Security Issues in Privacy and Key Management Protocols of 802.16

Sen Xu, Manton Matthews, Chin-Tser Huang, and Jingshan Huang

Computer Science and Engineering Department
University of South Carolina
Columbia, SC 29208, USA
{xu4, matthews, huangct, huang27}@cse.sc.edu

Abstract

Without physical boundaries, a wireless network faces many more vulnerabilities that a wired network does. IEEE802.16 provides a security sublayer in the MAC layer to address the privacy issues across the fixed BWA. Several articles have been published to address the flaws in IEEE 802.16 security after IEEE802.16-2001 was released. However, the IEEE802.16-2004 does not settle all these problems and additional flaws remain. This paper introduces the IEEE802.16 standard, focusing on the MAC layer and especially the security sublayer. We analyze the security flaws in the standard as well as in related works, and illustrate possible attacks to the authentication and key management protocols. Possible solutions are also proposed to prevent these attacks. Finally, we propose a security handover protocol that should be supported in the future 802.16e for mobility.

1. Introduction

As a member of IEEE802 group, IEEE802.16 is the standard to specify the air interface of fixed Broadband Wireless Access (BWA). IEEE802.16 was first designed to provide the last mile for Wireless Metropolitan Area Network (WMAN) with line-of-sight (LOS) working at 10-66GHz bands. The latest version, IEEE802.16-2004 [4], which consolidates previous standards, also supports non-line-of-sight (NLOS) within 2-11GHz bands and mesh nodes. The developing IEEE802.16e aims to provide mobility in BWA.

Security requirement for wireless network is similar to the wired network. However, wireless networks are inherently insecure compared to their wired counterparts due to the lack of physical infrastructure. Although digital systems are much harder to tap, with the cheap and freely available chips, it is not very hard for the attackers to break into the system. On the other hand, more and more people use wireless access to the Internet and use wireless networks for e-commerce, transactions, etc. Therefore, special attention should be paid to the wireless security.

Privacy or confidentiality is the straightforward requirement for secure communication, which provides resistance to interception. Message authentication provides integrity of the message and sender authentication, corresponding to the security attacks of message modification and impersonation. Non-repudiation is against denial and fabrication. Access control prevents masquerading. Availability ensures that the resources or communications are not prevented from access by DoS attack. Detailed discussion of the security requirements, together with corresponding attacks and possible solutions, can be found in [7] and [8].

In WMAN, both the Base Station (BS) and Subscriber Station (SS) face almost all those attacks mentioned above. 802.16 implements a security sublayer at the bottom of the MAC layer. Security sublayer provides SS with privacy and protects BS from theft of service. There are two component protocols in security sublayer: an encapsulation protocol for encrypting packet data across the fixed BWA; a Privacy and Key Management Protocol (PKM) providing the secure distribution of keying data from BS to SS as well as enabling BS to enforce conditional access to network services.

The PKM protocol uses X.509 digital certificates, RSA public-key algorithm, and strong encryption algorithm to perform key exchanges between SS and BS, at client/server model. IEEE802.16 employs two-tier key systems. The PKM protocol first authenticates SS to BS, establishing a shared secret (Authentication Key—AK) via public-key cryptography, then SS registers to the network, during which AK is used to secure the exchange of Transport Encryption Keys (TEK).

A certificate sent by SS allows BS to authenticate legal SS. On the other hand, SS also needs to authenticate BS to keep away from malicious ones. That is because through the opened air interface, SS has no other way to identify legal BS from malicious attacks. A few articles have been published to address the necessity of mutual authentication as well as mechanisms to anti-attack for 802.16. However, there are still some flaws in their protocols. This paper analyzes those possible attacks to both BS and SS, and proposes a revised PKM protocol to settle those problems.

WMAN also intends to support mobility in the developing 802.16e standard. Researchers in IEEE802.16e task group (TGe) have proposed some mechanisms for security roaming of key association for fast handover in the new standard. However, their schemas only support forward secrecy for the service BS (SBS), without backward secrecy for the target BS (TBS). Moreover, most vulnerabilities in 802.16 protocols are still applicable in 802.16e. In this paper, we propose a security roaming protocol. Our protocol can prevent those attacks. It also supports backward secrecy and forward secrecy to some extend.

This paper is organized as follows. In Section 2, we introduce some related works. Section 3 analyzes the Authentication Protocols in 802.16. In Section 4, we focus on the Key Management (Registration) Protocol. Possible attacks are illustrated and solutions are proposed in both Section 3 and 4. In Section 5, we propose a security roaming protocol for 802.16e. Finally, Section 6 concludes and describes some future work.

2. Related Works

Since the first version of IEEE802.16 [1], a few papers have been published to introduce this new standard and address the security issues. In [2], Roger Marks gives a technical overview of 802.16. There are also some other papers to review this standard, such as [5]. Some books such as [9] and [10] aim to enable operators to deploy and set up a network with standards-based equipment, and run it profitably as well. That is beyond the topic of this paper. Few of those papers and books tackle the security issues. Clearly, WMAN has gained much less attention than WLAN.

The authors in [3] review the standard, and analyze its security in many aspects, such as vulnerability in authentication and key management protocols, failure in data encryption, and lack of explicit definition. Mutual authentication is the major contribution proposed by [3], which enables SS to authenticate BS as well.

In fact, the need for mutual authentication in wireless network is not a novel idea. It has been widely studied in WLAN. A definitive guide [11] overviews the 802.11 management operation and brings forward mutual authentication in chapter 7. There are also many other papers dealing with this topic, such as [12] and [13]. In WLAN, WS needs to authenticate AP while AP authenticates WS. However, the authentication and key management protocols in 802.11 and 802.16 are based on different methods. IEEE802.11 applies the shared-key authentication method, while IEEE802.16 is based on public-key authentication algorithm, specifically, X.509 certificate. Therefore, the authentication and key management in IEEE802.16 needs separate study. The general discussion of mutual authentication protocols (in wired network) can be found in [6].

In the developing standard IEEE802.16e, mobility is supported in WMAN. [14] gives an overview of Handoff Schemes on different kinds of networks, such as GSM, UMTS, 802.11, HIPERLAN 2, and proposes the requirements for handoff procedures in 802.16. [15] proposes a draft for IEEE 802.16e handoff. Some comments have been submitted to the TGe for inter-BS handoff, [16] for example. Due to the limited capability of the wireless device, such as power source and computation ability, it is important to decrease the computation for encrypting or decrypting. Thus a fast handover is proposed, which establishes and exchanging the keying information inside the wireless access network, so as to get fast and efficient intra-domain mobility or micro-mobility control. The fast handover is based on the extension authentication protocol, which is implemented in 802.16 PKMv2 [20]. [19] applies this micro mobility protocol and proposes LPM scheme (last packet marking), which aims to minimize the handover delay and eliminates packet losses during handover.

Based on previous works, [17] proposes a secure roaming of key association for fast handover in IEEE802.16, which provides perfect forward secrecy. [18] gives some comments to modify some keying materials which should be exchanged during the roaming. Several kinds of attacks mentioned before are also applicable to this protocol.

3. Weakness and Enhancement of Authentication Protocol

3.1. General Attacks on Authentication Protocols

Before we start to analyze the authentication protocol of 802.16, we would like to introduce some typical attacks on authentication protocols. Message Replay Attack is the most common attack on authentication and authenticated key establishment protocols. It seems that we have already established a good awareness to this kind of attack. However, replay attack is still the most severe one and it happens even when the subtle designers know the errors very well in a different context. Man-in-the-Middle Attack is another classic attack and is generally applicable in a communication protocol where mutual authentication is absent. Other familiar attacks include Parallel Session Attack, Reflection Attack, Interleaving Attack, Attack Due to Type Flaw, Attack Due to Name Omission, and Attack Due to Misuse of Cryptographic services. The detailed discussion and examples of these attacks can be found in [6].

3.2. Authentication Protocol in 802.16

An SS begins authorization by sending an Authentication Information message which contains the SS manufacturer's X.509 certificate. This message is strictly informative and the BS may choose to ignore it. Afterwards the SS sends an Authorization Request message (Auth-REQ) to its BS. In response to Auth-REQ, the BS validates the requesting SS' identity, determines the encryption algorithms and protocols shared with the SS, generates an AK, and sends the AK to SS. The authentication protocol is illustrated in Figure 1.

Message 1. SS -> BS : Cert (SS. Manufacturer)
Message 2: SS -> BS : Cert (SS) | Capabilities | BCID
Message 3: BS -> SS :KU_{ss} (AK) | SeqNo | Lifetime | SAIDList

Figure 1. Authentication Protocol Scenario in 802.16

In Figure 1, Cert (SS. Manufacturer) is the X.509 certificate of SS' manufacturer, and Cert (SS) is SS' X.509 certificate. The X.509 basic fields includes: the certificate version, serial Number, signature, issuer, validity, subject, subject Public Key Info, issuer Unique ID, subject Unique ID, and extensions. Capabilities are the SS-supported authentication and data encryption algorithms. BCID is the Basic CID of SS, which equals to its primary SAID. KU_{SS} (AK) is the Authentication Key encrypted by SS' public key. SeqNo is a 4-bit sequence number for AK. And lifetime gives the number of seconds before AK expires (32bits). SAIDList contains the identities and the properties of the single primary SA and zero or more static SAs for which SS is authorized to obtain keying information.

Since message 1 is optional and only informative, we begin the security analysis from the next message. Message 2 is sent in plaintext but eavesdrop is not a problem since the information is almost public and preferred sent in plaintext to facilitate authentication. However, BS will face replay attack from malicious SS, who intercepts and saves the message sent by a legal SS previously. Although the malicious SS cannot get the AK from message 3 because he does not have the corresponding private key, this attacker still can replay message 3 multiple times and thus, either exhausting BS' capabilities or forcing BS to deny the SS who owns that cert (SS). The reason is that, if BS set a timeout value which makes him to reject Auth-REQ from the same SS in a certain period, the legal request from the victim SS will also be ignored. Therefore, the Deny of Service occurs to the victim SS.

To avoid these replay attacks, we suggest adding time stamps in message 2, together with a signature of SS which provides the message authentication and non-repudiation. The signature uses SS' private key to encrypt the critical information in message 2.

Similarly, message 3 also endangers SS in replay attacks. Even worse, SS also faces the fraudulence from the malicious BS who intercepts his Auth-REQ message. This BS can make his own Auth-Reply message with the AK generated by itself, thus gaining the control of the communication of the victim SS. This is a typical Man-in-the-Middle attack, which brings forward the need of mutual authentication, i.e., SS needs to authenticate BS as well. This can be done by adding BS' certificate in message 3. The time stamp received from message 2 is also replied in message 3 to ensure SS that the message 3 responds to its request. Time stamp from BS assures its liveness and freshness. Signature of BS is added at the end of message 3, which provides the authentication and non-repudiation of this message. The modified protocol is show in Figure 2.

Message 1. SS -> BS : Cert (SS. Manufacturer)
Message 2. SS -> BS : T_S | Cert (SS) | Capabilities | SAID | SIG_{SS} (2)
Message 3. BS -> SS : T_S | T_B | KU_{SS} (AK) | Lifetime | SeqNo | SAIDList | Cert (BS) | SIG_{BS} (3)

Figure 2. Modified Authentication Protocol

In Figure 2, T_S and T_B are timestamps generated by SS and BS respectively; SIG_{SS} (2) is the signature of SS over message 2; SIG_{BS} (3) is the signature of BS over message 3.

Nonce is a possible alternative to timestamp in the authentication protocol. In [3], the authors use nonce instead of timestamp. Their protocol is shown in Figure 3.

Message 1. SS -> BS : Cert (SS. Manufacturer)
Message 2. SS -> BS : N_S | Cert (SS) | Capabilities | SAID |
Message 3. BS -> SS : N_S | N_B | KU_{SS} (pre-AK) | Lifetime | SeqNo | SAIDList | Cert (BS) | SIG_{BS} (3)

Figure 3. Authentication Protocol with nonce in [3]

However, the exchange of nonces only assures SS that message 3 is replied to its corresponding request. The BS still faces the replay attack because he cannot tell whether message 2 is recently sent or it is just an old message.

The authors also suggest passing the pre-AK to SS instead of AK, and let SS and BS derive AK from the pre-AK at both ends. If the AK's generation exhibits significant bias, adding freshness in the AK may prevent the expose of the AK. However, this cannot provide freshness as they claimed. If the pre-AK is compromised, the attacker can easily derive the AK by the same algorithm applied by the SS and BS, with the same freshness identities (such as nonce or timestamp) which are sent in plaintext. Thus the distributed-derived AK is barely more secure than the BS-generated AK.

Nonce and timestamp are two major methods to provide message freshness and principal liveness. The main drawback of timestamp is that it needs the communication parties to maintain time synchronization, which is considered to be difficult over the network. However, in 802.16, the SS and BS have already synchronized during the initial ranging, right before they begin authentication procedure. Thus the synchronization is not a problem for applying timestamp here due to the inherence of 802.16.

4. Analysis and modification of Key Management Protocol

After achieving authentication, SS begin to request keying materials. SS sends Key-Request message to the BS periodically, corresponding to one of his legal SAIDs. The BS responds with a Key-Reply message, containing the BS's active keying material for the specific SAID. This procedure is shown in Figure 4.

Message 1. BS -> SS: SeqNo | SAID | HMAC (1)
Message 2. SS -> BS: SeqNo | SAID | HMAC (2)
Message 3. BS -> SS: SeqNo | SAID | OldTEK | NewTEK | HMAC (3)

Figure 4. Key Management Protocol in 802.16

Again, message 1 is optional. BS sends re-key message (message 1) to SS only if he regards it necessary to re-key before SS requests it. BS will choose a SAID from the SAIDList which the SS is allowed to access. SeqNo is the sequence number of AK provided by BS to this SS in the authentication protocol previously. This number allows the SS (and BS in the next message) to determine which HMAC_KEY_D (HMAC_KEY_U in the next message) was used to authenticate the message. HMAC(1) is the digest of message 1 under HAMC_KEY_D. Both of this downlink HMAC key (HMAC_KEY_D) and the uplink HMAC key (HMAC_KEY_U) are derived from the AK. By computing the value HMAC(1), it allows SS to detect forgeries.

Upon receiving message 1, SS will reply with the Key-Request message (message 2). If SS does not receive message 1 from BS and timeout, SS will send the normal Key-Request message, where the SAID is chosen by himself from the SAIDList, to request a refresh of keying material for this specific SAID. HMAC(2) is the digest of message 2 under HMAC_KEY_U, which assures BS the message authentication.

BS will reply with the Key-Reply message (message 3) immediately after receiving the request from SS, which includes keying materials. At all times BS maintains two active sets of keying material per SAID. The OldTEK is the keying materials for the old (current used) TEK, NewTEK is the keying materials for the new (to be used after the current one expired) TEK. The keying materials include the TEK encrypted by the KEK (Key Encryption Key), which is also derived from the AK. In addition, the set of keying materials also includes the CBC initialization vector and the remaining lifetime of each set of keying materials. HMAC(3) is the digest of message 3 under HMAC_KEY_D. As in message 1, HMAC(3) assures SS that message 3 is from BS and has not been modified.

Message Replay Attack is also the major threat to the key management protocol. In [3], the authors claim the SS cannot recognize reused data SAs, just like it cannot recognize reused authorization SA in authentication protocol. However, if the attack resends message 3 to SS after the SS has already exchanged some keying materials with BS, the SS can easily tell whether message 3 is relative to his request. That is because each SAID maintains two set of keying materials, and the OldTEK in the recently received Key Reply Message should be the NewTEK in the previous Key Reply Message. Therefore, in order to put the replay attack, the attacker must fool the SS at the very beginning, i.e., the first time SS requests keying materials. But now the attacker will face another obstacle. The correct use of the AK provides a way for both SS and BS to check the validity of the Key Management Protocol instance. If the attacker intends to replay an old Key Reply message, the HAMC_KEY_D used in HMAC(3) must be derived from the AK that the SS currently used. So the only chance for this replay attack to succeed is that the attacker intercepted and saved a former sequence of exchanged key request and reply messages, and the Key Management Protocol is reset which makes SS requesting totally new keying materials. This attack can be simply avoided by forcing SS request a new AK every time the current Key Management Protocol instance fails and is reset. But this requires re-authentication.

In [3], the authors also suggest it should tie messages to a particular protocol instance in order to prevent replays from succeeding against the key management. Their solution is adding the nonces exchanged in the previous Authentication Protocol as the instance identifier. However, the correct use of the AK already provides a way to identify these instances. The SeqNo of AK provides some relationship between the instance of Authentication protocol and the instance of related Key Management Protocol. Although this 4-bit number is prone to be reused thus endangers the Key Management Protocol into replay attack, the digest of these messages exchanged during Key Management provides a way to ensure both parties the validation of the legal AK. In order to succeed this replay attack, the attacker should not only replay the SeqNo, but also be able to replay the correct HMAC message whose encryption key is derived from the current used AK in the corresponding Authentication Protocol instance. The failure of bounding Key Management Protocol instance to its corresponding Authentication protocol instance will happen only if there is a coincidence that another instance of Authentication Protocol happened to have the same AK and the same SeqNo. Due to the random generation of AK, this can be regarded impossible.

Although SS is somewhat free from the replay attacks on message 3, BS still faces it on replayed message 2. The reason is the Key-Request Message does not have the Keying Material like the Key-Reply Message, which allows the receiver to compare with its previously received message. Thus if an attacker replays the Key-Request message to BS, the BS has no way to recognize whether it is a fresh request from SS or an old one. Therefore, BS should reply with message 3 that assigns new Keying Materials to SS, which SS did not request at all. This can cause frequent exchange of Keying materials, resulting in exhausting BS' capabilities, or the confusion in the use of TEK. This situation is quite the same as the one BS faced in the Authentication protocol.

Similar replay attacks happen to SS on message 1 as well. This replayed message will make SS send message 2. Besides the effects on BS, this will eventually make SS and BS exchange the keying materials which they do not want to. That is because SS will think it is the BS who requests the re-key by sending message 1, which is indeed, send by the attacker; while on the other hand, the BS will think it is the SS who request the re-key.

A timestamp is also a suitable identity to be added in these Key Management messages to provide freshness. But the signature is unnecessary since digest already provides the message authentication. The revised protocol is illustrated in Figure 5.

Message 1:BS -> SS: T_{B2} | SeqNo | SAID | HMAC (1)
Message 2:SS -> BS: T_{B2} | T_{S2} | SeqNo | SAID | HMAC (2)
Message 3:BS -> SS: T_{S2} | T_{B2} | SeqNo | SAID | OldTEK | NewTEK | HMAC (3)

Figure. 5. Revised Key Management Protocol

Since message 1 is optional, SS will set T_{B2} to 0 in message 2 if he initiates the re-key; and T_{B2} in message 3 is generated by BS in responding SS' request to assure SS the freshness and liveness. If BS initiates the re-key, T_{B2} is generated in message 1 by BS and SS should include it in message 2 to assure BS the freshness and liveness, but BS can omit it in message 3 by setting it to 0.

5. Security Roaming of Key Association during Handover

In view of the limitation of mobile (subscriber) stations (MSS), it will be too expensive for the MSS to re-authenticate every time he hands over to another BS because the authentication protocol is based on public key infrastructure. Thus Fast handover is proposed, and the security roaming of key association becomes crucial.

The security roaming of key association scheme proposed in [17] supports perfect backward secrecy, i.e., the TBS cannot derive keys used by the SBS from the roaming key association sent by it, thus is kept from the communication between the roaming MSS and its serving BS. However, this scheme does not support forward secrecy. Since the PKMv2 is still under development, we propose a security roaming of keying materials for handover scheme here based on the basic PKM protocol. Our protocol supports both backward secrecy and forward secrecy, and prevents many attacks mentioned before as well. Though it is not based on PKMv2, the idea is similar and it can be easily modified to be implemented in PKMv2.

We skip the handover procedure for conciseness, and focus on the security roaming of the keying materials. The details of handover procedure can be found in the references such as [15]. Keying Materials should be encrypted and sent from serving BS to target BS through the backhaul. Obviously, this can be done by letting serving SS encrypt the message by target BS' public key and add its signature at the end. However, this requires two public key encryptions for both communication parties, which could be very expensive. Due to the frequent communication between BSs, it is desirable to distribute a shared secret key (SK) to each pair of BSs within the network domain. This secret key can also be used in many other applications, such as multicast. There are many ways to establish and distribute the SK, which is beyond the topic of this paper. Here we assume TBS and SBS already have the SK.

If TBS accepts the Handover Request from SBS, the latter will send keying materials which is required by the former to communicate with the roaming MSS.

Message 1. SBS -> TBS: T_1, MSS, SK (MSS, T_1, RAK)

where RAK is the Roaming Authentication Key, which is derived from the AK shared by SBS and MSS. T_1 is timestamp. And TBS replies with Acknowledgement (ACK)

Message 2. TBS -> SBS: T_1, N_1, SK (N_1, T_1)

where N_1 is nonce from TBS. It provides freshness in ACK. The nonce will be also used as the identity for this roaming protocol instance. After receiving ACK from TBS, SBS will notify MSS that TBS is ready to accept his roaming. This message also includes the RAK.

Message 3. SBS->MSS: T_2, N_1, Ready-to-Roam TBS, AK (TBS, RAK, $T2_2$ N_1)

After exchanging handoff messages with its SBS, MSS begins initial ranging with the TBS, and achieves re-authentication without sending the X.509 certificate.

Message 4. MSS -> TBS: T_3, N_1, re-auth, RAK (T_3, N_1)

Message 5. TBS -> MSS: T_3, RAK (new-AK, T_3)

where new-AK is the current AK shared by MSS and TBS.

Notice that the SBS is still able to intercept the future communication between MSS and TBS. SBS may intercept message 5 and get the new-AK as long as he has the RAK, therefore decrypt the following messages exchanged between MSS and TBS. But it is a little better than simply using the RAK. A possible enhancement is letting TBS and MSS derive the new-AK instead of letting TBS generate and distribute it. Both MSS and TBS contribute to the new-AK (possibly by the exchanged nonce). However, the threat to the forward secrecy still exists. So it only provides the forward secrecy to some extend.

6. Conclusion and Future Works

In this paper, we analyze the vulnerability in authentication and key management protocols of 802.16. Our modified protocols can prevent many kinds of attacks, such as replay attacks to BS and SS. We also propose a security roaming protocol for 802.16e, which provides fast handover and guarantees backward and forward secrecy to some extend.

As we discussed before, the 802.16e is still under development. The mobility will bring up more problems in authentication and key management protocols and make them more vulnerable. Therefore, we should pay more attention to the security issues in the drafts from TGe before it is made as standard. Security Roaming in PKMv2 needs more works to finish. Mesh network in 802.16 also needs separate study. Multicast is another issue in the new standard, where authentication and key management should be revised to facilitate the multicast functions.

7. References

[1] IEEE std 802.16-2001: Air Interface for Fixed Broadband Wireless Access Systems (2002)

[2] Roger Marks: A technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access, IEEE C802.16-02/05 (2002)

[3] David Johnston, Jesse Walker: Overview of IEEE 802.16 Security, IEEE Security & Privacy (2004)

[4] IEEE std 802.16-2004: Air Interface for Fixed Broadband Wireless Access Systems (2004)

[5] Intel white paper: IEEE802.16 and WiMax: Broadband Wireless Access for everyone (2004)

[6] Wenbo Mao: Modern Cryptography: Theory and Practice, Pearson Education, Prentice Hall PTR (2004)

[7] William Stalling: Cryptography and Network Security: Principles and Practices. 3rd edition, Pearson Education, Prentice Hall PTR (2003)

[8] Kaveh Pahlavan, Prashant Krishnamurthy: Principles of Wireless Networks: A unified Approach, Pearson Education, Prentice Hall PTR (2002)

[9] Daniel Sweeney: WiMax Operator Manual: building 802.16 Wilreless Networks, Apress (2005)

[10] Ron Olexa: Implementing 802.11, 802.16 and 802.20 Wireless network, ELSEVIER (July 2004)

[11] Matthew S. Gast: 802.11 Wireless Networks: The Definitive Guide, O'Reilly (2002)

[12] Richard R. Joos, Anand R. Tripathi: Mutual Authentication in Wireless network (June 1997); http://cs.engr.uky.edu/~singhal/CS685-papers/joos97mutual.pdf

[13] William A. Arbaugh, Narendar Shankar, Y.C. Justin Wan: Your 802.11 Wireless network has No clothes (March 2001); http://www.cs.umd.edu/~waa/wireless.pdf

[14] Avi Freedman, zion Hadad: Handoff Schemes Overview and Guidelines for handoff Procedures in 802.16, IEEE C802.16sgm-02/24 (2002).

[15] Itzik Kitroser: IEEE 802.16e handoff Draft, IEEE C802.16e-03/20r1 (2003)

[16] Changhoi Koo, Sohyun Iim, Jungje Son: Inter-BS communication for IEEE 802.16e handoff, IEEE 802.16e-03/29 (2003)

[17] Kihun Hong, Souhwan Jung, Ki Jun Lee, Brian Lee, Jungwook Wang: Secure Roaming of Key Association for Fast handover, IEEE C802.16e-04/407 (2004)

[18] Feng Tian, DongXin Lu, Rui Li: Comment on Security Roaming of Key association for Fast Handover, C802.16e-04/571r1 (2005)

[19] Kyung-ah Kim, Chong-Kwon Kim, Tongsok Kim: A seamless handover Mechanism for IEEE 802.16e Broadband Wireless Access, International Scientific-Practical Conference (ISPC) "Communication-2004" (August 22~29, 2004)

[20] Jeff Mandin: 802.16e Privacy Key Management (PKM) version 2, IEEE C802.16e-02/131r1 (2002)