Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**The Enhancement For The AK Lifetime**
Date Submitted	**2005-04-24**
Source(s)	Li Rui, Tian Feng, Chen JianYong, Zhao Jie, Li YuanWei ZTE corporation ZTE Plaza , Keji Road South , Hi-tech Industrial Park , Nanshan District , Shenzhen , P.R.China , 518057 Voice: [86-0755-26772016] Fax: [86-0755-26772004] [mailto:li.rui2@zte.com.cn]
Re:	Response to Sponsor Ballot on IEEE802.16e/D7 document
Abstract	This contribution describes the enhancement of AK lifetime.
Purpose	To incorporate the text changes proposed in this contribution into the 802.16e/D8 draft.
Notice	
Release	
Patent Policy and Procedures	

The Enhancement For The AK Lifetime

Li Rui, Tian Feng, Chen JianYong, Zhao Jie, Li YuanWei
ZTE corporation

1. Problem Statement

The AK lifetime is the minimum between PAK lifetime and PMK lifetime in current 802.16e/D7. The PAK lifetime is sent from BS to MS in the PKMv2-Auth Reply message. But there is not description about how the PMK lifetime is generated and the MS gets the PMK lifetime in this current specification. Since the EAP protocol does not provide for explicit key lifetime negotiation (seen RFC 3748, Page 51), the exchange of PMK lifetime needs to be added after the EAP authentication process in the EAP-based authorization. So it brings the additional exchange in the EAP-based authorization and modification of the EAP-based authorization flow.
In order to resolve the above-mentioned problem we put forward the method that can not only remain the existing EAP-based authorization flow but also assure that both MS and BS can get the AK lifetime successfully.

2. Proposed solutions

The purpose of PAK lifetime generation and PMK lifetime generation is to derive the AK lifetime. So we suggest that the AK lifetime shall be directly generated in the BS and sent from BS to MS in the SA-TEK Response message after both sides finish the authentication and get the AK. Since the PAK lifetime and PMK lifetime are not necessary, the PAK lifetime and PMK lifetime may be deleted.

3. Specific text changes

=== Start text changes ====

6.3.2.3.9.13 Auth-Reply message

Sent by the BS to a client MS in response to an Authorization Request, the Authorization Reply message contains an AK, the key's lifetime, 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 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 message, along with a random number supplied by the BS, thus enabling assurance of key liveness.
Code: 22
Attributes are shown in Table 37c.

Table 37c— Auth-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
EncryptedAK	RSA-OAEP-Encrypt(PubKey(MS), pre-PAK | Id(MS))
~~AK Lifetime~~	~~AK Aging timer~~
AK Sequence Number	64-bit AK 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 proper-ties of the SA (optional, only if there is no EAP phase afterwards)
CertBS	The BS Certificate
SigBS	An RSA signature over all the other attributes in the message

6.3.2.3.9.18 SA-TEK-Response message

The BS transmits the SA-TEK-Response message as a final step in the 3-way handshake.

Table 37h—SA-TEK-Response message attributes

Attribute	Contents
NonceSS	The number received from the MS
RandomBS	A freshly generated random number of 64-bits. This is optional
AKID	This identifies the AK to the MS that was used for protecting this message.
AK lifetime	The lifetime of AK is generated in the BS.
SA_TEK_Update	A compound TLV list each of which specifies an SA identifier (SAID) and additional properties of the SA that the MSS is authorized to access. Additionally, in case of HO, for each active SA in previous serving BS, corresponding TEK, GTEK and GKEK parameters are also included.
OMAC/HMAC	Message integrity tuple for this message.

7.2.2.4.1 AK context

The context of AK includes all the parameters connected to AK and keys derived directly from it.

When one parameter from this context expires, a new AK should be obtained in order to start a new context.

Obtaining of new AK means re-authentication - doing the whole EAP and/or PAK due to the authorization policies negotiated between the MS and BS until obtaining a new PMK and/or PAK which AK may be derived from.

Derivation of AK after HO is done separately in the MS and network from a common PMK, PAK, SSID and BSID. The PMK and/or PAK may be used to derive keys to several BSs sharing the same PMK and/or PAK.

In HO scenario, if the MS was previously connected to the TBS, the derived AK will be identical to the last one, as long as the PMK stays the same. In order to maintain security in this scenario: the context of the AK must be cached by both sides and to be used from the point it stopped, if context lost by one side, re-authentication is needed to establish new PMK and new AK context.

The AK context is described in the table:

Table 133—AK Context

Parameter	Size (bits)	Usage

Primary AK (PAK)	160	A key yielded from the mutual authorization exchange. Only present at initial network entry and only if the certificated RSA exchange took place, as a result of the mutual authorization policy negotiation.
PAKID	64	Derived from the mutual authorization, present when PAK is present.
~~PAK lifetime~~	—	~~Derived from the mutual authorization, present when PAK is present.~~
PMK	160	A key yielded from the EAP authentication.
~~PMK lifetime~~	—	~~The lifetime of PMK derived from EAP.~~
PMKID	64	hash 64(EAP session-id)
AK	160	The authentication key, calculated as f(PAK,PMK), if only EAP, AK=f(PMK).
AKID	64	Calculated according to the keys that contributed to AK: -If AK=f(PMK,PAK) then AKID=hash 64(EAP session-id | PAKID BSID) -If AK=f(PMK) then AKID=hash 64(EAP session-id | BSID) -If AK=PAK then AKID = PAKID
AK Lifetime	—	~~This is the time this key is valid, it is calculated AK life-time=MIN(PAK lifetime, PMK lifetime) – when this expires re-authentication is needed.~~ The lifetime of AK is generated in the BS.
H/OMAC_KEY_U	160 or128	The key which is used for signing UL management messages.
H/OMAC_PN_U	32	Used to avoid UL replay attack on management messages – when this expires re-authentication is needed.
H/OMAC_KEY_D	160 or 128	The key which is used for signing DL management messages.
H/OMAC_PN_D	32	Used to avoid DL replay attack on management messages – when this expires re-authentication is needed.
KEK	160	Used to encrypt transport keys from the BS to the SS.

7.8.1 SA-TEK 3-way handshake

Depending on mutual authorization/EAP, AK can be derived in three different ways as documented in section XXX. Before the 3-way handshake begins, the BS and MS shall both derive a shared AK, KEK and HMAC/OMAC as per 7.2.2.2.

The SA-TEK 3-way handshake sequence proceeds as follows:

1. During initial network entry or reauthorization, the BS shall send SA-Challenge (including a random number RandomBS) to the MS after protecting it with the OMAC/HMAC tuple. If the BS does not receive SA-TEK-Request from the MS within SAChallengeTimer, it shall resend the previous SA-Challenge. The BS may send SA-Challenge up to SAChallengeMaxResends

times. If the BS reaches its maximum number of resends, it shall discard the AK and may initiate full re-authentication or drop the MS.

2. During network re-entry or handover, the BS begins the 3-way-handshake by appending the SaChallenge TLV to the RNG-RSP. If the BS does not receive SA-TEK-Request from the MS within SaChallengeTimer (suggested to be several times greater than the length of SaChallengeTimer), it shall discard the AK and may initiate full re-authentication or drop the MS. If the BS receives RNG-REQ during the period that SA-TEK-Request is expected, it shall send a new RNG-RSP with another SaChallenge TLV.

3. The MS shall send SA-TEK-Request to the BS after protecting it with the OMAC/HMAC. If the MS does not receive SA-TEK-Response from the BS within SATEKTimer, it shall resend the request. The MS may resend the SA-TEK-Request up to SATEKRequestMaxResends times. If the MS reaches its maximum number of resends, it shall discard the AK and may do full re-authentication or decide to connect to another BS or take some other action. The message shall include RandomBS, NonceSS, AKID, SS's Security Capabilities and OMAC/HMAC.

4. Upon receipt of SA-TEK-Request, a BS shall confirm that the supplied AKID refers to an AK that it has available. If the AKID is unrecognized, the BS shall ignore the message. The BS shall verify the OMAC/HMAC. If the OMAC/HMAC is invalid, the BS shall ignore the message. Meanwhile the AK lifetime is generated in the BS.

   Upon successful validation of the SA-TEK-Request, the BS shall send SA-TEK-Response back to the MS. The message shall include the AK lifetime and a compound TLV list each of which identifies the Primary and static SAs, their SA identifiers (SAID) and additional properties of the SA (e.g., type, cryptographic suite) that the MS is authorized to access. In case of HO, the details of any Dynamic SAs that the requesting MS was authorized in the previous serving BS are also included.

   === End text changes ====

4. References

[1] IEEE Standard 802.16e/D7-2004
[2] IEEE Standard 802.16-2004