Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Corrections for the MS's Authorization Flow**
Data Submitted	**2005-05-04**
Source(s)	Seokheon Cho Sungcheol Chang Chulsik Yoon, ETRI 161, Gajeong-dong, Yuseong-Gu, Daejeon, 305-350, Korea Voice: +82-42-860-5524 Fax: +82-42-861-1966 chosh@etri.re.kr
Re:	IEEE P802.16e/D7
Abstract	The existing PKMv2 is somewhat unorganized and insecure security framework. This contribution provides a resolution for unorganized and insecure MS's authorization flow in the PKMv2.
Purpose	Adoption of proposed changes into P802.16e/D7
Notice	
Release	
Patent Policy and Procedures	

Corrections for the MS's Authorization Flow

Seokheon Cho, Sungcheol Chang, and Chulsik Yoon
ETRI

Introduction

The existing PKMv2 is somewhat in disorder and provides unorganized and insecure security framework.
This contribution supports the backward compatibility with the PKMv1 and security framework of the PKMv2.

This contribution provides a resolution for those problems in the PKMv2.

0.1 IEEE P802.16e/D7 Status

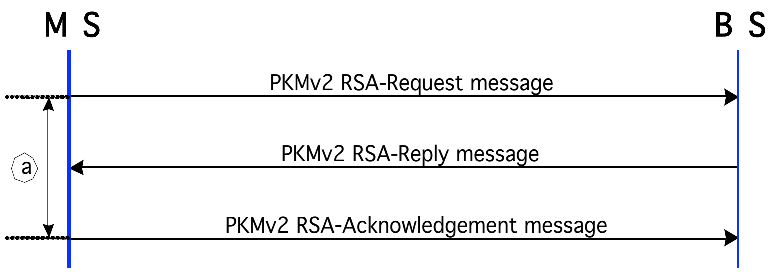
The AK is derived from the PAK or/and the PMK. The PAK and the PMK are obtained from the RSA-based Authorization procedure and the EAP-based Authorization procedure, respectively.

0.2 Problems

- For the RSA-based Authorization procedure:
  - The Authorization Request message and the Authorization Reply message in the RSA-based Authorization procedure are in the same rank as the EAP Transfer message in the EAP-based Authorization procedure. The Authorization Request/Reply messages contain Security-Capabilities, SAID, and SA-Descriptors, but the EAP Transfer message doesn't contain those parameters.
  - The sequence number and lifetime included in the RSA-based Authorization procedure are not an AK sequence number and AK lifetime.
  - When the RSA-based Authorization procedure is selected, there is no message for informing the MS's authentication failure from BS to MS and the BS's authentication result (such as success or reject) from MS.
- For the EAP-based Authorization procedure:
  - The Key Sequence Number used in the Authenticated EAP-Transfer message is not an AK Sequence Number.
  - The BS doesn't know the completion time of the EAP-based Authorization procedure in case that EAP protocol doesn't yield AAA-key and whether an MS receives the last EAP Transfer message (such as "EAP Success" used in EAP-TLS) or not. Both the BS and the MS cannot simultaneously share the AK derived from PMK, when an MS doesn't receive the last EAP Transfer message.
- AK generation process needs more secure mechanism.

0.3 Solutions

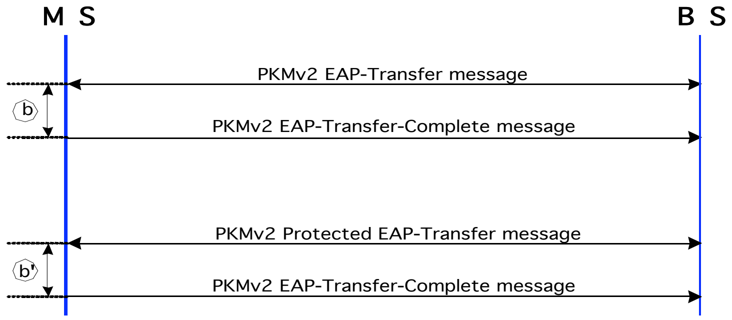
a) For the RSA-based Authorization procedure:
  - Since the RSA-based Authorization is the same rank as the EAP-based Authorization, several attributes (such as Security-Capabilities, SAID, and SA-Descriptors) which are present in the EAP-based Authorization procedure, are omitted in the RSA-based Authorization procedure. The RSA-based Authorization procedure still supports mutual authentication. The messages in the RSA-based Authorization procedure are as follows:
  - The sequence number and lifetime included in the RSA-based Authorization procedure is not an AK sequence number and AK lifetime but the PAK sequence number and PAK lifetime.
  - To inform the MS's authentication failure from BS to MS, the PKMv2 RSA-Reject message is added. Also, to inform the BS's authentication result (such as success or reject) from MS to BS, Auth Result Code and Error-Code attributes shall be included in the PKMv2 RSA-Acknowledgement message.



i. PKMv2 RSA-Request message: MS_Random, MS_Cerfificate
ii. PKMv2 RSA-Reply message: MS_Random, BS_Random, Encrypted pre-PAK, Key lifetime (PAK), Key Sequence Number (PAK), BS_Certificate, SigBS

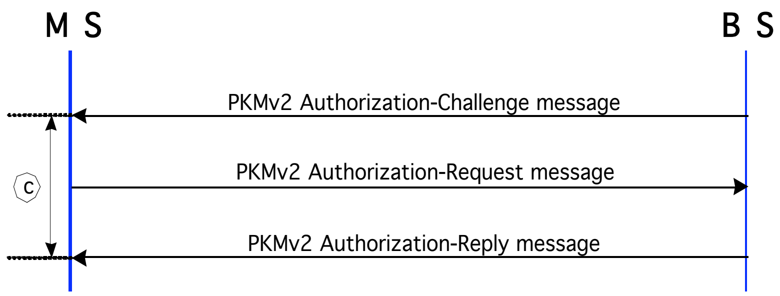
iii. PKMv2 RSA-Reject message: MS_Random, BS_Random, Error-Code, Display-String, SigBS
iv. PKMv2 RSA-Acknowledgement message: BS_Random, Auth Result Code, Error-Code, Display-String, SigMS

b) For the EAP-based Authorization procedure:
- The messages used in the EAP-based Authorization procedure are as follows
- The Key Sequence Number used in the PKMv2 Authenticated EAP-Transfer message is not an AK sequence number but PAK sequence number.
- In order that BS knows the completeness time of EAP-based Authorization procedure in case that EAP protocol doesn't yield AAA-key and whether a MS receives the last EAP Transfer message (such as "EAP Success" used in EAP-TLS) or not, an MS shall send the PKMv2 EAP-Transfer-Complete message to report EAP-based Authorization completeness. Therefore, both MS and BS can synchronize the AK.



i. PKMv2 EAP-Transfer message: EAP Payload
ii. PKMv2 Authenticated EAP-Transfer message: Key Sequence Number (PAK), EAP Payload, OMAC Digest (from EIK)
iii. PKMv2 EAP-Transfer-Complete message:

c) For MS's Authorization Key Generation procedure:
- To assure more secure AK, seeds (such as MS_Nonce and BS_Nonce) should be used to derive AK.
- To protect from replay-attack, the messages needed in MS's AK Generation procedure should contains random number (such as MS_Nonce and BS_Nonce) and message authentication function (such as OMAC Digest).
- The messages with important parameters, e.g. Security-Capabilities, SAID, and SA-Descriptors, should be authenticated.
- This procedure supports secure 3 way handshake.



i. PKMv2 Authorization-Challenge message: BS_Nonce
ii. PKMv2 Authorization-Request message: Key Sequence Number (PAK), MS_Nonce, BS_Nonce, Security_Capabilities, SAID, OMAC Digest (from AK)
iii. PKMv2 Authorization-Reply message: Key Sequence Number (AK), Key Lifetime (AK), BS_Nonce, (one or more) SA-Descriptor(s), OMAC Digest (from AK)
iv. PKMv2 Authorization-Reject message: Error-Code, Display-String, BS_Nonce, OMAC Digest (from AK)

Proposed Changes into IEEE P802.16e/D7

[Change sub-clauses 6.3.2.3.9.15 as follows]

6.3.2.3.9.12 Auth-Request message

6.3.2.3.9.11 PKMv2 RSA-Request message

A client MS sends a PKMv2 RSA-Request message to the BS in order to request mutual authentication in the RSA-based authorization.

Code: 21 13

Attributes are shown in Table 37e 37a.

Table 37e 37a-Auth-Request attributes PKMv2 RSA-Request attributes

Attribute	Contents
SS_Random	A 64 bit random number generated in the MS
SS_Certificate	Contains the MS's X.509 user certificate
Security_Capabilities	Describes requesting MS's security capabilities
AAID/SAID	Either the AAID or the Basic CID if in initial network entry
SigSS	An RSA signature over all the other attributes in the message

The MS-certificate attribute contains an X.509 MS certificate (see 7.6) issued by the MS's manufacturer. The MS's X.509 certificate and Security Capabilities attribute is as defined in 6.3.2.3.9.2.

The SigSS indicates an RSA signature over all the other attributes in this message, and the SS's private key is used to make an RSA signature.

[Change sub-clauses 6.3.2.3.9.16 as follows]

6.3.2.3.9.13 Auth-Reply message

6.3.2.3.9.12 PKMv2 RSA-Reply message

Sent by the BS to a client MS in response to an Authorization Request a PKMv2 RSA-Request message, the Authorization Reply PKMv2 RSA-Reply message contains an AK an encrypted pre-PAK, the key's lifetime, and the key's sequence number, and a list of SA-Descriptors identifying the Primary and Static SAs the requesting MS is authorized to access and their particular properties (e.g., type, cryptographic suite). The AK pre-PAK shall be encrypted with the MS's public key. The SA-Descriptor list shall include a descriptor for the Basic CID reported to the BS in the corresponding Auth-Request. The SS_Random number is returned from the auth-req PKMv2 RSA-Request message, along with a random number supplied by the BS, thus enabling assurance of key liveness.

Code: 22 14

Attributes are shown in Table 37f 37b.

Table 37f 37b-Auth-Reply attributes PKMv2 RSA-Reply attributes

Attribute	Contents
MS_Random	A 64 bit random number generated in the MS
BS_Random	A 64 bit random number generated in the BS
Encrypted pre-PAK	RSA-OAEP-Encrypt(PubKey(MS), pre-PAK|Id(MS) SS ID)
Key Lifetime	AK PAK Aging timer
Key Sequence Number	64 bit AK PAK sequence number
(one or more) SA-Descriptor(s)	The primary SA and zero or more static SAs.Each compound SA-Descriptor attribute specifies an SAID and additional properties of the SA (optional, only if there is no EAP phase afterwards)
CertBS BS_Certificate	The BS Certificate Contains the BS's X.509 certificate
SigBS	An RSA signature over all the other attributes in the message

The SigBS indicates an RSA signature over all the other attributes in this message, and the BS's private key is used to make an RSA signature.

[Insert the following sub-clause in 6.3.2.3.9:]

6.3.2.3.9.13 PKMv2 RSA-Reject message

The BS responds to an SS's authorization request with an PKMv2 RSA-Reject message if the BS rejects the SS's authorization request.

Code: 15

Attributes are shown in Table 37c.

Table 37c-PKMv2 RSA-Reject attributes

Attribute	Contents
MS_Random	A 64 bit random number generated in the MS
BS_Random	A 64 bit random number generated in the BS
Error-Code	Error code identifying reason for rejection of authorization request
Display-String (optional)	Display string providing reason for rejection of authorization request
SigBS	An RSA signature over all the other attributes in the message

The Error-Code and Display-String attributes describe to the requesting MS the reason for the RSA-based authorization failure.

The SigBS indicates an RSA signature over all the other attributes in this message, and the BS's private key is used to make an RSA signature.

[Insert the following sub-clause in 6.3.2.3.9:]

6.3.2.3.9.14 PKMv2 RSA-Acknowledgement message

The MS sends the PKMv2 RSA-Acknowledgement message to BS in response to a PKMv2 RSA-Reply message or a PKMv2 RSA-Reject message. Only if the value of Auth Result Code is failure, then the Error-Code and Display-String can be included in this message.

Code: 16

Attributes are shown in Table 37d.

Table 37d-PKMv2 RSA-Acknowledgement attributes

Attribute	Contents
BS_Random	A 64 bit random number generated in the BS
Auth Result Code	Indicates result (Success or Failure) of authorization procedure.
Error-Code	Error code identifying reason for rejection of authorization request
Display-String (optional)	Display string providing reason for rejection of authorization request
SigSS	An RSA signature over all the other attributes in the message

The SigSS indicates an RSA signature over all the other attributes in this message, and the SS's private key is used to make an RSA signature.

[Change sub-clauses 6.3.2.3.9.11 as follows]

~~6.3.2.3.9.11 EAP Transfer message~~

6.3.2.3.9.15 PKMv2 EAP-Transfer message

When an MS has an EAP message received from an EAP method for transmission to the BS or when a BS has an EAP message received from an EAP method for transmission to the MS, it encapsulates it ~~in an EAP Transfer~~ a PKMv2 EAP Transfer message.

Code: ~~13~~ 18

Attributes are shown in Table ~~37a~~ 37e.

Table ~~37a~~ 37e ~~–EAP Transfer attributes~~ PKMv2 RSA-Acknowledgement 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 section 4 of RFC 2284bis.

[Change sub-clauses 6.3.2.3.9.18 as follows]

~~6.3.2.3.9.15 Authenticated EAP message~~

6.3.2.3.9.16 PKMv2 Authenticated EAP-Transfer message

If EIK is available and an MS or BS has an EAP message received from an EAP method for transmission, it encapsulates EAP message in ~~a Authenticated EAP Transfer message~~ a PKMv2 Authenticated EAP Transfer message. In other words, this message may be used in case that both an MS and BS negotiate RSA-based authorization and Authenticated EAP-based authorization as authorization policy support.

Code: ~~24~~ 19

Attributes are shown in Table ~~37h~~ 37f.

Table ~~37h~~ 37f ~~–Authenticated EAP message attributes~~ PKMv2 Authenticated EAP Transfer attributes

Attribute	Contents
Key Sequence Number	~~AK~~ PAK Sequence Number
EAP Payload	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 MS 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~~ EIK.

[Insert the following sub-clause in 6.3.2.3.9:]

6.3.2.3.9.17 PKMv2 EAP Transfer-Complete message

A MS sends the PKMv2 EAP-Transfer-Complete message to the BS to report completeness of EAP-based authorization procedure (PKMv2 EAP-Transfer message and PKMv2 Authenticated EAP-Transfer message). This message doesn't contain any attributes.

Code: 20

Attributes are shown in Table 37g

Table 37g- PKMv2 EAP-Transfer-Complete attributes

Attribute	Contents

Key Sequence Number	AK sequence number
MS_ Nonce	A 64 bit random number generated in a MS
OMAC Packet Number Counter	Sequential counter for OMAC packet number
HMAC Digest/OMAC Digest	Message Digest calculated using AK

The MS_Nonce shall be included to protect the replay attack, when the HMAC-Digest is included in the message.

If the OMAC-Digest is included in the message, then the OMAC Packet Number Counter should be also included.

The HMAC-Digest attribute or the OMAC-Digest attribute shall be the final attribute in the message's attribute list.

Inclusion of the HMAC-Digest or the OMAC digest allows the MS and BS to authenticate the PKMv2 Key-Request message. The HMAC-Digest or the OMAC-Digest's authentication key is derived from the AK.

[Delete 7.2.2.1] and [Insert 7.2.2.1 as follows] ***(Informative: Add the deleted contents in 7.2.2.1 into 7.2.2.3)***

7.2.2.1 Security Associations

Upon achieving authorization, an MS starts a separate TEK state machine for each of the SAIDs identified in the Authorization Reply message. Each TEK state machine operating within the MS is responsible for managing the keying material associated with its respective SAID. TEK state machines periodically send Key Request messages to the BS, requesting a refresh of keying material for their respective SAIDs.

The BS responds to a Key Request with a Key Reply message, containing the BS's active keying material for a specific SAID.

The TEK is encrypted using appropriate KEK derived from the AK.

TEKs and KEKs may be either 64 bits or 128 bits long. SAs employing any ciphersuite with a basic block size of 128 bits shall use 128 bit TEKs and KEKs. Otherwise 64 bit TEKs and KEKs shall be used. The name TEK-64 is used to denote a 64 bit TEK and TEK-128 is used to denote a 128 bit TEK. Similarly, KEK-64 is used to denote a 64 bit KEK and KEK-128 is used to denote a 128 bit KEK.

For SAs using a ciphersuite employing DES-CBC, the TEK in the Key Reply is triple DES (3-DES) (encrypt-decrypt-encrypt or EDE mode) encrypted, using a two-key, 3-DES KEK derived from the AK.

For SAs using a ciphersuite employing 128 bits keys, such as AES-CCM mode, the TEK in the key Reply is AES encrypted using a 128 bit key derived from the AK and a 128 bit block size.

Note that at all times the BS maintains two active sets of keying material per SAID. The lifetimes of the two generations overlap such that each generation becomes active halfway through the life of it predecessor and expires halfway through the life of its successor. A BS includes in its Key Replies both of an SAID's active generations of keying material.

The Key Reply provides the requesting SS, in addition to the TEK and CBC initialization vector, the remaining lifetime of each of the two sets of keying material. The receiving SS uses these remaining lifetimes to estimate when the BS will invalidate a particular TEK, and therefore when to schedule future Key Requests such that the SS requests and receives new keying material before the BS expires the keying material the SS currently holds.

For SAs using a ciphersuite employing CBC mode encryption the Key Reply provides the requesting MS, in addition to the TEK and CBC initialization vector, the remaining lifetime of each of the two sets of keying material. For SAs using a ciphersuite employing AES-CCM mode, the Key Reply provides the requesting MS, in addition to the TEK, the remaining lifetime of each of the two sets of keying material. The receiving MS uses these remaining lifetimes to estimate when the BS will invalidate a particular TEK, and therefore when to schedule future Key Requests such that the MS requests and receives new keying material before the BS expires the keying material the MS currently holds. For AES-CCM mode, when more than half the available PN numbers in the 31 bit PN number space are exhausted, the MS shall schedule a future Key Request in the same fashion as if the key lifetime was approaching expiry.The operation of the TEK state machine's Key Request scheduling algorithm, combined with the BS's regimen for updating and using an SAID's keying material (see 7.4), ensures that the MS will be able to continually exchange encrypted traffic with the BS.

The operation of the TEK state machine's Key Request scheduling algorithm, combined with the BS's regimen for updating and using an SAID's keying material (see 7.4), ensures that the SS will be able to continually exchange encrypted traffic with the BS.

~~A TEK state machine remains active as long as~~

~~a) the MS is authorized to operate in the BS's security domain, i.e., it has a valid AK, and~~

~~b) the MS is authorized to participate in that particular SA, i.e., the BS continues to provide fresh keying material during rekey cycles.~~

[Add the whole contents in 7.2.2.1 into 7.2.2.3 as follows]

7.2.2.3 ~~Associations~~ Security Associations

Upon achieving authorization, an MS starts a separate TEK state machine for each of the SAIDs identified in the Authorization Reply message. Each TEK state machine operating within the MS is responsible for managing the keying material associated with its respective SAID. TEK state machines periodically send Key Request messages to the BS, requesting a refresh of keying material for their respective SAIDs.

The BS responds to a Key Request with a Key Reply message, containing the BS's active keying material for a specific SAID.

~~The TEK is encrypted using appropriate KEK derived from the AK.~~

TEKs and KEKs may be either 64 bits or 128 bits long. SAs employing any ciphersuite with a basic block size of 128 bits shall use 128 bit TEKs and KEKs. Otherwise 64 bit TEKs and KEKs shall be used. The name TEK-64 is used to denote a 64 bit TEK and TEK-128 is used to denote a 128 bit TEK. Similarly, KEK-64 is used to denote a 64 bit KEK and KEK-128 is used to denote a 128 bit KEK.

For SAs using a ciphersuite employing DES-CBC, the TEK in the Key Reply is triple DES (3-DES) (encrypt-decrypt-encrypt or EDE mode) encrypted, using a two-key, 3-DES KEK derived from the AK.

For SAs using a ciphersuite employing 128 bits keys, such as AES-CCM mode, the TEK in the key Reply is AES encrypted using a 128 bit key derived from the AK and a 128 bit block size.

Note that at all times the BS maintains two active sets of keying material per SAID. The lifetimes of the two generations overlap such that each generation becomes active halfway through the life of it predecessor and expires halfway through the life of its successor. A BS includes in its Key Replies both of an SAID's active generations of keying material.

~~The Key Reply provides the requesting SS, in addition to the TEK and CBC initialization vector, the remaining lifetime of each of the two sets of keying material. The receiving SS uses these remaining lifetimes to estimate when the BS will invalidate a particular TEK, and therefore when to schedule future Key Requests such that the SS requests and receives new keying material before the BS expires the keying material the SS currently holds.~~

For SAs using a ciphersuite employing CBC mode encryption the Key Reply provides the requesting MS, in addition to the TEK and CBC initialization vector, the remaining lifetime of each of the two sets of keying material. For SAs using a ciphersuite employing AES-CCM mode, the Key Reply provides the requesting MS, in addition to the TEK, the remaining lifetime of each of the two sets of keying material. The receiving MS uses these remaining lifetimes to estimate when the BS will invalidate a particular TEK, and therefore when to schedule future Key Requests such that the MS requests and receives new keying material before the BS expires the keying material the MS currently holds. For AES-CCM mode, when more than half the available PN numbers in the 31 bit PN number space are exhausted, the MS shall schedule a future Key Request in the same fashion as if the key lifetime was approaching expiry.The operation of the TEK state machine's Key Request scheduling algorithm, combined with the BS's regimen for updating and using an SAID's keying material (see 7.4), ensures that the MS will be able to continually exchange encrypted traffic with the BS.

~~The operation of the TEK state machine's Key Request scheduling algorithm, combined with the BS's regimen for updating and using an SAID's keying material (see 7.4), ensures that the SS will be able to continually exchange encrypted traffic with the BS.~~

A TEK state machine remains active as long as

a) the MS is authorized to operate in the BS's security domain, i.e., it has a valid AK, and

b) the MS is authorized to participate in that particular SA, i.e., the BS continues to provide fresh keying material during rekey cycles.

Keying material is held within associations. There are three types of association: The security associations (SA) that maintain keying material for unicast connections, group security associations (GSA) that hold keying material for multicast groups and MBSGSAs which hold keying material for MBS services.