EUROPEAN TELECOMMUNICATION STANDARD

DRAFT
pr **ETS 300 393-7**

July 1996

Source: ETSI TC-RES

Reference: DE/RES-06004-7

ICS: 33.060, 33.060.50

Key words: TETRA, PDO, security

Radio Equipment and Systems (RES); Trans-European Trunked Radio (TETRA); Packet Data Optimized (PDO); Part 7: Security

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Postal address: F-06921 Sophia Antipolis CEDEX - FRANCE
Office address: 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE
X.400: c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

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

Contents

Blank Page

Foreword

This draft European Telecommunication Standard (ETS) has been produced by the Radio Equipment and Systems (RES) Technical Committee of the European Telecommunications Standards Institute (ETSI), and is now submitted for the Public Enquiry phase of the ETSI standards approval procedure.

This ETS is a multi-part standard and will consist of the following parts:

Part 1: "General network design".

Part 2: "Air Interface (AI)".

Part 3: "Inter-working ", (DE/RES-06004-3).

Part 4: "Gateways ", (DE/RES-06004-4).

Part 5: "Terminal equipment interface", (DE/RES-06004-5).

Part 6: "Line connected stations", (DE/RES-06004-6).

Part 7: "Security".

Part 8: "Management services", (DE/RES-06004-8).

Part 9: "Performance objectives", (DE/RES-06004-9).

Part 10: "SDL Model of the air interface", (DE/RES-06004-10).

Part 11: "PICS proforma", (DE/RES-06004-11).

Proposed transposition dates	
Date of latest announcement of this ETS (doa):	3 months after ETSI publication
Date of latest publication of new National Standard or endorsement of this ETS (dop/e):	6 months after doa
Date of withdrawal of any conflicting National Standard (dow):	6 months after doa

Blank page

1 Scope

This ETS describes the security mechanisms in the Trans-European Trunked Radio (TETRA) Packet Data Optimised (PDO) standard. It provides mechanisms for authentication and key management mechanisms for the PDO air interface.

The following two authentication services are specified for the PDO air interface as determined in ETR 086-3 [1], based on a threat analysis:

- authentication of a user by the Radio Packet Data Infrastructure (RPDI);

- authentication of the RPDI by a user.

2 Normative references

This ETS incorporates by dated and 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] ETR 086-3: "Trans-European Trunked Radio (TETRA) systems; Technical requirements specification; Part 3: Security aspects".

3 Definitions and abbreviations

3.1 Definitions

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

Authentication Code (AC): A (short) key to be entered by the user into the terminal.

authentication Key (K): The primary secret, the knowledge of which has to be demonstrated for authentication. On the infrastructure side, it is stored in a secure place of the home network. In the terminal it is generated in one of three ways:

1) K may be generated from an Authentication Code (AC) that is manually entered by the user;

2) K may be generated from a User Authentication Key (UAK) stored in a module (detachable or not);

3) K may be generated from both the UAK stored in a module and a Personal Identification Number (PIN) entered by the user.

Personal Identification Number (PIN): Entered by the user into the terminal and used to generate K together with the UAK.

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

RANDom challenge (RAND1, RAND2): A random value generated by the infrastructure to authenticate a user or in a terminal to authenticate the infrastructure, respectively.

Random Seed (RS): A random value used to derive a Session authentication Key (KS, KS') from K.

RESponse (RES1, RES2): A value calculated in the terminal from RAND1 and the KS to prove the authenticity of a user to the infrastructure or by the infrastructure from RAND2 and the KS' to prove its authenticity to a user, respectively.

Session authentication Key (KS, KS'): Generated from K and RS for the authentication of a user. It has a more limited lifetime than K, can be stored in less secure places, and forwarded to visited networks.

spoofer: An entity attempting to obtain service from or interfere with the operation of the system by impersonation of an authorised system user or system component.

User Authentication Key (UAK): Stored in a (possibly detachable) module within the terminal and used to derive K (with or without a PIN as an additional parameter).

3.2 Abbreviations

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

AC	Authentication Code
AI	Air Interface
BS	Base Station
K	authentication Key
KS	Session authentication Key
MM	Mobility Management
MS	Mobile Station
PDU	Protocol Data Unit
PIN	Personal Identification Number
RAND1	RANDom challenge 1
RAND2	RANDom challenge 2
RES1	RESponse 1
RES2	RESponse 2
RPDI	Radio Packet Data Infrastructure
RS	Random Seed
SAP	Service Access Point
UAK	User Authentication Key
XRES1	eXpected RESponse 1
XRES2	eXpected RESponse 2

4 Air interface authentication and key management mechanisms

NOTE: The algorithms referred to in this clause may be the same as those defined in the TETRA Voice plus Data standard with some outputs ignored.

4.1 Security mechanisms

4.1.1 Requirements

The following requirements shall be taken into account:

- the mechanisms used to realise the authentication services and the key management functions shall be based on symmetric cryptographic algorithms. They assume that both parties share a common secret key, that has to be distributed before the authentication process. Authentication shall be successfully performed by proving the knowledge of the secret key to the other party;

- in order not to be vulnerable to a replay attack, the exact way of delivering this proof shall change unpredictably from instance to instance of an authentication;

- a synchronised real time clock shall not be required within the terminals by the authentication mechanism;

- there shall be separate mechanisms for authentication of a user by the infrastructure and for authentication of the infrastructure by a user. It shall be possible to combine these two mechanisms to achieve mutual authentication of user and infrastructure;

- there shall be a key hierarchy of master and session authentication keys. There shall not be a necessity to store master keys in Base Stations (BSs) or forwarded to visited networks;

- a spoofer who is able to compromise one of the BSs (could be an untrustworthy operator of a visited network) but not the central authentication module shall not be able to falsely authenticate

himself as a legitimate user to any other but this particular BS and only then until a new session authentication key is chosen.

4.1.2 Authentication of a user

In this subclause, a mechanism is described that should be used to achieve the authentication of a user of a TETRA terminal by the RPDI. This shall be done using a challenge response protocol, with a session authentication key derived from an authentication key that shall be shared by the user and the infrastructure. The session authentication key shall be provided by an authentication centre of the home system.

The computation of the session authentication key should be done by an algorithm called TA11. The computation of the response should be done by another algorithm TA12P, which at the same time produces a Derived Cipher Key (DCK).

The infrastructure shall produce a random number as a challenge called RAND1. The terminal shall compute a response called RES1 and the RPDI shall compute an expected response called XRES1. The entire protocol is summarised in figure 1.

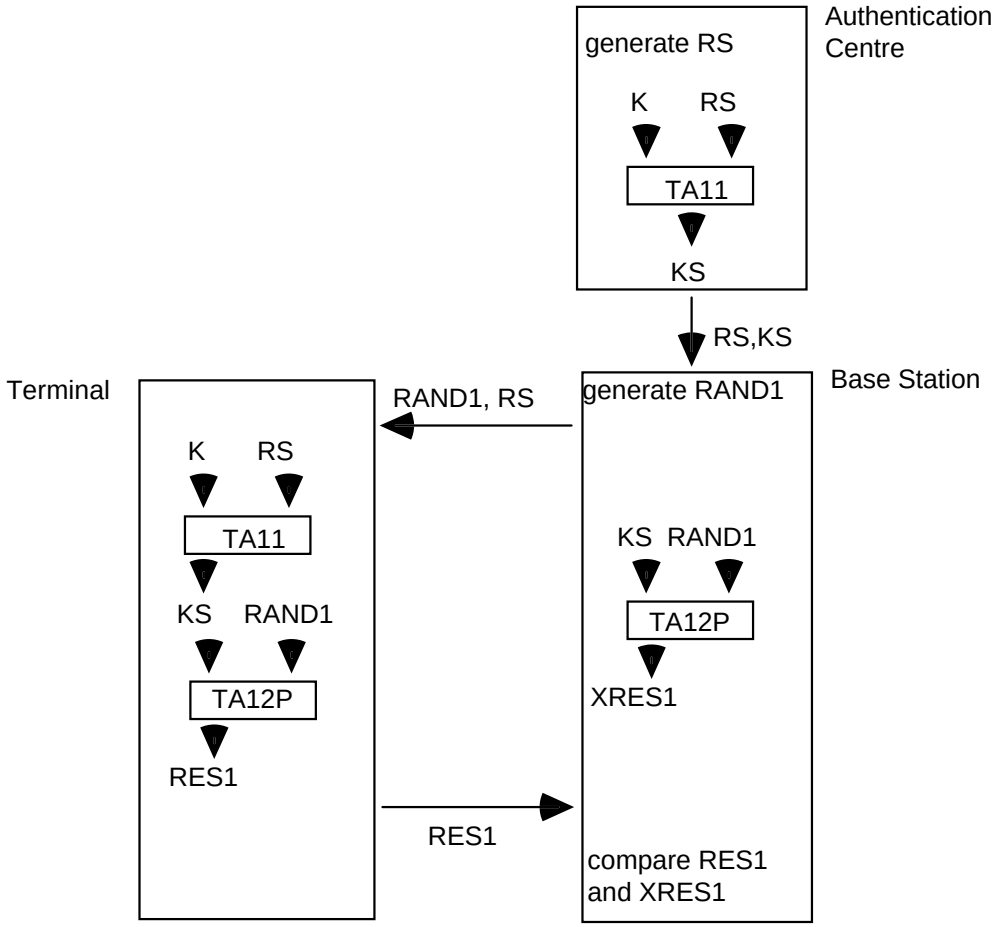

Figure 1: Authentication of a user by the infrastructure

4.1.3 Authentication of the infrastructure

Authentication of the infrastructure by a user shall be done in the same way as described in subclause 4.1.2 with the roles of the claimant and verifier reversed. The terminal shall generate a challenge called RAND2, the RPDI shall generate an actual response, and the terminal shall generate an expected response, called RES2 and XRES2 respectively.

The same authentication key K and random seed RS shall be used as in the case of authentication of the user by the infrastructure. However, the algorithms should be different: TA11 shall be replaced by TA21

and TA12P by TA22P. Hence, there should also be a different value for the session authentication key, called KS'. The process is summarised in figure 2.

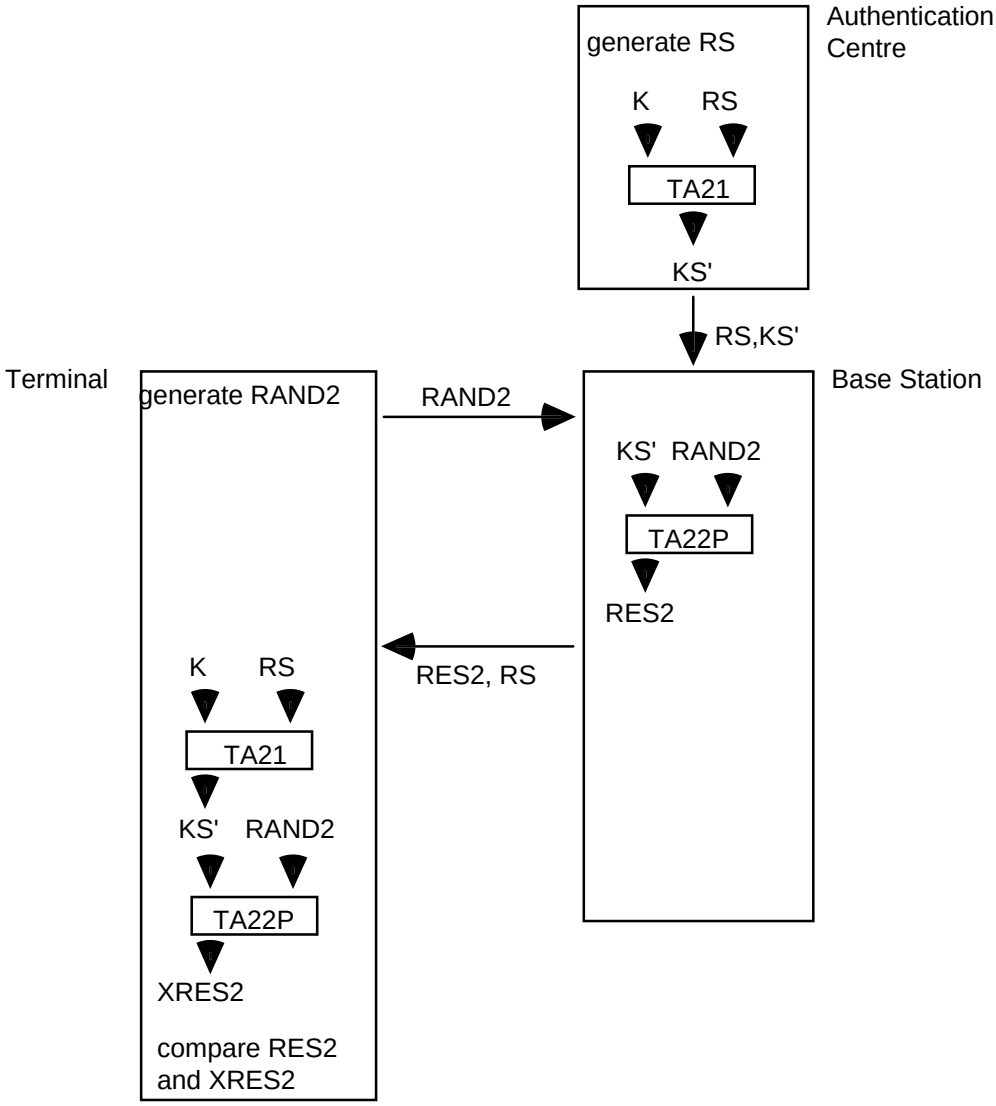

Figure 2: Authentication of the infrastructure by a user

4.1.4 Mutual authentication of user and infrastructure

Mutual authentication of user and infrastructure should be achieved by combining the two mechanisms described in subclauses 4.1.2 and 4.1.3.

4.1.5 Generation of K

Users should be authenticated by a process that is carried out in the terminal, as described in subclause 4.1.2. Therefore, the user should be able to control K. One way to exercise this control may be to enter K either directly or through a detachable module. The key may be stored in a module, detachable or not. Protection should be provided against the use of lost or stolen modules/terminals. This may be done by requiring the user to enter a personal identification number (PIN).

The PIN may be checked in either of two ways. It may either be checked locally against a reference stored in the module or it may be used to generate the authentication key in conjunction with other data stored within the module. In the latter case the actual checking shall be carried out by the infrastructure.

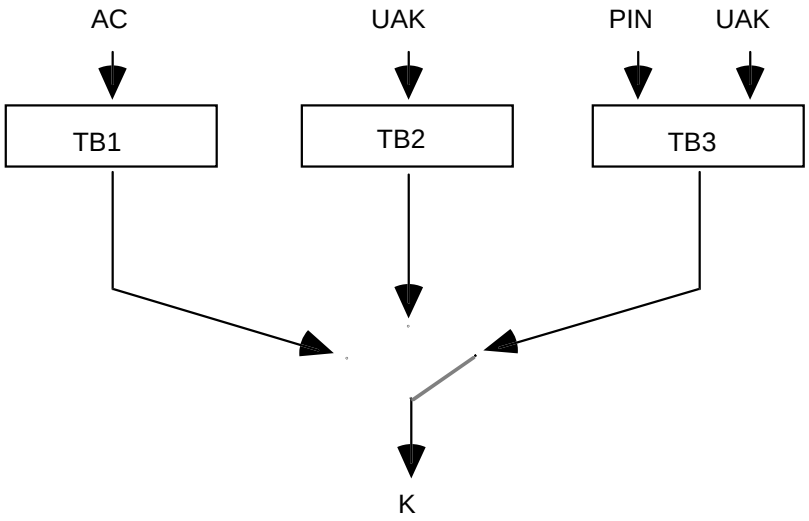

Figure 3: Generation of the authentication key

There shall be three different cases for the generation of the authentication key, which are summarised in figure 3:

1) K may be generated from an AC that is manually entered by the user. In this case the code shall be remembered by the user and should not normally be longer than a few digits. The procedure to generate K from AC is labelled TB1;

2) K may be generated from a UAK stored in a module (detachable or not). In this case the UAK can be a random value of a desirable length (e.g. 128 bits). However the user should be required to enter a PIN that can be checked before the UAK can be used for authentication. The checking of the PIN should not be part of the mechanism and, therefore, not shown in figure 3. The procedure to generate K from UAK is labelled TB2;

3) K may be generated from both the UAK stored in a module and the PIN entered by the user. The procedure to generate K from UAK and PIN is labelled TB3.

4.2 Definition of protocols

The air interface authentication protocol shall involve layer 3 Mobility Management (MM).

4.2.1 Service description and primitives

4.2.1.1 Authentication service

At the TNMM SAP, a specific service shall be provided to allow an application to initiate an authentication exchange and to receive its result. The Mobile Station (MS) application shall also respond to an authentication demand from the RPDI. The primitives required shall be as follow:

- TNMM-MS_AUTHENTICATE indication shall be used to indicate to the MS application a demand for authentication received from the RPDI;

- TNMM-MS_AUTHENTICATE response shall be used to provide the appropriate response to the challenge received in the demand for authentication;

- TNMM-MS_AUTHENTICATE confirm shall be used to confirm successful or failed authentication of the MS by the RPDI;

- TNMM-BS_AUTHENTICATE request shall be used by the MS application to initiate an authentication of the RPDI;

- TNMM-BS_AUTHENTICATE indication shall be used to report to the MS application the response returned by the RPDI, and the RS to be used for DCK generation;

- TNMM-BS_AUTHENTICATE response shall be used by the MS application to signal success or failure of the authentication to the RPDI.

Table 1: TNMM AUTHENTICATE service primitives

Generic name	Specific name	Parameters
TNMM-MS_AUTHENTICATE	indication	random challenge, random seed
TNMM-MS_AUTHENTICATE	response	value, mutual authentication flag
TNMM-MS_AUTHENTICATE	confirm	result
TNMM-BS_AUTHENTICATE	request	random challenge
TNMM-BS_AUTHENTICATE	indication	response value, random seed, mutual authentication flag
TNMM-BS_AUTHENTICATE	response	result

4.2.2 Protocol functions

An authentication exchange can be requested, either explicitly or as part of the registration procedure. It can be initiated by the MS or RPDI side. The initiating side shall send an Authentication Demand PDU that shall always be answered by the other side with an Authentication Response PDU. Success or failure of the authentication shall be communicated either by a specific Authentication Result PDU or as part of the registration procedure.

4.2.3 Protocol Data Units (PDUs)

D-Authentication Demand: shall be used by the infrastructure to start the authentication procedure of a user.

Direction: RPDI to MS;
Service used: MM;
Response to: U-Location Update Demand or none;
Response expected: U-Authentication Response.

Table 2: D-Authentication Demand PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Random challenge	80	M		
Random seed	80	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

U-Authentication Response: shall be used by the MS to reply to a demand message. The "Mutual Authentication Flag" shall be used to indicate to the infrastructure whether it should expect a U-Authentication Demand message after sending the D-Authentication Result.

Direction: MS to RPDI;
Service used: MM;
Response to: D-Authentication Demand;
Response expected: D-Authentication Result.

Table 3: U-Authentication Response PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Response Value	32	M		
Mutual Authentication Flag	1	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

D-Authentication Result: shall be used by the infrastructure to notify the MS about the result of the authentication exchange.

Direction: RPDI to MS;
Service used: MM;
Response to: U-Authentication Response;
Response expected: U-Authentication Demand or none (depending on whether the Mutual Authentication Flag was set).

Table 4: D-Authentication Result PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Authentication result	1	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

U-Authentication Demand: shall be used by the MS to start the authentication procedure of the infrastructure.

Direction: MS to RPDI;
Service used: MM;
Response to: D-Authentication Result or none;
Response expected: D-Authentication Response.

Table 5: U-Authentication Demand PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Random challenge	80	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

D-Authentication Response: shall be used by the RPDI to reply to a demand message. The "Mutual Authentication Flag" shall be used to indicate to the terminal whether it should expect a D-Authentication Demand message after sending the U-Authentication Result.

Direction: RPDI to MS;
Service used: MM;
Response to: U-Authentication Demand;
Response expected: U-Authentication Result.

Table 6: D-Authentication Response PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Response value	32	M		
Random seed	80	M		
Mutual Authentication Flag	1	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

U-Authentication Result: shall be used by the MS to notify the RPDI about the result of the authentication exchange.

Direction: MS to RPDI;
Service used: MM;
Response to: D-Authentication Response;
Response expected: D-Commonkey Provide or D-Authentication Demand (depending on whether the Mutual Authentication Flag was set).

Table 7: U-Authentication Result PDU contents

Information element	Element length	Element type	Reference	Remark
Message identifier	4	M		
Authentication result	1	M		
KEY: M: Mandatory; C: Conditional; O: Optional; -: Not used				

4.2.4 Protocol sequences

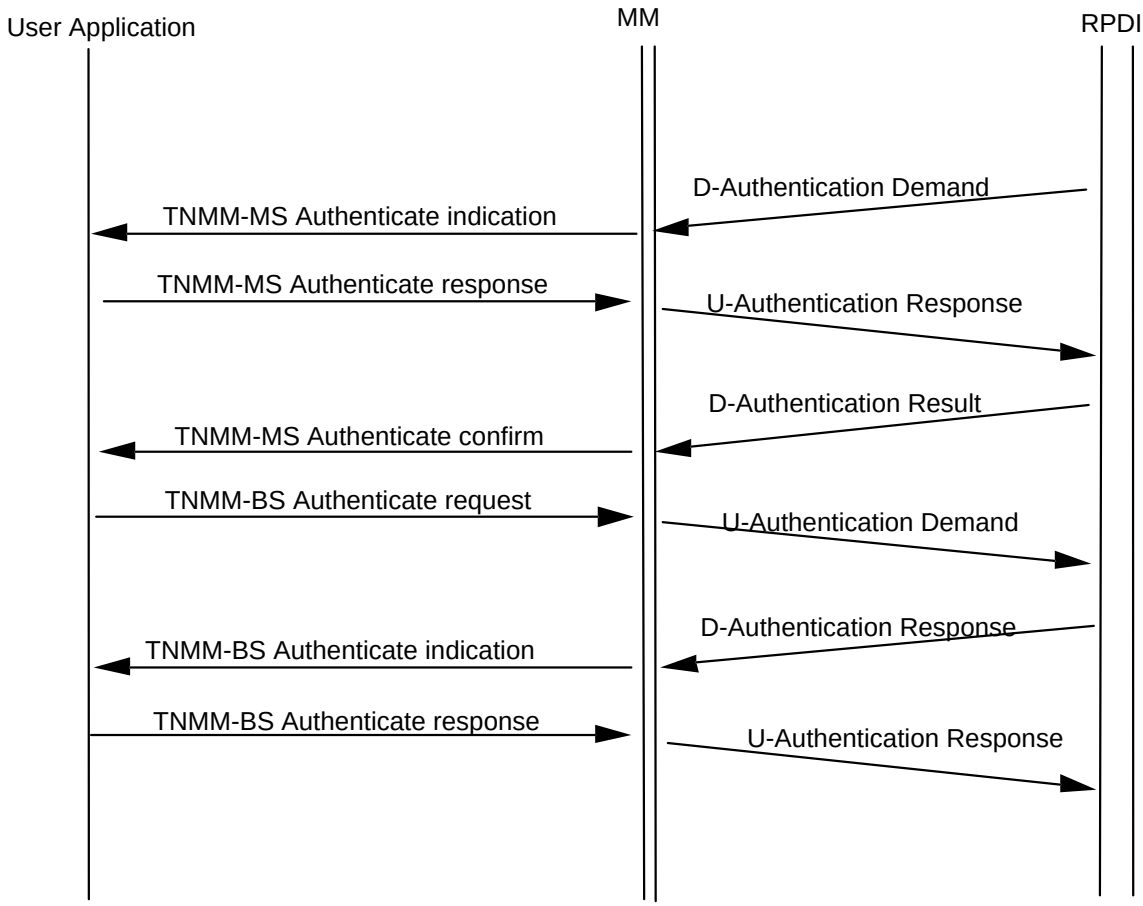

Figure 4: Protocol sequence of authentication without registration

Figure 4 shows the protocol sequence for mutual authentication in the case where the process is started by the infrastructure, not within the context of a registration procedure. Figure 5 shows the protocol sequence within the context of a registration procedure.

Figures 4 and 5 show the most general and complete examples in the sense of mutual authentication. By deleting the respective messages, sub-sequences can be generated that apply to more special cases.

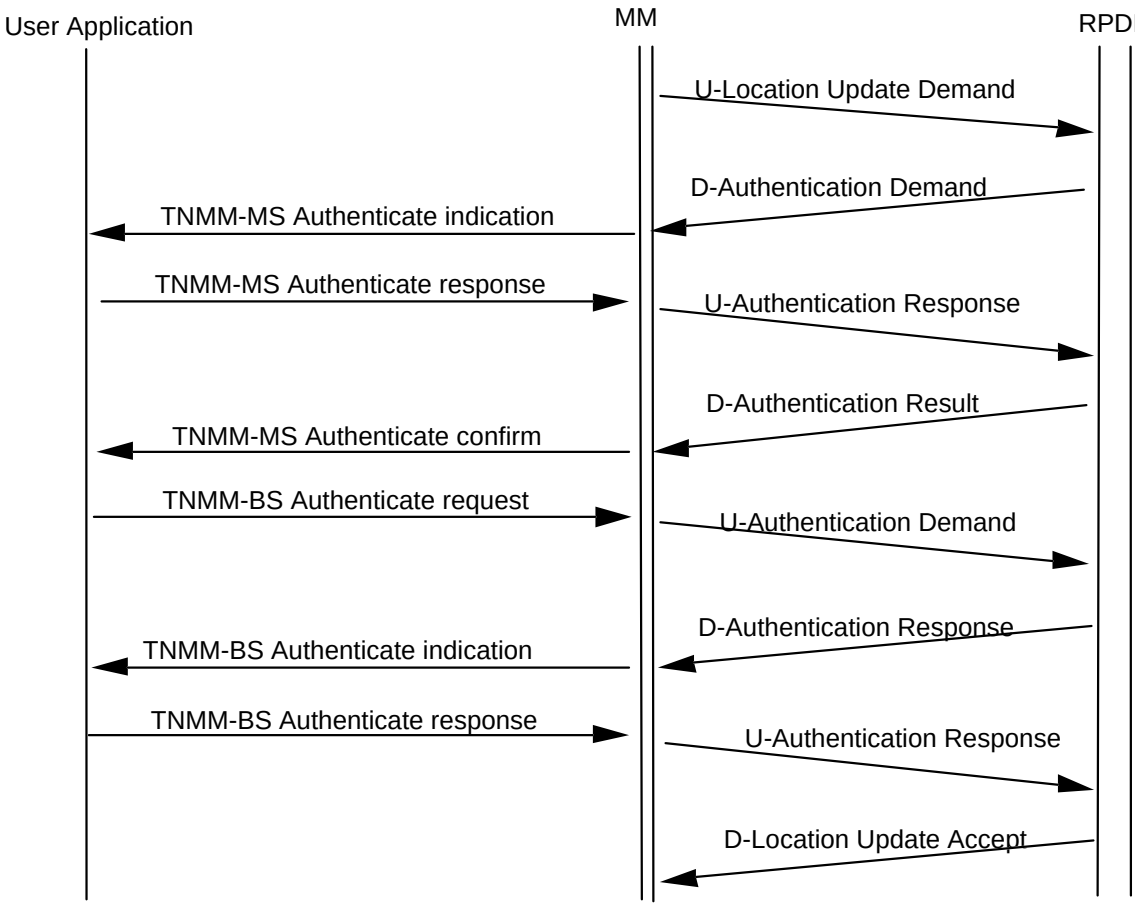

Figure 5: Protocol sequence of authentication registration

4.3 Boundary conditions for the cryptographic algorithms and procedures

In the following the symbol |XYZ| shall be used to denote the length of the parameter XYZ. If the length of a parameter can vary, |XYZ| denotes the range between the shortest and the longest possible values for XYZ.

TA11: shall be used to compute the session authentication key KS from the authentication key K and the random seed RS. The algorithm shall have the following properties:

Input 1: Bit string of length |K|;
Input 2: Bit string of length |RS|;

Output: Bit string of length |KS|.

The algorithm should be designed such that it is difficult to infer any information about Input 1 from the knowledge of Input 2 and the Output (even if the details of the algorithm are known).

TA21: shall be used to compute the session authentication key KS' from the authentication key K and the random seed RS. The algorithm shall have the following properties:

Input 1: Bit string of length |K|;
Input 2: Bit string of length |RS|;

Output: Bit string of length |KS'|.

The algorithm should be designed such that it is difficult to infer any information about Input 1 from the knowledge of Input 2 and the Output (even if the details of the algorithm are known).

TA12P: shall be used to compute the (expected) response (X)RES1 from KS and the challenge RAND1. The algorithm shall have the following properties:

Input 1: Bit string of length |KS|;
Input 2: Bit string of length |RAND1|;

Output 1: Bit string of length |(X)RES1|.

The algorithm should be designed such that it is difficult to infer any information about Input 1 from the knowledge of Input 2 and Output 1 (even if the details of the algorithm are known).

TA22P: shall be used to compute the (expected) response (X)RES2 from KS' and the challenge RAND2. The algorithm shall have the following properties:

Input 1: Bit string of length |KS'|;
Input 2: Bit string of length |RAND2|;

Output 1: Bit string of length |(X)RES2|.

The algorithm should be designed such that it is difficult to infer any information about Input 1 from the knowledge of Input 2 and Output 1 (even if the details of the algorithm are known).

TB1: shall be used to compute K from the AC. The algorithm shall have the following properties:

Input: Bit string of length |AC|;

Output: Bit string of length |K|.

The algorithm should be designed such that the Output is dependent on every bit of the Input.

TB2: shall be used to compute K from UAK. The algorithm shall have the following properties:

Input: Bit string of length |UAK|;

Output: Bit string of length |K|.

The algorithm should be designed such that the Output is dependent on every bit of the Input.

TB3: shall be used to compute K from UAK and PIN. The algorithm shall have the following properties:

Input 1: Bit string of length |PIN|;
Input 2: Bit string of length |UAK|;

Output: Bit string of length |K|.

The algorithm should be designed such that the Output is dependent on every bit of both Inputs.

4.4 Dimensioning of the cryptographic parameters

Table 8: Dimensioning of cryptographic parameters

Abbreviation	Name	No. of Bits
AC	Authentication Code:	16 - 32 bit
K	Authentication key:	128 bit
KS	Session authentication Key:	128 bit
KS'	Session authentication Key:	128 bit
PIN	Personal Identification Number:	16 - 32 bit
RAND1	Random challenge 1:	80 bit
RAND2	Random challenge 2:	80 bit
RES1	Response 1:	32 bit
RES2	Response 2:	32 bit
RS	Random Seed:	80 bit
UAK	User authentication key:	128 bit
XRES1	Expected response 1:	32 bit
XRES2	Expected response 2:	32 bit

4.5 Summary of the cryptographic processes

Figure 6 gives a summary of the authentication mechanisms explained in the previous subclauses.

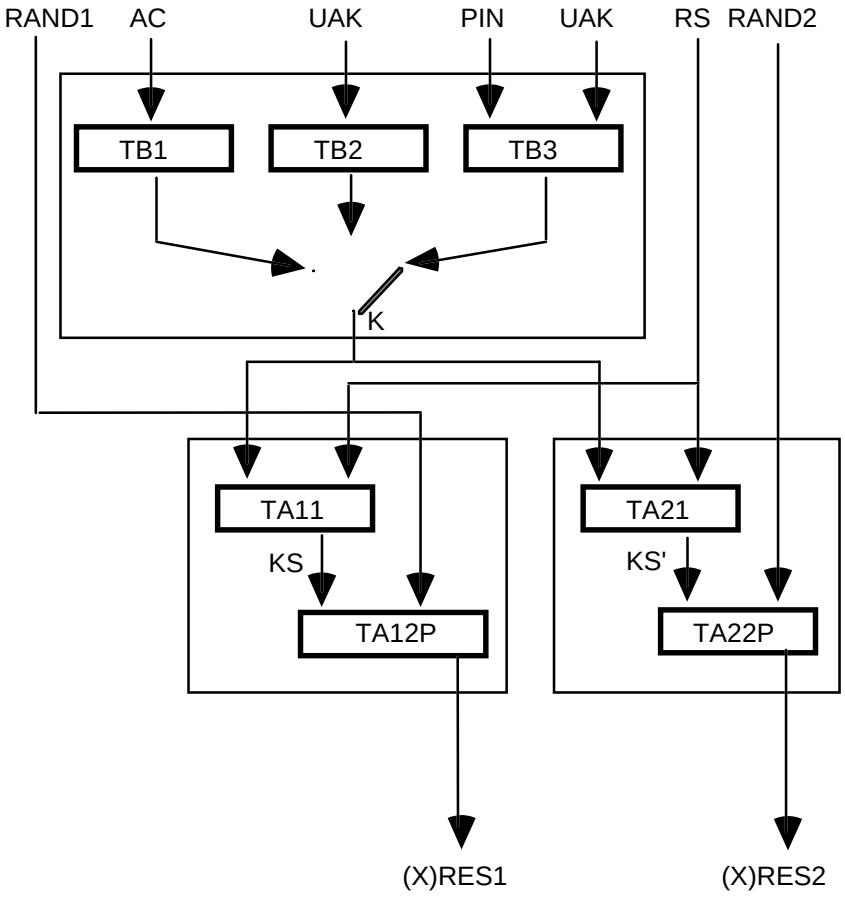

Figure 6: Overview of air interface authentication and key management

History

Document history			
July 1996	Public Enquiry	PE 110:	1996-07-22 to 1996-11-15