3GPP TSG SA WG3 Security — S3#28 **S3-030187**
6 - 9 May 2003
Berlin, Germany

TSG-SA WG1 #20 **S1-030546**
Seoul, Korea, 7th–11th April 2003 **Agenda Item: WLAN**

Title:	Reply LS on 'Request for Information Regarding WLAN Interworking Impacts to UICC applications'
Response to:	LS (S1-030320 / T3-030116) on 'WLAN Interworking Impacts to UICC applications' from T3
Release:	
Work Item:	
Source:	SA1
To:	T3, SA3
Cc:	SA2, EP SCP

Contact Persons:

Name:	Jan Ignatius	Christophe Dubois
Tel. Number:		
E-mail Address:	jan.ignatius@nokia.com	christophe.dubois@slb.com

Attachments: draft-urien-eap-smartcard-01.txt

---

1. Overall Description:

SA1 thanks T3 for their LS requesting guidance on the need for a work item and on its content to support WLAN interworking.

There are requests from operators for a secured SIM based WLAN authentication solution, but SA1 thinks that it could be difficult to achieve, considering the short time-frame and a possible impact to frozen releases (Rel-4 and Rel-5).

At the moment WLAN authentication based on legacy SIMs can be done via proprietary methods or EAP SIM, but with security issues raised by SA3 (S1-030329 / S3-030161). If USIM is utilized for WLAN authentication EAP AKA is used.

Therefore, SA1 foresee a need for T3 to start a work item to support the WLAN interworking efforts.
SA1 would like T3 to focus on a standardised solution for a secured WLAN authentication based on (U)SIM.

SA1 is aware of the IETF proposal "EAP-Support in smartcard" attached within the present Liaison, which T3 may wish to consider.

SA1 would like to note that these enhancements shall be considered as optional and should not produce impacts in the authentication mechanisms already proposed for WLAN inter-working.

2. Actions:

To T3 and SA3 group.

ACTION: SA1 asks SA3 and T3 groups to make efforts to find a standardised solution for a secured WLAN authentication based on (U)SIM.

3. Date of Next TSG SA WG1 Meetings:

Title	Date	Location	Country
SA1#21 SWG	May 12-16, 2003	San Diego	USA
SA1#21	July 7-11, 2003	Sophia Antipolis	France
SA1#22	October 27-31, 2003	Asia	TBD

Internet Draft P.Urien
Document: draft-urien-eap-smartcard-01.txt A.J. Farrugia
M.Groot
G.Pujolle
Expires: August 2003

EAP-Support in smartcard

Status

Abstract

This document will describe the interface to the EAP protocol in smartcards, which could store multiple identities associated to Network Access Identifiers.

Table of Contents

Overview

All technologies derived from 802.11 specifications such as 802.11a, 802.11b, 802.11g need a strong security protocols for data privacy, integrity and network access. Where the 802.1X [8] specification describes the risks and the protocols for the protection of the exchanged data during the network connection. The very same

specification remains compatible with other standard for the authentication and the network access.
802.1X specification requires the Extensible Authentication Protocol (EAP) to be used as the framework for application dependent authentication processes with a mutual authentication between the supplicant and the authenticator. It is obvious that the role of the supplicant in this specification has partly been implemented in the smart card has an authentication processing mean. The flexibility of EAP (RFC 2284) specification does not provide a Mandatory-to-implement solution. The structure of the EAP frames allows the applications to identify the EAP type of consequently to operate the appropriate authentication.

Terms

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

Authentication Agent: A piece of software implemented in the supplicant that processes the authentication sequence.

AS
Authentication Server

Authenticator: See the IEEE 802.1X specification for a definition of this concept.

EAP
Extensible Authentication Protocol.

GSM
Global System for Mobile communications.

IMSI
International Mobile Subscriber Identifier, used in GSM to

identify subscribers.

NAI
Network Access Identifier

PMK
Pairwise Master Key

SIM
Subscriber Identity Mobile

Supplicant: an IEEE 802.1X concept, which in the context of IEEE 802.11 represents a STA (station) seeking to attach to an IEEE 802 LAN via an IEEE 802.1X Port. See the IEEE 802.1X specification for a complete definition

Identification label

802.1X specification [5] requires an authentication between the authenticator or the authentication server (AS) and the supplicant. The authentication is embedded in the Extensible Authentication Protocol (EAP) RFC2284 [1] specification. The authentication consists of a challenge response between both parties without consideration of the involved crypto-suite. Before starting the mutual authentication, the AS needs the supplicant identity to establish the session. The AS or the authenticator sends an EAP Request Identity to the supplicant that returns its system identity. A user may own several identities likely associated to the network operators.

The identification label is a pointer to a system identity stored in smartcard; it may be of various types:

1. A network SSID as described in the 802.11 standard [4].
2. A userÆs identification (userid) e.g. an ASCII string. A network access identifier, NAI [6] may be used as userid.
3. A pseudonym, e.g. a friendly name.

According to the network environment, the supplicant software needs to set the appropriate identity and verifies if the smart card is able to mirror the authenticator.

If the smart card is not able to process the authentication related to the identity then any setting process is rejected by the NAK code.

The subsequent sections give the description of the methods used by a supplicant for processing an 802.1X authentication using the smart card.

Annex one provides a reference implementation example for a SIM based authentication. Annex two provides a reference implementation example for a MD5 based authentication. Annex three provides a

reference implementation for a TLS based authentication.

Identification Label Coding Rules

The Get-Next-Identity section didnÆt define the coding rules of the identification label. This section describes the structure and the architecture of the userid.

A userid consists of 2 fields separated by the Internet symbol "@". The right hand side of the "@" symbol is the userid realms while the left hand side is an application dependent and unique identification number. EAP/SIM has defined the userid where the application identification is "1IMSI". Other userid such as email address can be used by the application.

Add-Identity

This command and the Delete-Identity are part of the userÆs identity management protocols. The smart card is initially manufactured without any identification label. The personalization or the supplicant software adds in the smart card userÆs identification label that can be retrieved by other smart card command.

If the smart card manages pseudonyms the command does not allows setting the user pseudonyms. The smart card command only adds permanent identification label in the list.

Delete-Identity

This command and the add-Identity are part of the userÆs identity management protocols. The smart card contains a list of one or several identification labels that can be retrieved by the supplication software. The command deletes one entry of the smart card list.

Get-Preferred-Identity

The smart card contains at least one userÆs identity related to the userÆs network subscription. The supplicant software gets from the smart card the initial and preferred identification label. If the user has more than one identities the supplicant software uses the Get-Next-Identity to read all the available other userÆs identities. If the smart card manages pseudonyms and a pseudonym is available as preferred identity, the Get-Preferred-Identity shall return the pseudonym.

Get-Next-Identity

The smart card may contain one or more userÆs identities according to the userÆs network subscriptions. The supplicant software should prompt the userÆs identity and a subsequent selection allows the

smart card to process the appropriate EAP authentication type. The method Get-Next-Identity allows the supplicant software to read all the available userÆs identities.

The Get-Next-Identity method may inform the supplicant software when all userÆs identities have been read. Otherwise the supplicant software detects the identity list end when it gets again the first identity.

If the smart card contains a pseudonym management and the pseudonym is (are) available the Get-Next-Identity returns the appropriate pseudonym. If the pseudonym management is not supported, the smart card returns the permanent Identity according to the previous section.

Get-Subscriber-Profile

The Authentication Agent or the authenticator may request the subscriber profile information. The Get-Subscriber-Profile returns all related information available in the smart card. This specification does not provide the detail of the subscriber profile information. The implementation of the information may be ruled but ASN.1 BER coding specification [9] or by an XML dialect [10].

Set-Identity

Once the Identity selection is processed, the supplicant software needs to set the smart card EAP framework according to the selected userÆs identity. The Set-Identity sets or restarts the smart card EAP framework state machine for further processing using the EAP-Packets method.

The supplicant software can set the EAP framework using the pseudonym if available in the smart card. If the pseudonym is not available the supplicant software uses the permanent identity to set the EAP framework according to the previous section.

EAP-Packets

The EAP process is described in the RFC 2284 specification [1] and involves several EAP requests and responses packets,

1. EAP request/response Identity;
2. A suite of EAP request/response related to a particular authentication scenario; and
3. EAP success or failure.

The Set-Identity restarts the smart card EAP framework state machine for further processing using the EAP-Packets method.

The smart card receives the RFC 2284 frames. It retrieves the

appropriate EAP authentication type in the frame and the identifier. The smart card maintains the EAP state machine and returns an EAP NAK packet if the state sequence is broken. Any EAP request is silently ignored if the state machine was not started.

The last step of the protocol retrieving the Pairwise Master Key from the smart card can be accomplished only if the last EAP packet received from the authentication is an EAP success packet.

Get-Pairwise-Master-Key (PMK)

At the end of a successful authentication the supplicant needs to update the appropriate crypto suite using the master session key. The Get-Pairwise-Master-Key returns to the supplicant software the key to initialize radio security protocols like TKIP, WRAP or CCMP.

For obvious security reasons the Get-Pairwise-Master-Key is available only if the smart card has received an EAP success packet.

ISO 7816-4 APDUs

This section of the document provides an implementation of the previous descriptions for an ISO 78176-4 compatible smart card. The section does not preclude of the transport protocol used between the smart card and the reader. Thus, this specification does not mandate-to-implement any transport protocol such as T=0 or T=1, which are not in the scope of this document. It should be noted that all values are in hex representation.

The restriction and security related descriptions are not present in the document. Annexes of this document give implementation examples.

Add-Identity

This command adds an identification label as described in the section: Identification Label Coding Rules. The smart card list is managed by the smart card. The identification label is appended as the last element of the list if the parameter Po is 0x00. If this parameter has any value, it represents the identification label position.

Command	Class	INS	P1	P2	Lc	Le
	A0	16	81	Po	00	XX

Delete-Identity

This command deletes the identification label as described in the section: Identification Label Coding Rules. The command parameter gives the identification label to be deleted and the smart card

leave the space empty.

Command	Class	INS	P1	P2	Lc	Le
	A0	16	82	Po	00	00

Get-Preferred-Identity

This command returns the userÆs preferred identification label as described in the section: Identification Label Coding Rules

Command	Class	INS	P1	P2	Lc	Le
	A0	16	02	00	00	XX

Get-Next-Identity

This command returns a user identification label as described in the section: Identification Label Coding Rules.

Command	Class	INS	P1	P2	Lc	Le
	A0	16	01	00	00	XX

Get-Subscriber-Profile

The command returns the related subscriber profile information according to the application requirements and format.

Command	Class	INS	P1	P2	Lc	Le
	A0	16	08	00	00	YY

Set-Identity

The command resets and initializes the state machine for processing

the EAP Packets. The first step after this command is an EAP request identity packet. If a different EAP packet is sent to the smart card the smart card return an EAP NAK response.

Command	Class	INS	P1	P2	Lc	Le
EAP-Packets	A0	16	80	00	XX	00

The command is the method for EAP packet management. The smart card identifies the EAP packet type and processes the EAP authentication according to current state machine. The state machine sequences have

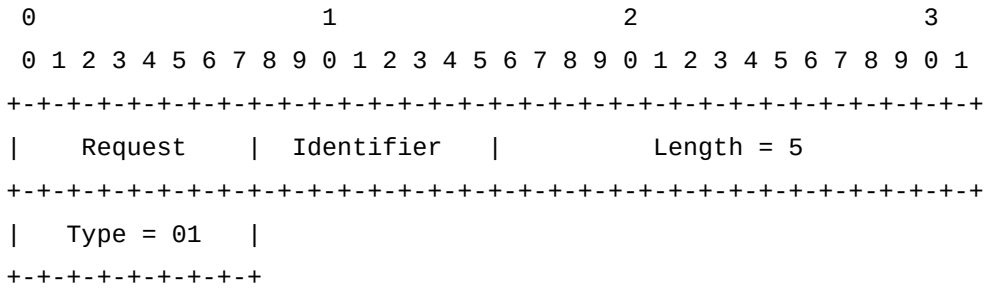
to be respected and the smart card enforces the EAP sequence processing.

Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	XX	YY

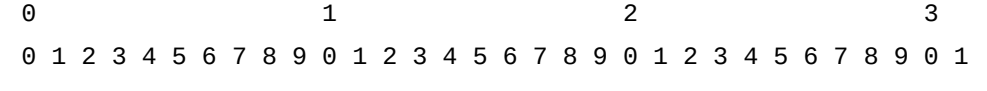
The EAP request or response packet lengths are represented by the unknown value XX and YY. The supplicant software should set these elements in accordance with the EAP packet types.

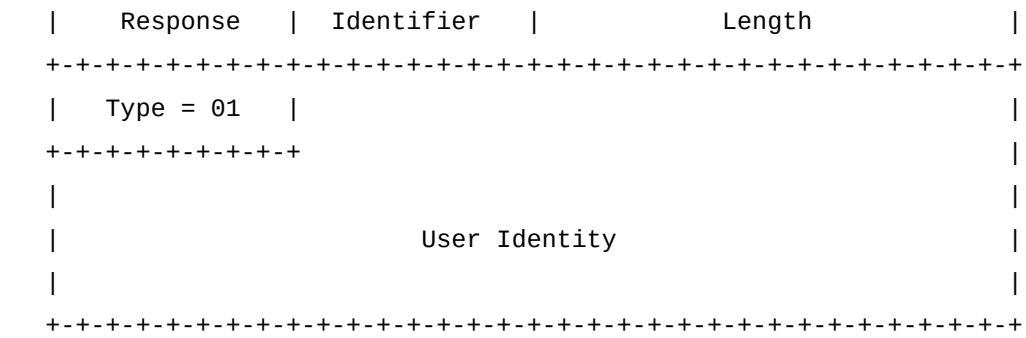
EAP Identity packets are independent of the authentication type and can be the same for any type of authentications. This section of the document provides the packet details. The rest of the EAP packet being authentication protocol dependent, they are detailed in the informative annex of this document.

The description of the EAP/Request/identity is detailed according to the IETF RFC 2284 [1].



The description of the EAP/Response/identity is detailed according to the IETF RFC 2284 [1].





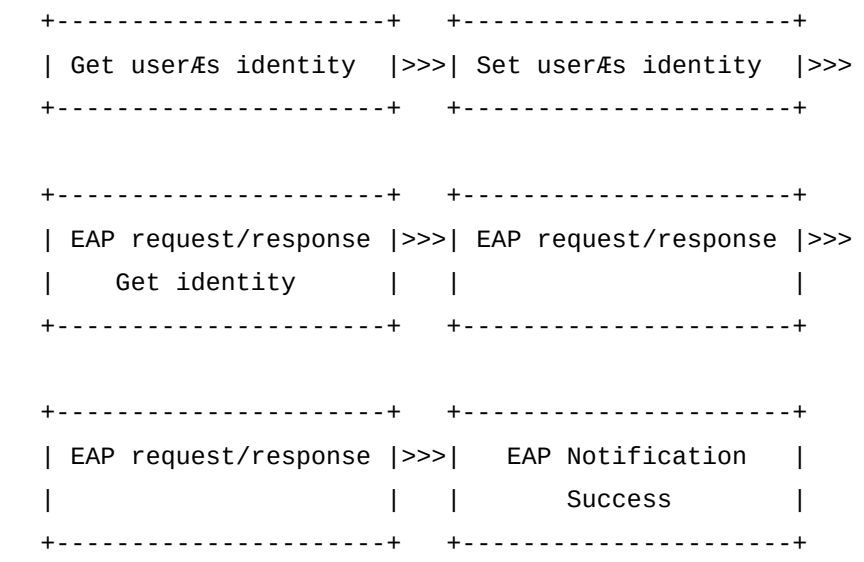
Get-Pairwise-Master-Key

Once the state machine has received the EAP Success packet the smartcard process is able to send the Master Key used by the 802.1X specification for the crypto-suite.

As an illustration the EAP SIM authentication [2] specifies the Pairwise Master Key usage according to the system cryptographic suite.

Command	Class	INS	P1	P2	Lc	Le
	A0	A6	00	00	00	20

State Machine Sequence



Security Considerations

General Considerations

As a reference implementation the previous section provides the details of the EAP authentication using the GSM SIM. This section of the document highlights the new potential risks providers of application may face by re-using deployed networks for other purposes. From the document [7] fatal flaw does exist when have physical access to the smart card.

The nature of the Internet network does no longer require getting physical access to the smart card. Worms, Trojan horses or viruses can move to the computing platforms and performs the jobs. It is important for a reference implementation to provide the relevant level of protection for the new applications but not to create other flaws.

Other consideration have been introduced in [2] to protect the smart card against crypto attack and recommends the authentication should take place in a PROTECTED ENVIRONMENT.

PEAP Consideration

Protected Extensible Authentication Protocol (PEAP) [12] is a pre-processing protocol that allows the privacy of data when processing EAP [1] protocol. EAP protocol, as defined in [1], starts by an EAP packet request/Identity. The EAP packet response Identity returns the userÆs identification label with no privacy being not part of [1].

PEAP protocol allows both part of the EAP packet exchange creating a session key that can be for privacy over the subsequent execution of the EAP protocol.

This implementation of EAP in the smart card shall allow performing a PEAP tunnel for privacy. Once PEAP first phase has been successfully preformed, the EAP protocol has defined shall be performed according the EAP smart card requirements.

Intellectual Property Right Notice

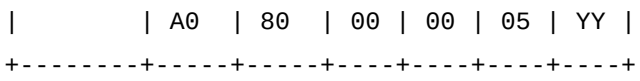
To be specify according to the author and participant.

Annex 1 (Informative) - EAP/SIM packet detail.

The protocol implementation is out of the scope of this document but as a reference implementation this section gives details using the SIM as specified by [3]. Other protocol can be implemented using ISO 7816-3 TPDU. This section of the document gives the APDU syntax and coding which makes the specification protocol free.

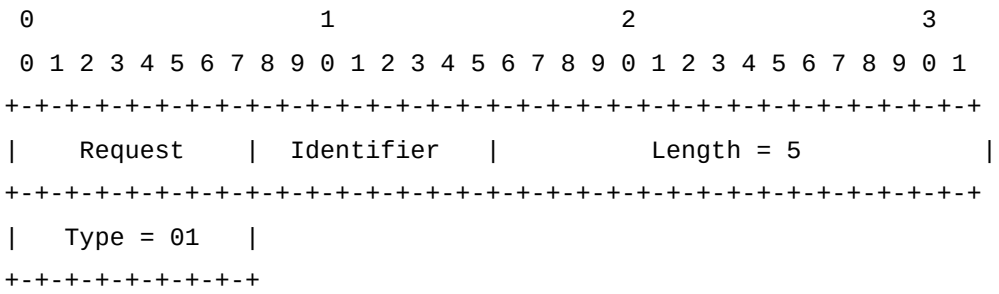
The first EAP packet is the EAP Request Identity. This initial packet format complies with [1]. The smart card returns an EAP response identity according to the IMSI length and the supported version according to [2].

Command	Class	INS	P1	P2	Lc	Le



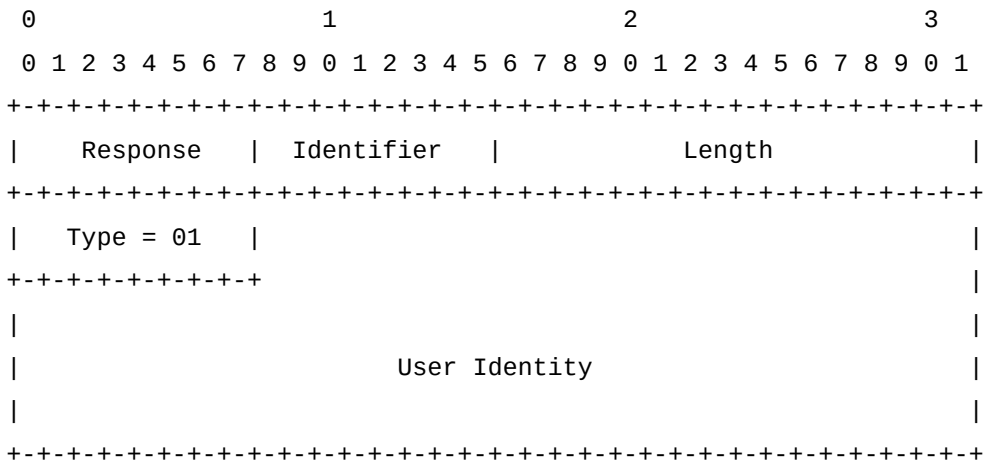
```
|        | A0  | 80  | 00 | 00 | 05 | YY |
+--------+-----+-----+----+----+----+----+
```

The description of the EAP/Request/identity is detailed according to the IETF RFC 2284 [1]. This EAP packet doesnÆt respect the EAP/SIM format since it is only part of [1].



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Request    |  Identifier   |          Length = 5           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = 01   |
+-+-+-+-+-+-+-+-+
```

The description of the EAP/Response/identity is detailed according to the IETF RFC 2284 [1].

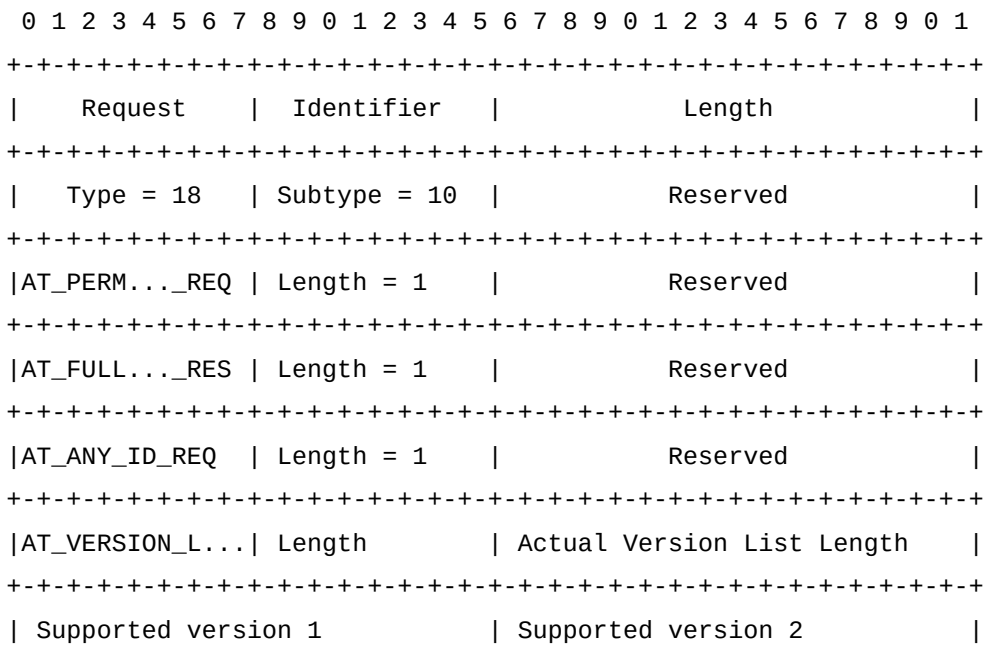


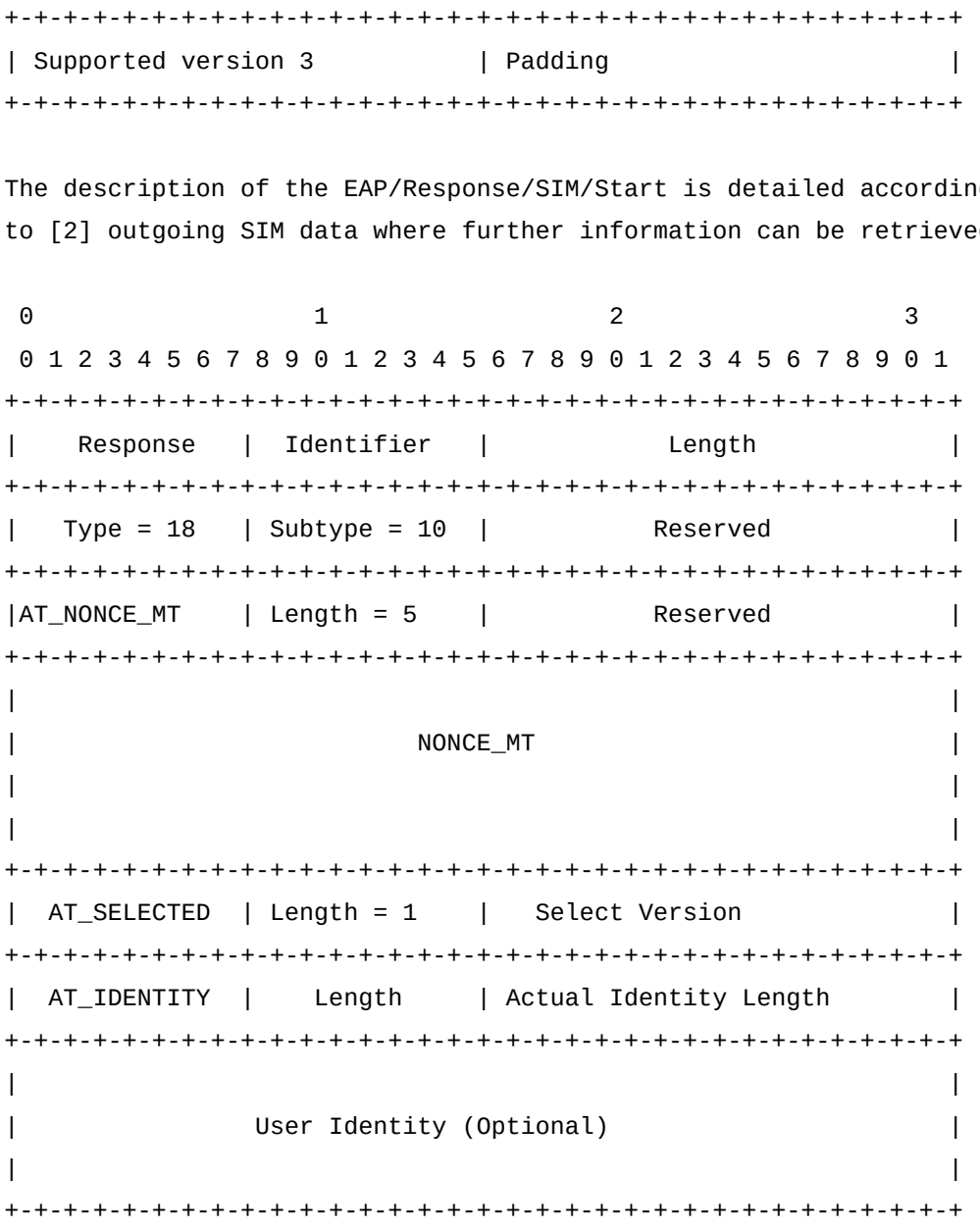
Note the EAP/Response/Identity when returning the userÆs identity that includes the IMSI includes the real coded IMSI in the EAP packet and not the IMSI coded for GSM network. Further information can be retrieved in [3] for the IMSI coding in the SIM during the SIM setting.

The user Identity field can contains the userÆs permanent pseudonym or re-authentication identity.
The second EAP Packet is the EAP request SIM start as represented in the IETF draft document [2].

Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	XX	YY

The description of the EAP/Request/SIM/Start is detailed according to [2] incoming SIM data where further information can be retrieved.


0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Request | Identifier | Length
Type = 18 | Subtype = 10 | Reserved
AT_PERM..._REQ | Length = 1 | Reserved
AT_FULL..._RES | Length = 1 | Reserved
AT_ANY_ID_REQ | Length = 1 | Reserved
AT_VERSION_L... | Length | Actual Version List Length
Supported version 1 | Supported version 2

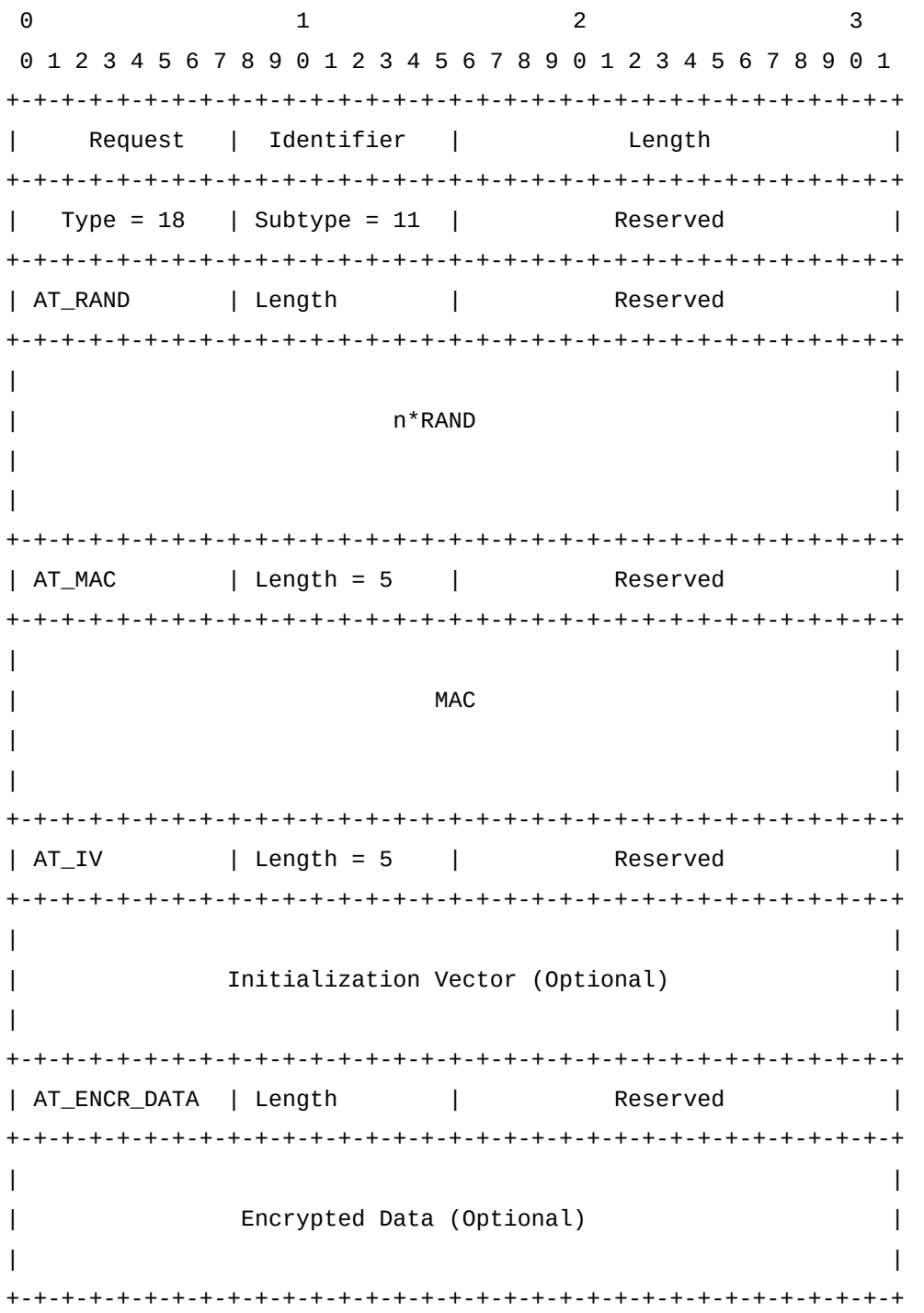


The description of the EAP/Response/SIM/Start is detailed according to [2] outgoing SIM data where further information can be retrieved.

The description of the EAP/Response/SIM/Start is detailed according to [2] outgoing SIM data where further information can be retrieved. The third EAP Packet is the EAP request SIM Challenge as represented in the IETF draft document [2].

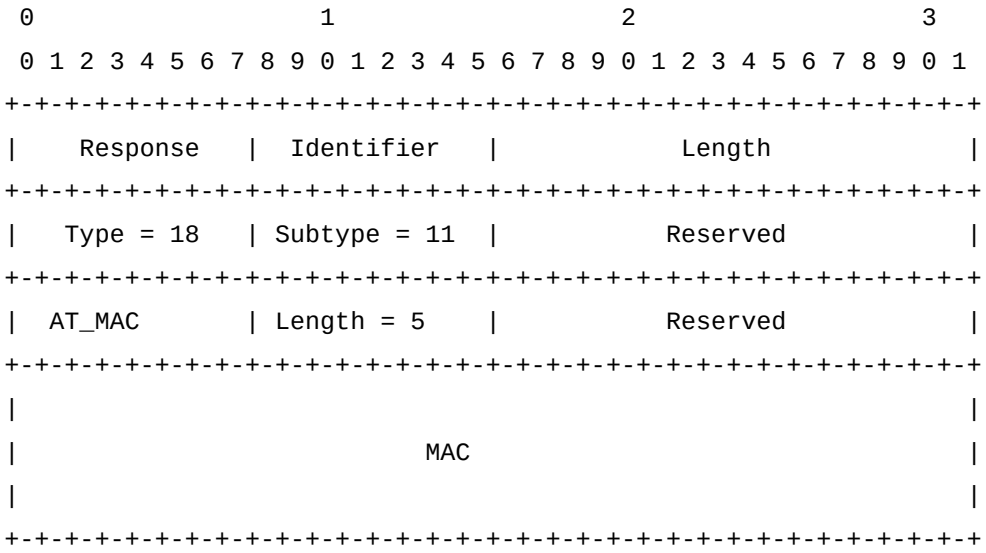
Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	XX	1C

The description of the EAP/Request/SIM/Challenge is detailed according to [2] incoming SIM data where further information can be retrieved.



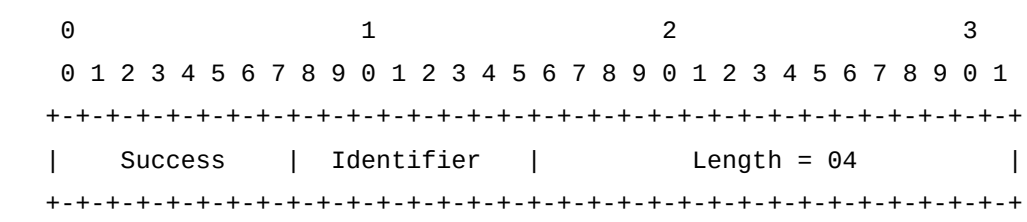
The description of the EAP/Response/SIM/Challenge is detailed according to [2] outgoing SIM data where further information can be

retrieved.


0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Response
Identifier
Length
Type = 18
Subtype = 11
Reserved
AT_MAC
Length = 5
Reserved
MAC

The last EAP Packet is the EAP success notification as represented in the IETF RFC 2284 [2].

Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	04	00

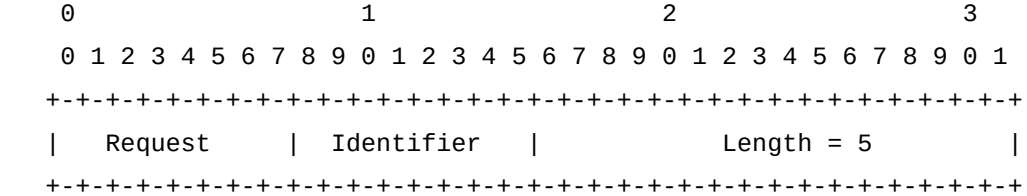


Annex 2 (Informative) - EAP/MD5 packet details

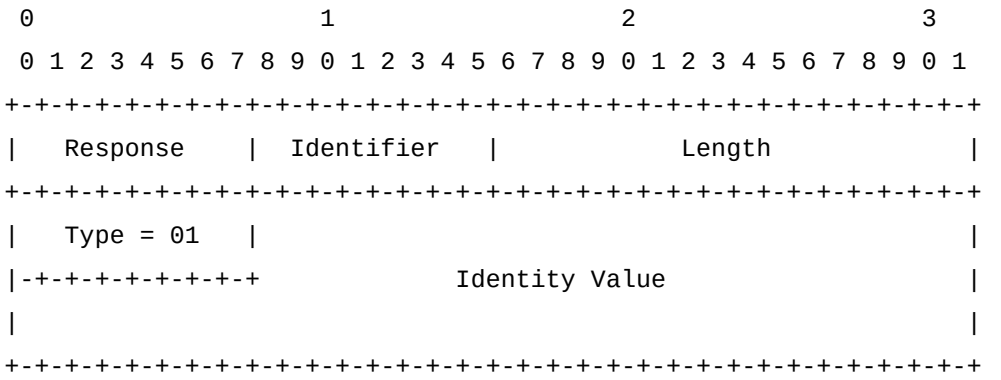
The first EAP packet is the EAP Request Identity. This initial packet format complies with the RFC 2284. The smart card returns an EAP response identity according to the NAI length.

Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	05	YY

The description of the EAP/Request/identity is detailed according to the IETF RFC 2284 [1].



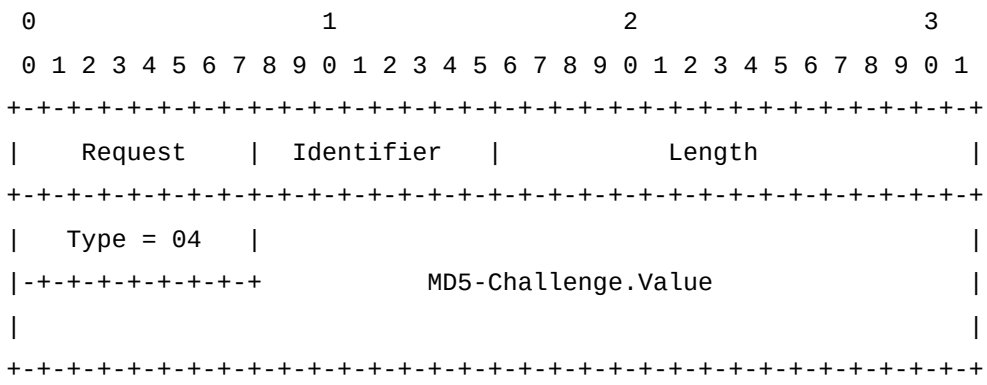
The description of the EAP/Response/identity is detailed according to the IETF RFC 2284 [1].



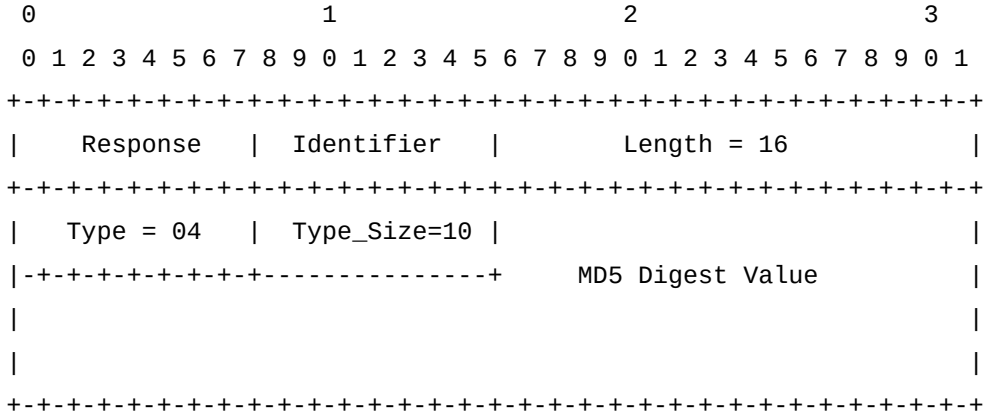
The second EAP Packet is the EAP/request/MD5/challenge as represented in the IETF RFC 2284 [1].

Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	XX	16

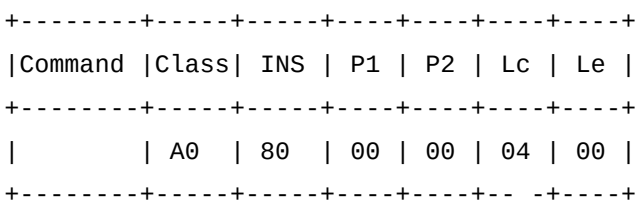
The description of the EAP/Request/MD5/challenge is detailed according to the IETF RFC 2284 [1].



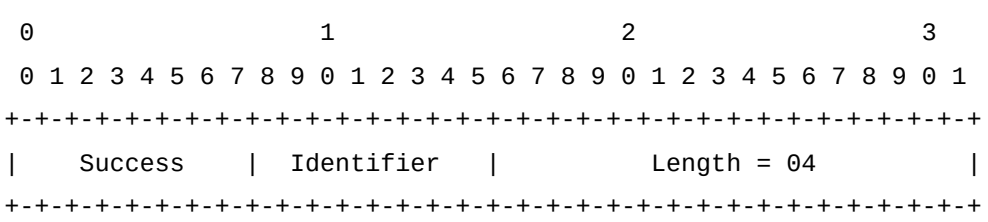
The description of the EAP/Response/MD5/challenge is detailed according to the IETF RFC 2284 [1].



The third EAP Packet is the EAP success notification as represented in the IETF RFC 2284 [1].



Command	Class	INS	P1	P2	Lc	Le
	A0	80	00	00	04	00



0	1	2	3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1			
Success	Identifier	Length = 04	

Further information can be retrieved from the IETF draft document [2].

Annex 3 (Informative) TLS support

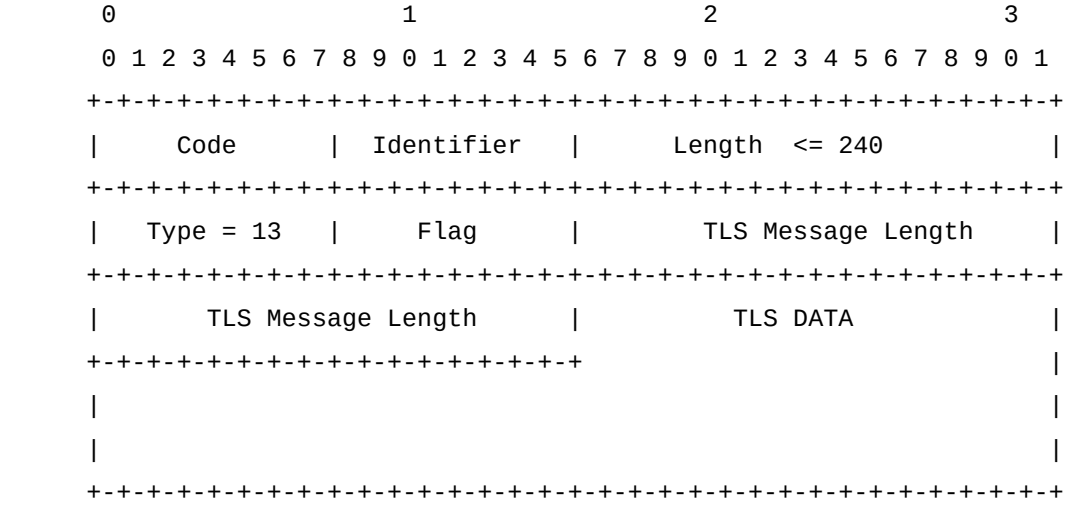
Fragment maximum size.

A single TLS record may be up to 16384 octets in length, but a TLS message may span multiple TLS records, and a TLS certificate message may in principle be as long as 16MB. The group of EAP-TLS messages sent in a single round may thus be larger than the maximum RADIUS packet size of 4096 octets, or the maximum 802 LAN frame size.

Due to smartcard constraints, the maximum EAP message length of a no fragmented packet is set to 240 bytes. For a fragmented EAP message, the maximum length value is 240 bytes.

When the smartcard receives an EAP-Request packet with the M bit set, it MUST respond with an EAP-Response with EAP-Type=EAP-TLS and no data. This serves as a fragment ACK.

EAP/TLS messages format.



```
  Flags
  0 1 2 3 4 5 6 7 8
  +-+-+-+-+-+-+-+-+-+
  |L M S R R R R R|
  +-+-+-+-+-+-+-+-+-+
  L = Length included.
  M = More fragments
  S = EAP-TLS start, set in an EAP-TLS Start message.
  R = Reserved
```

Example of EAP/TLS Authentication

```
     Smartcard           Authentication Server
                             <- EAP-Request/
                                Identity
```

```
  EAP-Response/
  Identity (MyID) ->
                              <- EAP-Request/
                              EAP-Type=EAP-TLS
                              (TLS Start)
  EAP-Response/
  EAP-Type=EAP-TLS
  TLS client_hello)->
                              <- EAP-Request/
                              EAP-Type=EAP-TLS
                              (TLS server_hello,
                               TLS certificate,
                               TLS certificate_request,
                               TLS server_hello_done)
  EAP-Response/
  EAP-Type=EAP-TLS
  (TLS certificate,
   TLS client_key_exchange,
   TLS certificate_verify,
   TLS change_cipher_spec,
   TLS finished) ->
                              <- EAP-Request/
                              EAP-Type=EAP-TLS
                              (TLS change_cipher_spec,
                               TLS finished)
     EAP-Response/
     EAP-Type=EAP-TLS ->
                              <- EAP-Success
```

Annex 4 (Normative) ASN.1 BER Tag coding for the subscriber profile information

To be defined according to the EAP type.

References

[1] L. Blunk, J. Vollbrecht, "PPP Extensible Authentication Protocol (EAP)", RFC 2284, March 1998. (NORMATIVE)

[2] EAP SIM Authentication draft version 8 (NORMATIVE)

[3] GSM Technical Specification GSM 11.11. Digital cellular telecommunications system (Phase 2+); Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME)

[4] Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications

[5] Standards for Local and Metropolitan Area Networks: Standard for Port based Network Access Control.

[6] "The Network Access Identifier" rfc 2486

[7] "Can you Clone a GSM Smart Card (SIM)? " From Charles Brookson Chairman GSM Association Security Group

[8] Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Specification for Enhanced Security

[9] ASN.1 standard 2002 edition ISO/IEC 8825.1. http://asn1.elibel.tm.fr/en/standards/index.htm

[10] Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation 6 October 2000

[11] B. Aboba, D. Simon, EAP TLS Authentication Protocol RFC 2716, October 1999.

[12] H. Andersson, S. Josefsson, G. Zorn, D. Simon, A. Palekar, "Protected EAP Protocol (PEAP)", draft-josefsson-pppext-eap-tls-eap-05.txt, work-in-progress, September 2002. (INFORMATIVE)

Author's Addresses

Pascal Urien
ENST
46 rue Barrault
75013 Paris Phone: NA
France Email: Pascal.Urien@enst.fr

Augustin J. Farrugia
Impasse des CAMEGIERS Phone: NA
Ceyreste, 13600 France Email: afarrugia@csi.com

Max de Groot
Gemplus
Avenue du Pic de Bertagne

BP 100, 13881 Gemenos Phone: +33 442 365 036
France Email: max.de-groot@gemplus.com

Guy Pujolle
LIP6 - University Paris 6
8 rue Capitaine Scott Phone: NA
Paris 75015 France Email: Guy.Pujolle@lip6.fr