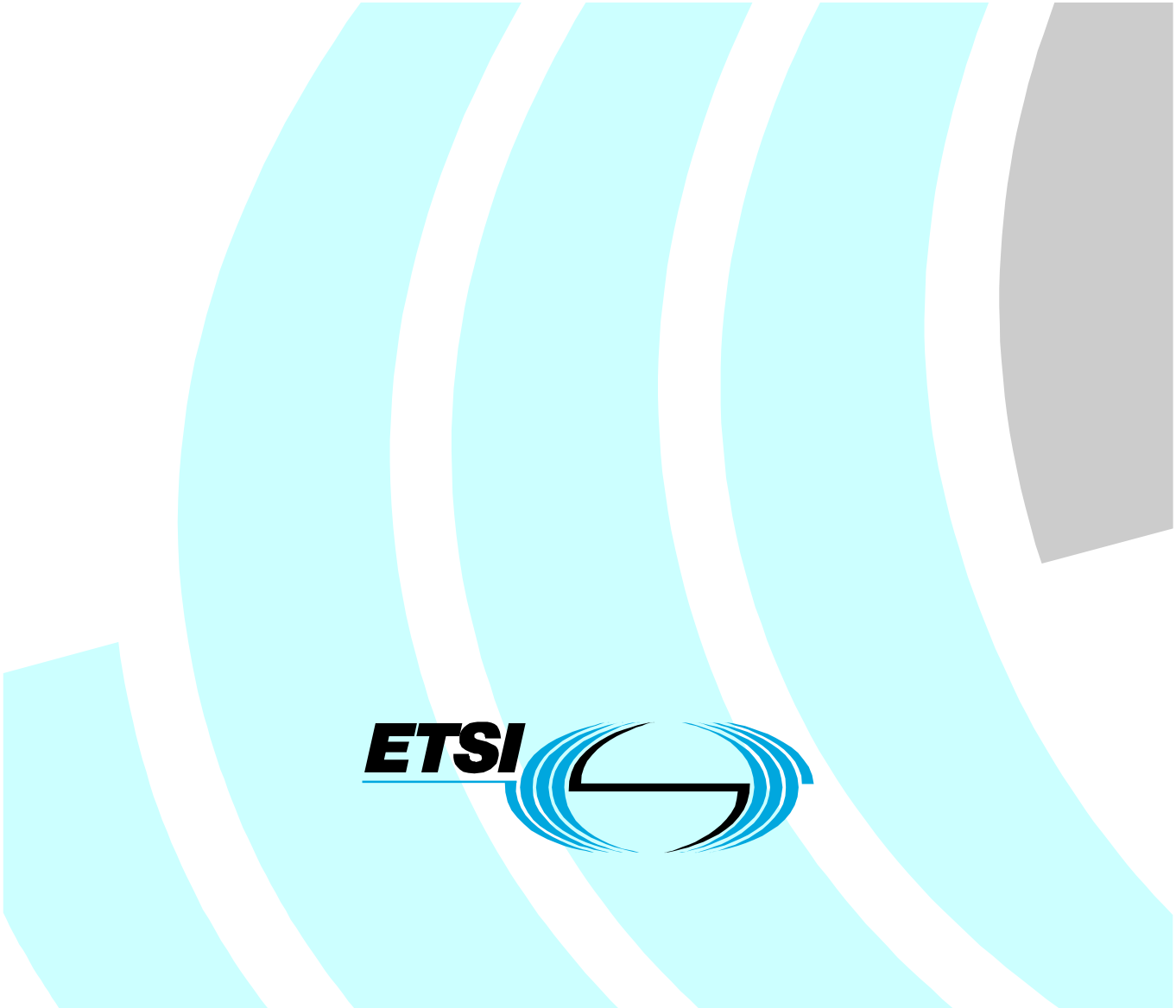
ETSI TS 102 310 V6.1.0 (2005-02)

Technical Specification

Smart Cards;
Extensible Authentication Protocol support in the UICC
(Release 6)



Reference
DTS/SCP-T0013r1

Keywords
card, protocol, security

ETSI

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

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

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

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

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

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Contents

Intellectual Property Rights 4
Foreword 4
1 Scope 5
2 References 5
3 Definitions and abbreviations 6
3.1 Definitions 6
3.2 Abbreviations 6
4 Introduction 6
5 Architecture 7
5.1 Architectural Principles 7
5.2 EAP clients discovery 7
5.3 EAP-capable-application selection 8
5.4 Key derivation 8
5.5 Authentication Status 9
6 EAP related Commands 9
6.1 EAP Authenticate 9
6.1.1 Command description 9
6.1.1.1 Command parameters and data 10
6.2 Specific status conditions returned 11
6.2.1 Status words 11
7 EAP Files 11
7.1 $EF_{EAPKEYS}$ (EAP derived keys) 11
7.2 $EF_{EAPSTATUS}$ (EAP Authentication STATUS) 12
7.3 EF_{PUId} (Permanent User Identity) 13
7.4 EF_{Ps} (Pseudonym) 13
Annex A (informative): Change history 14
History 15

Intellectual Property Rights

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

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

Foreword

This Technical Specification (TS) has been produced by ETSI Project Smart Card Platform (SCP).

The contents of the present document are subject to continuing work within EP SCP and may change following formal EP SCP approval. If EP SCP modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

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

1 Scope

The present document defines additional features that shall be provided by the UICC to support EAP authentication capabilities.

The goal of these new features is to adapt the UICC to provide support of different EAP methods, ensuring interoperability between the UICC and any terminal independently of their respective manufacturers.

The present document defines:

- The architectural framework.
- The additional commands required.

2 References

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

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

[1] IETF RFC 3748: "Extensible Authentication Protocol (EAP)". (http://www.ietf.org/rfc/rfc3748.txt).

[2] IETF RFC 2284: "PPP Extensible Authentication Protocol (EAP)". (http://www.ietf.org/rfc/rfc2284.txt).

[3] ETSI TS 102 221: "Smart cards; UICC-Terminal interface; Physical and logical characteristics (Release 7)".

[4] IETF RFC 2716: "PPP EAP TLS Authentication Protocol". (http://www.ietf.org/rfc/rfc2716.txt).

[5] IETF RFC 2486: "The Network Access Identifier". (http://www.ietf.org/rfc/rfc2486.txt).

[6] IETF RFC 2661: "Layer Two Tunneling Protocol". (http://www.ietf.org/rfc/rfc2661.txt).

[7] IETF RFC 1661: "The Point-to-Point Protocol (PPP)". (http://www.ietf.org/rfc/rfc1661.txt).

[8] IEEE Std 802.1X-2001: "IEEE Standard for Local and metropolitan area networks-Port-Based Network Access Control".

[9] IEEE Std 802.11-1999: "Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE, Sept. 1999".

3 Definitions and abbreviations

3.1 Definitions

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

authenticator: end of the EAP link initiating EAP authentication

peer or supplicant: end of the EAP Link that responds to the authenticator

3.2 Abbreviations

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

DF	Dedicated File
EAP	Extensible Authentication Protocol
EF	Elementary File
L2TP	Layer Two Tunnelling Protocol
LAN	Local Area Network
NAI	Network Access Identifier
PPP	Point-to-Point Protocol
TLS	Transport Layer Security
USIM	Universal Subscriber Identity Module
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network

4 Introduction

The Extensible Authentication Protocol is a general authentication framework, which supports multiple authentication methods. EAP typically may run directly over data link layers such as PPP (see RFC 1661 [7]) or IEEE 802 (see Std 802.1X-2001 [8] and 802.11-1999 [9]).

As described in RFC 3748 [1], EAP implementations consist of three main components:

- A **lower layer** that is responsible for transmitting and receiving EAP frames between the peer and the authenticator. EAP has been run over a variety of lower layers (including PPP, IEEE 802 LANs, IEEE 802.11 WLANs, and L2TP (see RFC 2661 [6])).
- An **EAP layer** that receives and transmits EAP packets via the lower layer, implements duplicate detection and retransmission, and delivers and receives EAP messages to and from EAP methods.
- **EAP methods** that implement the authentication algorithms and receive/transmit EAP messages via the EAP layer.

The UICC offers suitable possibilities for the implementation of some of these EAP methods in the peer side, since it provides the required protection of credentials and authentication algorithms. This is even more important when the following conditions apply:

- The authentication methods require the usage of credentials that are stored in the UICC.
- For security reasons, these credentials shall not be revealed in clear in an unprotected peer environment (e.g. a laptop or mobile terminal).

The present document defines the principles that shall be implemented in the UICC in order to enable that UICC applications may support one or more of these EAP methods.

5 Architecture

5.1 Architectural Principles

The following architectural principles are applied:

- The authenticator is able to perform an EAP authentication process (using an specific EAP method) with a UICC application implementing this method. That means that the authentication is performed end-to-end between the authenticator and the UICC application.
- The peer is composed of several components:
  - **The UICC EAP Framework** provides information to the terminal about the existing UICC applications that provide UICC EAP clients.
  - A **UICC application** provides one or more UICC EAP clients.
  - A **UICC EAP client** implements one specific EAP method.

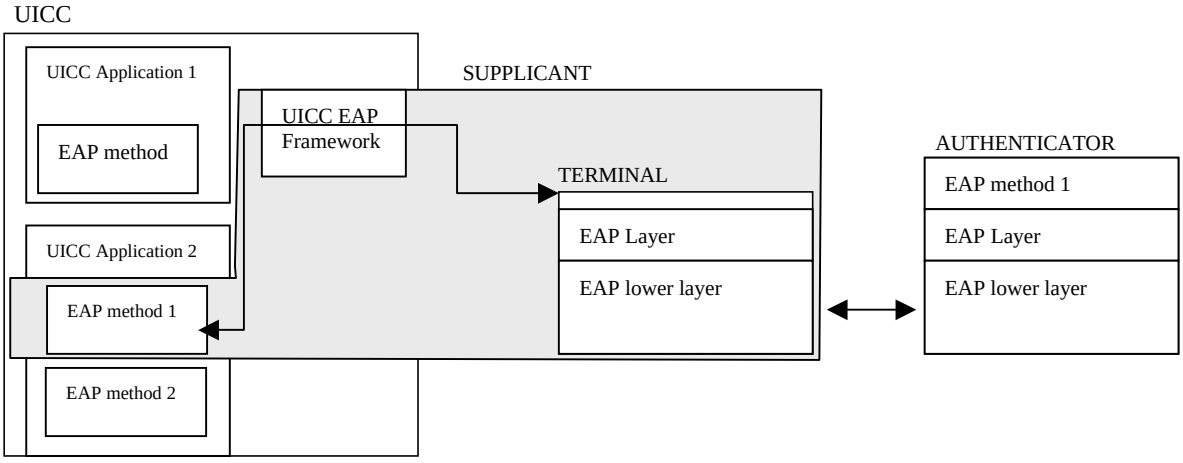


Figure 5.1: EAP architecture when supplicant is split between a UICC and a terminal

5.2 EAP clients discovery

When a UICC application implements one or more EAP clients, its corresponding record in EF_{DIR} shall contain the following EAP related Data Objects:

- Application EAP support types list: defining the EAP methods supported by the corresponding UICC application.
- Application EAP Dedicated File list: defining a list of Dedicated Files, each of them associated to one supported EAP method. Likewise, each EAP method is associated to one DF. Each of this DF are hereafter referred as DF_{EAP}.
- Application EAP Label: Defining a user readable label defining the EAP clients.

Table 5.1: Coding of EAP related DOs

Length	Description	Status
1	Discretionary template tag = "73"	M
1	Length of the discretionary template	M
1	EAP Application service specific data content tag (="A0")	M
1	EAP Application service specific data content length	M
1	Application EAP supported types list tag = "80"	M
1	Length of the Application EAP supported types list	M
A	Application EAP supported types list	M
1	Application EAP Dedicated file list tag = "81"	M
1	Length of Application EAP Dedicated file list	M
B	Application EAP Dedicated File list	M
1	Application EAP Label tag = "82"	M
1	Length of the Application EAP Label	M
C	Application EAP Label	M

Coding:

- Application EAP supported types list.

Contain a list of supported EAP type (as defined in RFC 3748 [1]) each of them coded in one byte except for expanded types that are coded on 8 bytes.

EXAMPLE 1: An UICC application supporting EAP-MD5 (see RFC 3748 [1]) and EAP-TLS (see RFC 2716 [4]) provides the following "Application EAP supported types list":

- "040D"corresponding to EAP-MD5 (Type=4) and EAP-TLS (Type=13).

- Application EAP Dedicated Files list:

Contain a list of file identifiers of each DF_{EAP} associated to a particular supported EAP type. Each of them coded in two bytes.

EXAMPLE 2: Using the previous example, A DF "6D34" for EAP-MD5 and a DF "6D35" for EAP-TLS will result in the following EAP Dedicated Files list:

- "6D346D35".

- Application EAP label:

The application label is a DO that contains a string of bytes provided by the application provider to be shown to the user for information.

5.3 EAP-capable-application selection

The terminal shall use the information in EF_{DIR} file if available to present the list of EAP-capable applications to the user or to any application that may request an EAP authentication.

The terminal shall then select the corresponding EAP-capable-application to start an EAP authentication. Once selected, all EAP-Client state machines of the application are reset.

5.4 Key derivation

It is possible for many EAP methods to derive key material after successful authentications. These keys may be used for subsequent processes (e.g. for WEP encryption in IEEE 802.11 [9]).

Keys derived from an authentication shall be retrieved by the terminal by inspecting the mandatory file $EF_{EAPKEYS}$.

5.5 Authentication Status

The terminal may retrieve the authentication status of the EAP client in the selected UICC application by inspecting the mandatory file $EF_{EAPSTATUS}$.

6 EAP related Commands

The following clauses specify the additional commands needed to implement the EAP framework in the UICC.

6.1 EAP Authenticate

6.1.1 Command description

The function is used to transfer the EAP packets from the terminal to the selected UICC EAP client (i.e. EAP client in the selected UICC application that corresponds to the given EAP type).

The UICC EAP client shall provide a response EAP packet (as defined in RFC 3748 [1]) or a warning status word according to the authentication method being used.

The UICC EAP client shall maintain the state machine of the authentication process as described for the particular EAP method used.

The function is related to a particular UICC application supporting EAP and shall not be executable unless this application has been selected and activated, and the current directory is a DF_{EAP} corresponding to a specific EAP method. The correspondence between EAP type and the current DF_{EAP} is defined in EF_{DIR}.

Each UICC application implementing a UICC EAP client may require different security conditions to execute this command (e.g. user PIN verification).

The format of the EAP packet is defined by the application implementing the EAP client and shall respect the conventions corresponding for the EAP method.

The following EAP packets are allowed input packets for this command: EAP packets with code field equal to 1 "Request", 3 "Success" or 4 "Failure" and EAP packets with code equal to 2 "Response" for EAP type 1 "Identity" (Code and type values as defined in RFC 3748 [1]).

NOTE: EAP Response Identity packet may be delivered to the UICC application when the identity is managed outside the UICC application and the method itself needs to have access to the chosen identity.

The command and response data may contain specific EAP method related data as an additional input/output parameter (e.g. gmt_unix_time for EAP-TLS implementations as defined in RFC 2716 [4]).

Input:

- EAP Packet;
- EAP method related data.

Output:

- Either none (i.e. if authentication successful: EAP success packet received).

Or:

- EAP Response Packet;
- EAP method related data.

6.1.1.1 Command parameters and data

Code	Value
CLA	As specified in ETSI SCP 102 221 [3]
INS	"88"
P1	"00"
P2	See table 6.1
Lc	Length of subsequent EAP command data
Data	See below
Le	Length of the response data

Table 6.1: Coding of P2

b8	b7	b6	b5	b4	b3	b2	B1	Meaning
1	-	-	-	-	-	-	-	Specific reference data (DF $_{EAP}$ application dependent KEY)
-	X	X	-	-	-	-	-	"00" (other values are RFU)
-	-	-	X	X	X	X	X	Reference data number ("01" to "1F")
NOTE: The reference data number assignment rule shall be defined in the application specification referencing the present document.								

NOTE 1: The reference data number assignment rule shall be defined in the application specification referencing the present document.

Command data:

Byte(s)	Description	Length
1 - Lc	EAP command data (see table 6.2)	Lc

Table 6.2: Coding of EAP command data

Byte(s)	Description	Status	Length
1-J	EAP packet (coded as defined for the method of EAP used as defined in RFC 3748 [1])	M	J bytes
J+1-J+K+1	EAP method related data (must be specified by each application specific document defining a particular EAP method implementation)	O	K bytes
NOTE: The length of an EAP packet is contained within the packet and can therefore be retrieved from it.			

NOTE 2: The length of an EAP packet is contained within the packet and can therefore be retrieved from it.

Response data:

Byte(s)	Description	Length
1 - Le	EAP Packet Response Data (see table 6.3)	Le

Table 6.3: Coding of EAP Response data

Byte(s)	Description	Status	Length
1-L	EAP packet	M	L bytes
L+1-L+N+1	EAP method related data (must be specified by each application specific document defining a particular EAP method implementation)	O	N bytes
NOTE: The length of an EAP packet is contained within the packet and can therefore be retrieved from it.			

NOTE 3: The length of an EAP packet is contained within the packet and can therefore be retrieved from it.

6.2 Specific status conditions returned

This clause specifies the coding of the specific status words SW1 and SW2.

6.2.1 Status words

The following table shows the meaning of possible status conditions returned.

Table 6.4: Status byte coding - warnings

SW1	SW2	Description
"62"	"00"	- No information given, state of non volatile memory unchanged (EAP Packet silently ignored)

Table 6.5: Status byte coding - application errors

SW1	SW2	Description
"98"	"62"	- Authentication error (EAP Failure Packet received)

7 EAP Files

This clause describes the files present in an application supporting an EAP type. The following files are situated under the corresponding DF_{EAP} of a particular UICC application.

7.1 $EF_{EAPKEYS}$ (EAP derived keys)

This EF contains the key material derived after a successful EAP authentication.

Structure of $EF_{EAPKEYS}$

Identifier: "4F01"	Structure: transparent		Conditional (see note)
File size: n		Update activity: high	
Access Conditions: READ PIN UPDATE ADM/NEVER DEACTIVATE ADM ACTIVATE ADM			
Bytes	**Description**	**M/O**	**Length**
1	1st Key Tag	O	1 bytes
2	1st Key Length	O	1 bytes
3 to L1+2	1st Key Value	O	L1 bytes
	...		
2(k-1) + L1 + + L(k-1) + 1	Kst Key Tag	O	1 bytes
2(k-1) + L1 + + L(k-1) + 2	Kst Key Length	O	1 bytes
(2(k-1) + L1 + + L(k-1) + 3) to (L1 + … + Lk + 2k)	Kst Key Value	O	LK bytes
NOTE: The presence of this file depends on the supported EAP method.			

NOTE: The presence of this file depends on the supported EAP method.

- Key Tag
  - Contents:
    - Identifier of the derived key.
  - Coding:
    - The assigned Key tag values are given in the following table.

Derived key description	Key tag value	Reference
Master Session Key (MSK)	"80"	RFC 3748 [1]
Extended Master Session Key (EMSK)	"81"	RFC 3748 [1]

- Key Length
  - Contents:
    - Length of the derived key.
- Key Value
  - Contents:
    - Derived key.

7.2 $EF_{EAPSTATUS}$ (EAP Authentication STATUS)

This EF contains the authentication status corresponding to the EAP client supported by the application.

Structure of $EF_{EAPSTATUS}$

Identifier: "4F02"		Structure: transparent		Mandatory
File size: 1		Update activity: high		
Access Conditions: READ PIN UPDATE ADM/NEVER DEACTIVATE ADM ACTIVATE ADM				
Bytes	Description		M/O	Length
1	Authentication Status		M	1 bytes

- Authentication Status
  - Contents:
    - Status of the corresponding EAP authentication.
  - Coding:
    - Authentication Status coded in one byte as below.

Value	Meaning
"00"	No authentication started
"01"	Authenticating
"02"	Authenticated
"03"	Held (Authentication failure)

7.3 EF_{PUId} (Permanent User Identity)

This EF contains the permanent user identity. Permanent User identity may be used as the username part of the Network Access Identifier (see RFC 2486 [5]).

This File is not mandatory if the Permanent user identity is derived by other means.

Structure of EF_{PUId}

Identifier: "4F03"	Structure: transparent		Optional
File size: n (where n ≥10 bytes)		Update activity: low	
Access Conditions: READ PIN UPDATE ADM DEACTIVATE ADM ACTIVATE ADM			
Bytes	**Description**	**M/O**	**Length**
1 to n	Permanent user identity	M	n bytes

- Permanent user identity.
  - Contents:
    - user identity to be used as the username part of the NAI.
  - Coding:
    - Binary. Unused bytes shall be set to "FF" and shall not be considered as a part of the value.

7.4 EF_{PS} (Pseudonym)

This EF contains a temporary user identifier (pseudonym) for subscriber identification. Pseudonyms may be provided as part of a previous authentication sequence. This may be used as the username part of the Network Access Identifier (see RFC 2486 [5]).

This file is not mandatory if pseudonyms are not managed by the application or they are derived by other means.

Structure of EF_{Ps}

Identifier: "4F04"	Structure: transparent		Optional
File size: n		Update activity: high	
Access Conditions: READ PIN UPDATE PIN DEACTIVATE ADM ACTIVATE ADM			
Bytes	**Description**	**M/O**	**Length**
1 to n	Pseudonym	M	n bytes

- Pseudonym.
  - Contents:
    - Pseudonym to be used as the username part of the NAI.
  - Coding:
    - Binary. Unused bytes shall be set to "FF" and shall not be considered as a part of the value.

Annex A (informative): Change history

This annex lists all change requests approved for the present document by ETSI SCP.

SCP#	SCP tdoc	VERS	CR	RV	CAT	SUBJECT	Resulting Version
19	SCP-040420	6.0.0	001		F	Clarification on references and clarification on the coding of P2 authenticate parameter	6.1.0
			002		F	Allocation of new tag values for EAP	

History

Document history		
V6.0.0	December 2004	Publication
V6.1.0	February 2005	Publication