

WLAN Smart Card Consortium

EAP-SIM Handler Specification Version 1.1



SUBJECT TO CHANGE
WITHOUT NOTICE

1	Introduction	3
2	References	3
3	Terminology	3
4	Architecture	4
5	Interface Component Manager	5
5.1	Select Component / Check Card Type	5
5.2	Get Identity	5
5.3	EAP Processing	5
6	All Interface Components	6
6.1	Card Activation	6
7	WLAN SIM Interface Component	6
7.1	WLAN-SIM Session	6
7.2	Check Card Type	6
	1) <i>Application Selection</i>	6
	2) <i>Check Version</i>	6
7.3	Component Get Identity	6
7.4	User Authentication	7
7.5	Processing of EAP-SIM messages	7
	General	7
	EAP Request/SIM/Start	7
	EAP Request/SIM/Challenge	9
	EAP Request/SIM/Re-authentication	9
	EAP Request/SIM/Notification	10
7.6	Protocol Overview	11
8	SIM component	12
8.1	SIM Session	12
8.2	Check Card Type	12
8.3	Component Get Identity	12
8.4	User Authentication	12
8.5	Startup	12
8.6	EAP Processing	12
9	UICC EAP component	12
9.1	Check Card Type	13
9.2	UICC EAP-SIM Session	13
9.3	Component Get Identity	13
9.4	User Authentication	13
9.5	EAP Processing	13
	General	13
	Status	14
	Successful Authentication	14
10	Security Considerations	14

1 Introduction

Several different types of smart cards support the EAP-SIM protocol for the purpose of authenticating a client device to a WLAN network. Currently each type of smart card requires different, non-interoperable, client software.

This document specifies an “EAP-SIM Handler” that ensures interoperability between a client application and any of the following smart cards:

- WLAN-SIM Smart Card [2] (within the SCP [3]context)
- Legacy SIM Smart Card [6,7]
- UICC supporting EAP-SIM [8]

The EAP-SIM handler specification defines the

- Exchange with the EAP-SIM server for authentication/re-authentication
- Use of the Smart Card applications to generate EAP-SIM Specific identities, Authentication Codes and session keys.

The integration of the handler into the operating systems or third party network layers, support of 802.1x or EAP or the interface to WLAN Network Interface Cards is out of scope of these specifications.

2 References

- [0] draft-haverinen-pppext-eap-sim-11.txt: H. Haverinen & All, Nokia.
- [1] draft-urien-eap-smartcard-04.txt: P.Urien & All, ENST.
- [2] WLAN-SIM Specification V.1.0: WLAN Smart Card Consortium
- [3] ETSI TS 102.221: "Smart Cards; UICC-Terminal interface; Physical and logical characteristics (Release 5)"
http://docbox.etsi.org/SCP/SCP/Specs/TS_102221/
- [4] RFC 2284bis: Extensible Authentication Protocol (EAP)
- [5] RFC 2119: Key words for use in RFCs to Indicate Requirement Levels
<ftp://ftp.rfc-editor.org/in-notes/rfc2119.txt/>
- [6] ETSI 11.11: Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
- [7] 3GPP 51.011: Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
- [8] ETSI TS 102.310 EAP support in UICC

3 Terminology

Acronym	Description
AID	Application Identifier

EAP	Extensible Authentication Protocol
GSM	Global System for Mobile Communication
MAC	Message Authentication Code
PIN	Personal Identification Number
RFU	Reserved for Future Use
SIM	Subscriber Identity Module
WLAN	Wireless LAN
WLAN SCC	Wireless LAN Smart Card Consortium
FFS	For Further Study

When words such as ‘MUST’, ‘SHOULD’ or ‘MAY’ are used in this document, their precise meaning is to be understood as described in the IETF RFC 2119.

All numbers are decimal by default. Hexadecimal numbers are represented between quotes such as ‘XY’.

4 Architecture

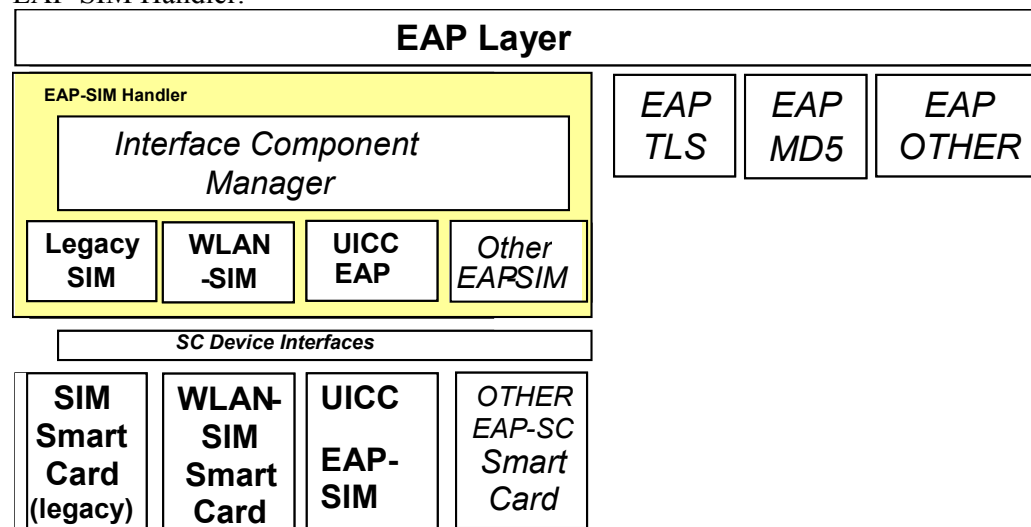
With EAP-SIM, the authentication functionality is split between the smart card and the EAP-SIM Handler which interfaces to the smart card. Together, the smart card and the handler form the EAP “METHOD” as defined in RFC 2284 [4].

The EAP-SIM Handler consists of an Interface Component Manager and several Interface components, one for each type of smart card used for EAP-SIM authentication.

The Interface Component Manager identifies the available smart card, determines which component will handle the interaction with card, and “routes” the EAP authentication messages to the appropriate component which then interfaces with the smart card and provides the response.

The upper network layers (EAP/802.1X) are out of scope. They may be integrated in the client application, or provided by the operating system (e.g. Windows XP).

The lower smart card interface, including the card interface drivers are also out of scope to the EAP-SIM Handler.



5 Interface Component Manager

5.1 Select Component / Check Card Type

After the client station is associated with an access point where an EAP SIM authentication applies (e.g. as indicated in the connection configuration), or after a previously selected component has returned an error, the Interface Component Manager **MUST** execute a Select Component procedure.

First the Interface Component Manager **MUST** check the availability of a smart card. If no smart card is available, the Interface Component Manager **MUST** inform the user, and **MAY** request the user to provide a smart card.

If multiple smart cards are available, the handler **MUST** select the smart card to work with.

When a smart card is available the Interface Component Manager **MUST** check the type of card by calling the Check Card Type function of the Interface Components in the following order:

1. UICC supporting EAP-SIM according to [8]
2. WLAN-SIM (on UICC) according to [2]
3. Legacy SIM according to [6] or [7]

The Interface Component Manager **MUST** select one of the components that returns **TRUE** to the Check Card Function as the active component.

If no usable card is available of the Interface Components return **TRUE**, then the handler **MUST** return an EAP-NAK packet.

5.2 Get Identity

The Interface Component Manager **MUST** request the preferred EAP identity of the active Interface Component using the Component Get Identity command and provide this identity to the EAP Layer.

If the EAP-Request-Identity message is available to the Interface Component Manager, then the Interface Component Manager **MUST** provide the EAP-Response-Identity to this message. Otherwise, the Interface Component Manager **SHOULD** provide the EAP Identity to the EAP Layer using whatever available interface.

5.3 EAP Processing

The Interface Component Manager **MUST** forward subsequent EAP messages to the active component until the active component returns an error.

6 All Interface Components

6.1 Card Activation

The Interface Component must check that the smart card is activated. If it is not activated, the card **MUST** be activated, e.g. cold or warm reset.

7 WLAN SIM Interface Component

7.1 WLAN-SIM Session

A WLAN-SIM Session starts with a Check Card Type call from the Interface Component Manager. With this call the Component selects the WLAN SIM application, and checks the version of the card. The component then waits for a Component Get Identity to check the User Authentication before processing this and any subsequent EAP messages.

This version of the WLAN-SIM Handler specification only supports WLAN-SIM smart cards in the SCP [3] context.

The active WLAN-SIM Session ends whenever the new VERIFY command is executed with a wrong PIN code, the application is deselected or when the card is reset.

The WLAN-SIM component **MUST** support multiple complete EAP-SIM authentications in the same WLAN-SIM session as described in the section “Processing of EAP-SIM Messages” below.

7.2 Check Card Type

The Check Card Type function is called by the Interface Component Manager to check whether the available card can be handled by the WLAN-SIM Interface Component. It consists of the following two routines

1) Application Selection

The WLAN-SIM component checks whether the WLAN-SIM smart card application has been selected. If the smart card application has not been selected, the WLAN-SIM component **MUST** activate a communication channel and select the WLAN-SIM smart card application. To select the WLAN-SIM application, the handler **MUST** either issue the SELECT command, using the AID defined in [2], with implicit channel selection, or issue a MANAGE CHANNEL command before a SELECT command.

Further commands will be sent to the card using the channel the WLAN-SIM is selected on.

2) Check Version

The application **MUST** check the version of WLAN-SIM card using the Check Current Version function.

Check Card Type **MUST** return FALSE if the component does not handle the version of the WLAN-SIM smart card application.

7.3 Component Get Identity

The WLAN-SIM Component **MUST** reply to this request with the identity string obtained with the GET PREFERRED IDENTITY command. Therefore, this command **MUST** be ‘freshly’ issued, i.e. after activation of the card and selection of the WLAN-SIM application

or after any correctly or incorrectly executed EAP/SIM authentication, as the pseudonym or re-authentication id might be changed.

7.4 User Authentication

Upon receipt of the Component Get Identity command, and if the PIN code is activated, the VERIFY [3] command MUST be issued with an empty data field, to obtain the number of PIN presentation retries left.

Then, the PIN code MUST be verified (with the VERIFY command) before any further transaction with the application is done. Once the VERIFY command has been successfully executed, the handler MUST not prompt the user for a PIN code more than once inside a session.

7.5 Processing of EAP-SIM messages

General

The WLAN-SIM Component MUST handle the exchange of EAP-SIM messages with the authentication server or Access Point through available EAP and 802.1x support provided by the Interface Component Manager. The reception of a message will trigger the execution of a specific process, combined with a command-response exchange with the WLAN-SIM. These processes are described below.

In general, the WLAN-SIM Component MUST construct EAP/SIM compliant messages from the responses obtained from the card, unless EAP/SIM compliant messages are returned in the card response (e.g. PROCESS EAP).

In general, the WLAN-SIM Component MUST check the format, type and consistency of all EAP-SIM messages received from the network

After the end of the authentication process, the handler MUST provide the session keys to the necessary entity.

EAP Request/SIM/Start

There can be up to three roundtrips of EAP Request/SIM/Start messages with related responses.

First round:

The server might not understand the identity received in the EAP Response Identity message and include optional attributes AT-ANY-ID-REQ or AT-FULLAUTH-ID-REQ in the EAP Request SIM/Start message.

If AT-ANY-ID-REQ or AT-FULLAUTH-ID-REQ attribute are NOT contained in the request, the WLAN-SIM Component MAY issue the SELECT-IDENTITY command on the card indicating the same identity previously returned with the GET PREFERRED ID command (P2=00), followed by the GET VERSION and GET RANDOM command and reply EAP Response SIM/Start to the server.

If the AT-ANY-ID-REQ or AT-FULLAUTH-ID-REQ attribute is contained in the request, the WLAN-SIM Component MUST issue the SELECT-IDENTITY command on the card indicating the identity requested by the server.

Unless the identity returned is the RE-AUTHENTICATION IDENTITY, the WLAN-SIM Component MUST issue the GET CURRENT VERSION (EAP-SIM version) and GET RANDOM command and reply EAP Response SIM/Start to the server. It MUST include the returned identity in the AT-IDENTITY attribute and include the AT-NONCE-MT and AT-SELECTED-VERSION attribute.

If the identity returned is the RE-AUTHENTICATION IDENTITY, the WLAN-SIM Component MUST reply EAP Response SIM/Start to the server. It MUST include the returned identity in the AT-IDENTITY attribute and NOT include any other attribute.

Second round:

The server might not agree with the identity received in the EAP Response SIM/Start message and reiterate the EAP Request SIM/Start message, include the attribute AT-FULLAUTH-ID-REQ or AT-PERMANENT-ID-REQ.

If a second EAP Request SIM/Start message is received, containing the AT- FULLAUTH-ID-REQ attribute, the WLAN-SIM Component MUST issue the SELECT-IDENTITY command on the card indicating to use Pseudonym or IMSI (P2=17).

If a second EAP Request SIM/Start message is received, containing the AT- PERMANENT-ID-REQ attribute, the WLAN-SIM Component MUST issue the SELECT-IDENTITY command on the card indicating to use IMSI (P2=10).

If the identity type returned by the card in above-mentioned command is not the permanent id, as requested (the card wishes not to reveal the IMSI), the WLAN-SIM Component MUST NOT respond to the server. Otherwise, it MUST include the returned identity in the AT-IDENTITY attribute in the EAP Response SIM/Start message.

Alternatively, if a second EAP Request SIM/Start message is received, containing the AT-PERMANENT-ID-REQ attribute, in case the permanent identity is known in the WLAN-SIM Component already, the WLAN-SIM Component can omit issuing the SELECT IDENTITY command if in the PROCESS EAP command the use of IMSI is enforced (P2 = '01'). Nevertheless, the permanent identity MUST be included in the AT-IDENTITY attribute.

The WLAN-SIM Component MUST issue the GET CURRENT VERSION (EAP-SIM version) and GET RANDOM command and include the returned identity in the AT-IDENTITY attribute and the random in the AT-NONCE-MT and the version in AT-SELECTED-VERSION attribute in the EAP Response SIM/Start message.

If a second EAP Request SIM/Start message is received, NOT containing any identity request attribute, the WLAN-SIM Component MUST issue the SELECT-IDENTITY command on the card indicating to use IMSI (P2=17). If the card effectively responds with the permanent ID, the WLAN-SIM Component MUST reply to the server with the EAP Response SIM/Start including the random in the AT-NONCE-MT and the version in AT-SELECTED-VERSION attribute, but NOT including the AT-IDENTITY attribute. In fact, the server recognized the re-authentication id but wants to fall back on full authentication.

Third round:

The server might not agree with the pseudonym identity received in the EAP Response SIM/Start message and reiterate the EAP Request SIM/Start message, include the attribute AT-PERMANENT-ID-REQ.

If a third EAP Request SIM/Start message is received, containing the AT- PERMANENT-ID-REQ attribute, the WLAN-SIM Component MUST issue the SELECT-IDENTITY command on the card indicating to use IMSI (P2=10).

If the identity type returned by the card in above-mentioned command is not the permanent id as requested, the WLAN-SIM Component MUST NOT respond to the server, because the card wishes not to reveal the IMSI. Otherwise, it MUST include the returned identity in the AT-IDENTITY attribute in the EAP Response SIM/Start message.

Alternatively, if a third EAP Request SIM/Start message is received, containing the AT-PERMANENT-ID-REQ attribute, in case the permanent identity is know in the WLAN-SIM Component already, the WLAN-SIM Component can omit issuing the SELECT IDENTITY command if in the PROCESS EAP command the use of IMSI is enforced (P2 = '01'). Nevertheless, the permanent identity MUST be included in the AT-IDENTITY attribute.

The WLAN-SIM Component MUST issue the GET CURRENT VERSION (EAP-SIM Version) and GET RANDOM command and include the returned identity in the AT-IDENTITY attribute and the random in the AT-NONCE-MT and the version in AT-SELECTED-VERSION attribute in the EAP Response SIM/Start message.

EAP Request/SIM/Challenge

The WLAN-SIM Component MUST forward this message unaltered to the card with the PROCESS EAP command.

The EAP Response SIM/Challenge packet returned in the PROCESS EAP command MUST be sent unaltered to the server.

The WLAN-SIM Component MUST return the session keys, provided in the command response.

EAP Request/SIM/Re-authentication

The WLAN-SIM Component MUST forward this message unaltered to the card with the PROCESS EAP command.

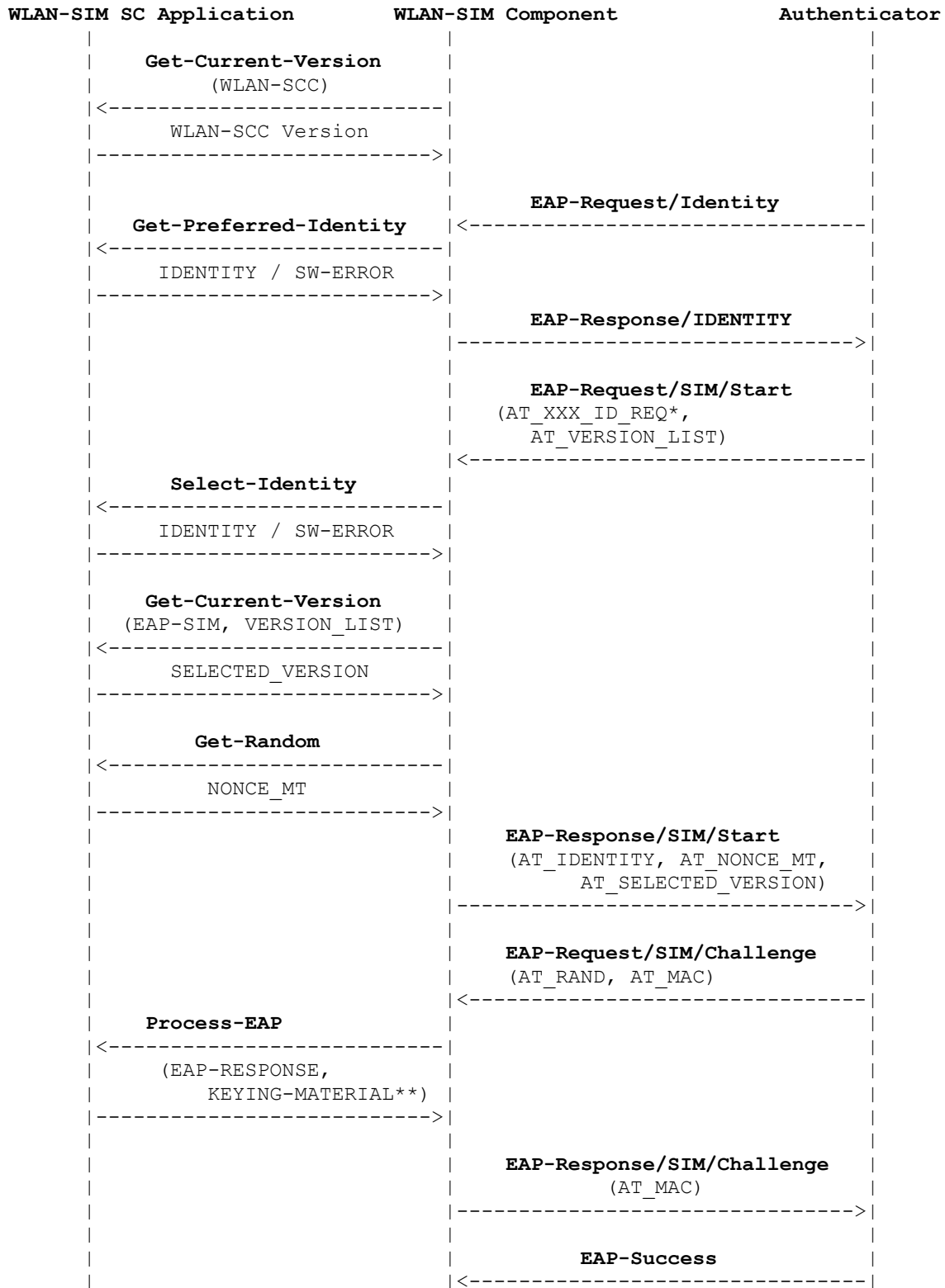
The EAP Response SIM/Re-authentication packet returned in the PROCESS EAP command MUST be sent unaltered to the server.

The WLAN-SIM Component MUST return the session keys, provided in the command response.

EAP Request/SIM/Notification

This message is silently discarded.

7.6 Protocol Overview



This overview does not show all EAP-SIM command attributes. For a complete overview see [0]

*AT_ANY_ID_REQ, AT_PERMANENT_ID_REQ, AT_FULLAUTH_ID_REQ, see [0]
** master session key and extended master session key

8 SIM component

This component MUST interface to a smart cards compliant to 3GPP 11.11 [6] and smart cards compliant to 3GPP 51.011 [7].

8.1 SIM Session

A SIM Session starts with the activation of the SIM smart card application and ends with the de-activation of the application, as described in the sections below.

The session start is initiated with a Check Card Type call from the Interface Component Manager. With this call the Component selects the SIM application. The component then waits for a Component Get Identity to check the User Authentication before processing this and any subsequent EAP messages.

The active SIM Session ends whenever the new VERIFY CHV command is executed with a wrong PIN code, or when the card is reset.

8.2 Check Card Type

The SIM Component selects the DF_{GSM} [6]. If the DF_{GSM} selection succeeds, the Check Card Type returns TRUE, otherwise it returns FALSE.

8.3 Component Get Identity

The Check Card Type method has established the current smart card DF context as DF_{GSM} [6]. The SIM Component selects the EF_{IMSI} [6] file. The SIM Component read the EF_{IMSI} [6] file using READ binary commands and builds the identity according to [0].

8.4 User Authentication

Upon receipt of the Component Get Identity command, and if the CHV1 code is activated, the VERIFY CHV [7] command MUST be issued before any further transaction with the application is done. Once the VERIFY CHV command has been successfully executed, the handler MUST NOT prompt the user for a CHV1 code more than once inside a SIM session.

8.5 Startup

Refer to [6], [7].

8.6 EAP Processing

Refer to [0].

9 UICC EAP component

This component MUST interface to a smart card compliant to [3] and holding an application handling EAP-SIM as defined in [8]. In the rest of this section this application is referred as EAP SIM application.

9.1 Check Card Type

The UICC EAP Component **MUST** select the EF_{DIR} [3] and **MUST** check if an UICC application supporting EAP-SIM is present (as defined in [8]).

If an EAP-SIM application is present in EF_{DIR}, the Check Card Type returns TRUE, otherwise it returns FALSE.

9.2 UICC EAP-SIM Session

A UICC EAP-SIM Session starts with the selection of the EAP-SIM application and the selection of the EAP-SIM dedicated file (EAP-SIM DF).

UICC EAP-SIM component **MUST** select the EAP-SIM application using the AID contained in EF_{DIR} EAP data objects as parameter of the SELECT command, using direct application selection as defined in [3]. The file identifier of the EAP-SIM DF inside this application is also indicated in the corresponding EAP related data objects in EF_{DIR} (as defined in [8]).

The session start is initiated with a Check Card Type call from the Interface Component Manager.

The active UICC EAP-SIM Session ends whenever the new VERIFY PIN command is executed with a wrong PIN code, when the EAP-SIM application is deselected or when the card is reset or removed.

9.3 Component Get Identity

The UICC EAP Component **SHOULD** reply to this request with the EAP-Response-Identity packet obtained with the EAP AUTHENTICATE command as defined in [8].

9.4 User Authentication

Upon receipt of the Component Get Identity command, access conditions to perform EAP Authenticate command shall be met.

Note: security conditions together with the PIN status are included in the File Control Parameters (FCP) returned by the UICC in the previous SELECT command as defined in [3].

VERIFY PIN [3] command **MUST** be issued before any further transaction with the application to satisfy the access conditions. Once the VERIFY PIN command has been successfully executed, the handler **MUST NOT** prompt the user for the PIN code more than once inside a UICC EAP-SIM session.

9.5 EAP Processing

General

The UICC EAP Component **MUST** forward all incoming EAP packets (as defined in [0]) unaltered to the card with the EAP AUTHENTICATE command (as defined in [8]).

The EAP response packets returned in the EAP AUTHENTICATE command MUST be sent unaltered to the server.

Status

The UICC EAP Component MAY find the status of the current authentication process by inspecting the EF_{EAPSTATUS} (EAP Authentication STATUS) file (as defined in [8]) using READ BINARY command [3].

Successful Authentication

Once a successful EAP-SIM is completed, The UICC EAP Component MUST return the session keys produced in the current authentication. They are retrieved from EF_{EAPKEYS} (as defined in [8]) by performing the READ BINARY command [3].

Note: The UICC EAP Component SHOULD detect the completion of EAP-SIM authentication once receiving the EAP-Success packet and by inspecting the authentication status as defined in the previous subsection.

10 Security Considerations

Keying material MUST be protected against eavesdropping by other applications.