WiMAX-Security – Assessment of the Security Mechanisms in IEEE 802.16d/e

Prof. Dr. -Ing. Evren Eren
University of Applied Sciences Dortmund, Emil-Figge-Str. 42
44227 Dortmund, Germany

and

Dr. -Ing. Kai-Oliver Detken
DECOIT GmbH, Fahrenheitstraße 9
28359 Bremen, Germany

ABSTRACT

The importance of IEEE-Standard 802.16 (WiMAX) is growing and will compete with technologies such as UMTS. As security plays a significant role for market acceptance, this article will deal with some of the basic security features of IEEE 802.16d (Fixed WiMAX). It will summarize the most important elements of the security architecture, present some of its weaknesses, potential attacks and viable counter measurements. Furthermore, we will introduce the basic improvements made by the IEEE 802.16e standard (Mobile WiMAX) compared to IEEE 802.16d (Fixed WiMAX).

Keywords: WiMAX, IEEE 802.16e, IEEE 802.16d, WiMAX-Security, Attacks, Shortcomings, Countermeasures.

1. ELEMENTS OF THE WIMAX SECURITY ARCHITECTURE

WiMAX (Worldwide Interoperability for Microwave Access) defines a Point-to-Multipoint-Wireless network which operates within a range of 2 to 66 GHz. Security is implemented in the so called Privacy Sublayer of the Reference Model. In the following, some essential elements of the IEEE 802.16 Security Architecture will be presented [1]. WiMAX defines the layers PHY and MAC within ISO-OSI. The physical layer (PHY) handles signal connectivity, error correction, initial ranging, registration, bandwidth requests, and connection channels for management and data. The MAC layer manages connections and security.

The WiMAX security architecture uses numerous components and mechanisms

- Digital certificate of the SS (X.509 standard – RFC 3280)
- Security Associations
- Encapsulation Protocol
- Privacy Key Management Protocol

Security Associations
A Security Association (SA) provides a set of security information by which secured communication between subscriber stations (SS) and base stations (BS) can be established. By means of the SA an SS is authorized for a WiMAX-service.

There are three different SAs (also denoted as Data SAs): Primary, Dynamic and Static SA. Different connections (Management Connections and Data Transport Connections) are mapped to these SAs and are secured according to the security mechanisms defined in the SAs. Dynamic SAs are dynamically generated by the BS and provided to the SS. They are only initiated if the SS intends to use a new service and are dynamically terminated when data transfer in the service ends.

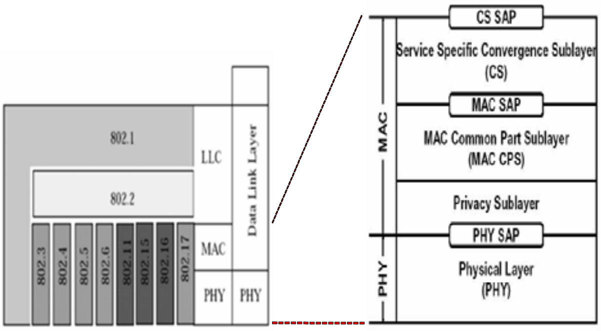

Figure 1: Communication workflow during the three phases and steps 1 - 4

An SA, comprising a set of different security information, is described by a SA-descriptor. This identifies the Primary SA, Static SA and Dynamic SA and contains the following information [1]:

- SAID (identifies the SA)
- SA-Type (Primary, Dynamic or Static SA)
- Cryptographic Suite: Data Encryption, Data Authentication, and TEK Encryption.

SA-Types
There are two SA-types: Authorization Security Associations and Data Security Associations. Authorization SAs are responsible for authorization of the SS. They are used by the BS in order to establish the Data SA between BS and SS. They consist of the following components [2]:

- X.509-Certificate: Digital certificate serving for the

identification of the SS.
- Authorization Key (AK): Generated by the BS and used for the generation of the Key Encryption Keys (KEK), calculation of HMAC-Digests and HMAC-Digest verification on receiver side.
- AK Sequence Number: Serves for the differentiation of successive AKs.
- AK-Lifetime (32 Bit): Validity duration of an AK.
- Key Encryption Key (KEK; 128 Bits): Used in the BS in order to encrypt the Traffic Encryption Key (TEK) from a Data SA and to transmit it. Furthermore, the SS uses the KEK to decrypt the encrypted TEKs, which is needed for data encryption.
- HMAC-Digest: For the integrity check of exchanged key material. It is derived from the AK and guarantees, that SS and BS have the same AKs.
- SA-Descriptor(s)

Key material has limited validity duration and is monitored and controlled by the BS. The BS informs the SS immediately after delivering key material. The SS is responsible for requesting new key material within the validity duration. However, the complete authentication procedure has not to be accomplished.

Data SAs protect transport connections. They consist of the following security information [2]:
- SA-Identification (SAID)
- AK-Sequence Number
- TEK-Parameter (two sets[1] of key material), having the following values:
    - Traffic Encryption Key (TEK) for data encryption (encrypted with the KEK)
    - TEK-Lifetime (remaining validity duration)
    - 2-Bit TEK-Sequence Number
    - Initialisation Vector (IV) – a block of random numbers
    - Encryption algorithm[2]
    - HMAC-Digest

Encapsulation Protocol

The Encapsulation Protocol enables encryption of data packets between BS and SS. For this purpose, it determines the cryptographic suites, which are cryptographic identifiers specifying encryption and authentication methods supported by the SS. They are presented to the BS in form of a list of consecutive cryptographic suites. A set consists of the packet data encryption/authentication algorithm and the encryption algorithm for the TEK. Out of this list the BS selects a single suite and applies it for the Primary SA of the requesting SS [1].

Privacy Key Management Protocol (PKM)

The Privacy Key Management Protocol (PKM) is responsible for normal authorization of the SS, periodic re-authorization and reception/renewal of key material. This protocol is comparable to a conventional Client-/Server-model where the SS (as PKM client) requests key material from the BS, which functions as PKM server. With this mechanism the client (and thus each SS) only receives key material for those services the client is allowed respectively authorized for.

With the extension IEEE 802.16e of the standard the protocol has been modified and is denoted as PKM Version 1. With further modifications it was transformed to Version 2 (PKMv2), which supports mobile SS exclusively in PMP networks[3] [2], [4].

2. SECURITY MECHANISMS: AUTHENTICATION, AUTHORIZATION AND ENCRYPTION

Authentication and authorization of the SS and the subsequent data encryption are based on a two-stage combination of symmetrical and asymmetrical encryption methods and PKM. Communication between the SS and BS takes place in three phases:

Phase I: SS Authorization

The first phase authorizes the SS within four steps [2], [1]:
- **Step 1:** In order to connect with the BS (initial authorization), the SS sends its authentication information within the Authentication Information Message (AuthenticationInfMess). This message contains the vendor certificate of the SS serving the BS to check the trustworthiness of the SS.
- **Step 2:** Immediately the SS sends an Authorization Request Message (AuthorizationReqMess) to the BS, by which it requests the Authorization Key (AK) – a shared secret – and the Security Association Identifications. Each SA-Descriptor within the Authorization Reply Message of the BS contains a SAID each. SAIDs are unique IDs for a single Primary SA and possible (optional) Static SAs. The AuthorizationReqMess contains the certificate[4] issued by the vendor, a description of encryption algorithms (Security Capabilities)[5] supported by the SS and furthermore their Basic CID[6]. By means of the certificate verification the SS is being authorized.
- **Step 3:** After having proved the identity of the SS and thus authorized it the BS identifies the SA-Descriptors (i.e. identities and properties of the Primary SA and the existing Static SAs, which the SS may have access). Now, the BS activates the Authorization Key (AK) and constructs the Authorization Reply Message (AuthorizationRepMess) with the following information:
    - AK (encrypted with the public key of the SS (S_{pub}-SS)),
    - 4-Bit AK Sequence Number (to differenciate between consecutive AKs),
    - AK-Lifetime
    - SA-Descriptor(s)
- **Step 4:** The SS calculates the KEK and the Message Authentication keys (HMAC_Key_D and HMAC_Key_U) from the AK. These are needed for the TEK exchange phase. The SS has to keep its authorization status to further receive current TEKs. Thus, it periodically renews the AK by sending Authorization Request Messages, before the current AK expires. This reauthorization is identical to the original authorization. However, the AuthenticationInfMess (i.e. Step 1) becomes obsolete.

[1] Each SAID has a set consisting TEKold and TEKnew.

[2] Data encryption is possible using Data Encryption Standard (DES) Algorithm in Cipher Block Chaining (CBC)-Mode with 56-Bit keys and also using Advanced Encryption Standard (AES) Algorithm in CCM-Mode with 128-Bit keys.

[3] The PKM Protocol is used both in Point-To-Multipoint (PMP) and in meshed networks.

[4] Each SS owns a X.509 certificate issued and installation by the vendor and the associated X.509 certificate of the vendor itself.

[5] The encryption methods supported by the SS are presented to the BS in the form of a list of consecutive cryptographical identifiers (Cryptographic Suites). Each Suite has a set of packet data encryption and packet data authentication algorithms and also the encryption algorithm for the TEK.

[6] The Basic CID (Basic Connection Identifier) of the SS is the first Static CID, which the BS assigns to the SS during the initial connection establishment (Initial Ranging).

Fig. 2 depicts the communication workflow during the three phases and respective steps 1 - 4.

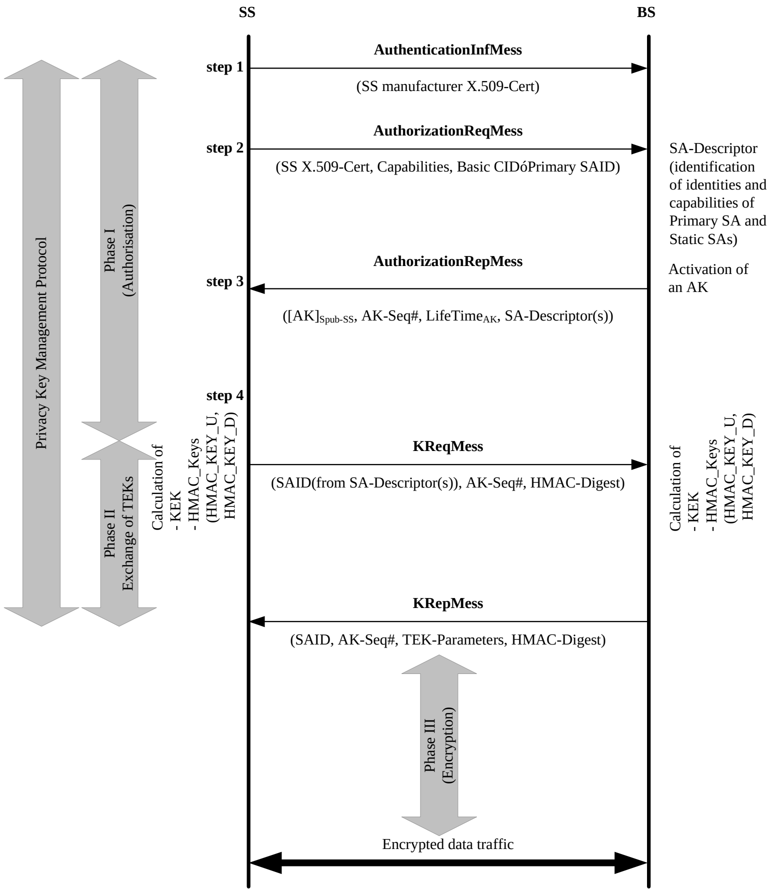
SS
BS
step 1
AuthenticationInfMess
(SS manufacturer X.509-Cert)
step 2
AuthorizationReqMess
(SS X.509-Cert, Capabilities, Basic CIDóPrimary SAID)
SA-Descriptor (identification of identities and capabilities of Primary SA and Static SAs)
step 3
AuthorizationRepMess
([AK]Spub-SS, AK-Seq#, LifeTimeAK, SA-Descriptor(s))
Activation of an AK
step 4
Calculation of - KEK - HMAC_Keys (HMAC_KEY_U, HMAC_KEY_D)
KReqMess
(SAID(from SA-Descriptor(s)), AK-Seq#, HMAC-Digest)
Calculation of - KEK - HMAC_Keys (HMAC_KEY_U, HMAC_KEY_D)
KRepMess
(SAID, AK-Seq#, TEK-Parameters, HMAC-Digest)
Phase III (Encryption)
Encrypted data traffic
Privacy Key Management Protocol
Phase I (Authorisation)
Phase II Exchange of TEKs

Figure 2: Communication workflow during the three phases and steps 1 - 4

Phase II: Exchange of Key Material
After successful authentication the second phase initiates the exchange of the TEKs, which are needed for data encryption.

The SS requests key material and starts a separate TEK state machine for each SAID (identified in the Authorization Reply Message). The state machine is responsible for the key material management and autonomously manages single SAs and associated key material. It periodically sends (for TEK renewal) corresponding requests to the BS (Key Request Message). The BS responds to the requests with a Key Reply Message, which contains the new key material needed by the state machines. The BS permanently maintains two active sets of key material for each SAID, which are denoted in the Key Reply Message as TEK-Parameter. They comprise the TEK (encrypted with the KEK), the TEK-Lifetime (remaining validity duration of the TEK), the TEK Sequence Number, and an Initialization Vector (64 Bits). One of the two TEK-Parameters contains an old TEK and the other a new one. The old TEK is being used by the BS in order to encrypt downlink data traffic whereas the new TEK is being used by the SS to encrypt uplink data traffic.

The SS uses the remaining validity durations (TEK-Lifetime) of the keys to estimate, when the BS will discard a specific TEK or when it will request a new TEK. The validity durations of the two TEKs overlap, consequentially a TEK is being activated before the lifetime of the predecessor TEK ends and is destroyed after the activation of the successor TEK. The lifetime of a TEK lies between 30 minutes minimum and 7 days maximum [1].

The Key Request Message is the first message from the SS requesting key material for data encryption. This message contains the following information [1]:

- SAID: A SAID started by the TEK state machine (the SAID is one of the SAIDs from the list with SA descriptions within the Authorization Reply Message)
- AK Sequence Number: With this number the BS is able to determine which HMAC_Key_U from the SS has been used for generating the HMAC-Digest.
- HMAC-Digest: This is the HMAC-Digest of the Key Request Message using HMAC_Key_U. It serves the BS as a message integrity check and ensures that SS and BS possess the same AK.

The BS receives the Key Request Message and checks, if the corresponding HMAC-Digest is valid and if the SAID herein matches the SA at the SS. If so, it responds with the Key Reply Message, containing an AK Sequence Number, a SAID, a HMAC-Digest, and also the TEK-Parameters. As with the first message, the HMAC-Digest shall assure the SS, that this message stems from the BS, and has not been modified (spoofed). However, if the SAID in the Key Request Message should be invalid, the BS reacts with a Key Reject Message, which contains an AK Sequence Number, the SAID (which the key material has been requested for), an error code with a display string (displaying the reason for rejection) and an HMAC-Digest. The SS again has to send a Key Request Message to the BS and receives completely new key material (TEK).

If the SS has already started to encrypt data with the existing key material, in uplink direction, the BS checks the key material used. In case it realizes an Uplink PDU (data stream) has been encrypted with an invalid TEK, (TEK Sequence Number for the SAID used is no longer in the TEK Sequence Number range known by the BS), it sends a TEK Invalid Message. This message contains the same parameters as the Key Reject Message [5].

Phase III: Encrpytion of the Data Stream
After completion of the SS-Authorization and TEK exchange, the data stream (MAC PDU) has to be encrypted. The MAC-Header itself is not involved, otherwise it would be impossible to directly forward the MAC PDU. Also, the optional CRC checksum of the MAC PDU remains unencrypted. Encryption takes place by means of DES (DES in CBC-Mode with 56 Bits) or AES (AES in CCM-Mode with 128 Bits). The encryption of the TEK can be done in three ways [4], [2]: 3DES in EDE-Mode with 128-Bit, RSA (PKCS #1 v2.0) or AES in ECB-Mode with 128-Bit.

3. SHORTCOMINGS, ATTACKS, AND COUNTERMEASURES

Although at first sight IEEE 802.16d appears to be secure WiMAX is vulnerable to some attacks. In this article we will not address general attacks such as cloning, brute force, jamming or scrambling, as this would go beyond the scope of this paper.

Instead, we will focus on the two essential phases which have shortcomings: “Authentication Phase” and “Key Material Exchange Phase”.

Authentication of the SS - Man-in-the-Middle and Forgery, Replay and DoS-Attack

One of the obvious weaknesses is a lack of mutual authentication between SS and BS. SS authenticates itself through its certificate, however, the BS does not. A potential attacker who pretends to be a BS (rogue BS) could place himself between SS and real BS and try to force SS to authenticate itself and initiate a session by transferring an AK (forgery-attack). The attacker can generate his own Authorization Reply Message containing a self-generated AK and thus gain control over the communication of the attacked SS. This is a typical Man-in-the-Middle-Attack (MiM). The SS is not able to recognize whether the messages of the authorization phase stem from a trustworthy BS. With the credentials of the SS the attacker could register himself at the BS [6], [1], [7].

Replay- and DoS-Attack

The SS begins with an Authentication Information Message and a subsequent Authorization Request Message, with the aim to transmit all relevant information to the BS. The latter responds to the last message with an Authorization Reply Message. Although the message is transmitted in plaintext, it does not constitute a problem since the information is public anyway. However, the BS can fall victim to a replay-attack by which the attacker intercepts an Authorization Request Message from an authorized SS and stores it. Even though he will not be able to derive the AK from the Authorization Response Message (since he does not possess the associated private key), he can repeatedly send the message to the BS, burdening the BS with the effect that this declines the real/authentic SS. This is a DoS-attack against the SS [6], [1], [7].

Countermeasures and Recommendations

A countermeasure against replay-/DoS-attacks is to equip the Authorization Request Message with a time stamp together with a signature of the SS. These additional parameters, would guarantee message authenticity. In this context, the signature should use the private key of the SS in order to protect sensible information within this message.

An authentication of the BS could be achieved if a certificate of the BS is appended to the Authorization Reply Message. The time stamp received within the Authorization Request Message should also be appended to this message, so that the SS can be sure, that the Authorization Reply Message corresponds with its Authorization Request Message. The appended signature of the BS inside the Authorization Reply Message would allow message authentication and non-repudiation of the origin.

A viable alternative to the time stamp would be using a nonce by which the SS can be sure, that the Authorization Reply Message is the corresponding response to the Authorization Request Message. However, the BS is exposed to replay-attacks since it can not recognize whether the Authorization Request Message has been sent recently or is an old message. An improvement can be achieved by replacing the AK with a pre-AK (preliminary AK) and sending it to the SS. SS and BS each would be able to derive the AK from the pre-AK.

However, if the pre-AK is compromised, an attacker could derive the AK with the same algorithms and a freshness-identifier (nonce or time stamp) of the SS and BS, which are transmitted in plaintext.

Nonce and time stamp are two essential methods for the verification of the timeliness (freshness) of messages. A main disadvantage of time stamps is that the communication partners have to synchronize their time base. However, SS and BS are already synchronized (system block) at initial connection establishment with the WiMAX network prior to the authorization phase. Thus, the synchronization should not pose a problem using time stamps [7], [1].

Key Material Exchange Phase - Attacks against the Key Sequence Number

After completion of the authorization phase, the SS requests key material (TEKs), necessary for data encryption. For this purpose, it periodically sends Key Request Messages referring to one of its valid SAIDs. The BS replies with a Key Reply Message containing valid key material for the given SAID. One potential replay-attack is possible due to the Key Sequence Number of the TEK, which has a length of only two Bits. This Sequence Number is part of the TEK parameter within the Key Reply Message. It is used in a circle buffer changing its values to the tiny range of 1 to 4. An attacker is able to capture TEK messages and replay them to gain information needed in order to decrypt data traffic [6], [1].

Countermeasures and Recommendations

A considerable mitigation of this vulnerability could be achieved by increasing the sequence number length so that a satisfactory amount of TEK Sequence Numbers can be generated and transmitted within the longest validity duration of the AK. Using 70 days as highest duration of an AK and 30 minutes as the smallest duration of a TEK, a Data SA could theoretically consume 3.360 TEKs over a complete AK-Lifetime [1].

4. COMPARISON WITH IEEE 802.16e

In chapter 3, we have discussed some weaknesses of IEEE 802.16d. Fortunately, considerable effort has been invested in the security architecture of the newer standard IEEE 802.16e (Mobile WiMAX). Compared to IEEE 802.16d it is by far more secure and the improvements are obvious. Most of the shortcomings have been eliminated. By adding encryption methods and by introducing mutual station authentication, Mobile WiMAX can be judged as much more secure.

However, this chapter can only outline the most essential improvements within IEEE 802.16e, and only the basic differences between these two standards will be presented, since even an overview of the security features would go beyond the scope of this paper.

Authentication and Authorization

Mutual Authentication

The most obvious weakness of IEEE 802.16d is that the SS is not able to check the identity of the BS. The probability of a classical Man-in-the-Middle-Attack is high: a potential intruder can act as BS (rogue BS). In IEEE 802.16e this shortcoming has been fixed by PKMv2 introducing mutual authentication, giving both stations the possibility to check each other's identity. The standard offers two methods for mutual authentication: RSA, EAP (RFC 3748) or even combined. RSA-based authentication uses digital certificates according to the X.509 standard whereas the latter offers EAP-methods such as EAP-TLS, EAP-SIM, EAP-AKA or EAP-MSCHAPv2, depending on the solutions provided by the WiMAX network operator, and also the version of the Privacy Key Management Protocol (PKM) (Table 1).

Authentication of User Data and Control Message Protection

For user data integrity Mobile WiMAX uses AES in CCM-Mode. Keys used for driving the cipher are generated from the EAP authentication mechanism. CCM-Mode combines counter mode encryption and cipher based message authentication (CMAC). Since each CMAC key is used only once an efficient protection against replay-attacks is given. Control data is protected using AES based CMAC, or HMAC schemes.

Authorization

Another important improvement has been achieved concerning the generation of the Authorization Key (AK) within the Authorization Phase leading to a higher security level. Assignment and transmission of the AK is of significance, since the AK provides key material for the Key Encryption Key (KEK) and the HMAC-/CMAC-Keys. Within Fixed WiMAX the AK has been generated by the BS only. In case of a weak random generator the AK is exposed to brute-force-attacks and with a compromised AK all data traffic can be decrypted[7]. In Mobile WiMAX both stations are involved in the calculation of the AK. Furthermore, the calculation of the AK depends on the authentication method.

In addition, all sensitive messages between the stations are secured by Nonces (which are random values) providing the messages with unique identities and thus protection against replay-attacks.

TEK 3-Way Handshake

As outlined in chapter 3 the TEK exchange phase has considerable weaknesses. Mobile WiMAX improves the key material exchange procedure by means of the PKMv2 SA-TEK 3-Way Handshake either during initial authentication or handover. Each message in the 3-Way Handshake is secured by Nonces, an AKID (which identifies the AK), and also a Message Authentication Code derived from the AK. With these new security features a MiM-Attack is impossible.

Encryption of Data

Data Encryption in IEEE 802.16d

In IEEE 802.16d data encryption can be accomplished with two different methods [8], [9]: DES in CBC-mode with 56 bit keys and AES in CCM-mode with 128 bit keys. By means of the first method data is fragmented into 64-bit plaintext blocks which are encrypted with the key each. With the second method using AES in CCM-mode data has to be segmented into fixed block sizes of 128 bits. The MAC PDU to be encrypted receives a prefix of 4 bytes – the packet number. An Integrity Check Value (ICV) with a length of 8 bytes is attached at the end of the payload. Payload and ICV are encrypted with the TEK according to the CCM-mode.

The DES-algorithm only allows block sizes of 64 bits. Due to the restriction of key length it can be easily comprised with 56. It is therefore recommendable to exclusively use AES in IEEE 802.18d. With variable block sizes of 128, 192, and 256 bits as well as variable key lenghts of 128, 192 und 256 bits AES offers a high degree of security for Fixed WiMAX.

Data Encryption in IEEE 802.16e

IEEE 802.16e allows data encryption in four different ways. However, the MAC PDU payload is encrypted, whereas the generic MAC header[8], and the optional CRC-checksum are transmitted in plaintext. Also, MAC Management Messages are transported unencrypted.

Encryption of Keys

TEK-encryption in IEEE 802.16d

In IEEE 802.16d TEK-encryption can be carried out in three different ways[9]:

- 3DES in EDE-mode using 128 bit key
- RSA-encryption (PKCS #1 v2.0)
- AES in ECB-mode using 128 bit key

Within a PMP network the KEK is with 3DES or AES. In a Meshed network the TEK-encryption is based on RSA using the public key of the SS [2], [4], [8].

- 3DES in EDE-mode using 128 bit key: The KEK, a 128 bit long 3DES-key and derived from the AK, is applied for TEK-encryption in EDE-mode[10]. The first (most significant) 64 bits are used for encryption and the least significant 64 bits for decryption. In the first cycle the TEK is encrypted with the most significant 64 bits of the KEK, in the second cycle with the least significant 64 bits, and in the third (and last) cycle again with the most significant 64 bits. In all three cases the DES-algorithm is used in ECB-mode, which encrypts the plaintext blocks consecutively and independently from each other.
- RSA-encryption (PKCS #1 v2.0): Since RSA is an asymmetric method two different keys are used: public and private key.
- AES in ECB-mode using 128 bit key: AES is a symmetric method. Hence, only one key is used for encryption and decryption. By means of the ECB-mode the plaintext blocks are encrypted consecutively and independently from each other using the KEK. However, unlike the 3DES-method the full KEK (128 bits) is used without any segmentation.

TEK-encryption in IEEE 802.16e

In Mobile WiMAX TEK-encryption within PKMv2 can be carried out in four ways:

- 3-DES in EDE-mode using 128-bit keys
- RSA-encryption using 1024-bit keys
- AES in ECB-mode using 128-bit keys
- AES Key Wrap using 128-bit keys

The AES Key Wrap has been added to IEEE 802.16e. All other methods are integral to IEEE 802.16-2004 [10].

Encryption of GKEK and GTEK in PKMv2

The GKEK is a random value generated by the BS, encrypted by the KEK and transmitted to the M-STA. Only one GKEK is assigned to a Group Security Association (GSA). The GKEK-encryption within PKMv2 is carried out identically to the TEK using the same encryption method within PKMv1 respectively PKMv2. Contrary to PKMv1, it additionally defines two further encryption methods. Therefore, the GTEK Encryption Algorithm Identifier is the same as the TEK Encryption Algorithm Identifier.

The following methods are applied:

- GKEK with 3DES within PKMv2
- GKEK with RSA within PKMv2
- GKEK with AES in ECB-mode
- GKEK with AES Key Wrap

In all four methods the GKEK is encrypted with the KEK both in the initial GTEK Request Exchange (within the PKMv2 Key

[7] Since data traffic is encrypted with the TEK and the TEK itself encrypted with the KEK, an attacker derives the KEK out of the AK and decrypts TEK and thus the data communicated between the stations.

[8] The Generic MAC Header is he prefix of the MAC PDU. It contains specific information with respect to the MAC PDU format such as MAC PDU length, and the information whether payload data is encrypted or not.

[9] This depends on the operation type of the WiMAX network: Meshed or im Point-To-Multipoint-Mode.

[10] EDE: Encrypt, Decrypt, Encrypt.

Reply Message) and also in the GKEK Update (within the Key Update Command Message). Also during the SA-TEK 3-Way-Handshake, which takes place after an initial authentication (or re-authentication) in order to exchange key material, the GKEK is encrypted with the KEK. The GKEK is used to derive the HMAC_/CMAC_KEY_GD[11] (only for the downlink direction) and also to encrypt the GTEK.

Table 1 summarizes the basic features and compares Fixed and Mobile WiMAX.

		802.16d Fixed WiMAX	***802.16e Mobile WiMAX***
Crypto-graphic Suites	***Data Encryption/ Decryption***	DES in CBC-Mode with 56 Bit key	DES in CBC-Mode with 56 Bit key
			AES in CCM-Mode with 128 Bit key
		AES in CCM-Mode with 128 Bit key	AES in CBC-Mode with 128 Bit key
			AES in CTR-Mode with 128 Bit key for Multicast-/Broadcast-Services with 8 Bit Roll-over Counter
	Data-Authentication	not supported	AES in CCM-Mode with 128 Bit key
	Key Generation	not supported	Dot16KDF
	Key Encryption/ Decryption	3DES in EDE-Mode with 128 Bit key	3DES in EDE-Mode with 128 Bit key
		RSA-Encryption with 1048 Bit key	RSA-Encryption with 1048 Bit Key
		AES in ECB-Mode with 128 Bit key	AES in ECB-Mode with 128 Bit key
			AES Key Wrap with 128 Bit key
Methods	***Station Authentication***	only SS	SS and BS (mutual)
		X.509-Certificate	RSA or EAP
	Key Management (TEK)	unsecured	Secured with 3-Way-Handshake
	Station Authorization (Generation of AK)	BS	BS and SS Calculation depends on authentication method

Table 1: Comparison of Fixed and Mobile WiMAX

5. CONCLUSIONS

IEEE 802.16d provides a security architecture which basically secures the wireless link using different components such as X.509-certificates (identity of the communications parties BS and SS), Security Associations (SA), encryption methods (securing data transmission against attacks), the Encapsulation Protocol (specifying the encryption and authentication algorithms supported by the SS), and the Privacy Key Management Protocol (PKM).

The security analysis in this paper depicted that IEEE 802.16d is mainly vulnerable in two phases: authentication and key exchange phase. Suggestions for countermeasures made by security analysts and researchers have to be considered respectively implemented in products. Alternatively, the security mechanisms from IEEE 802.16e should be deployed in IEEE 802.16d implementations. If applicable, users or network operators should deploy IEEE 802.1e.

6. REFERENCES

[1] Bundesamt für Sicherheit in der Informations-technik 2006. “*Drahtlose Kommunikations-systeme und Ihre Sicherheitsaspekte”,* 2006: http://www.bsi.bund.de/literat/doc/drahtkom/drahtkom.pdf

[2] H. Chin-Tser, M. Matthews, X. Sen. *“Security Issues in Privacy and Key Management Protocols of IEEE 802.16”,* Department of Computer Science and Engineering - University of South Carolina Columbia, USA, 2006: http://www.cse.sc.edu/~huangct/acmse06cr.pdf#search=acmse06cr.pdf

[3] S. Wattanachai. “*Security Architecture of the IEEE 802.16 Standard for Mesh Networks”,* Department of Computer and Systems Sciences Stockholm University/Royal Institute of Technology, 2006. http://www.dsv.su.se/research/seclab/pages/pdf-files/2006-x-360.pdf#search=2006-x-360.pdf

[4] A.-V. Aikaterini. “*Security of IEEE 802.16 Master of Information and Communication Systems Security”,* Department of Computer and Systems – Science Royal Institute of Technology, 2006: http://www.dsv.su.se/research/seclab/pages/pdf-files/2006-x-332.pdf

[5] D. Johnston, J. Walker. “*Overview of IEEE 802.16 Security”,* IEEE Computer Society, 2004: http://mia.ece.uic.edu/~papers/WWW/Bubbles/segment/WiMax_Security.pdf#search=WiMax_Security.pdf

[6] M. Barbeau. “*WiMAX/802.16 Threat Analysis”,* School of Computer Science Carleton University, Ottawa, Ontario, Canada, 2005: http://www.scs.carleton.ca/~barbeau/Publications/2005/iq2-barbeau.pdf#search=iq2-barbeau.pdf

[7] F. Ohrtmann. Wimax Handbook. *“Building 802.16 Wireless Network”,* McGraw-Hill Communications, 2005.

[8] IEEE Standard for Local and Metropolitan Area: “*Networks-Part 16: Air Interface for Fixed Broadband Wireless Access Systems”,* 2006: http://standards.ieee.org/getieee802/download/802.16e-2005.pdf

[9] Network Working Group. Request for comments: 3394, 2002: http://www.ietf.org/rfc/rfc3394.txt

[10] F. Pugliese. *“Sicherheitsarchitektur des Standards IEEE 802.16-2004 für Point-To-Multipoint Netzwerke”*, 2007.

[11] The CMAC_KEY_GD is used to authenticate broadcast messages such as the Key Update Command Message in the GTEK Update Mode.