	IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>
Title	**Enhanced EAP-based User Authentication coexisting with PKM based Device Authentication**
Date Submitted	**2004-05-04**
Source(s)	Dongkie Lee, DongIl Moon, DongRyul Lee, JongKuk Ahn, Sungho Ha SK Telecom 15F, Seoul Finance Center, 84, Taepyungpro 1 ga, Chung-gu, Seoul, 100-768, Korea Voice: +82-2-6323-3147 Fax: +82-2-6323-4493 [mailto: {galahad,dimoon,drlee,jgahn,ss23}@sktelecom.com]
Re:	Response to IEEE 802.16-04/19 (Recirculation Ballot #14a Announcement)
Abstract	To minimize impact on the current standard and achieve two-tiered device/user authentication, EAP is performed after PKM Key exchange. EAP-based User Authentication is separate from Device Authentication and operators can choose any EAP-based User Authentication method based on their needs.
Purpose	Discuss and Adopt as the enhanced authentication procedure
Notice	s 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>.

Enhanced EAP-based User Authentication coexisting with PKM based Device Authentication

Dongkie Lee, DongRyul Lee, DongIl Moon, JongKuk Ahn
SK Telecom

1. Problem Statements

With 802.11 WLAN, several EAP methods are developed and widely used due to the WEP's security weakness. It suffered also from the static key provisioning problem. That's why the so-called Dynamic WEP is introduced to WLAN. With Dynamic WEP, WEP keys are refreshed periodically using EAP-TLS, EAP-TTLS, PEAP, etc. Where Client and AAA negotiate master key, and the master key is sent to the AP from the AAA. Again as mentioned, proliferation of EAP is driven by WEP's static key provisioning problem and security weakness.

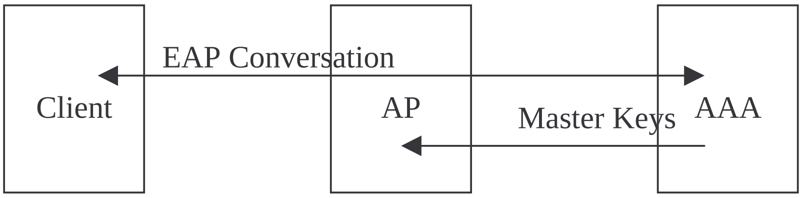


Figure 1 Transfer of EAP Master Key

But with 802.16, it's a different story. Although, PKM have somethings left to be enhanced, it's quite well defined and does not suffer from the problems of WEP. Public key systems which requires certificates of both sides such as EAP-TLS have suffered deployment and management problems. But with PKM, off-the-shelf SS and BS have embedded certificate and does not suffer from the deployment and management problem. TEK is refreshed periodically and does not suffer from the static key provisioning problem. And PKM does not suffer from the security flaw which is found in WLAN. This is the first reason EAP doesn't need not be tied to or tweaked with PKM.

Secondly EAP-MD5 does not have any master key generation mechanism and cannot be used. If we stick to PKM EAP which is proposed already, EAP-MD5 cannot be used. ANY EAP methods SHALL be supported for user authentication by IEEE standard. EAP-MD5 is not a exception.

Thirdly, if Authorization Key is derived from EAP AAA key, it'll make BS difficult to manage several timers. According to P802.16-REVd/D4, below 7 timer values are forwarded with Auth Reply message. If EAP is tweaked into PKM, 4 of these values should come from AAA and BS should parse EAP message, extract TEK-related values and somehow combine these 7 timers and forward to MSS in the Auth Reply. That is, AK related timer management entity and TEK timer management entity should be separated.

PKM configuration	Relation	
Authorize wait timeout		AK
Reauthorize wait		AK
Authorization grace		AK
Operational wait	TEK	
Rekey wait timeout	TEK	
TEK grace time	TEK	
Authorize reject wait		AK

Therefore separation of PKM and user authentication has minimal impact on the current standard and it'll benefit both operators and vendors also. According to this contribution, EAP method for user authentication shall be performed after PKM key exchange phase is complete. So shared key or master key is not transferred from AAA to BS. EAP-MD5, EAP-TTLS, EAP-TLS, EAP-AKA etc whatever may be used and is up to the operators. If operators would like to use light-weight method, there's EAP-MD5. If operators would like to use integrated method with CDMA 2000, there's EAP-AKA or EAP-CAVE. If operators would like to use certificate based authentication, there's EAP-TLS/TTLS/PEAP.

Table 1Comparison between PKM EAP and this Proposal

	ETRI & Streetwaves Networking Proposal	This Proposal
Impact on 802.16 standard	AK is derived from EAP AAA key.	PKM AK is used as is defined in standard.
EAP Usage	EAP Key Exchange and user authentication is done and AAA key is used as AK.	EAP Key Exchange and user authentication is done after PKM.
Device Authentication	Is not performed.	Is performed as is defined in standard.
AK/TEK state machine	Is managed by BS.	AK state machine is managed by AAA and TEK state machine is managed by BS. Timer values in Auth Reply should be separately defined between BS, AAA.
Mutual Authentication	depends on EAP method	depends on EAP method
BS/AAA Overhead	BS : Light AAA : Heavy	BS : Same as PKM AAA : Light
Negotiation	SBC-REQ/RSP	PKM Ciphersuite

2. Overview of Proposed Solutions

2.1 Option 1 - Alternative to ETRI Proposal

For user authentication, EAP method is performed after native PKM Key exchange is complete. There is no restriction which EAP method is used. EAP exchange may be crypto-graphically protected using Data Encryption method negotiated with PKM if Secondary Management CID is used[1]. But if EAP exchange does not require encryption for example EAP-TTLS/TLS, Primary Management CID is used. In this

[1] IETF PANA(Protocol for carrying Authentication for Network Access) WG has not yet standardized EAP over IP. But Secondary Management CID is used for messages carried IP packets. So after PANA WG standardize EAP over IP, i.e. EAP over UDP

contribution, only the Primary Management CID is used.

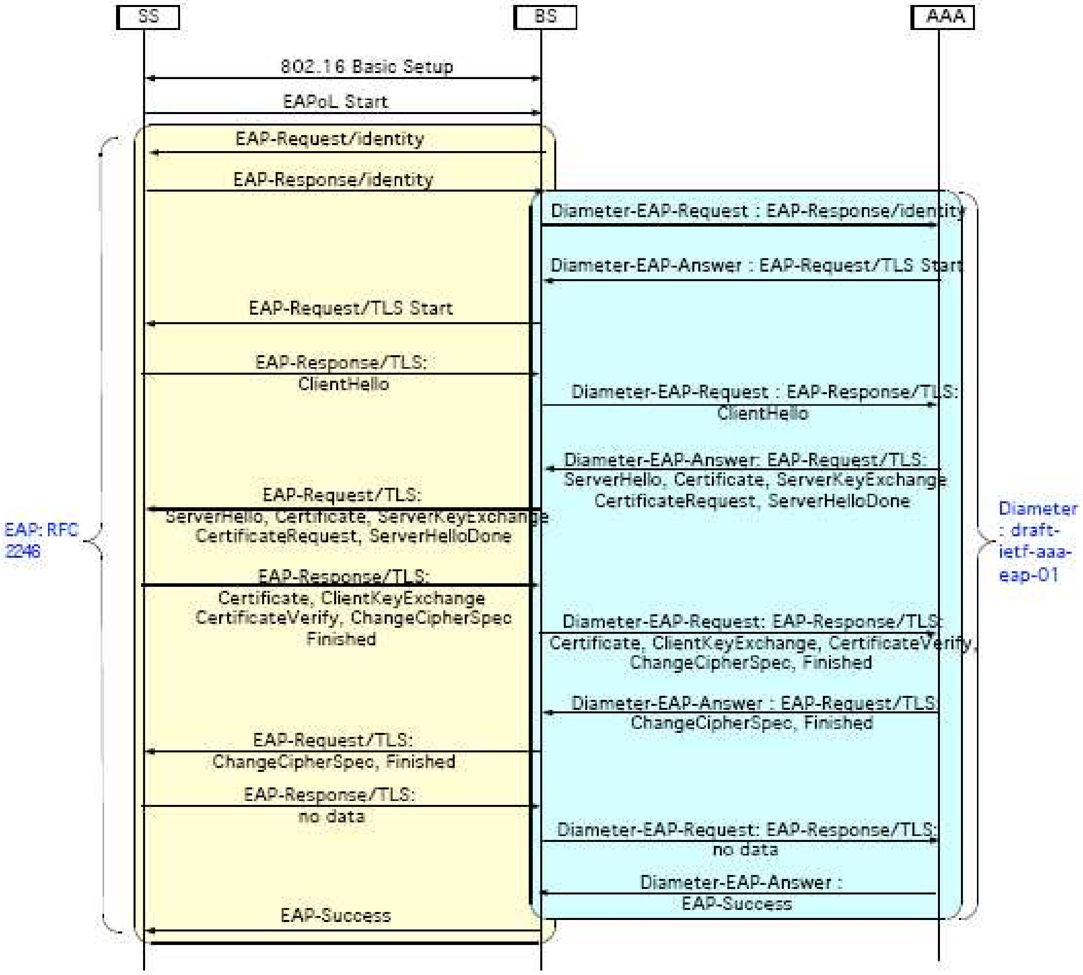
2.2 Option 2 - Coexistence with ETRI Proposal

There are two authentication scheme. One is to go with EAP-PKM. The other is to go with PKM and then EAP. With this option, only user authentication procedure is added after PKM and some modifications to PKM message arrow direction and usage.

3. Proposed Changes to IEEE 802.16e/D2

3.1 Proposed Changes for Option 1

[Reference] Call flow


SS
BS
AAA
802.16 Basic Setup
EAPoL Start
EAP-Request/identity
EAP-Response/identity
Diameter-EAP-Request : EAP-Response/identity
Diameter-EAP-Answer : EAP-Request/TLS Start
EAP-Request/TLS Start
EAP-Response/TLS: ClientHello
Diameter-EAP-Request : EAP-Response/TLS: ClientHello
Diameter-EAP-Answer: EAP-Request/TLS: ServerHello, Certificate, ServerKeyExchange CertificateRequest, ServerHelloDone
EAP-Request/TLS: ServerHello, Certificate, ServerKeyExchange CertificateRequest, ServerHelloDone
EAP: RFC 2246
Diameter : draft-ietf-aaa-eap-01
EAP-Response/TLS: Certificate, ClientKeyExchange CertificateVerify, ChangeCipherSpec Finished
Diameter-EAP-Request: EAP-Response/TLS: Certificate, ClientKeyExchange, CertificateVerify, ChangeCipherSpec, Finished
Diameter-EAP-Answer : EAP-Request/TLS: ChangeCipherSpec, Finished
EAP-Request/TLS: ChangeCipherSpec, Finished
EAP-Response/TLS: no data
Diameter-EAP-Request: EAP-Response/TLS: no data
Diameter-EAP-Answer : EAP-Success
EAP-Success

EAP Request message is sent from BS to MSS, however PKM Request message is sent from MSS to BS. So EAP Request message is not mapped to PKM Request message. If BS sends EAP Request message, it should send it in *unsolicited* PKM Response message, which is not described in the standard. It’s better to newly define EAP-REQ/RSP, which could be used for downlink/uplink and EAP-

Request/Response/Success/Failure. Finally, EAP-Success, which dose not trigger response, should also be considered in designing protocol.

6.3.2.3 MAC Management messages

[Change Table 14 as shown:]

Type	Message	Message Description	Connection
60	MOB-HO-IND	HO indication message	basic
63	EAP-REQ	EAP Request Transfer message	primary
64	EAP-RSP	EAP Response Transfer message	primary
~~61~~65-255	*reserved*		

[Change/Delete the following as shown]

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

~~*[Insert the following rows to 28 in section 6.4.2.4.9, and change the last line in the table:]*~~

~~Table 28a – PKM Message codes~~

	~~PKM Message Type~~	~~MAC Message Type~~
~~13~~	~~EAP Transfer Request~~	~~PKM-REQ~~
~~14~~	~~EAP Transfer Reply~~	~~PKM-RSP~~
	~~reserved~~	

~~*[Add the following to section 6.4.2.4.9:]*~~

~~6.3.2.3.9.11 EAP Transfer Request message~~

~~When an SS has an EAP message received from an EAP method for transmission to the BS, it encapsulates it in an EAP Transfer Request message.~~

~~Attributes are shown in Table 39a.~~

~~Table 39a-EAP Transfer Request attributes~~

~~Attribute~~	~~Contents~~
~~EAP Protocol~~	~~Contains the EAP Request, not interpreted in the MAC~~

~~The EAP Payload field carries data in the format described in RFC2284bis (see section 4).~~

~~6.3.2.3.9.12 EAP Transfer Response message~~

~~When a BS has an EAP message received from an EAP method for transmission to the SS, it encapsulates it in an EAP Transfer Request message.~~

~~Code: 14~~
~~Attributes are shown in Table 39b.~~

~~**Table 39b—EAP Transfer Response attributes**~~

~~Attribute~~	~~Contents~~
~~EAP Payload~~	~~Contains the EAP authentication data, not interpreted in the MAC~~

~~The EAP Payload field carries data in the format described in RFC2254bis (or successor RFC) section 4.~~

[Add the following before section 6.3.2.3.10 DSA-REQ messages:]

6.3.2.3.10 EAP-REQ message

A EAP-REQ is sent by a BS to carry the encapsulated EAP-request authentication data. The format of a EAP-REQ shall be as shown in Table XX.

Table XX—EAP-REQ message format

Syntax	Size	Notes
EAP-REQ_Message_Format(){		
Management Message Type = 63	8 bits	
Transaction ID	16 bits	
TLV Encoded Information	Variable	TLV specific
}		

Parameters shall be as follows:

CID (in the generic MAC header)
SS's Primary Management CID.
Transaction ID
Unique identifier for this transaction assigned by the sender.

All other parameters are coded as TLV tuples.
SAID
Security Association ID
EAP Payload
Encapsulated EAP-Request Payload
HMAC Tuple (see 11.1.2)
The HMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the EAP message's attribute list).

6.3.2.3.11 EAP-RSP message

A EAP-RSP shall be generated in response to a received EAP-REQ and contains EAP-response. When EAP-REQ is received which does not trigger EAP-RSP, i.e. EAP-Success/Failure, BS sends EAP-RSP with no EAP payload.

Table XX—EAP-RSP message format

Syntax	Size	Notes
EAP-REQ_Message_Format(){		
Management Message Type = 64	8 bits	
Transaction ID	16 bits	
TLV Encoded Information	Variable	TLV specific
}		

Parameters shall be as follows:
CID (in the generic MAC header)
SS's Primary Management CID.

Transaction ID
Transaction ID from corresponding EAP-REQ.

All other parameters are coded as TLV tuples.

SAID
Security Association ID

EAP Payload
Encapsulated EAP-Response Payload, or Null if EAP-REQ contains EAP-Success/Failure

HMAC Tuple (see 11.1.2)
The HMAC Tuple attribute contains a keyed message digest (to authenticate the sender). The HMAC Tuple attribute shall be the final attribute in the EAP message’s attribute list).

[Change the following as shown below:]

7. Privacy sublayer

The privacy sublayer provides subscribers with privacy, authentication or confidentiality across the broadband wireless network. It does this by applying cryptographic transforms to MPDUs carried across connections between SS and BS.

In addition, Privacy provides operators with strong protection from theft of service. The BS protects against unauthorized access to these data transport services by securing the associated service flows across the network. Privacy employs an authenticated client/server key management protocol in which the BS, the server, controls distribution of keying material to client SS. Additionally, the basic privacy mechanisms are strengthened by adding digital-certificate-based SS device-authentication to its key management protocol.

7.1 Architecture

Privacy has ~~two~~three component protocols as follows:

a) An encapsulation protocol for securing packet data across the fixed BWA network.This protocol defines (1) a set of supported *cryptographic suites*, i.e., pairings of data encryption and authentication algorithms, and (2) the rules for applying those algorithms to a MAC PDU payload.

b) A key management protocol (PKM) providing the secure distribution of keying data from BS to SS. Through this key management protocol, SS and BS synchronize keying data; in addition, the BS uses the protocol to enforce conditional access to network services.

c) A user authentication protocol(EAP) providing the user authentication using EAP. Through user authentication, BS authenticates MSS and MSS may authenticates BS based on the EAP method.

7.1.1 Packet data encryption

Encryption services are defined as a set of capabilities within the MAC Privacy Sublayer. MAC Header information specific to encryption is allocated in the generic MAC header format.

Encryption is applied to the MAC PDU payload when required by the selected ciphersuite; the generic MAC header is not encrypted. All MAC management messages described n subclause 6.4.2.3 shall be sent in the clear to facilitate registration, ranging, and normal operation of the MAC. The format of MAC PDUs carrying secured packet data payloads is specified in 6.4.3.6.

7.1.2 Key management protocol

The PKM protocol facilitates mutual authentication of the SS and BS, as well as distribution of traffic keying material from the BS to the SS. It also supports periodic reauthentication/reauthorization and key refresh. The key management protocol uses ~~either EAP [IETF RFC 2284], or~~ X.509 digital certificates [IETF RFC 3280] together with RSA public-key encryption algorithm [PKCS #1] to perform authentication. It uses strong symmetric algorithms to perform key exchanges between SS and BS.

The PKM’s authentication protocol establishes a shared secret (i.e., an AK) between SS and BS. The shared secret is then used to secure subsequent PKM exchanges of TEKs. This two-tiered mechanism for key distribution permits refreshing of TEKs without incurring the overhead of computation-intensive public-key operations.

A BS authenticates a client SS during the initial authorization exchange. Each SS presents its credentials, which will be a unique X.509 digital certificate issued by the SS’s manufacturer ~~(in the case of RSA authentication) or a vendor-specific credential (in the case of EAP-based authentication)~~.

The BS associates an SS’s authenticated identity to a paying subscriber, and hence to the data services that subscriber is authorized to access. Thus, with the AK exchange, the BS establishes an authenticated identity of a client SS and the services (i.e., specific TEKs) the SS is authorized to access.

Since the BS authenticates the SS, it can protect against an attacker employing a cloned SS, masquerading as a legitimate subscriber’s SS.

The traffic-key management portion of the PKM protocol adheres to a client/server model, where the SS (a PKM “client,”) requests keying material, and the BS (a PKM “server”) responds to those requests, ensuring that individual SS clients receive only keying material for which they are authorized.

The PKM protocol uses MAC management messaging, i.e., PKM-REQ and PKM-RSP messages defined in 6.4.2.3. The PKM protocol is defined in detail in 7.2.

7.1.3 Authentication Protocol

An SS uses the PKM protocol to obtain authorization and traffic keying material from the BS, and to support periodic reauthorization and key refresh.

~~PKM supports two distinct authentication protocol mechanisms:~~

- ~~RSA [PKCS #1] (support is mandatory in all devices)~~
- ~~Extensible Authentication Protocol (support is optional as described in xx)~~

~~7.1.3.1 PKM RSA Authentication~~

The PKM ~~RSA~~ authentication protocol uses X.509 digital certificates [IETF RFC 3280], the RSA public-key encryption algorithm [PKCS #1].

A BS authenticates a client SS during the initial authorization exchange. Each SS carries a unique X.509 digital certificate issued by the SS’s manufacturer. The digital certificate contains the SS’s Public Key and SS MAC address. When requesting an AK, an SS presents its digital certificate to the BS. The BS verifies the digital certificate, and then uses the verified Public Key to encrypt an AK, which the BS then sends back to the requesting SS.

All SSs shall have factory-installed RSA private/public key pairs or provide an internal algorithm to generate such key pairs dynamically. If an SS relies on an internal algorithm to generate its RSA key pair, the SS shall generate the key pair prior to its first AK exchange, described in 7.2.1. All SSs with factory-installed RSA key pairs shall also have factory-installed X.509 certificates. All SSs that rely on internal algorithms to generate an RSA key pair shall support a mechanism for installing a manufacturer-issued X.509 certificate following key generation.

~~7.1.3.2 PKM EAP Authentication~~

~~PKM EAP Authentication uses Extensible Authentication Protocol [IETF RFC 2284bis] in conjunction with a vendor-selected standardized EAP Method (eg. EAP-TLS [IETF RFC 2716]). The EAP method will use a particular kind of credential – such as an x.509 certificate in the case of EAP-TLS, or a Subscriber Information Module in the case of EAP-SIM (Draft xxxxx). The particular credentials and EAP methods that are to be used are outside of the scope of this specification, but they should be selected with awareness of the security issues described in [IETF RFC 2284bis] section 7. Figure xx shows the relationship between the lower levels of the 802.16 MAC and the generic EAP components (and the interface between them).~~

7.2 PKM protocol

[Change the baseline document as indicated below:]

7.2.1 SS authorization and AK exchange overview

SS authorization, controlled by the Authorization state machine, is the process of

a) the BS authenticating a client SS's identity

b) the BS and SS establishing a shared providing the authenticated SS with an AK, from which a key encryption key (KEK) and message authentication keys are derived

c) the BS providing the authenticated SS with the identities (i.e., the SAIDs) and properties of primary and static SAs the SS is authorized to obtain keying information for

After achieving initial authorization, an SS periodically reauthorizes with the BS; reauthorization is also managed by the SS's Authorization state machine. TEK state machines manage the refreshing of TEKs.

~~7.2.1.1 Authorization via PKM RSA Authentication Protocol~~

An SS begins authorization by sending an Authentication Information message to its BS. The Authentication Information message contains the SS 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 SS.

The SS 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 SS 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 SS supports; an SS'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 SS supports

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

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

a) an AK encrypted with the SS'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 SS 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 SS's Basic CID, the Authorization Reply shall not identify any Dynamic SAs.

The BS, in responding to an SS's Authorization Request, shall determine whether the requesting SS, 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 SS's user has subscribed for. Note that the protected services a BS makes available to a client SS can depend upon the particular cryptographic suites SS and BS share support for.

An SS shall periodically refresh its AK by reissuing an Authorization Request to the BS. Reauthorization is identical to authorization with the exception that the SS 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 SS's AKs have overlapping lifetimes. Both SS 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 SS's AKs (see 7.4), ensures that the SS can refresh.

~~**7.2.1.2 Authorization via PKM Extensible Authentication Protocol**~~

~~The first steps of the authorization flow are as follows:~~

~~1) 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.~~

~~2) 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.~~

~~3) 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.~~

~~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:~~

~~4) Upon success, EAP on the authenticator transmits a “success” signal on the logical control interface to fully activate the airlink.~~

~~5) EAP on the authenticator transmits EAP-success, which is then encapsulated in a MAC management message and transmitted to the supplicant.~~

~~6) EAP on the supplicant transmits a “success” indication on the logical control interface to fully activate the airlink.~~

~~7) 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:~~

~~8) BS sends Auth-handshake msg (structure TBD) to SS to supply a nonce, and includes its ciphersuite capabilities and a list of SAIDs that are available to the SS.~~

~~9) SS sends Auth-handshake reply msg (structure TBD) to supply its nonce and includes an HMAC based on its TBD-function derived AK~~

~~The Authorization Key (AK) is derived from the AAA-Key (Derivation algorithm TBD)~~

[Add Section 7.6 after 7.5.6 Digital Signatures]

7.6 User Authentication

After SS authorization and AK exchange is complete, a BS begins user authentication by sending an EAP Request Transfer message to SS. After several EAP Request Transfer and EAP Response Transfer message exchanges, EAP Success or Failure message is sent to the SS from the BS. If user authentication fails, BS terminates the whole session information related with that SS or may retain the session information. Which EAP method is used depends on operator’s needs requirements and any method is not excluded by this standard.

3.2 Proposed Changes for Option 2

[Change/Delete the following as shown]

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

PKM employs two MAC message types: PKM Request (PKM-REQ) and PKM Response (PKM-RSP), as described in Table 24.

Table 24—**PKM MAC messages**

Type Value	Message name	Message description
9	PKM-REQ	Privacy Key Management Request [SS <-> BS]
10	PKM-RSP	Privacy Key Management Response [BS < -> SS]

These MAC management message types distinguish between PKM requests (SS–to–BS, or BS-to-SS) and PKM responses (BS–to–SS, or SS-to-BS). Each message encapsulates one PKM message in the Management Message Payload.

PKM request~~ protocol~~ messages~~ transmitted from the SS to the BS~~ shall use the form shown in Table 25. They are transmitted on the SSs Primary Management Connection.

PKM response~~protocol~~ messages~~ transmitted from the BS to the SS~~ shall use the form shown in Table 26. They are transmitted on the SSs Primary Management Connection.

Table 25—PKM request (PKM-REQ) message format

Table 26—PKM response (PKM-RSP) message format

The parameters shall be as follows:

Code

The Code is one byte and identifies the type of PKM packet. When a packet is received with an invalid Code, it shall be silently discarded. The code values are defined in Table 27.

PKM Identifier

The Identifier field is one byte. ~~An ~~MSS and BS uses the identifier to match a ~~BS ~~response to the~~ SS's~~ requests.

The MSS and the BS shall increment (modulo 256) the Identifier field whenever ~~it~~they issue~~s~~ a new PKM message. A "new" message is an Authorization Request,~~ or~~ Key Request or EAP Request/Success/Failure Transfer that is not a retransmission being sent in response to a Timeout event. For retransmissions, the Identifier field shall remain unchanged.

The Identifier field in Authentication Information messages, which are informative and do not effect any response messaging, shall be set to zero. The Identifier field in a ~~BS's ~~PKM-RSP message shall match the Identifier field of the PKM-REQ message~~ the BS is responding to~~. The Identifier field in TEK Invalid messages, which are not sent in response to PKM-REQs, shall be set to zero. The Identifier field in unsolicited Authorization Invalid messages shall be set to zero.

On reception of a PKM-RSP message, the SS associates the message with a particular state machine (the Authorization state machine in the case of Authorization Replies, Authorization Rejects, and Authorization Invalids; a particular TEK state machine in the case of Key Replies, Key Rejects, and TEK Invalids).

An SS shall keep track of the identifier of its latest, pending Authorization Request. The SS shall discard Authorization Reply and Authorization Reject messages with Identifier fields not matching that of the pending Authorization Request.

An SS shall keep track of the identifiers of its latest, pending Key Request for each SA. The SS shall discard Key Reply and

Key Reject messages with Identifier fields not matching those of the pending Key Request messages.

Attributes

PKM attributes carry the specific authentication, authorization, and key management data exchanged between client and server. Each PKM packet type has its own set of required and optional attributes. Unless explicitly stated, there are no requirements on the ordering of attributes within a PKM message. The end of the list of attributes is indicated by the LEN field of the MAC PDU header.

Table 28a – PKM Message codes

	PKM Message Type	MAC Message Type
13	EAP Transfer Request	PKM-REQ
14	EAP Transfer Reply	PKM-RSP
15	EAP Transfer Success	PKM-REQ
16	EAP Transfer Failure	PKM-REQ
	reserved	

[Add the following to section 6.4.2.4.9:]

6.3.2.3.9.11 EAP ~~Transfer~~ Request Transfer message

When a~~n SS~~ BS has an EAP message received from an EAP method for transmission to the MSS~~BS~~, it encapsulates it in an EAP ~~Transfer~~ Request Transfer message.

Attributes are shown in Table 39a.

Table 39a-EAP ~~Transfer~~ Request Transfer attributes

Attribute	Contents
EAP Protocol	Contains the EAP Request, not interpreted in the MAC

The EAP Payload field carries data in the format described in RFC2284bis (see section 4).

6.3.2.3.9.12 EAP ~~Transfer~~ Response Transfer message

When a MSS~~BS~~ has an EAP message received from an EAP method for transmission to the BS~~SS~~, it encapsulates it in an EAP ~~Transfer~~ Response Transfer message.

Code: 14
Attributes are shown in Table 39b.

Table 39b—EAP ~~Transfer~~ Response Transfer attributes

Attribute	Contents
EAP Payload	Contains the EAP authentication data, not interpreted in the MAC

The EAP Payload field carries data in the format described in RFC2254bis (or successor RFC) section 4.

7.6 User Authentication

When PKM version 2 is negotiated which includes User Authentication, MSS and BS perform user authentication using EAP method. Based on to the operator's local policy, BS may not allow subsequent connection establishment if MSS denies user authentication.

After SS authorization and AK exchange is complete, a BS begins user authentication by sending an EAP Request Transfer message to SS. After several EAP Request Transfer and EAP Response Transfer message exchanges, EAP Success or Failure message is sent to the SS from the BS. On receiving EAP Success or Failure Transfer message, MSS responds with PKM-RSP which has no EAP Payload.

If user authentication fails, BS terminates the whole session information related with that SS or may retain the session information. Which EAP method is used depends on operator's needs requirements and any method is not excluded by this standard.

11.9.16 Version

Table 333—Version attribute values

Value	**Description**
0	*Reserved*
1	PKM (Initial standard release)
2	**PKM and EAP-based User Authentication**
~~2~~3-255	*Reserved*