Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Simplified PKM EAP Procedure**
Date Submitted	**2004-11-04**
Source(s)	Bryan Kim, Dongkie Lee, JungPyo Han, DongIl Moon, KangIl Koh SK Telecom Voice: +82-31-710-5329 [mailto: {kkhoon, galahad, jphan, dimoon, melomo}@sktelecom.com]
Re:	Recirculation Ballot #15a Announcement
Abstract	PKM EAP Procedure is simplified into two step procedure initiated by MSS not by BS. Benefits coming from the change are explained from the two major points.
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>.

Simplified PKM EAP procedures

Brian Kim, Dongkie Lee, JungPyo Han, DongIl Moon, KangIl Koh
SK Telecom

1. Problem Statements

Current PKM EAP procedure is defined as three-step, BS initiated AK key establishment. As such, this approach has two major drawbacks.

First, MSS is the one, which initiates handover procedure most of the times. So after the MSS is handed over to a new BS and if it's required to do authentication and authorization, then MSS has to wait until BS sends EAP Establish-Key Request message. But it's more appropriate for the MSS to initiate re-authentication and re-authorization because the MSS can decide the re-authentication time better than BS.

Second, from the state machine viewpoint three-step procedure introduces much more states and events and it leads to much more complex implementation of state machines.

Reducing three step into two step does not impair re-authorization itself and authorization message contents but rather it simplifies the reauthorization procedure. Changing the initiator of AK key establishment procedure from the BS to MSS simplifies the state machine implementation and leads to minimal impact on current IEEE 802.16d state machine text. By the above rathional, PKM EAP is changed to be two-step, MSS initiated AK key establishment procedure.

2 Proposed Changes

[Change into the following:]

6.3.2.3.9 Privacy key management (PKM) messages (PKM-REQ/PKM-RSP)

[Insert the following rows into Table 26 in section 6.3.2.3.9, and change the last line in the table:]

Table 26a—PKM Message Codes

13	EAP Transfer	PKM-REQ/PKM-RSP
14	EAP Establish-Key Request	PKM-REQ
15	EAP Establish-Key Reply	PKM-RSP
16	EAP Establish-Key Reject	PKM-RSP
17	Pre-Auth-Request	PKM-REQ
18	Pre-Auth-Reply	PKM-RSP
19	Pre-Auth-Reject	PKM-RSP
21-255	*Reserved*	

6.3.2.3.9.12 EAP Establish-Key Request message

The MSS transmits the EAP Establish-Key Request message forestablishing an AK after EAP-based authentication.
Code: 14
Attributes are shown in Table 37b.

Table 37b—EAP Establish-Key Request attributes

Attribute	Contents
EAP-Master-Key-	A unique handle for the Master Key supplied by the EAP exchange. For use after

Id (optional)	handover or drop/reentry situations when a MSS wants to use a PMK which MSS believes that BS has for the MSS and can proceed immediately to the Establish/Install Key phase. Derivation of the Master Key Id is described in 11.9.20
Nonce	A fresh, randomly generated bit string
Security-Capabilities	Describes MSS's security and ciphersuite capabilities
Primary SAID	MSS's primary SAID (equal to the Basic CID)

6.3.2.3.9.13 EAP Establish-Key Reply message

The BS transmits the EAP Establish-Key Reply message in response to an EAP Establish-Key Request for establishing an AK after EAP-based authentication.
Code: 15
Attributes are shown in Table 37c.

Table 37c—EAP Establish-Key Reply attributes

Attribute	Contents
Nonce	A fresh, randomly generated bit string
Key-Lifetime	AK's active lifetime
Key-Sequence-Number	Sequence Number for established AK
(one or more) SA descriptors	Each Compound SA-Descriptor attribute specifies an SAID and additional properties of the SA
PKM Configuration settings(optional)	PKM timer values
HMAC-Tuple	The cryptographic hash for the message. The key used to generate the hash is the KCK (key confirmation key) as described in 7.2.1.2.

6.3.2.3.9.14 EAP Establish-Key Reject message

The BS transmits the EAP Establish-Key Reject message if the BS rejects the MSS's EAP Establish-Key Request.
Code: 16
Attributes are shown in Table 37d.

Table 37d—EAP Establish-Key Reject attributes

Attribute	Contents
Error Code	Error Code identifying reason for rejection of EAP Establish-Key Reply

7.2.1.2 Authorization via PKM Extensible Authentication Protocol

The first steps of the authorization flow are as follows:

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.
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.
7) EAP on the supplicant receives EAP-Request, passes it to the local EAP method for processing,and transmits EAP Response.

Steps 2 and 3 (EAP-Request/Response exchange) continue as many times as needed.

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:

8) Upon success, EAP on the authenticator transmits a “success” signal on the logical control interface to fully activate the airlink.
9) EAP on the authenticator transmits EAP-success, which is then encapsulated in a MAC management message and transmitted to the supplicant.
10) EAP on the supplicant transmits a “success” indication on the logical control interface to fully activate the airlink.
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:

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.
2) MSS sends the EAP-Establish-Key-Request PKM message (including a 32-byte nonce) to the BS. The BS 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)
return LeastSignificant-384-bits(R).
and “|” denotes bitstring concatenation.
The BS then derives Key Confirmation Key (KCK) and Authorization Key (AK) as follows:
KCK = bits 0-127 (ie. lowest order) of the TK
AK = bits 224-383 of the TK

The MSS 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 BS should use for AK establishment if it also has the MK cached.

3) BS sends the EAP-Establish-Key-Reply PKM message (including the 32-byte nonce that it used to derive TK) to the MSS to supply the MSS with its SA information. 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 MSS computes the TK, KCK, and AK as above. MSS then validates the HMAC Tuple. If the HMAC tuple is incorrect, MSS discards the message and restarts authorization procedure again with its BS. If the BS elects not to proceed with key establishment (eg. the EAP-Establish-key-request specified an unknown MKID), the BS sends EAP-Establish-Key-Reject instead.

11.9.10 Error Code

Type	Length	Value(uint8)	
16	1	Error-Code	Authorization Reject, Authorization Invalid, Key Reject, TEK Invalid, EAP Establish-Key Reject

Table 371—Error-code attribute code values

Error Code	Messages	Description

0	All	No information
1	Auth Reject, Auth Invalid, EAP Establish-Key Reject	Unauthorized SS
2	Auth Reject, Key Reject	Unauthorized SAID
3	Auth Invalid	Unsolicited
4	Auth Invalid, TEK Invalid	Invalid Key Sequence Number
5	Auth Invalid	Message (Key Request) authentication failure
6	Auth Reject, EAP Establish-Key Reject	Permanent Authorization Failure
7	EAP Establish-Key Reject	Unrecognized MKID