A Comparison of Security in Wireless Network Standards with a Focus on Bluetooth, WiFi and WiMAX

Günther Lackner
(Corresponding author: Günther Lackner)

Institute for Applied Information Processing and Communications, IAIK, University of Technology Graz
(Email: guenther.lackner@iaik.tugraz.at)

(Received Dec. 17, 2010; revised and accepted Mar. 26, 2011)

Abstract

As wireless networks are finally coming of age, people and organizations start to implement critical applications and infrastructures based on them. As most wireless network standards have been designed with security as an afterthought, severe security shortcomings were the results and several improvements and amendments were necessary to fix the worst. Founded on a series of insecure implementations and design faults, recent standards and amendments show some improvements. To cover personal area, local area and wide area wireless networks, the following standards have been chosen as examples: IEEE 802.15.1 Bluetooth, IEEE 802.11 WiFi and IEEE 802.16 WiMAX. This article provides a detailed overview, analysis and discussion of state-of-the-art security mechanisms in wireless networks and briefly presents their development and history allowing the reader to quickly gain detailed insight into the topic.

Keywords: Bluetooth, WiFi, WiMAX, wireless network security

1 Introduction

The number of deployed wireless networks increases every day. Due to the low cost and convenience of deploying wireless networks, they replace hardwired networks in many fields of application.

The shift from hardwired to wireless networks invalidates many established security concepts. Hardwired networks are usually integrated within structural measures, and can be protected by building security or perimeter protection. With a state-of-the-art intrusion prevention system (IPS) to protect the connection to the Internet, hardwired networks can thus be considered closed and secure, as illustrated in Figure 1.

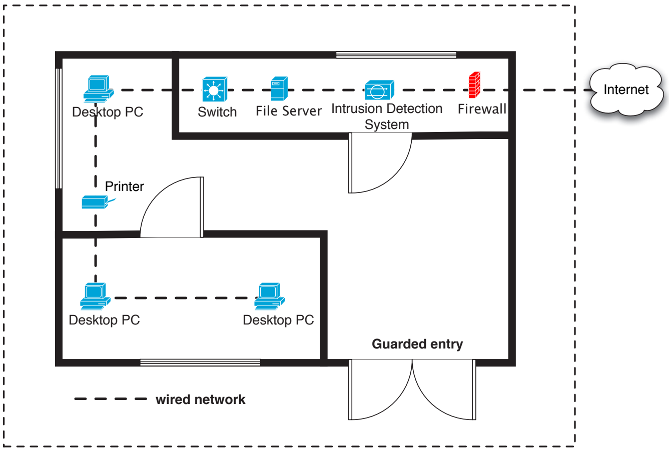


Figure 1: Wired-only environment with perimeter protection

The nature of radio propagation makes it possible to attack wireless networks from outside the established perimeter protection. Figure 2 illustrates how wireless network coverage could extend to a public domain outside of a controlled building (protected area).

As building security and perimeter protection are not sufficient to avoid attacks against the wireless network, the general approach is to secure these infrastructures by cryptographic measures. Almost all state-of-the-art wireless computer network technologies provide strong cryptographic mechanisms to provide confidentiality and integrity.

This article describes and discusses security mechanisms in *personal-area*, *local-area* and *wide-area* networks, each represented by a popular implementation namely IEEE 802.15.1 (Bluetooth), IEEE 802.11 (WiFi) and IEEE 802.16 (WiMAX). The focus lies on confidentiality, integrity and accountability.

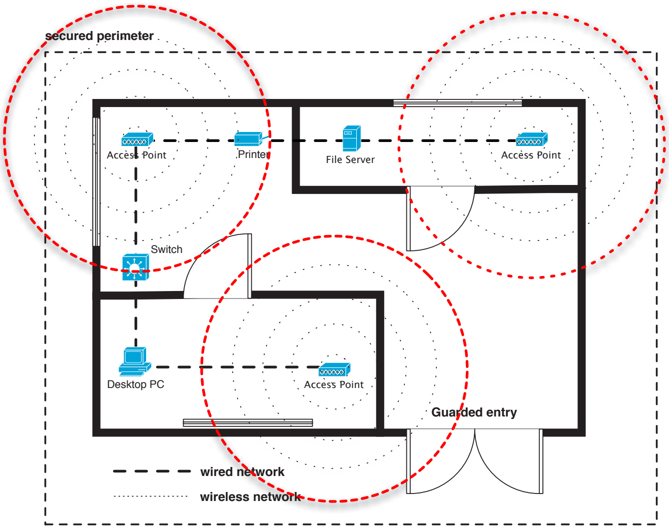


Figure 2: Environment with wireless components

2 Security in IEEE 802.11 (WiFi)

2.1 Wired Equivalent Privacy Algorithm (WEP)

Right from the release of the first IEEE wireless LAN standard 802.11, a security mechanism called *wired equivalent privacy* was integrated. The primary goal of this mechanism was to protect the confidentiality of user data from eavesdropping. This should be gained by enforcing three properties [11]:

Confidentiality: Prevent casual eavesdropping by a non-authorized clients.

Access control: Only authorized clients should be allowed to join the network.

Data integrity: It should be recognized if data was altered during the transmission.

All these properties are gained by using a secret key. The security of the WEP protocol only relies on the difficulty of discovering the secret key. If this difficulty only relies on the length of the key, and the only possibility of getting the key is an exhaustive search, the protocol is cryptographically secure.

WEP was initially designed for 40-bit keys with a resulting keyspace of $2^{40} = 1.099E9$. Using modern hardware it is no infeasible problem to discover the key with a brute-force approach in a reasonable time. As a consequence, the key length has been raised to 128-bit and an overall keyspace of $2^{128} = 3.402E38$. This extension renders an exhaustive key-search attack impossible, even with the most powerful hardware available [11].

Nevertheless, WEP owns some very critical design flaws that leave the standard practically futile. Although some feeble attempts to improve WEP were made like [15], the main vulnerabilities remained unchanged.

2.1.1 WEP Encryption/Decryption Process

Before taking a closer look at the encryption/decryption process, some terms need to be declared:

- Pseudo random-number generator (PRNG)
  Cryptography always needs some kind of random number source. In WEP, this task is done by the RC4 stream cipher.Seeded by some initialization value it creates a stream of *pseudo random-numbers*. But like all stream ciphers it will create the same keystream again if given the same seed.

- The initialization vector (IV)
  The IV is used to provide some diversion to the RC4 PRNG. It is 24-bits long and concatenated to the 40-bit secret key. In order to keep the PRNG from producing the same numbers for every packet, this IV needs to be changed as often as possible. There exist only $2^{24} = 16.777E3$ different IVs.

- The integrity check value (ICV)
  In order to provide data integrity, WEP uses the CRC32 algorithm. Before a packet gets encrypted, a *cyclic redundancy check value* with 32-bit length is computed and concatenated to the message. CRC32 is a linear function and does not provide any cryptographic security.

Figure 3 illustrates the message encryption process in WEP. The WEP-PRNG gets seeded by the secret key and some IVs and as the result it provides the so called *key sequence*. This key sequence is XORed with a concatenation of the plain text data and its CRC32 (ICV) value. Finally, the encrypted message is concatenated with the plaintext IV and transmitted [1].

The receiving client only needs to reverse the process to retrieve the plaintext massage, compute a CRC32 value of its on (ICV') and verify the integrity of the message by comparing the ICV and ICV'. The process is illustrated in Figure 4.

2.1.2 WEP Security Analysis

Several different attacks have been published during the last years. Most of them are based on the insecurity of the used RC4 stream-cipher. Although, RC4 was believed to be secure when it was integrated to WEP, it turned out to have some design flaws. While first attacks needed a high amount of collected data, more recent approaches like the attack of Andreas Klein [21] only need a relatively small number of transmitted packets. Klein's approach targeted flaws of the RC4 cipher. Erik Tews et al. [31] designed a process using Klein's approach and massive packet injection to generate enough traffic for breaking 128-bit WEP[1] in less than 60 seconds. Furthermore they do not need powerful special-purpose hardware, any contemporary personal-computer suffices. But not only

[1] Due to the 24-bit plaintext IV concatenated to the key, the effective key-length is only 104-bit.

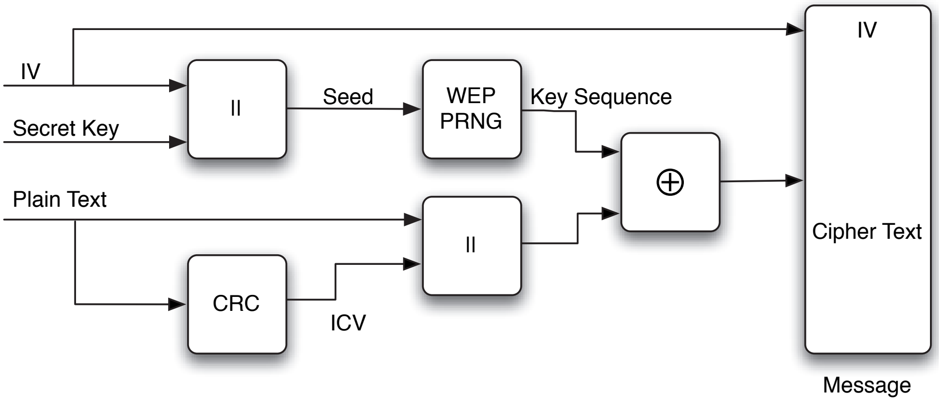


Figure 3: WEP encryption block diagram

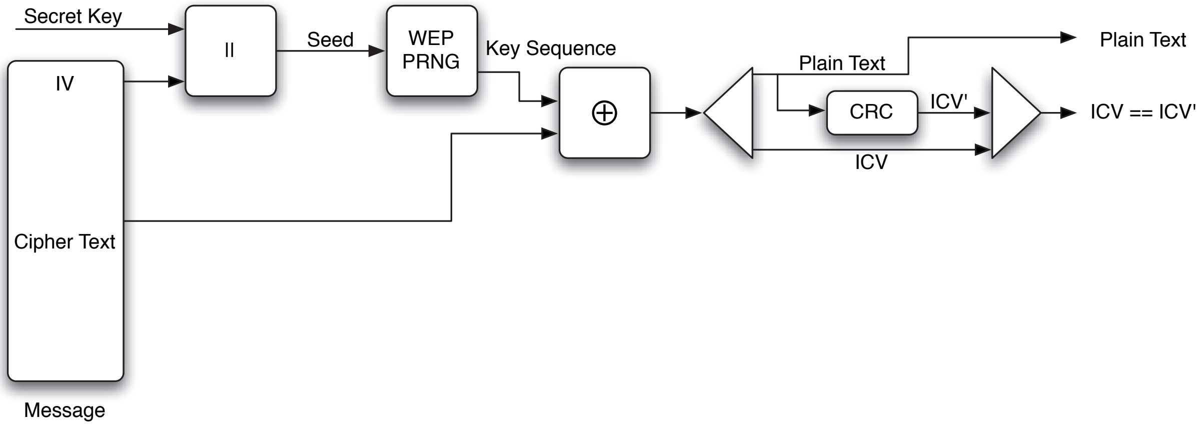


Figure 4: WEP decryption block diagram

RC4 may be exploited to break WEP. Also the very small number of IVs and their plaintext transmission offer a weak point. Another major vulnerability arises from the usage of the linear integrity check function CRC32. A detailed analysis of the components used in WEP is described in [11].

As a short conclusion it can be stated that WEP is highly insecure and should not be used if any other mechanism is available.

2.2 IEEE 802.11i (WPA, WPA2)

Since the publication of the WEP vulnerabilities and the upcoming of very effective attack implementations, the IEEE has begun the work on a replacement standard. On June 24th 2004, IEEE 802.11i ratified in order to provide enhanced security for wireless networks. A formal verification of this standard may be found in [13]. The standard specifies two classes of security algorithms:

- Robust Security Network Association (RSNA).
- Pre Robust Security Network Association (Pre-RSNA).

Pre-RSNA consists of WEP and 802.11 entity authentication while RSNA implements two new data confidentiality protocols known as *Counter-Mode-CBC-MAC Protocol* (CCMP) and *Temporal Key Integrity Protocol* (TKIP) and the RSNA establishment procedure that includes the use of the IEEE 802.1X authentication and key management protocol [3].

TKIP is meant to bring more security to legacy hardware by using available RC4 implementations, while CCMP demands AES compatible hardware.

The WiFi-Alliance[2] certified TKIP compatible hardware under the name *Wi-Fi Protected Access* (WPA).

2.2.1 Wi-Fi Protected Access (WPA)

WPA may be seen as a short-time fix to secure legacy hardware based WLANs. TKIP is based on RC4 and includes the keyed hash-function Michael [3] (cf. Section 2.2.1). TKIP can be described as a *"wrap"* around the existing WEP encryption/decryption to shield it's worst vulnerabilities. Due to the inherited insecurities and flaws, it does not provide sufficient security in the long-term [3].

[2] Nonprofit international association certifying interoperability of wireless local area network products based on IEEE 802.11 specification. http://www.wi-fi.org/

Figure 5 illustrates the TKIP encryption process while Table 1 explains the used notations.

Table 1: TKIP notations

Symbol	**Description**
TA	Transmitter address
TTAK	TKIP mixed transmitter address and key
TK	Temporal key
TSC	Sequence Number
IV	Initialisation vector
DA	Destination address
SA	Source address
MSDU	MAC service data unit
MPDU	MAC protocol data unit

The block *WEP encryption* corresponds with the WEP data encryption scheme presented in Figure 3. The TKIP extensions gain the security improvements only by modifying the input for the WEP encryption process. The most important change to classic WEP is that a new temporal key for each packet is used. This key is created by mixing together a base key, the MAC address of the transmitting station and a 48-bit serial number. The base key is newly created any time a station associates with the network and the mixing operation can be done with little computing power but provides a significant rise in cryptographic security. By adding the serial number into the key, it is assured that it will be different for each packet. An the 48-bit space for the serial number prevents WEP-collision attacks and replay attacks as well. Together with IEEE 802.1X, the secret keys are securely distributed between the participating STAs.

The second major vulnerability in WEP was the use of the linear CRC32 integrity check function. By implementing the Michael keyed hash-function, this problem was diminished but not solved as Michael also possesses some design flaws [33] (cf. Section 2.2.1).

Figure 6 shows the TKIP decryption process that can be seen as a *"wrap"* around the WEP decryption scheme. It works exactly the other way round as the TKIP encryption process.

Details of Michael Message Integrity Code (MIC) In 2004 the IEEE ratified the draft of the IEEE 802.11i standard. It is an amendment to 802.11 and should replace WEP in the long run. Besides a complete new design (*Counter-Mode-CBC-MAC Protocol*, CCMP), MIC also provides a compatibility mode for legacy hardware (*Temporal Key Integrity Protocol*, TKIP). TKIP implements a keyed hash-function called *Michael* that is meant to provide message integrity [17].

Michael is a *message integrity code* and was designed by Niels Ferguson in 2002 [14]. It is a keyed hash-function that takes a message of arbitrary length and a 64-bit Michael key. The key is converted into two 32-bit words and the output message is partitioned in blocks of 32-bit length and padded that the message length is a multiple of four.

Like any keyed hash-function Michael should fulfill the basic requirements [32]:

1) The message digest code (MDC) $h(m)$ can be calculated very quickly.

2) h must be a *one-way* function.
   Given a y it must be computationally infeasible to find an m' with $h(m') = y$. We are not trying to find the message. y is a MDC of some message.

3) It must be computationally infeasible to find messages m_1 and m_2 with $h(m_1) = h(m_2)$. The function is then called *strongly collision-resistant.*

Even the author of Michael knew about this flaw right from the release. Its is even mentioned in [14] on Page 6:

> A known-plaintext attack will reveal the key stream for that IV, and if the second packet encrypted with the same IV is shorter than the first one, the MIC value is revealed, which can then be used to derive the authentication key.

Avishai Wool was able to create a simple function that is capable of inverting Michael, and he proposed a related-message attack [33]. In [18], Huang et al. proved that Michael is also not *collision-resistant.* In fact it is not very hard to find a collision and furthermore launch a packet-forgery attack.

Although these attacks are not practical yet, they reveal weaknesses in Michael that render it as not secure on the long run.

TKIP Security Analysis Due to the inherited WEP vulnerabilities and the fact that some parts of TKIP (like Michael) posseses known security relevant flaws, WPA can not be assumed to be secure in the long run. However, it has always be seen as a short-time fix for WEP and it does its job pretty well. But as mentioned before, it is just a fixture and not a perfect solution. So, whenever possible, the use of WPA2 has to be preferred.

2.2.2 Wi-Fi Protected Access 2 (WPA2)

The Wi-Fi Alliance certified systems in compliance to IEEE 802.11i's *Robust Security Network Association* (RSNA) algorithm *Counter-Mode-CBC-MAC* (CCMP) under the name *Wi-Fi Protected Access 2* (WPA2). WPA2 may be seen as the first wireless network protocol that provides real cryptographic security. The only shortcoming is the need of new hardware because the WEP standard cipher RC4 has been replaced by the Advanced Encryption Standard (AES) [3].

The use of AES brings some very significant advances. With one single 128-bit AES key one is able to encrypt all

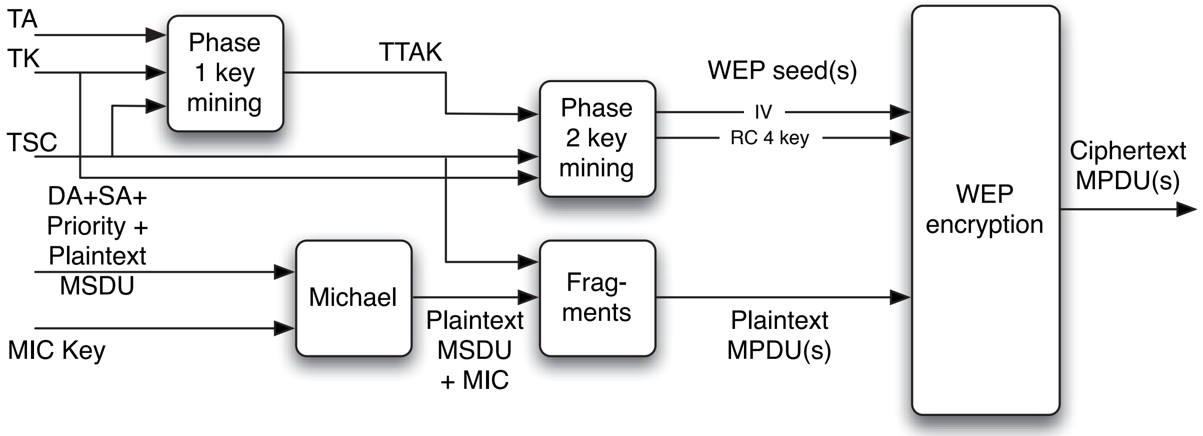


Figure 5: TKIP encryption block diagram

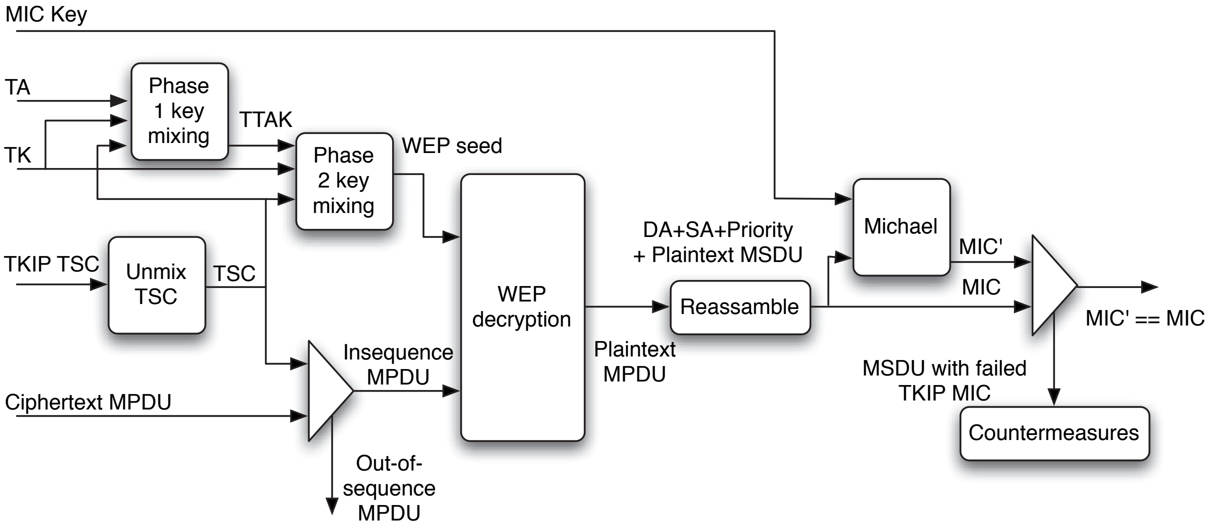


Figure 6: TKIP encryption block diagram

packets, eliminating the key scheduling problems of WEP and TKIP. CCMP also provides an AES based *Message Integrity Code* (MIC) over the frame body and nearly the complete MAC header. Message confidentiality and integrity are both gained by the use of the same 128-bit AES key. Like in TKIP, CCMP also implements a 48-bit serial number (PN) to prevent replay attacks and PN collisions.

Figure 8 illustrates the CCMP encryption process while Table 2 explains the used notations.

The following steps explain the CCMP encryption of the payload of a plaintext MPDU and the encapsulation of the ciphertext in a MAC frame:

1) In order to obtain a new PN for each MPDU respectively for the temporal key creation, it is incremented after each packet.

2) The additional authentication data (AAD) is created from the MAC header and provided to the CCM encryption module.

Table 2: CCMP notations

Symbol	**Description**
PN	Packet number
A2	MPDU address 2
AAD	Additional authentication data
TK	Temporal key
KeyId	Key identifier
MPDU	MAC protocol data unit

3) The CCM Nonce is formed of the incremented PN, the A2 and the Priority field.

4) The key identifier (keyId) and the PN are placed in the CCMP header.

5) The TK, AAD, Nonce and MPDU data is taken by the CCM encryption to form the ciphertext and MIC. This step is also known as *CCM originator process-*

ing.

6) The final step is to combine the results of the former steps to form the packet including the MPDU header, the CCMP header, the encrypted data and the MIC.

Figure 7 shows the format of the WPA2 packet after CCMP encryption.

The CCMP decryption process shown in Figure 8 works exactly the other way round as the decryption process.

AES in CBC mode provides mathematically proven security. Without the knowledge of the key, an adversary is not able to break data confidentiality or integrity. Even with a *known-plainttext-attack*, it is not possible to obtain any information about the key [16].

But like any relevant cryptographic mechanism, CCMP relies on the privacy of the key. It is well known that *pre-shared key schemes* are very vulnerable. Therefore, IEEE 802.11i defines the RSNA establishment procedure to ensure strong mutual authentication by using the 802.1X protocol. This mechanism is not only restricted to CCMP but may also be integrated in TKIP.

CCMP Security Analysis

The usage of the AES introduced mathematically proven cryptographic security to wireless networks. Without the knowledge of the key, an adversary is not able to break CCMP data confidentiality or data integrity. Supported by the (proper) use of IEEE 802.1X the temporal keys may be exchanged securely between the communicating stations and it is not possible for an attacker to obtain a key. CCMP in connection with IEEE 802.1X is the best available security solution for wireless networks. The fact that CCMP does not protect MAC control- and management-frames leaves some inherited WEP vulnerabilities unaddressed.

3 Security in IEEE 802.15.1 (Bluetooth)

Bluetooth is an open standard for short-range radio frequency communication. It has been designed to easily establish wireless personal area networks (WPAN), often referred to as ad-hoc or peer-to-peer networks. Initially integrated into personal computers and mobile phones, Bluetooth can nowadays be found in a wide variety of devices as headphones, portable music-players or even in cars [28].

There have been several versions of Bluetooth, with the most recent released definition being Bluetooth 4.0. The released versions differ greatly in bandwidth and the provided security. Being most of the available devices still implemented according to Bluetooth 2.1 and earlier, this section will focus on their analysis [28].

Like WiFi, Bluetooth operates in the unlicensed 2.4 GHz ISM frequency band. Therefore it is primarily vulnerable to all physical layer Denial of Service (DoS) attacks like channel jamming. As BT implements channel-hopping at a very high rate, changing frequencies about 3200 times per second, it shows some resistance against these DoS attacks.

The BT standard specifies the following three security services [35]:

- **Authentication:** This service authenticates the communicating devices. User authentication is no natively provided by Bluetooth.
- **Confidentiality:** Ensuring that only authorized devices can access transmitted data and therefore prevent all kinds of eavesdropping.
- **Authorization:** As bluetooth allows to control connected resources (printers, headphones, etc.), this service assures a devices authorization before allowing it to do so.

Other security services as *non-repudiation* are not provided by BT [28].

3.1 Bluetooth Security Modes

Cumulatively, the BT versions up to 2.1 define four modes of security. Each of these version support some of these modes but none of them supports all four.

3.1.1 Security Mode 1

This mode is non-secure. Authentication and encryption are bypassed leaving this mode without any security measures at all. Mode 1 is only supported in BT 2.0 + EDR and earlier versions [28].

3.1.2 Security Mode 2 (Service-level Enforced)

Mode 2 is designed as a *service-level enforced security-mode*. It is possible to grant access to some services without providing access to others. It introduces the *notion of authorization*, the process of deciding if a specific device is allowed to have access to a specific service. A centralized security manager (as defined in the BT architecture) controls access to specific services and devices. The security measures take place after the physical link has been established. Security Mode 2 is supported by all Bluetooth devices [28].

3.1.3 Security Mode 3 (Link-level Enforced)

This mode mandates authentication and encryption for all connections to and from the device. All security measures take place before the physical link is fully established. Security Mode 3 is only supported in Bluetooth 2.0 + EDR and earlier devices [28].

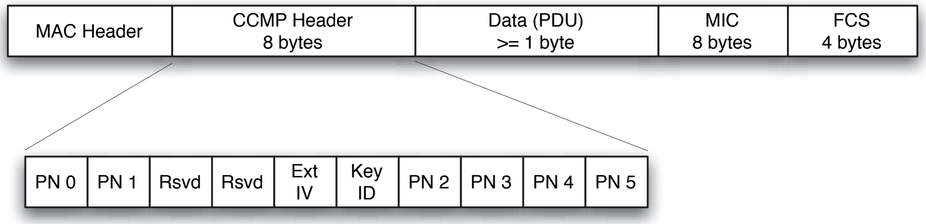


Figure 7: WPA2 packet format

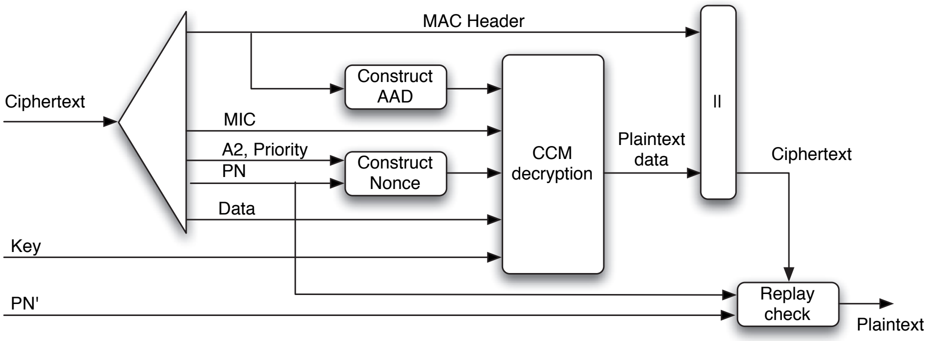


Figure 8: CCMP encryption block diagram

3.1.4 Security Mode 4 (Service-level Enforced)

Similar to security Mode 2, this mode is enforced on the service level, after the physical link has been established.The pairing mechanism uses Elliptic Curve Diffie Hellman (ECDH) techniques. Services supported by Mode 4 must be classified as one of the following:

- Authenticated Link Key required.
- Unauthenticated Link Key required.
- No security required.

Security Mode 4 is mandatory for communication between devices in compliance to Bluetooth 2.1 + EDR or newer versions [28].

3.2 Bluetooth Key Management

The various defined Bluetooth security mechanisms require several different keys. According to the used security mode, some of them are used to establish the connection and derive a Link Key between two devices. This Link Key can be semi-permanent or temporary. A semi-permanent key might be stored in the nonvolatile memory of a device and therefore used for multiple sessions, while the lifetime of a temporary key is limited to the current session [35].

- **K**$_{AB}$ - Combination Key
  The Combination Key is derived from information in both connecting devices A and B. It therefore depends on two devices. K$_{AB}$ is derived for each new combination of two devices.

- **K**$_{A}$ - Unit Key
  Contrary to K$_{AB}$, K$_{A}$ is only derived from the information of a single device. It is generated at the installation of the device and usually very rarely changed.

- **K**$_{master}$ - Master Key
  In a point-to-multipoint (Broadcast or Multicast) scenario, a common encryption key (K$_{master}$) may be used to replace the current Link Keys.

- **K**$_{init}$ - Initialization Key
  The Initialization Key should be used to as the Link Key during the initialization process, when no combination or unit keys have been exchanged yet. It protects the transfer of initial parameters. In security modes 2 and 3, this key is derived from tre triple of random number, a PIN code and the devices hardware address.

- **K**$_{link}$ - Link Key
  The Link Key is usually a 128-bit random number which is shared between two ore more parties as the base for all cryptographic transactions. It is used in the authentication routine and to derive the Encryption Key K$_{c}$.

- **K**$_{c}$ - Encryption Key
  The Encryption Key is used for encrypting all transmissions during a session. It is usually derived from the Link Key K$_{link}$.

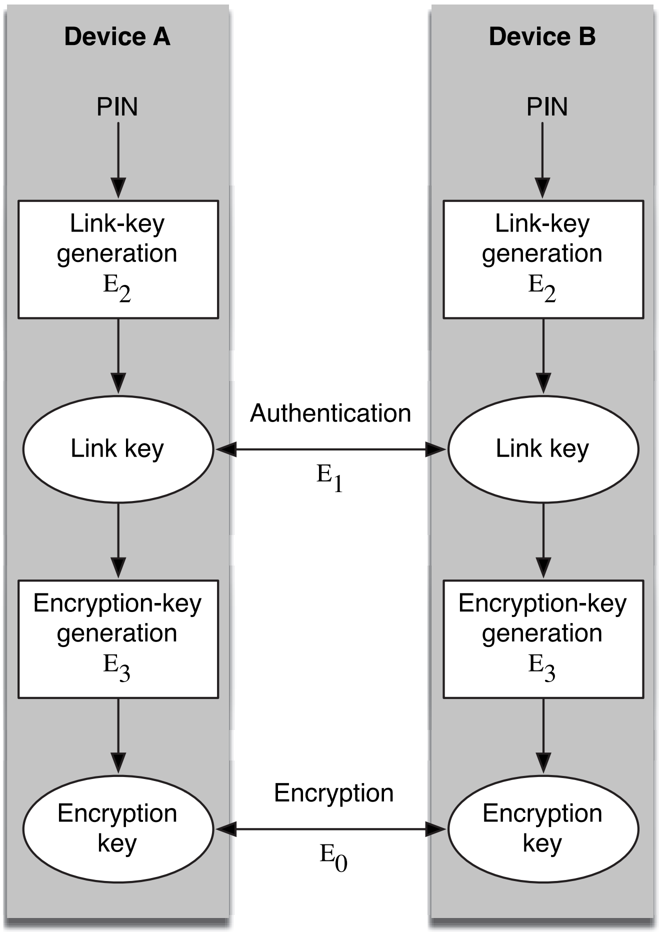

Device A
Device B
PIN
PIN
Link-key generation E_2
Link-key generation E_2
Link key
Link key
Authentication
E_1
Encryption-key generation E_3
Encryption-key generation E_3
Encryption key
Encryption key
Encryption
E_0

Figure 9: Overview of the Bluetooth key generation routines for security Modes 2 and 3 [20]

3.2.1 Link Key Generation in Security Modes 2 and 3

As the Link Key must be distributed among the communicating devices in order to allow the authentication procedure, it has to be created during the initialization phase. This procedure is also called pairing and consist of the following five steps:

1) Generation of an Initialization Key;

2) Generation of a Link Key;

3) Link Key exchange;

4) Authentication;

5) Generation of encryption keys (optional).

Bluetooth standards define a number of generic cryptographic building blocks called E_0, E_1, E_2 and E_3 [35].

- E_0 - a stream cipher function
- E_1 - the authentication function
- E_2 - the Link Key generation function
- E_3 - the Encryption Key generation function

These building blocks are mainly based on the block cipher SAFER+ and Linear Feedback Shift Registers (LFSR). Figure 9 provides an overview of the Bluetooth key generation process and the used cryptographic building blocks for security Modes 2 and 3.

3.2.2 Secure Simple Pairing (SSP) in Security Mode 4

SSP was introduced in Bluetooth 2.1 + EDR for the use with security Mode 4. It simplifies the pairing process by providing four flexible association models [28]:

- **Numeric Comparison**
  During pairing the user is shown a six digit number allowing her to enter a “yes” or “no” response if the numbers do match on both devices.
- **Passkey Entry**
  One of the devices shows a six digit number which the user has to enter on the second device in order to allow pairing.
- **Just works**
  Is designed for the use of devices without displays or an input possibility. Keys are exchanged in plaintext leaving a vulnerability for man-in-the-middle attacks.
- **Out of Band (OOB)**
  OOB is an extension that allows devices with additional wireless techniques like near field communication (NFC), to use them for device discovery and cryptographic value exchange. Devices can therefore be paired by simply “tapping” one device against the other.

Figure 10 provides an overview of the Bluetooth Secure Simple Pairing process for security Mode 4.

3.3 Authentication in Bluetooth

Authentication in Bluetooth is based on a challenge-response scheme as shown in Figure 11. The authentication procedure takes the following steps [28]:

1) The verifier transmits a 128-bit random challenge (AU_RAND) to the claimant.

2) The claimant applies the E_1 authentication function using his unique 48-bit Bluetooth device address (BD_ADDR$_A$), the Link Key and AU_RAND as inputs. The verifier performs the same procedure. The 32 most significant bits of the E_1 output (SRES) are used for the authentication output while the remaining 96 bits (Authenticated Ciphering Offset - ACO) will be used later to create the Bluetooth encryption key.

3) The claimant returns the SRES to the verifier.

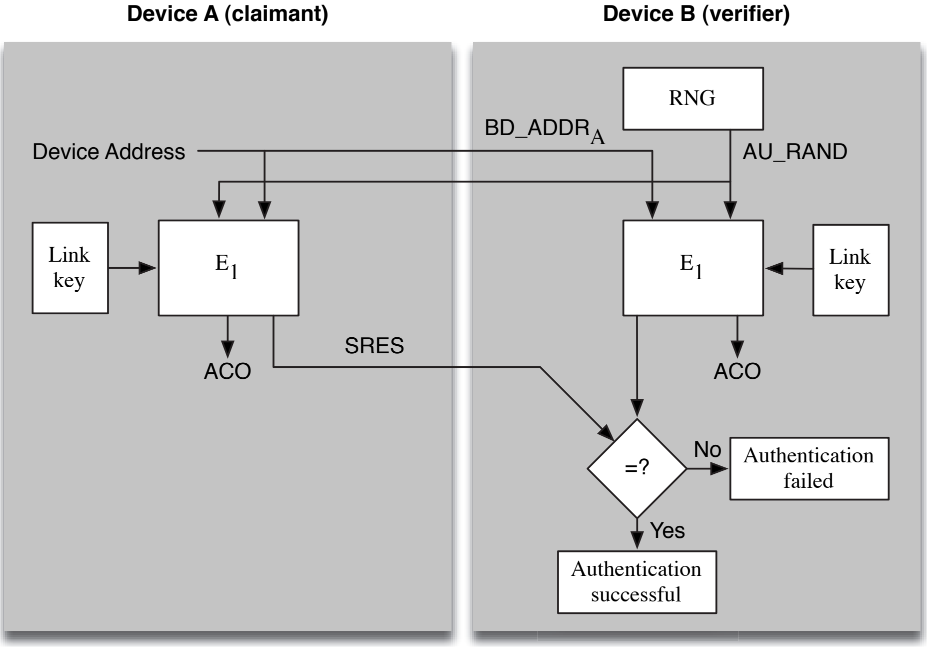


Figure 11: Bluetooth authentication [28]

4) The verifier compares the received SRES with its own outcome of the E_1 algorithm.

5) If the two SRES values are equal, the authentication process is successful in one direction. To achieve mutual authentication, this process needs to be repeated with switched roles.

3.4 Bluetooth Encryption Concept

As already mentioned before, encryption is not mandatory for all bluetooth connections and devices. Bluetooth defines three encryption modes [28]:

1) **Encryption Mode 1**
No encryption is performed at all.

2) **Encryption Mode 2**
Broadcast traffic is not encrypted. Only individually traffic is encrypted using keys based on individual link keys.

3) **Encryption Mode 3**
All traffic is encrypted using an encryption key based on the master Link Key.

Figure 12 illustrates the Bluetooth encryption procedure as implemented in BT versions 2.0 + EDR and earlier. Newer versions differ in the key derivation (cf. Section 3.2).

The key stream K_{cipher} is generated by the stream cipher function E_0, which is based on the block cipher SAFER+. This key stream is XOR'ed with the data and transmitted to the receiver. According to the symmetric cryptography paradigm, decryption is achieved by applying the same cipher key as used for encryption.

3.5 Bluetooth Trust and Service Levels

Additionally to the four security modes, Bluetooth allows two *trust levels* and three *service security levels.* Trust levels are *trusted* and *untrusted.* Trusted devices have full access to all services provided by the connected devices while untrusted devices only receive restricted access [28].

Service Security Levels allow to configure and alter the requirements for authorization, authentication and encryption independently.

Bluetooth Service Security Levels [28]:

- **Service Level 1**
Authorization and authentication are required. Trusted devices are allowed to automatically connect to all services. Untrusted devices need manual authorization for all services.

- **Service Level 2**
This level requires authentication only. Access to services is granted only after the authentication procedure.

- **Service Level 3**
Access is granted automatically and to all devices with no authentication required.

Trust and service levels allow the definition of policies to set trust relationships and may also be used to initiate user-based authentication. Bluetooth core protocols usually only provide device authentication.

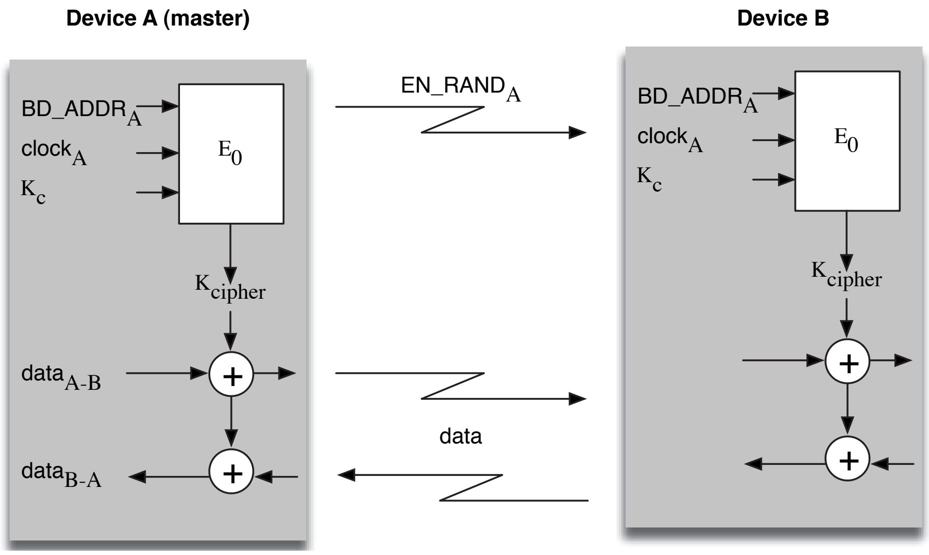


Figure 12: Functional description of the bluetooth encryption procedure [1]

3.6 Analysis of Security Measures in Bluetooth

Security matters differ very strongly between the single versions of Bluetooth. Bluetooth security always depends on the weakest BT device in the communication chain. As legacy-standard devices are still widespread this section will take their vulnerabilities in account as well as of state-of-the-art implementations. Later on, this section lists and shortly describes common Bluetooth related attacks.

3.6.1 Bluetooth Version Related Vulnerabilities

Versions before Bluetooth 1.2

- **Unit Key and Link Key Vulnerability**
  The Unit Key is reusable and becomes public after once used. This could be circumvented by using temporary broadcast keys, derived from the Unit Key which is kept secret. The same problem occurs if a corrupt or malicious device that has communicated with either device of a new communication pair, wants to eavesdrop on this communication. The Link Key stays the same for the same device. Various kinds of replay attacks are possible.

Versions before Bluetooth 2.1 This section presents vulnerabilities in Bluetooth standards prior to version 2.1 + EDR. As newer versions, namely 3.0 and 4.0, are still in the process of being standardized, no vulnerabilities have been published yet.

- **Short PIN codes are allowed**
  Short PIN codes can easily be guessed and all derived Link end Encryption keys compromised.

- **No PIN management**
  It is hardly possible to use adequate PINs in an enterprise setting as no PIN management capabilities are defined.

- **Keystream reoccurrence**
  The keystream (as created in Figure 12) repeats after 23.3 hours due to a clock overrun allowing various cryptographic attacks on the ciphertext.

Regarding All Versions

- **No User Authentication**
  By default, no user authentication is defined by BT standards. Application-level security and authentication needs to be added.

- **E$_0$ stream cipher function is weak (SAFER+)**
  The used stream cipher function SAFER+ has been subject to vulnerabilities and needs to be replaced by a more robust solution to prevent cryptographic attacks.

- **One Way Device Authentication**
  One-way challenge-response authentication can easily be exploited my man-in-the-middle (MITM) attacks. Mutual authentication should be enforced.

- **No End-to-End Encryption**
  No end-to-end encryption is provided in multi-hop scenarios. Transmissions are only encrypted between to nodes. Higher level solutions need to be deployed.

- **Limited Security Services**
  Services as nonrepudiation are not defined by BT standards. They can only be implemented in an overlay fashion.

3.6.2 Bluetooth Related Attacks

BT attacks are best classified using the following definitions [12]:

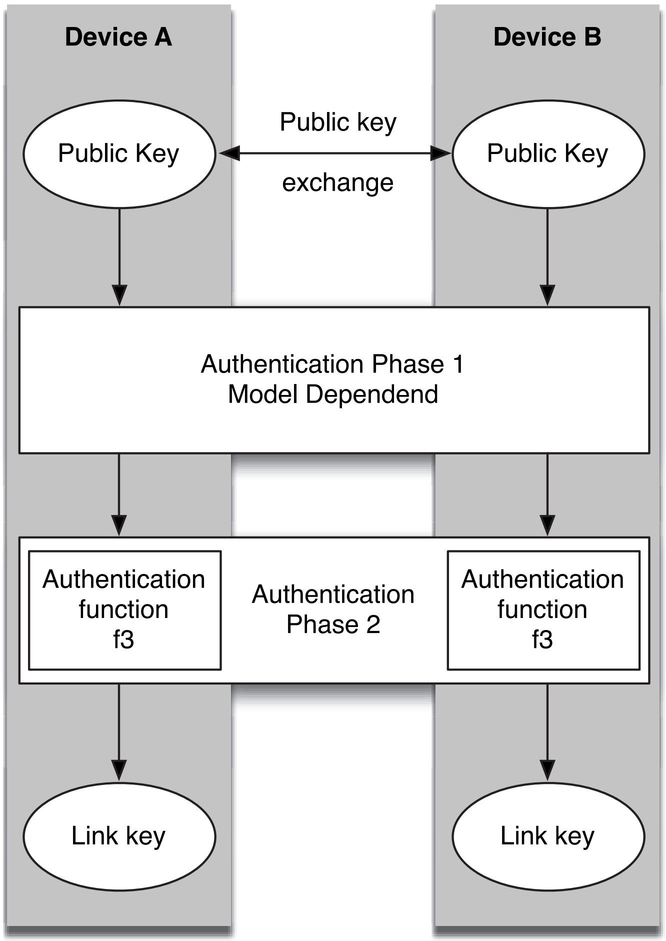


Figure 10: Overview of the bluetooth secure simple pairing routines for security Mode 4

- **Surveillance**
  Collecting information about a BT device like the provided services, device address, location and so on. No direct adverse effects to the target caused. Location tracking of users is a great potential threat.

- **Range extension**
  The range of BT devices is limited by their device class between 1 and 100 meter. Extending the transmission range of BT devices is in general against authority regulations. Attackers can use strong directional antennas to conduct BT all kinds related attacks from a great distance, even up to some kilometers.

- **Obfuscation**
  Attackers can forge their Bluetooth identities by spoofing the 48-bit device address, the device name and the device class. This can be used to obfuscate attacks.

- **Fuzzer**
  Bluetooth stack implementations are sometimes not very robust against nonstandard inputs. An attacker can create malformed data packets causing buffer-overflows or system failures at the target devices.

- **Sniffing**
  Attackers can capture all BT traffic due to its open space propagation nature in order to launch offline cryptographic attacks to recover the plaintext.

- **Denial of Service (DoS)**
  DoS attacks can target the media (i.e. channel jamming) or the devices (i.e. the energy consumption in mobile devices).

- **Malware**
  Malware is a form of malicious software that carries out various attacks as data mining or password theft on the targeted devices. This malware can be self-replicating in form of worms.

- **Unauthorized direct data access (UDDA)**
  UDDA attacks can gather all kinds of private data, and further on use all resources of the attacked device. They can i.e. place phone calls or send text messages if the attacked device provides these services.

- **Man in the middle (MITM)**
  An attacker could place himself between two communicating devices, relaying all their communication to each other. If the attacker is i.e. placed between a computer and a printer it can obtain all traffic sent to the printer. This attack mainly concerns the *Just Works* authentication method.

Concluding it has to be said, that the deployment of Bluetooth poses a serious security risk especially for enterprise settings. Even though BT can be regarded secure if all devices are configured properly, the probability of the occurrence of vulnerabilities is too high to allow its implementation in security-critical systems.

There exist some guidelines for securing Bluetooth as [28] or [12]. Further information of the security of Bluetooth can be obtained from the following references [26, 27, 29, 30].

4 Security in IEEE 802.16 (WiMAX)

Whereas WiFi and Bluetooth have been around for many years now, WiMAX is a young and emerging standard. For a better understanding of its principles, the following section will provide a short introduction.

4.1 WiMAX at a Glance

WiMAX stands for *worldwide interoperability for microwave access* and is a certification mark for the IEEE 802.16 standard family. It was designed for point-to-multipoint broadband wireless access. Its original main

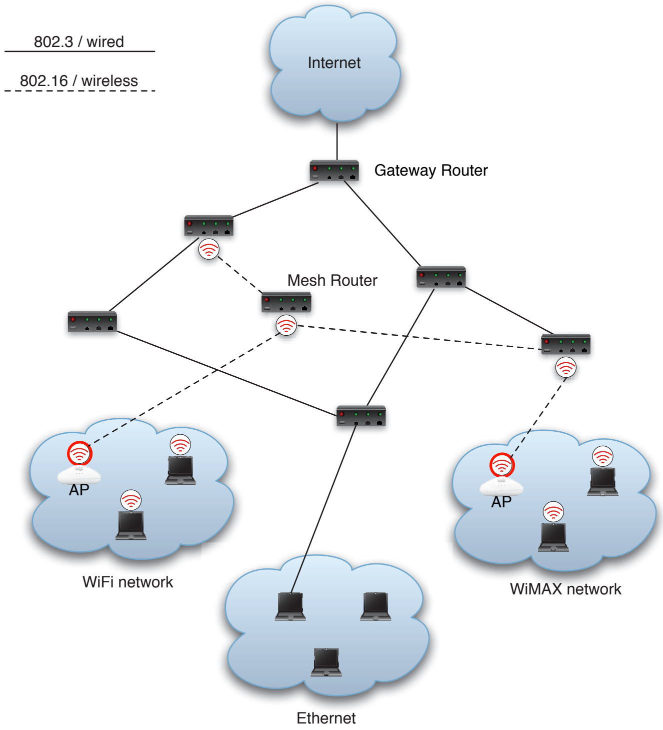


Figure 13: Possible WiMAX network setup

purpose was not to connect end-users with an access-point, but to interconnect access-points with each other. It could be seen as a kind of wireless backbone network and states an alternative to cable and DSL to provide broadband access to groups of end-users [25].

In the last years, as a response to customer and industry needs, WiMAX was extended to support connections between mobile end-nodes and base-stations.

WiMAX devices are usually organized in a mesh network (cf. Figure 13). A mesh network consist of two different kinds of nodes, which perform the necessary routing tasks: *mesh routers* and *mesh users*.

The fact that mesh users and mesh routers are able to perform the same operations and therefore may switch roles, renders mesh networks very powerful and flexible. Mesh networks are usually not limited to IEEE 802.16. They are designed to integrate other standards as IEEE 802.11 or IEEE 802.15.1 and form so called *metropolitan* and *enterprise networks*.

The most significant benefits of mesh networks are:

- **Scalability**
  The whole infrastructure is designed to be scalable as the need for resources might increase over time.

- **Ad hoc networking support**
  Devices are able to join and leave the network all the time. Routing can be self organizing.

- **Mobility support of end nodes**
  End node roaming is supported.

- **Connectivity to wired infrastructure**
  Heterogeneous networks may be interconnected by mesh routers.

The IEEE 802.16 standard uses the frequency range from 10 GHz up to 66GHz which states another significant difference to WiFi, which is using the 2.4 GHz band. WiMAX is able to cover up to 50 km of connectivity services between nodes without a direct line of sight, although the practically used distance is about 5 to 10 km. The data rate provided is up to 70 Mb/s which is enough to serve about 60 T-1-type links simultaneously [25].

Probably the most significant differences between WiMAX and WiFi standards may be found at the MAC layer. WiMAX offers a remarkable improvement as it defines a MAC layer that supports multiple physical-layer specifications. This renders WiMAX as a great framework for wireless broadband communications.

The MAC layer is a so called *scheduling* MAC layer where devices need to compete for the initial entrance to the network. Once joined the network, the base station dedicates a time slot to the device which can be variable but must not be used by any other user. This method offers better bandwidth efficiency and allows the base station to offer QoS by balancing the assignments of connected devices [25].

Some of the IEEE 802.16 MAC layer properties to support mesh networking are:

- It is designed to support multi-hop communication.

- It is designed for multipoint-to-multipoint communication.

- Self-organizing features are provided.

WiMAX was initially released as IEEE 802.16-2001 in April 2002 [2]. After some amendments, IEEE 802.16-2004, also known as IEEE 802.16d [4], was released and fixed many errors and initial security vulnerabilities. In 2005, IEEE 802.16e-2005 [5] was released, enabling mobility support in WiMAX networks and fixing further security issues. IEEE 802.16j [6] is the latest major release in this standard family. It mainly extends mobile support and does mot introduce new security functionality.

4.2 Overview of IEEE 802.16 Security

Lessons learned from weaknesses in WiFi security have been incorporated in WiMAX right from the beginning of its design. WiMAX provides right out-of-the-box the following security services [8]:

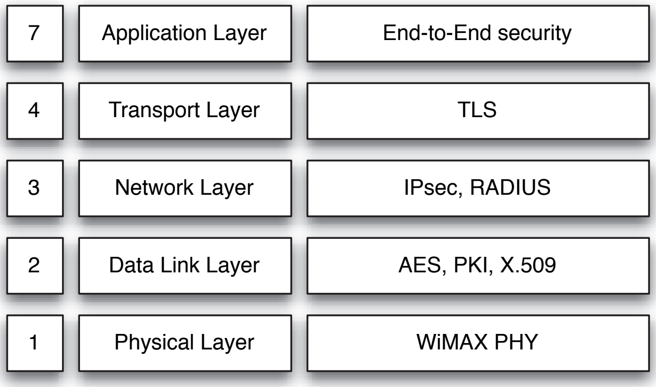
- Privacy - Protect from eavesdropping;

- Data integrity - Protect data from being tampered in transit;

- Authentication - At the user and the device level;

- Authorization - At the service level.

As Figure 14 illustrates, IEEE 802.16 allows the incorporation of security functions at various network layers [8]:



7	Application Layer	End-to-End security
4	Transport Layer	TLS
3	Network Layer	IPsec, RADIUS
2	Data Link Layer	AES, PKI, X.509
1	Physical Layer	WiMAX PHY

Figure 14: WiMAX supported security functions at various network layers

Right from the beginning of the WiMAX design process, a special layer, as part of the MAC layer has been introduced. The so called *security sublayer* should provide all necessary security functionality, securing all communication on the higher layers (cf. Figure 15).

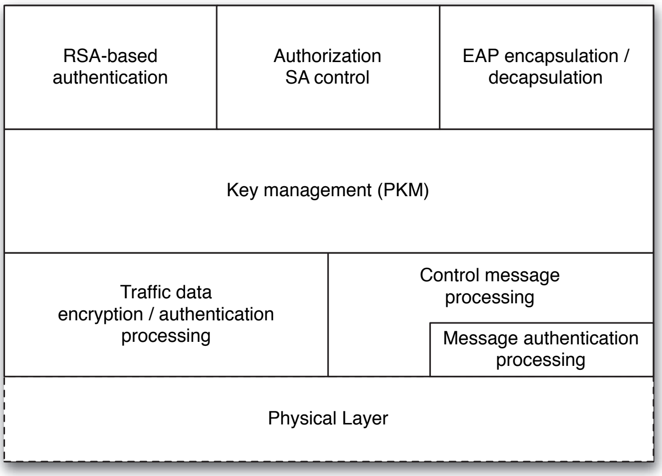

RSA-based authentication
Authorization SA control
EAP encapsulation / decapsulation
Key management (PKM)
Traffic data encryption / authentication processing
Control message processing
Message authentication processing
Physical Layer

Figure 15: WiMAX security sublayer

As this chapter is about security in wireless networks, it will focus on security measures which are part of the IEEE 802.16 security sublayer.

4.2.1 Authentication and Authorization in WiMAX

Authentication and Authorization in WiMAX is completely implemented at the security sublayer. It is achieved using a *public key interchange protocol* that ensures authentication and establishment of the cryptographic keys. A key pair, consisting of a private and a public key is needed for each party in the public key interchange scheme.

Key interchange and key management in general had several vulnerabilities in the original IEEE 802.16 standard. As IEEE 802.16e-2005 corrected most of these problems, this section will focus on this state-of-the-art standard.

IEEE 802.16e-2005 defines two *Privacy Key Management* (PKM) protocols, PKMv1 and an enhanced version PKMv2. They basically allow three types of authentication (cf. Figure 15):

- RSA based authentication - based on X.509 certificates and RSA encryption;
- Extensible Authentication Protocol (EAP);
- RSA based authentication followed by EAP authentication.

All security information between communicating parties are part of so called *Security Associations* (SA). SAs are a set of parameters used for authentication, authorization and encryption. The shared information depends on the chosen cryptographic suite and usually includes the encryption keys and *initialization vectors* (IV) needed for the encryption process. Three different types of SAs are defined by IEEE 802.16e-2005 [5]:

- **Primary SA**
  Each SS establishes a primary SA during its initialization process.
- **Statics SA**
  They are provisioned within each BS.
- **Dynamic SA**
  They are established and eliminated, on the fly, in response to the initiation and termination of the specific service flows.

Each SS establishes an exclusive Primary SA with its BS and dynamic SAs for each new service flow. The lifetime of SAs is limited by the standard. Each new SA has to be newly authorized before its establishment.

The PKM establishes a shared key called *Authorization Key* (AK) between the subscriber (SS) and the base station (BS). After this shared AK is established between the parties, a *Key Encryption Key* (KEK) is derived from it. This KEK is then used to encrypt subsequent PKM exchanges of *Traffic Encryption Keys* (TEK). All payload encryption is based on TEKs.

Table 3 provides an excerpt of the cryptographic keys used in WiMAX.

Figure 16 illustrates the authentication and authorization protocol as originally integrated in IEEE 802.16-2001.

The SS uses the first message to push its manufacturer X.509 certificate to the BS allowing it to validate its identity via a Certification Authority (CA). The second message is send right after the first and includes the SS's X.509 certificate its security capabilities and the ID

Table 3: Overview of cryptographic keys used in WiMAX (excerpt)

Key Name	Description	Derived from
AK Authorization Key	Shared private key (between SS and BS)	not clearly defined by the standard
KEK Key Encryption Key	Key used for encrypting TEKs in the key exchange	derived from the AK
TEK Traffic Encryption Key	Used for encrypting all end to end traffic	derived from the AK
PK Public Key	public key of the BS and the SS respectively	stored in the X.509 certificate of the BS and SS respectively

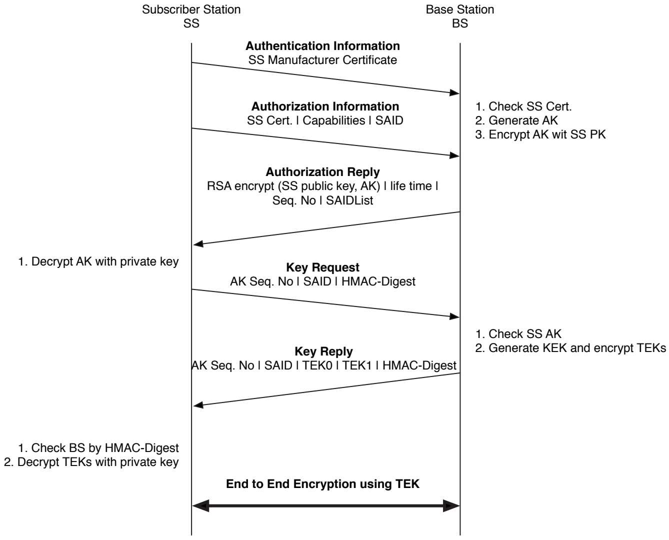

Subscriber Station
SS
Base Station
BS
Authentication Information
SS Manufacturer Certificate
Authorization Information
SS Cert. I Capabilities I SAID
1. Check SS Cert.
2. Generate AK
3. Encrypt AK wit SS PK
Authorization Reply
RSA encrypt (SS public key, AK) I life time I
Seq. No I SAIDList
1. Decrypt AK with private key
Key Request
AK Seq. No I SAID I HMAC-Digest
1. Check SS AK
2. Generate KEK and encrypt TEKs
Key Reply
AK Seq. No I SAID I TEK0 I TEK1 I HMAC-Digest
1. Check BS by HMAC-Digest
2. Decrypt TEKs with private key
End to End Encryption using TEK

Figure 16: WiMAX privacy key management protocol (PKM) v1

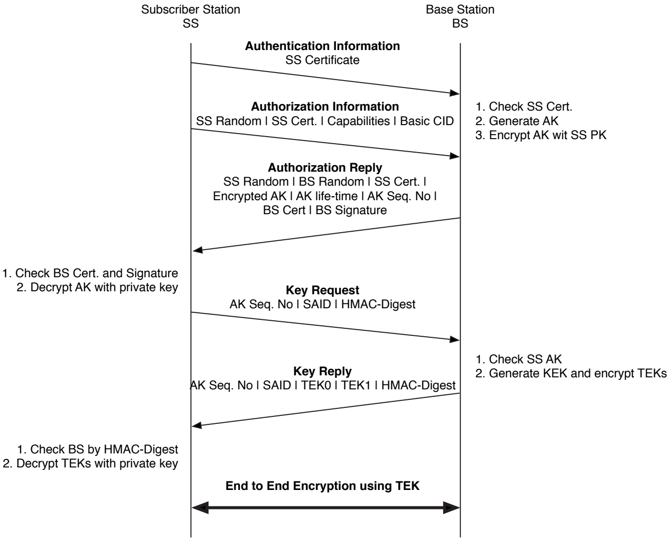

Subscriber Station
SS
Base Station
BS
Authentication Information
SS Certificate
Authorization Information
SS Random I SS Cert. I Capabilities I Basic CID
1. Check SS Cert.
2. Generate AK
3. Encrypt AK wit SS PK
Authorization Reply
SS Random I BS Random I SS Cert. I
Encrypted AK I AK life-time I AK Seq. No I
BS Cert I BS Signature
1. Check BS Cert. and Signature
2. Decrypt AK with private key
Key Request
AK Seq. No I SAID I HMAC-Digest
1. Check SS AK
2. Generate KEK and encrypt TEKs
Key Reply
AK Seq. No I SAID I TEK0 I TEK1 I HMAC-Digest
1. Check BS by HMAC-Digest
2. Decrypt TEKs with private key
End to End Encryption using TEK

Figure 17: WiMAX privacy key management protocol (PKM) v2 [7]

of the Primary Security Association (SAID). By using the SS certificates public key (PK), the BS is able to construct the Authorization Reply including the Authorization Key (AK). The following messages are to establish the keys needed for encryption [8].

PKMv1 lacks mutual authentication as only the SS provides a certificate. Problems arising due to this fact are discussed in the security analysis of WiMAX later in this chapter.

IEEE 802.16e-2005 introduced an improved version of the Privacy Key Management Protocol called PKMv2, targeted to provide mutual authentication based on X.509 certificates and to correct the vulnerabilities of PKMv1. As illustrated in Figure 17, the *Authorization Reply* is extended by the BS's certificate an digital signature and random seeds from the SS and BS respectively. These additional parameters aim to harden the protocol against replay and man-in-the-middle-attacks [19].

PKMv2 also allows the usage of *Cipher based Message Authentication Codes* (CMAC) instead oh *Hashed Message Authentication Codes* (HMAC) [23].

Additionally to RSA based authentication, WiMAX allows the use of the *Extensible Authentication Protocol* (EAP). The EAP method can use a particular kind of credential, such as an X.509 certificate in the case of EAP-TLS or a *Subscriber Identity Module* (SIM card) in the case of EAP-SIM [5].

The definition of the EAP protocol is outside of the WiMAX standard and can be obtained from RFC 4017 [9].

4.2.2 Encryption in WiMAX

The initial standard defined encryption based on the *Data Encryption Standard* (DES) with a default key length of 56 bit. Figure 18 illustrates the encryption process of IEEE 802.16-2001.

DES is operated in *Cipher Block Chaining* (CBC) mode using the TEK as encryption key, an *initialization vector* derived from the SA's IV and the value of a field in the PHY header. Both of these last named values are predictable.

IEEE 802.16e-2005 introduced the usage of the *Advanced Encryption Standard* (AES) in *Counter mode with*

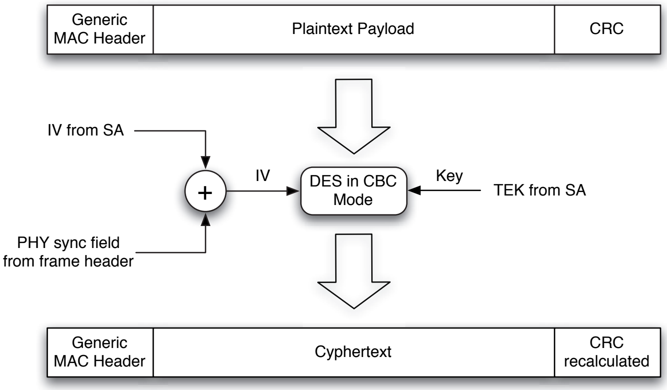


Figure 18: IEEE 802.16-2001 encryption process [19]

CBC-Message Authentication Code (CCM) mode for authentication and AES in *Counter* mode (CTR) for encryption purposes (cf. Figure 19).

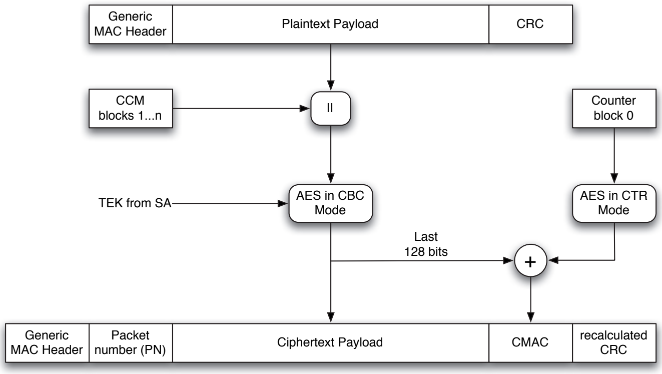


Figure 19: IEEE 802.16e-2005 encryption process based on AES [23]

AES-CCM and AES-CTR are slightly slower in their operation than 3DES but the security increase is significant.

4.3 Analysis of IEEE 802.16 Security

As mentioned before, WiMAX was originally developed to address the *last mile* problem. The IEEE 802.16 Working Group tried to avoid design mistakes like done by defining WiFi standards by incorporating a pre-existing standard, *Data Over Cable Service Interface Specifications* (DOCSIS). DOCSIS was designed to solve the last mile problem for wired connections. This fact allows the assumption, that it might not work in wireless networks without problems. The result was, that IEEE 802.16-2001 failed to properly protect the wireless links [19].

The major security flaws of the initial standard are the following [19]:

- Only data transport is encrypted, leaving management frames vulnerable for attacks.
- The focus on the encryption of the packet payload left the authorization protocol neglected and thus vulnerable.
- The standard allowed one-way authentication leaving many loop-holes for replay attacks.
- Several security related parts of the standard as key generation, lacked explicit definitions and could therefore be implemented imperfect by hardware vendors.
- Tripple DES (3DES) with a key length of 56 bit was used in CBC mode. While DES itself is not unbreakable anymore, very short keys as used in IEEE 802.16-2001 are a serious vulnerability. Further on, the encryption process (cf. Figure 18) exhibits a severe error by using predictable initialization vectors (IV). CBC mode would require a random IV to secure the scheme [22].
- Vulnerabilities introduced by the weak encryption scheme and lacking mutual authentication allow several attacks on the privacy and integrity of the communications. It furthermore leaves the topology of the network exposed to mesh-network attacks. The interested reader is referred to [10, 19, 24, 34, 36].

IEEE 802.16e-2005 corrects these errors described above by incorporating the following mechanisms:

- Encryption of management frames;
- Improving the authentication protocol by introducing PKMv2;
- Implementing mutual, PKI based authentication;
- Rendering definitions on key generation more precise;
- Replacing DES-CBC with AES-CBC;
- Introducing AES-CCM for message authentication.

As mentioned before, IEEE 802.16e-2005 is still a young standard and currently a lot of security related research is conducted around it. As history has shown with related wireless networks, this research will uncover further vulnerabilities and design flaws.

5 Conclusion

This article provides a detailed overview of security mechanisms implemented in Bluetooth, WiFi and WiMAX. It discusses authentication, key-agreement and cryptographic concepts and their security features and flaws.

Concluding this survey, we can state, that recent developments in wireless network security are pointing in the right direction. Standards become more and more robust and secure allowing the implementation of critical applications based wireless technologies. The standard bodies seem to have recognized the need for high quality security design in the early stages of standard development, avoiding to repeat mistakes of the past. Future releases will show if these measurements are effective.

Acknowledgments

This study was supported by the National Science Council of Taiwan under grant NSC 95-2416-H-159-003. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] "IEEE standards for information technology telecommunications and information exchange between systems local and metropolitan area network specific requirements part 11: Wireless lan medium access control,". IEEE 802.11 1999 Edition ISO/IEC 8802-11: 1999, IEEE, 1999.

[2] "IEEE standard for local and metropolitan area networks part 16: Air interface for fixed broadband wireless access systems,". Tech. Rep. 802.16-2001, IEEE, 2001.

[3] "IEEE standard for information technology telecommunications and information exchange between system local and metropolitan area networks specific requirements part 11: Wireless LAN,". tech. rep., IEEE, 2004.

[4] "IEEE standard for local and metropolitan area networks part 16: Air interface for fixed broadband wireless access systems,". Tech. Rep. 802.16-2004, IEEE, 2004.

[5] "IEEE standard for local and metropolitan area networks part 16: Air interface for fixed and mobile broadband wireless access systems amendment 2: Physical andmedium access control layers for combined fixed,". tech. rep., IEEE, 2005.

[6] "IEEE standard for local and metropolitan area networks part 16: Air interface for broadband wireless access systems amendment 1: Multiple relay specification,". tech. rep., IEEE, 2009.

[7] S. Adibi, B. Lin, P.H. Ho, G. Agnew, and S. Erfani, "Authentication authorization and accounting (AAA) schemes in wimax," in *2006 IEEE International Conference on Electro/Information Technology*, pp. 210–215, May 2006.

[8] Airspan. "Mobile wimax security,". tech. rep., 2007.

[9] D. Stanley andJ. Walker and B. Aboba. "Extensible authentication protocol (EAP) method requirements for wireless lan,". Tech. Rep. RFC 4017.

[10] M. Barbeau, "Wimax/802.16 threat analysis," in *International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, p. 8, 2005.

[11] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: The insecurity of 802.11," in *International Conference on Mobile Computing and Networking*, pp. 180, 2001.

[12] J. Dunning, "Taming the blue beast: A survey of bluetooth based threats," *IEEE Security & Privacy Magazine*, vol. 8, no. 2, pp. 20–27, 2010.

[13] S. Zafar E. Sithirasenan and V. Muthukkumarasamy, "Formal verification of the IEEE 802.11i wlan security protocol," in *Australian Software Engineering Conference (ASWEC '06)*, pp. 181–190, 2006.

[14] N. Ferguson. "Michael: An improved MIC for 802.11 WEP,". tech. rep., IEEE, 2002.

[15] H. Hassan and Y. Challal, "Enhanced WEP: An efficient solution to WEP threats," in *Second IFIP International Conference on Wireless and Optical Communications Networks*, pp. 594–599, 2005.

[16] C. He and J. C. Mitchell, "Security analysis and improvements for IEEE 802.11i," in *Proceedings of the 12th Annual Network and Distributed System Security Symposium*, pp. 90–110, 2005.

[17] J. Huang, W. Susilo, and J. Seberry. "Observations on the message integrity code in IEEE802.11 wireless LANs,". tech. rep., 2008.

[18] J. Huang, W. Susilo, J. Seberry, and M. Bunder. "On the security of the IEEE 802.11i message integrity code michael,". tech. rep., 2004.

[19] D. Johnston and J. Walker, "Overview of IEEE 802.16 security," *IEEE Security & Privacy Magazine*, vol. 2, no. 3, pp. 40–48, 2004.

[20] P. Kitsos, N. Sklavos, K. Papadomanolakis, and O. Koufopavlou, "Hardware implementation of bluetooth security," *IEEE Pervasive Computing*, vol. 2, no. 1, p. 21, 2003.

[21] A. Klein, "Attacks on the rc4 stream cipher," *Designs, Codes and Cryptography*, vol. 48, no. 3, p. 269, 2008.

[22] R. Laboratories. "Pkcs #1: RSA cryptography standard,". tech. rep., RSA Laboratories.

[23] C. Luo, "A simple encryption scheme based on wimax," in *2009 International Conference on E-Business and Information System Security*, pp. 1–4, May 2009.

[24] L. Maccari, M. Paoli, and R. Fantacci, "Security analysis of IEEE 802.16," in *2007 IEEE International Conference on Communications*, pp. 1160–1165, Glasgow, June 2007.

[25] N. F. Mir, *Computer and Communication Networks.* Prentice Hall, 2006.

[26] S. Pasanen, "New efficient rf fingerprint-based security solution for bluetooth secure simple pairing," *Security*, pp. 1–8, 2010.

[27] N. Sastry and D. Wagner, "Security considerations for IEEE 802.15.4 networks," in *Proceedings of the 2004 ACM Workshop on Wireless Security*, p. 32, New York, USA, 2004.

[28] K. Scarfone and J. Padgette. "Guide to bluetooth security,". Tech. Rep., 2008.

[29] Y. Shaked and A. Wool, "Cracking the bluetooth pin," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, pp. 39–50, 2005.

[30] D. Singelee. *Study and Design of a Security Architecture for Wireless Personal Area Networks*. PhD thesis, KU Leuven, 2008.

[31] E. Tews, R. P. Weinmann, and A. Pyshkin, "Breaking 104 bit WEP in less than 60 seconds," in *Proceedings of the 8th International Conference on Information Security Applications*, pp. 188–202, 2007.

[32] W. Trappe and L. C. Washington, *Introduction to Cryptography with Coding Theory (2nd Edition)*. Prentice Hall, 2005.

[33] A. Wool, "A note on the fragility of the michael message integrity code," *IEEE Transactions on Wireless Communications*, vol. 3, no. 5, pp. 1459–1462, 2004.

[34] S. Xu and C. T. Huang, "Attacks on pkm protocols of IEEE 802.16 and its later versions," in *The 3rd International Symposium on Wireless Communication Systems*, pp. 185–189, Sep. 2006.

[35] T. C. Yeh, J. R. Peng, S. S. Wang, and J. P. Hsu, "Securing bluetooth communications," *International Journal of Network Security*, vol. 14, no. 4, pp. 229–235, 2012.

[36] Y. Zhou and Y. Fang, "Security of IEEE 802.16 in mesh mode," in *MILCOM 2006*, pp. 1–6, Oct. 2006.

Günther Lackner is currently working on his Ph.D in the area of security and privacy aspects in wireless networks with Prof. Vincent Rijmen. He received his B.Sc and M.Sc degrees in Telematics, supervised by Prof. Reinhard Posch, at the Univer- sity of Technology Graz, Austria. He collaborated in several network security-related projects during the last years as a member of the Network Security Group at the Institute for Applied Information Processing and Communications (IAIK) at the University of Technology Graz. He is currently as a visiting researcher at the Information Security Institute of the Queensland University of Technology..