Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Clarification on the Key Hierarchy for the PKMv2**
Data Submitted	**2005-06-08**

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/D8

Abstract	Both an MS and the BS can share the PAK from the RSA-based authorization and the PMK from the EAP-based authorization in the PKMv2. Two keys, the PAK and the PMK, are used to derive the AK. The PAK is used as input data, however, the PMK is used as input key. Since the PAK and the PMK are root keys to derive the AK, both of them should be used as not input data but input keys. This contribution provides key hierarchy for the PKMv2.

Purpose	Adoption of proposed changes into P802.16e/D8

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:chiar@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>.

Clarification on the Key Hierarchy for the PKMv2

Seokheon Cho, Sungcheol Chang, and Chulsik Yoon

ETRI

Introduction

0.1 IEEE P802.16e/D8 Status and Problems

The PKMv2 supports the RSA-based authorization and the EAP-based authorization. Both an MS and BS can share the PAK from the RSA-based authorization and the PMK from the EAP-based authorization. Two keys, the PAK and the PMK, shall be used to derive the AK. In other words, two keys should be used as the equal-level keys.

When the AK is derived, however, the PAK is used as an input data but the PMK is used as an input key.

```
If (PAK and PMK)
        AK <= Dot16KDF (PMK, SSID | BSID | PAK | "AK", 160)
Else
        If (PAK)
        AK <= Dot16KDF (0, SSID | BSID | PAK | "AK", 160)
        Else      // PMK only
                   AK <= Dot16KDF (PMK, SSID | BSID | "AK", 160);
        Endif
Endif
```

Since the PAK and the PMK are root keys to derive the AK, both of them should not be used as input data but as input keys.

0.2 Solutions

The input keys for generating the AK should be both PAK and PMK. The exclusive-or (XOR:) value of PAK and PMK as input key is used to derive the AK. The generation method of the AK is proposed as follows.

```
If (PAK and PMK)
        AK <= Dot16KDF (PAK    PMK, SSID | BSID | "AK", 160)
Else
        If (PAK)
        AK <= Dot16KDF (PAK, SSID | BSID | "AK", 160)
        Else      // PMK only
```

AK <= Dot16KDF (PMK, SSID | BSID | "AK", 160)

Endif

Endif

Proposed Changes into IEEE P802.16e/D8

[Change sub-clauses 7.2.2.2.3 as follows]

7.2.2.2.3 Authorization Key (AK) derivation

The AK will be derived by the ~~authenticator~~ BS and the MS from the PMK (from ~~EAP exchange~~ EAP-based authorization procedure) and/or the PAK (from ~~RSA exchange~~ RSA-based authorization procedure). Note that PAK and/or PMK can be used according to the value of Authorization Policy Support field included in the SBC-REQ/RSP messages.

The exclusive-or (XOR:) value of PAK and PMK is mainly used to generate the AK.

If (PAK and PMK)

~~AK <= Dot16KDF (PMK, SSID | BSID | PAK | "AK", 160)~~

AK <= Dot16KDF (PAK PMK, SSID | BSID | "AK", 160)

Else

If (PAK)

~~AK <= Dot16KDF (0, SSID | BSID | PAK | "AK", 160)~~

AK <= Dot16KDF (PAK, SSID | BSID | "AK", 160)

Else // PMK only

~~AK <= Dot16KDF (PMK, SSID | BSID | "AK", 160);~~

AK <= Dot16KDF (PMK, SSID | BSID | "AK", 160)

Endif

Endif

[Change sub-clauses 7.2.2.2.10 as follows]

7.2.2.2.10 Key Hierarchy

Figure 131 outlines the process to calculate the AK when the RSA-based authorization process has taken place, but where the EAP-based authentication process hasn't taken place, or the EAP method used has not yielded an ~~AAA-key~~ MSK:

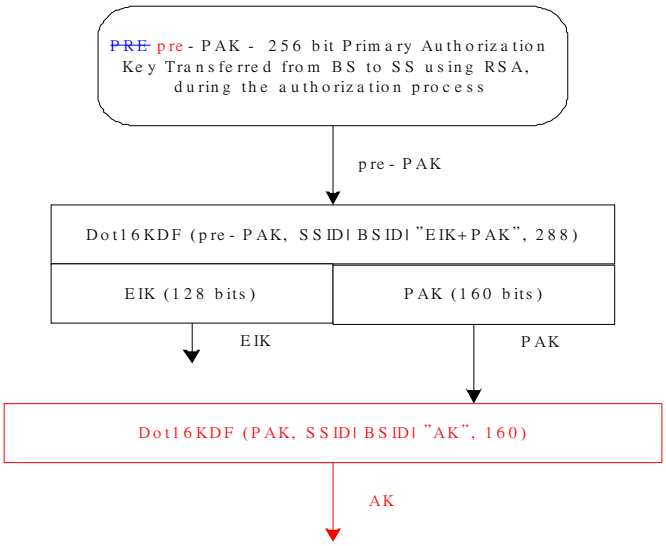

Figure 131-AK with the only PAK (from RSA-based ~~only~~ authorization process)

Figure 132 outlines the process to calculate the AK when both the RSA-based authorization exchange has taken place, yielding a PAK and the EAP based authentication exchange has taken place, yielding an ~~AAA-key~~ MSK:

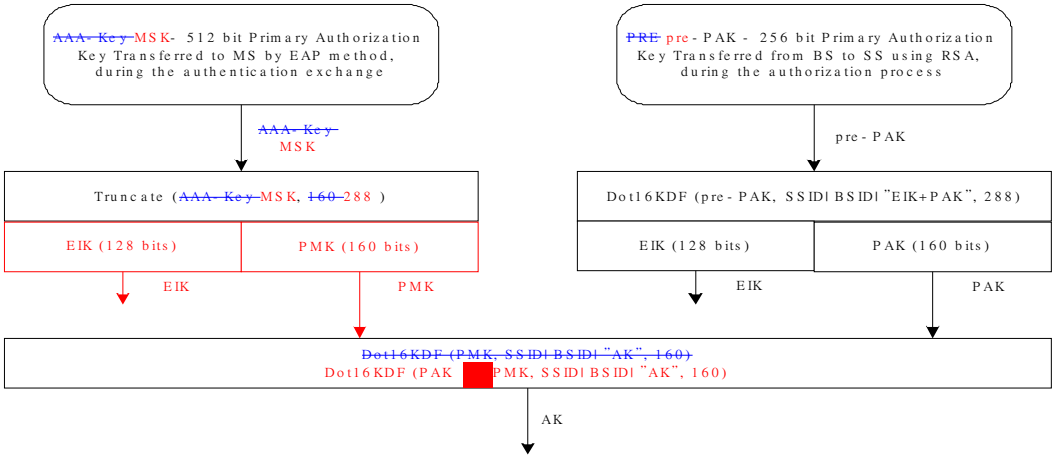

Figure 132-AK with PAK and PMK

(RSA-based and EAP-based authorization process)

Figure 133 outlines the process to calculate the AK when only the EAP based authentication exchange has taken place, yielding an ~~AAA-key~~ MSK:

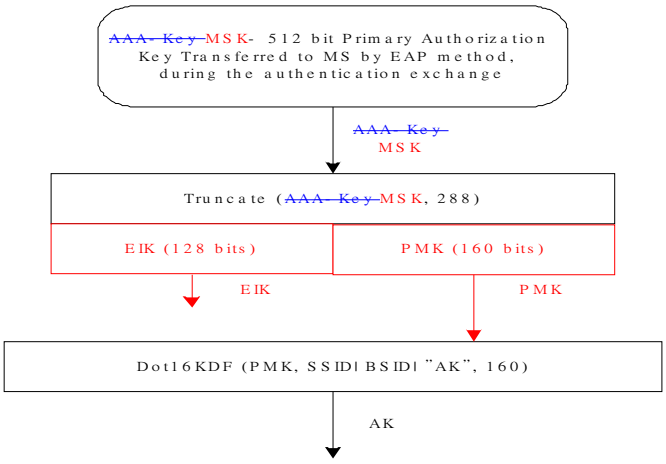

Figure 133-AK with the only PMK (from EAP-based ~~only authentication~~ authorization process)

[Change sub-clause 7.2.2.2.1 as follows]

7.2.2.2.1 ~~Certificated RSA authorization~~ RSA-based authorization

When the RSA-based authorization is negotiated as authorization policy, the PKMv2 RSA-Request, the PKMv2 RSA-Reply, the PKMv2 RSA-Reject, and the PKMv2 RSA-Acknowledgement messages are used to share the pre-PAK (Primary Authorization Key).

The pre-PAK ~~(Primary Authorization Key)~~ is sent by the BS to the MS encrypted with the public key ~~from the~~ of the MS certificate. Pre-PAK is mainly used to generate the PAK. The optional EIK for ~~EAP exchange~~ transmitting authenticated EAP payload (see 7.2.2.2.2) are also generated from pre-PAK:

~~|~~EIK | PAK = Dot16KDF(pre-PAK, SSID | BSID | "EIK+PAK", 288)

PAK will be used to generate the AK (see below) if RSA authorization was used. PAK is 160 bits long.

[Change sub-clause 7.2.2.2.2 as follows

7.2.2.2.2 ~~EAP authentication~~ EAP-based authorization

There are two kinds of EAP-based authorization; EAP payload exchange (using the PKMv2 EAP-Transfer message) and authenticated EAP payload exchange following the RSA-based authorization procedure or the EAP-based authorization procedure (using the PKMv2 Authenticated EAP-Transfer message).

In case of the EAP payload exchange, the MS authorization is achieved by transferring only EAP payload between an MS and a

BS.

Contrary to the only EAP payload exchange, in case of the authenticated EAP payload exchange, the MS authorization is executed by exchanging a PKMv2 Authenticated EAP-Transfer messages. ~~If a mutual authorization took place before the EAP exchange, the EAP messages~~ These messages may be protected using EIK ~~- EAP Integrity Key~~ (EAP Integrity Key) derived from pre-PAK (see 7.2.2.2.1) or from MSK (see below). EIK ~~and EEK are~~ is 128 bits long.

The product of the EAP payload exchange which is transferred to ~~802.16~~ MAC security sub-layer is the ~~AAA-key~~ MSK. This key is derived (or may be equivalent to the 512-bits Master Session Key (MSK)). This key is known to the AAA server, to the Authenticator* (transferred from AAA server) and to the MS. The MS and the authenticator (the serving BS or certain network node) derive a PMK (Pairwise Master Key) and optional EIK by truncating the ~~AAA-key~~ MSK to 288 bits.

The PMK and EIK derivation from the ~~AAA-key~~ MSK is as follows:

EIK | PMK = truncate (~~AAA-key~~ MSK, 288)

If more keying material is needed for future link ciphers, the key length of the PMK may be increased.

After successful EAP-based authorization, if the MS or BS wants to run additional EAP-based authorization ~~authentication~~ (Note that this authenticated EAP-based authorization ~~EAP authentication method~~ shall not derive key materials and PMK), the PKMv2 Authenticated EAP Transfer messages ~~authenticated EAP messages~~ shall carry EAP payload ~~message~~. It shall cryptographically bind previous ~~RSA EAP authentication~~ RSA-based authorization or EAP-based authorization and following authenticated EAP-based authorization ~~EAP authentication session~~, while protecting ~~second EAP messages~~ EAP payload used in the authenticated EAP-based authorization procedure.

[Change sub-clause 7.2.2.2.7 as follows

7.2.2.2.7 Group Traffic Encryption Key (GTEK)

The GTEK is used to encrypt multicast data packets and it is shared between all MSs that belongs to the multicast group. There are 2 GTEKs per GSA.

The GTEK is randomly generated at the BS or at certain network node and is encrypted using ~~AES_KEY_WRAP~~ same algorithms applied to encryption for TEK and transmitted to the MS in multicast or unicast messages. ~~In multicast the message will be encrypted by the GKEK. In unicast, it will be encrypted by the KEK.~~ The GTEK in a PKMv2 Key-Request and PKMv2 Key-Reply messages will be encrypted by the KEK. And, the GTEK in a PKMv2 Group Key Update Command message will be encrypted by the GKEK.