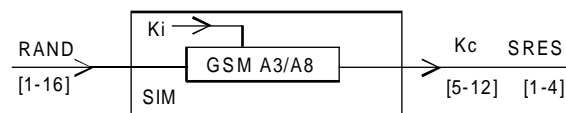The security functionality of a SIM is depicted in the figure below.

Figure G.1

The SIM is activated by a specific command (RUN_GSM_ALGORITHM).

The single input of the SIM is a 16 byte value RAND which is loaded into the SIM by the handset microprocessor.

Using the GSM authentication key Ki and the GSM algorithm(s) A3/A8 two values are calculated:

- SRES (4 bytes); and

- Kc (8 bytes).

These values are serially transferred from the SIM to the handset microprocessor where the first four bytes (1-4) are the SRES and the following eight bytes (5-12) the Kc (cipher key). See ETS 300 608 (Bibliography) for detailed information.

Though a DAM is in process of being specified, the needed functionality of a DAM with respect to a PP authentication is shown in the figure below. It should be noted that a DAM could support more security functions than a SIM (e.g. authentication of a FP, mutual authentication). These new security functions should be activated by new commands.
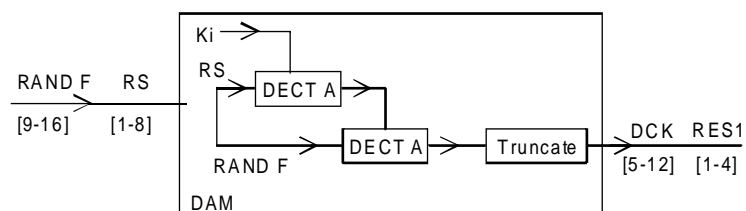

Figure G.2

For authentication of the PT the DAM should be activated by a specific command which is identical to the corresponding SIM command (RUN_GSM_ALGORITHM).

Two inputs, RS and RAND_F, are loaded to the DAM by the handset microprocessor. When the DSAA is used as algorithm A, both values are eight bytes long. In this case bytes [1-8] form RS and bytes [9-16] form RAND_F.

Using the authentication key K and DECT algorithm A, two values, RES1 and DCK, are obtained via the intermediate value KS and a possible truncation.

These values are serially transferred from the DAM to the handset microprocessor. If the DECT standard algorithms are used, the RES1 is four bytes long and the DCK eight bytes. In this case the first four output bytes ([1-4]) form RES1 and the following eight bytes ([5-12]) DCK.

G.3   Using an SIM for DECT authentication

An SIM can be used for DECT PP authentication provided that the following hold:

1)   the FP knows that a GSM authentication (but DECT encryption and protocols) will be performed by the handset;

2)   the "standard" lengths for the DECT parameters are used; i.e., RS, RAND_F and DCK each have a length of eight bytes and RES1 has a length of four bytes;

3)   SIM authentication and DAM PP authentication are activated by an identical command;

4)   the protocol and interface between the DECT PP and the DAM are identical to the corresponding protocol and interface between the GSM mobile station and the SIM;

5)   no session key mechanism for DECT is used.

The handset microprocessor using the command which is equivalent to (RUN_GSM_ALGORITHM) serially inputs RS and RAND_F to the SIM. This serial input is interpreted by the SIM as the RAND. The SIM performs a GSM calculation and outputs the SRES and Kc which are interpreted by the handset as RES1 and DCK, respectively, and used as such in the DECT protocols.

G.4   Using a DAM for GSM authentication

A DAM can be used for GSM authentication provided that the following hold:

1)   the FP knows that a DECT authentication (but GSM encryption and protocols) will be performed by the mobile station and also knows the DECT authentication key K;

2)   the "standard" lengths for the DECT parameters are used: i.e. RS, RAND_F and DCK have a length of eight bytes and RES1 has a length of four bytes;

3)   SIM authentication and DAM PP authentication are activated by an identical command;

4)   the protocol and interface between the DECT PP and the DAM are identical to the corresponding protocol and interface between the GSM mobile station and the SIM.

The mobile station microprocessor using the command (RUN_GSM_ALGORITHM) serially inputs RAND (16 bytes) to the DAM. The first eight bytes of this value are interpreted by the DAM as the RS and the second eight bytes as RAND_F. The DAM performs a DECT calculation and outputs RES1 and DCK, which are interpreted by the mobile station as SRES and Kc, respectively, and used as such in the GSM protocols.

Annex H (informative): DECT standard authentication algorithm

The DECT standard authentication algorithm is only available on a restricted basis. For further information please contact ETSI.

Annex J (informative): DECT standard cipher

The DECT standard cipher is only available on a restricted basis. For further information please contact ETSI.

Annex K (informative):   Bibliography

-      ETR 048: "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Common interface Services and facilities requirements specification".

-      ETR 015: "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); Reference Document".

-      ETR 056: "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); System description document".

-      ETR 042: "Radio Equipment and Systems (RES); Digital Enhanced Cordless Telecommunications (DECT); A guide to the DECT features that influence the traffic capacity and the maintenance of high radio link quality, including the results of simulations".

-      ETS 300 608: "European digital cellular telecommunications system (Phase 2); Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (GSM 11.11)".

History

<table>
<tr><th colspan="4">Document history</th></tr>
<tr><td>October 1992</td><td>First Edition</td><td></td><td></td></tr>
<tr><td>August 1995</td><td>Public Enquiry</td><td>PE 90:</td><td>1995-08-21 to 1995-12-15</td></tr>
<tr><td>June 1996</td><td>Vote</td><td>V 106:</td><td>1996-06-24 to 1996-08-30</td></tr>
<tr><td>September 1996</td><td>Second Edition</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>