Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**EAP Enhancement**
Date Submitted	**2005-01-10**
Source(s)	JUNHYUK SONG, Yong Chang Samsung Electronics Voice: +82-31-279-3639 junhyuk.song@samsung.com
Re:	Re: IEEE P802.16e/D5a
Abstract	Proposal to secure EAP transfer message
Purpose	Discuss and Adopt as the baseline text
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.

PKM version 2

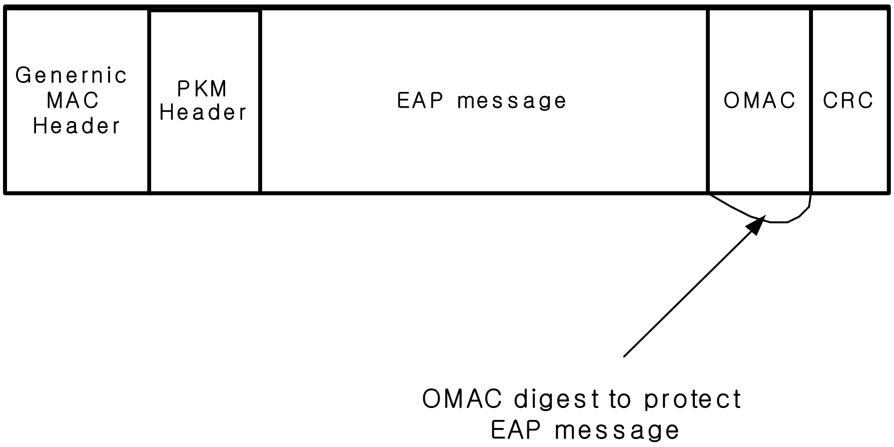
EAP Enhancement

JUNHYUK SONG,, YONG CHANG
Samsung Electronics

INTRODUCTION

RFC3748 (EAP) describes possible threat model against EAP message running over wireless network. Most of the attack is caused by lacking of link layer security. Current PKMv2 has a consideration for protecting EAP message if the link layer security is enabled by either RSA or EAP. PKMv2 Key Hierarchy defined key derivation for EAP Integrity Key (EIK), however it is missing the related message format, text and procedure. This contribution defined additional PKM EAP messages for protecting EAP message that will cryptographically bind device and user if the link layer security is enabled.

Proposed solution



Attribute	Contents
Key Sequence Number	AK Sequence Number
EAP Protocol	Contains the EAP authentication data, not interpreted in the MAC
OMAC Digest	Message Digest calculated using EIK

Changes to 802.16e D5a text

[Newly insert the following code and message name into Table 26 PKM Message Codes]

Code	PKM Message Type	MAC Management Message Name
24	Protected EAP	PKM-REQ/PKM-RSP
25-255, ~~24-255~~	reserved	-

[Newly add the following message after 6.3.2.3.9.21]

6.3.2.3.9.22 Protected EAP messages

If EIK is available and an MSS or BS has an EAP message received from an EAP method for transmission, it encapsulates EAP message in a Protected EAP Transfer message

Code: 24

Attributes are shown in Table 37l

Table 37l Protected EAP message Attributes

Attribute	Contents
Key Sequence Number	AK Sequence Number
EAP Protocol	Contains the EAP authentication data, not interpreted in the MAC
OMAC Digest	Message Digest calculated using EIK

The EAP Payload field carries EAP data in the format described in RFC 3748

The OMAC-Digest attribute shall be the final attribute in the message’s attribute list.
Inclusion of the OMAC digest allows the MSS and BS to cryptographically bind previous authorization and following EAP authentication by authenticating the EAP message. The OMAC-Digest’s authentication key is derived from the AK

[Changes to the section 7.2.1.3 as followings]

7.2.1.3 MSS authorization and AK exchange overview

MSS authorization, controlled by the Authorization state machine,

~~a)~~ is the process of the BS authenticating a client MSS's identity

a~~b~~) T~~t~~he BS and MSS establishing a shared AK by either RSA or EAP, from which a key encryption key (KEK), EAP Integrity Key (EIK) and message authentication keys are derived
b~~c~~) T~~t~~he BS providing the authenticated MSS with the identities (i.e., the SAIDs) and properties of primary and static SAs the MSS is authorized to obtain keying information for
c)

After achieving initial authorization, an MSS periodically reauthorize with the BS; reauthorization is also managed by the MSS's Authorization state machine. TEK state machines manage the refreshing of TEKs.
f) The MSS or BS may run optional protected EAP messages for additional authentication

7.2.1.3.1 Authorization via RSA authentication protocol

An MSS begins authorization by sending an Authentication Information message to its BS. The Authentication Information message contains the MSS manufacturer's X.509 certificate, issued by the manufacturer itself or by an external authority. The Authentication Information message is strictly informative; i.e., the BS may choose to ignore it. However, it does provide a mechanism for a BS to learn the manufacturer certificates of its client MSS.

The MSS sends an Authorization Request message to its BS immediately after sending the Authentication Information message. This is a request for an AK, as well as for the SAIDs identifying any Static Security SAs the MSS is authorized to participate in. The Authorization Request includes

a) a manufacturer-issued X.509 certificate

b) a description of the cryptographic algorithms the requesting MSS supports; an MSS's cryptographic capabilities are presented to the BS as a list of cryptographic suite identifiers, each indicating a particular pairing of packet data encryption and packet data authentication algorithms the MSS supports

c) the MSS's Basic CID. The Basic CID is the first static CID the BS assigns to an MSS during initial ranging—the primary SAID is equal to the Basic CID

In response to an Authorization Request message, a BS validates the requesting MSS's identity, determines the encryption algorithm and protocol support it shares with the MSS, activates an AK for the MSS, encrypts it with the MSS's public key, and sends it back to the MSS in an Authorization Reply message. The authorization reply includes:

a) an AK encrypted with the MSS's public key

b) a 4-bit key sequence number, used to distinguish between successive generations of AKs

c) a key lifetime

d) the identities (i.e., the SAIDs) and properties of the single primary and zero or more static SAs the MSS is authorized to obtain keying information for

While the Authorization Reply shall identify Static SAs in addition to the Primary SA whose SAID matches the requesting MSS's Basic CID, the Authorization Reply shall not identify any Dynamic SAs.

The BS, in responding to an MSS's Authorization Request, shall determine whether the requesting MSS, whose identity can be verified via the X.509 digital certificate, is authorized for basic unicast services, and what additional statically provisioned services (i.e., Static SAIDs) the MSS's user has subscribed for. Note that the protected services a BS makes available to a client MSS can depend upon the particular cryptographic suites MSS and BS share support for.

An MSS shall periodically refresh its AK by reissuing an Authorization Request to the BS. Reauthorization is identical to authorization with the exception that the MSS does not send Authentication Information messages during reauthorization cycles. Subclause 7.2.4's description of the authorization state machine clearly indicates when Authentication Information messages are sent.

To avoid service interruptions during reauthorization, successive generations of the MSS's AKs have overlapping lifetimes. Both MSS and BS shall be able to support up to two simultaneously active AKs during these transition periods. The operation of the Authorization state machine's Authorization Request scheduling algorithm, combined with the BS's regimen for updating and using a client MSS's AKs (see 7.4), ensures that the MSS can refresh.

After successful RSA based authorization if the MSS or BS wants to run additional EAP authentication, the protected EAP message shall carry EAP message. It shall cryptographically bind RSA and further EAP authentication.

7.2.1.3.2 Authorization via PKM Extensible Authentication Protocol

If MSS requests EAP based authorization and BS allows it, an MSS begins authorization by following steps:

The first steps of the authorization flow are as follows:
a) ~~5)~~ Upon successful completion of ranging (and capabilities exchange), a logical signal (ie. "link activation") is sent upwards on the Logical Control Interface at the BS (ie. the EAP authenticator). This will cause the authenticator to begin the authentication sequence.

b~~6~~) EAP on the Authenticator sends an EAP-Request message to the supplicant. This Request might be an EAP identity request or the beginning of an EAP method. The message is encapsulated in a MAC management PDU and transmitted.

c~~7~~) EAP on the supplicant receives EAP-Request, passes it to the local EAP method for processing,and transmits EAPResponse. ~~Steps 2 and 3 (~~EAP-Request/Response exchange~~)~~ continue as many times as needed based on EAP authentication method.
After one or more EAP-Request/Response exchanges, the authentication server (whether local to the Authenticator or connected remotely via an AAA protocol) determines whether or not the authentication is successful.

The next steps of the authorization flow are as follows:

d~~8~~) Upon success, EAP on the authenticator transmits a "success" signal on the logical control interface to fully activate the airlink.

e~~9~~) EAP on the authenticator transmits EAP-success, which is then encapsulated in a MAC management message and transmitted to the supplicant.

f~~10~~) EAP on the supplicant transmits a "success" indication on the logical control interface to fully activate the airlink.

g~~11~~) Both EAPs (authenticator and supplicant) export the AAA-key across the logical control interface.

As detailed in [3], the AAA-key is the shared "master key" that is derived by the two sides in the course of executing the EAP inner method. The authentication part of the authorization flow (and the involvement of the generic EAP layer) is now complete.

The final steps of the authorization flow:

h~~1~~) The BS and MSS each derive the EAP Master Key from the AAA-Key. The EAP Master Key is derived simply the taking the 32 lowest order octets of the AAA-Key.

i~~2~~) BS sends the EAP-Establish-Key-Request PKM message (including a 32-byte nonce) to the MSS. The MSS then generates its own 32-byte nonce, and derives a Transient Key (TK) as follows:
TK = PRF-384(*EAP Master Key*, “Pairwise key expansion”,
Min(*BSId, SSId*) |
Max(*BSId, SSId*) |
Min(*BS-Generated-Nonce, MSS-Generated-Nonce*) |
Max(*BS-Generated-Nonce, MSS-Generated-Nonce*))
where

```
PRF-384 (K, A, B) :=
for i = 0 to 3 do
R = R | HMAC-SHA-1(K, A | 0 | B | I i )
return LeastSignificant-384-bits(R).
```

and “|” denotes bitstring concatenation.
The MSS then derives Key Confirmation Key (KCK) and Authorization Key (AK) as follows:
KCK = bits 0-127 (ie. lowest order) of the TK (first 16 octets)
AK = bits 224-383 of the TK (last 20 octets)
The BS can attempt to use a cached or handover-transferred Master Key and avoid a full reauthentication.
To do this, it sends EAP-Establish-Key-Request specifying the MKID attribute, which identifies by name the Master Key that the MSS should use for AK establishment if it also has the MK cached.

j~~3~~) SS sends the EAP-Establish-Key-Reply PKM message (including the 32-byte nonce that it used to derive TK) to the BS. EAP-Establish-Key-Reply includes an HMAC Tuple TLV, which must be calculated using the KCK derived above.
Upon receipt of the EAP-Establish-Key-Reply, the BS computes the TK, KCK, and AK as above.
BS then validates the HMAC Tuple. If the HMAC tuple is incorrect, BS discards the message without responding.
If the MSS elects not to proceed with key establishment (eg. the EAP-Establish-key-request specified an unknown MKID), the MSS sends EAP-Establish-Key-Reject instead.

k~~4~~) BS sends the EAP-Establish-Key-Confirm PKM message to supply the MSS with its SA information and activate the AK.

After successful EAP based authorization if the MSS or BS wants to run additional EAP authentication, the protected EAP message shall carry EAP message. It shall cryptographically bind previous RSA authorization and further EAP authentication, while protecting following EAP message.