CHAPTER ONE

INTRODUCTION

1.1 Preview

This research study TV White Space (TVWS) database authentication and key management protocol. This introduction is presented into three parts, part one give a background about the TVWS and the security issue about it. The second part presents some TVWS authentication's protocols. And lastly discussed the key management in term of key generations and distributions

1.1.1 TV White Space

A well-known issue in modern wireless communications is spectrum scarcity. To solve the dilemma between the increasing bandwidth demands and the actual underutilization of spectrum resource [1], the Federal Communications Commission [2] has allowed unlicensed users to opportunistically access the temporarily unoccupied television (TV) bands, namely TV white spaces (TVWSs) [3] on the basis of noninterference of the licensed users.

In particular, the switchover to digital television frees up large areas between about 50 MHz and 700 MHz which called TV white space. This is because digital transmissions could be compressed in packages into adjacent channels, while analog ones cannot. This means that the band can be "compressed" into fewer channels, while still allowing for more transmissions. This spectrum is located between existing TV stations and is called TV White Space. This new spectrum provides the ideal platform for longer-range, but still local, wireless broadband services and will be used for bridging gaps between wired cable, and fiber connections and locations that cannot be economically served by either wide-area or Wi-Fi systems [4].

In other words, TV White Space is ideally suited to fill the gap between wide-area and local-area systems. Today, the rules to manage this spectrum are finished in some countries. The FCC and other rules working to make sure these new networks and devices do not interfere with TV receivers in homes and commercial establishments. These rules specify how much spectrum needs to be available for TV White Space operations [1]. One of the most important problems was appear is how to manage the new TV White Space devices to ensure they are operating on the correct portion of the spectrum and not on channels occupied by or close to existing TV stations. This is important because each area of the nation has TV stations licensed on different channels, so the available TV White Space and the part of the spectrum is in differs from area to area [1].

Those leading the researchers and trial systems in TV White Space are divided about how best to accomplish the spectrum management portion of the system. Some researchers believe that each TV White Space device deployed should contain a computer that will search the spectrum and determine the best channel on which to operate. The downside of this approach is that it will add to the cost of the devices, and the devices will still have to communicate among themselves in order to work together.

Another alternative, TVWS unlicensed can use cognitive radio (CR) techniques for sharing the spectrum. These techniques are similar to Wi-Fi techniques but the difference is CR is cover wide area, but still lack of coordinator and centralized device to avoid the interference.

The other approach is to develop a database of channels available in every area, and have each TV White Space device contact the database, provide its location, and be assigned spectrum that is available in that area. The advantage to this approach is that TV White Space devices are simpler, do not make erroneous decisions, and can be built without the expensive logic required to track its location, resulting in devices estimated to cost about the same as today’s Wi-Fi access points.

These database technologies are being tested in trials across many countries and the results are quite positive [3]. These trials demonstrate the viability of the central database approach for TV White Space use and provide real-world experience with this new and important way of allocating spectrum in a dynamic, real-time manner. This has implications for other wireless spectrum in use today. For unlicensed spectrum such as Wi-Fi, there is increasing interference because there is no coordination among access points [4]. Database-driven management of that spectrum could solve this problem. Today’s licensed spectrum is statically allocated, and there are times when some spectrum is lightly used while some is overloaded. Again, though not required, database-driven management could dynamically allocate the spectrum for more efficient utilization.

Many countries allow TV white spaces (TVWS) to be used by unlicensed devices. TV White Spaces is considered as an important step towards providing broadband access to millions of digital dividend household around the world and enabling a wide range of innovative wireless devices and services. TVWS coexistence should be performed for peaceful working with incumbent users in TV bands as well as other TVWS license-exempt technologies [5]. To utilize the TVWS channels the users can either use cognitive radio and/or database. The TV White Space Database (TVWSDB) is a database of authorized services in the TV frequency bands that is used to determine the available channels at a given location for use by White Space devices (WSD).

In general, the lower frequency band used, the more distance can be achieved using the same power levels, so TV White Space is ideally suited for city and town-wide systems where existing Wi-Fi spectrum cannot be effectively used. This is because the TV White Space spectrum is lower in frequency than the existing unlicensed bands used by Wi-Fi service (2.4 GHz and 5 GHz). The other property of the lower frequencies of TV White Space as compared to Wi-Fi is that it is significantly better at penetrating foliage, buildings, and other obstructions.

Thus the number of wireless data subscribers and the amount of data used per subscriber is set to significantly grow over the coming years [1]. The worldwide total addressable market for Intelligent Spectrum Management as encompassed by TV White Space and Database Networking is projected by our own estimates in addition to research from ABI Research, In-Stat, and Spectrum Bridge, to be more than $4 billion in annual expenditures with more than 280 million units shipped by 2015 [1]. The growth in TV White Space devices was occurring first for high-power devices deployed in fixed locations. This is because a natural use of the TV White Space spectrum is to bring broadband Internet access to locations where conventional service is costly and difficult. These devices will work with existing Wi-Fi and other networks to complete the connection to the user’s device.

Starting around 2013, the low-power devices making up the first volume shipments [2]. The growth of this market will be slower initially. As the value of dynamically database-managed spectrum via Intelligent Spectrum Management is proven, low-power device numbers should dramatically increase by the end of the decade.

1.1.2 Sensing vs. Database

Whatever methods used the most important issues must ensure that the devices will not cause interference to TV receivers. One approach to this problem is to add sensing circuits and logic into every low-power device [5]. In this way, the device would be smart enough to sense the presence of TV stations and avoid channels that are in use in any given area. The other approach is to use the master databases that track acceptable TV White Space spectrum for a given location [6]. This database is accessed by the master device on the network and, in turn, the lower-powered slave units are directed away from interfering channels.

If the use of sensing circuits is required in each device in addition to database access, the cost of TV White Space devices will be considerably higher than if the database approach is used exclusively, and the device will be prone to false positive detection. Building both smart sensing capabilities and database capabilities into each and every device does not appear to be the best approach. If this method is required, all devices will cost more and be less reliable, which could slow or kill the adoption of TV White Space systems [1]. So the database approach is a much better solution to ensure that TV White Space devices do not cause interference. Further, devices that rely exclusively on the database solution can be built at substantially less cost than if each device is required to be “smart” as well.

1.1.3 TVWHITE SPACE SECURITY ISSUES

As any applications that use the internet there must be confidentiality between the users and the service provider, so that to convince the users about their security information and data they are exchange while they are using this application. Because the attacks on web applications are one of crucial problems against our everyday lives that depend more and more on the world wide web (www). TV white space may encounter authentication problem for information exchange between licensed WSDB and TVWS users. Also key management (generation and distribution) is one of the security concerns in the TVWSDB.

1.1.4 TVWHITE SPACE DATABASE TERMINOLOGIES

According to FCC part 15.700, WSDs can have one of three modes of operations:

1.1.4.1Master/Fixed Mode II: An operating mode in which the WSD has the capability to transmit without receiving an enabling signal. The WSD is able to select a channel itself based on a list provided by the database and initiate a network by sending enabling signals to other devices. Fixed mode WSDs are usually working with maximum transmission power 4W [5]. According to FCC rules, fixed WSD needs to access the WSDB at least once a day. Figure 1 depicts the master/fixed mode WSDB access.

1.1.4.2 Mode I Operation: It is also called sensing only mode, it is an operation of a personal/portable WSD operating only on the available channel identified by either the fixed WSD or Mode II WSD that enables its operation. Mode I operation does not require use of a geo-location capability or access to the TV bands database and requires operation in client mode. Mode I WSDs are usually working with maximum transmission power 50 to 100m [5].

1.1.5 Research Motivation

The research [1] confirms this coming shortfall in wireless network capacity: The wide-area networks will not be able to meet the demand of the coming years [1]. It is, of course, far easier to build millions of new devices that consume bandwidth than to expand the networks to meet that increased demand.

1.1.6 The Scope

This research study the mutual authentication process in IEEE protocols and Process to Access White Space Protocol (PAWS) and the key management in terms of key generation and key management. And design a new confident and mutual authentication protocol which provides new methods of key generation and key management.

1.2 The problem statement

The transformation from analog transmission to digital transmission in the TV makes free frequencies called TV White Space (TVWS), which can reuse these frequencies in transmission of the data. This can be done, either by using cognitive radio (CR), sensing and transmit or make a user’s database. In both cases CR or geolocation database there must be a protocol to organize the channels utilization. When the network growing the security becomes highly demand specially the authentication and key management to prevent unauthorized user's misbehaving and to convince the users that their transmission data is save.

Now days in the available protocols, the authentication is either from a network point of view or from a database perspective and some protocols (such as PAW) apply the authentication in two layers.

PAWS pretend to specify both a database identification mechanism (how can a device know what database it has to connect to) and contents of the queries and responses (XML is an option). This protocol did not state any type of authentication procedure but just state that "This messaging between the device and the database needs to be secure (authentication, integrity of the content, prevent from man-in-the-middle attacks etc.), requiring some authentication and security measures".

The PAWS protocol depend on tow layer authentication (HTTP/TLS) and this will become more complex and overhead.

After the authentication process is complete the key management will start to complete the authentication protocol. The main drawbacks in the available key management process are, the master/DS and the user exchange some data that the attackers might access and use it to obtain the key, and even to generate new keys when the key life time is expired. Also, the process of generating the key is inefficient in time consuming.

This study presents a new confident and mutual authentication protocol for TVWS which can utilize the availability of the database security and the network link security together to be in one protocol. So that the users (Mode I) devices can use one secure protocol to authenticate themselves to the Database server, and generate and exchange shared secrete key. And according to the small capabilities of the TVWS deceives, this study present a new method of key generation and key distribution to overcome the key management problems.

1.3 Aims and Objectives

This study aims to enhance IEEE 802.22 protocol’s authentication and key management for TVWS database security to attain these objectives:

- Attain Base station and client mutual authentication
- Introduce more secure protocol.
- Reduce time in key generation and exchange.
- New method for key management which include:
  - Avoid sending any information has a relation with the key generation.
  - High speed calculation of key generation.
  - The attackers can’t generate a new key even if they could know the available key.
  - Save the power consumption.
  - Easy to implement
  - Do not need any hardware modification to implement.

1.4 Literature Review

The authentication can be maintained by the authentication process by passing user’s information to/from WSDB providers under appropriate security protocol, TV white space device (WSD) and database (DB) connection protocol is used to register and sending information's between the master/mode II devices and DB. In general Hypertext Transfer Protocol (HTTP and HTTPS) protocol **[9]** are used, and can provide secure connection, but still WSDs authentication and confidentiality is considered a problematic issue. This research present and analyzed some authentication protocols as follow:

1.4.1 Security mechanism in IEEE 802.22

Security features defined in IEEE 802.22 provide protection for the users, service providers and most importantly, the incumbents, who are the primary users of the spectrum. As a result, the protection mechanisms in IEEE 802.22 are divided into two security sublayers that target non-cognitive as well as cognitive functionality of the system and the interactions between the two. IEEE 802.22 does not discuss the methods to protect the access to the IEEE 802.22 system. [10]

The security sublayer 1 provides subscribers with authentication, or confidentiality for user’s data and MAC management messages transmitted across the broadband wireless network. It does this by applying cryptographic transforms to MAC PDUs carried across connections between CPE and BS. The security sublayers employ an authenticated client/server key management protocol in which the BS operator controls distribution of keying material to client customer premise equipment (CPE). This material is used to protect MAC management messages, and may be optionally used to protect user’s data. The basic security mechanisms are strengthened by adding EAP-based CPE device-authentication to the key management protocol.

To enhance the security for the cognitive functionality in IEEE 802.22, security sublayer 2 is introduced. These security mechanisms validate the availability of spectrum for the primary and the secondary users by employing mechanisms such as distributed sensing and decision making. This includes authentication of the incumbent sensing information to avoid Denial of Service (DoS) attacks, authentication of the IEEE 802.22.1 beacon frame utilizing the security features that are already embedded in it, authentication of the geolocation and co-existence information, etc. Some cognitive plane security related mechanisms are integral part of other cognitive functions required for the system implementation such as Spectrum Sensing Function, geolocation, spectrum manager, Spectrum Automaton, Management Plane procedures and functions etc.

1.4.2 Protocol Access White Space (PAWS)

Because the Protocol Access White Space (PAWS) did not restrict the authentication, this protocol uses HTTP protocol **[7]**. HTTP is usually used as one of the good secure protocols in internet application and transactions. However this protocol is not suitable for TVWS authentication and spectrum management applications, since the connection between the WSDB and users are inconvenient for the users to submit his\her username and password in every HTTP session. As for the current protection schemes, the HTTP has two user authentication methods, basic and digest, that respectively pass a plain and a hashed username-password pairs to the server, allowing optional server authentication after the user authenticated. These methods, however, have a few disadvantages: First, a TVWS attacker can steal the username and password pair since the server sends one of the plain names, “Basic” and “Digest”, of the schemes. Then the attacker can intercept the response to change the scheme’s name to “Basic” to trick the browser into using the Basic scheme **[9].** Second, server authentication is performed after user authentication that needs the username-password pair registered to the server, which is not recommended by spectrum management regulators. So we can not authenticate TV WSDB servers as in other conventional WSDB servers, such as e-shopping. To cope with the HTTP problems above, servers usually start the Secure Socket Layer (SSL) [11] protocol (or its standardized version, Transport Layer Security (TLS) protocol; the HTTP with the SSL protocol is called HTTPS. The SSL supports server authentication and optional user authentication. However, the server usually authenticates the user by the username-password pair otherwise the users have to register themselves to Certificate Authorities (CAs). On the other hand, the server authentication is not so convenient to be used due to complex manipulations on the records in CAs; in establishing a secure channel of SSL, the browser confirms the server’s domain-name certificate signed by the CA, next chooses a random key shared with the server and used to protect the confidentiality and integrity of requests and responses.

1.4.3 EAP-based authentication Protocol Framework

EAP offers the operator to select an EAP Method (e.g., EAP-TLS; RFC 2716) to execute the authentication. Each EAP Method specifies a credential that is used to perform authentication and verify the device’s/user’s identity. For example, EAP-TLS uses a X.509 certificate, while EAP-SIM uses a Subscriber Identity Module. EAP-TLS or EAP-TTLS shall be used to define the profile for the X.509 credential [12].

During initial authentication EAP transfer messages are not protected. For reauthentication, the EAP transfer messages are protected (encrypted and authenticated) using the management message protection key (MMP_Key). If EAP reauthentication messages fail their authentication verification or are not protected, they shall be ignored by the BS and CPE.

The authentication, authorization, and accounting (AAA) server and a client CPE authenticate each other during the initial authentication exchange. The AAA and CPE present their credentials to each other. Since the AAA and CPE mutually authenticate each other, there is protection against an attacker employing a cloned CPE that masquerades as a legitimate subscriber’s CPE. Once authentication is completed, the BS and CPE have a key that is used to protect management messages (e.g., MMP_Key) and keying used in transportation of keys for protection of user data (e.g., Key Encryption Key (KEK). During authentication exchange, if a CPE indicates that it does not support protection of user data, no key exchange and state machines used to maintain keying to protect user data will be executed.

1.4.4 Key Management and Authentication

The security control management (**SCM)** protocol allows for mutual authentication where both the network and CPEs authenticate each other [10]. It also supports periodic reauthentication and key refresh. It uses strong encryption algorithms to perform key exchanges between a CPE and BS.

The SCM’s authentication protocol establishes a shared secret (i.e., the authorization key AK) between the CPE and the BS. The shared secret is then used to secure subsequent SCM exchanges of traffic encryption key (TEKs). This two-tiered mechanism for key distribution permits refreshing of TEKs without incurring the overhead of computation-intensive operations.

The key distribution and management protocols which are used to establish secure communication between two principals, and authentication protocols which verify that the communicating principle is who it is supposed to be are one of the main issues that the applications of formal methods in the analysis of cryptographic protocols have been mainly concerned with [13]. The tools that have been constructed based on the theoretical developments have successfully located subtle bugs in many cases, even in protocols that have been considered secure for several years. One of the most famous success stories is the Lowe's attack [14, 15] on the Needham Schroeder public key protocol [16] using the process algebra Communicating Sequential Processes (CSP) and the Failures-Divergences Refinement (FDR) which is the model checker for CSP [17]. Also, Shmatikov and Stern [18] used Murphi, and Corin et al. [9] used symbolic traces and Pure-past Security - Linear Temporal Logic (PS-LTL) successfully.

1.5 Motivation for New Protocol

There are three reasons explain why the TVWS needs a new protocol, the first reason is the available protocols does not have mechanisms to protect the incumbent channels. The second reason the protocols distance except (IEEE

802.22) is small range. The third problem is all the available protocols can not work with the database because the network protocol is designed to work in the network layer only.

1.6 The Methodology

This research design a new confident authentication protocol in two phases, phase one authenticate the master with the server, and after that the master ask the server about the user's list which the master is allow to authenticate. In the second phase the user request to authenticate with the master and here three cases will happen. In the first case the master found the user's data in the list that he get from the server in phase one, in this case the authentication will follow the same way as phase one. In the second case if the user is not in the master's list then the master send the user's authentication request to the server asking about the user's data and the server reply by sends the user's data to the master. Then the master updates its list, and then follows the same process as phase one to authenticate the user. In the last case if the user in not registered in the server that means the user is not allow to use the spectrums in this location, so the server sends authentication reject to the master and the master ignore the user request.

1.7 The Proposed Solution

To enhance the security for the cognitive functionality in IEEE 802.22, security sublayer 2 is introduced. These security mechanisms validate the availability of spectrum for the primary and the secondary users by employing mechanisms such as distributed sensing and decision making.

But this protocol can not utilize the availability of the database for authentication, so when the user mode I want to authenticate with the database server, this authentication needs two types of authentication protocol. The first protocol to authenticate the mode I device with master mode II device using one of the network authentication's protocols. The second protocol is a database authentication protocol to authenticate the link between the master mod II with the database server.

To overcome these weaknesses in these protocols (IEEE 802.22 & security sublayer 2) this research designed a new mutual authentication protocol which can allow the user mode I to authenticate themselves with the database server and generate and exchange a shared secrete key using one protocol.

1.7 Thesis Outline

The rest of this research is organized as follow, chapter 2 discussed the TVWS access protocol background, chapter 3 previews some related works they are conducted to solve the TVWS authentication problems in the available protocols. Chapter 4 present the methodology used in this research. Chapter 5 shows the simulation result and discussion about these results. The research concluded and present some future works in the chapter 6.

CHAPTER TWO
PROTOCOL ACCESS WHITE SPACE BACKGROUND

2.1 Overview

Wireless Local Area Networks (WLAN) are widely used in our everyday life. Users are adopting the technology to save time and cost of running wires in providing high speed network access. The IEEE 802.11 is the most widely used WLAN standard, but it suffers from the weakness of its security protection [19].

The needs for wireless network is increased every day Figure 2.1 specify the demand versus capacity wireless network [1]. The TVWS frequencies were proposed to meet these requirements. These frequencies can be reused in broadband communication. TVWS can be licensed by auction or freely unlicensed, which is preferred by many parties around the world. TVWS unlicensed can use cognitive radio (CR) techniques for sharing the spectrum. Many standards agreed two ways for spectrum sharing, either by spectrum sensing and/or spectrum database. Figure 2.2 determined the global database growing opportunity.

The IEEE standard defined many protocols that can utilize the available spectrums. This chapter reviews some of these protocols terminologies in brief and IEEE 802.11i and IEEE802.11af and IEEE 802.22 in more details. This because IEEE802.11af and IEEE 802.22 are proposed to be TVWS protocols; and IEEE 802.11i is discussed the authentication and key management in the protocol architecture. Another protocols access white space non IEEE protocols are discussed, and also some database protocols.

2.2. IEEE Protocols

2.2.1 IEEE 802.11 WEP

Wired Equivalent Privacy (WEP) was an encryption algorithm designed to provide wireless security for users who implement 802.11 wireless networks [10]. WEP uses the RC4 stream cipher, combining a 40-bit WEP key with a 24-bit random number known as an Initialization Vector (IV) to encrypt the data. The sender XORs stream ciphers with the actual data to produce cipher text. The packet, combined with the IV and with the cipher text that sent to the receiver. The receiver decrypts the packet using the stored WEP key and the attached IV [12].

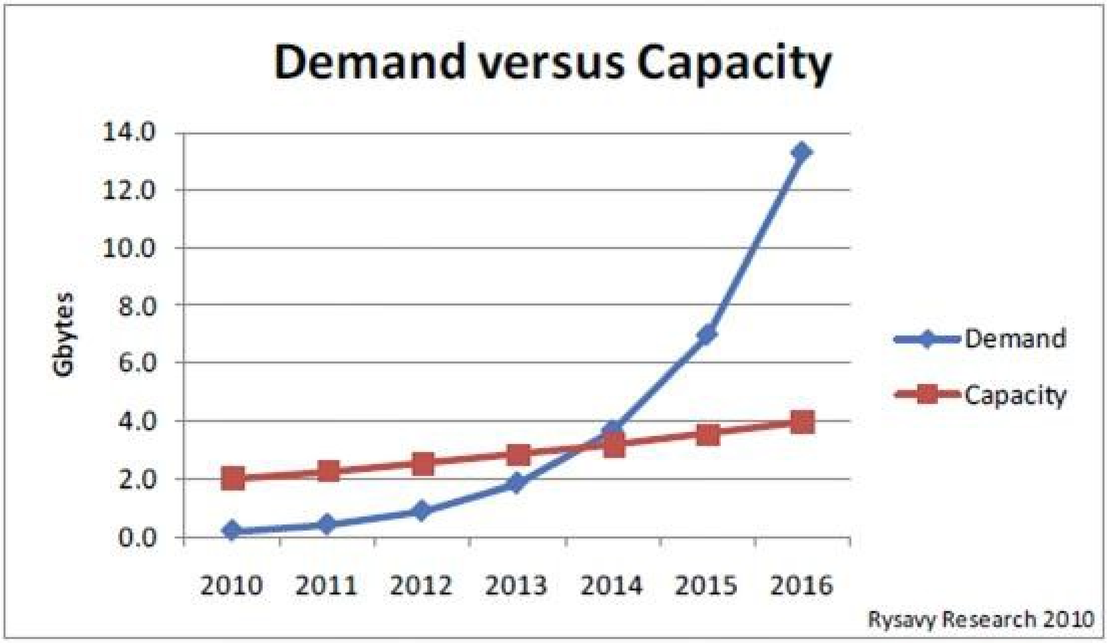
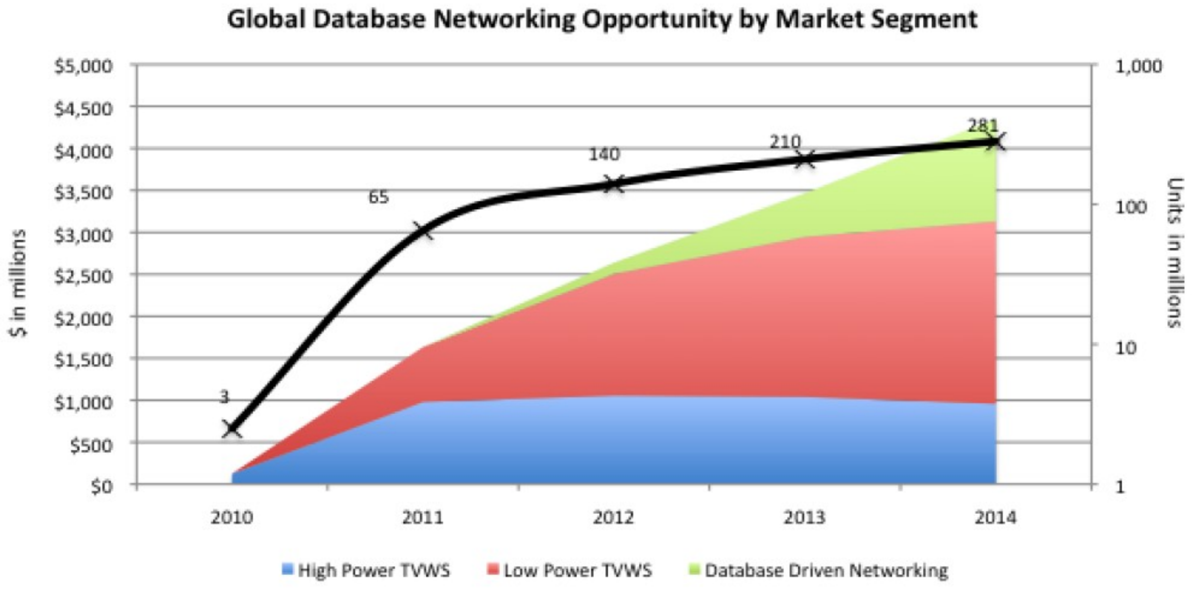


Figure 2.1 the wireless network demand and network capacity



Source: 2010 ABI Research, In-Stat, Andrew Seybold and Spectrum Bridge estimates.

Figure 2.2 global database growing opportunity by market segment

WEP has several security issues, such as weak key usage, reuse of initial vectors, exposure to replay and packet forging and problems with the encryption algorithm. Other than that, key management and updating is poorly designed in WEP [13]. These keys are weak and can be cracked, even in few hours or minutes using freely available software. The ability to modify packets, even without knowing the encryption key allows an attacker to modify or alter packets undetectably [14].

The 802.11 family [15, 16, 17] consists of a series of half-duplex over-the-air modulation techniques that use the same basic protocol. 802.11-1997 was the first wireless networking standard in the family, but 802.11b was the first widely accepted one, followed by 802.11a, 802.11g, 802.11n, and 802.11ac. Other standards in the family (c–f, h, j) are service amendments and extensions or corrections to the previous specifications. The security of IEEE 802.11 WEP is use unilateral authentication considered broken.

2.2.2 IEEE 802.1X

The IEEE 802.1X standard defines a mechanism for port-based network access control to provide compatible authentication and authorization mechanisms for devices interconnected by various 802 LANs [18]. It could also be used to distribute security keys for 802.11 WLANs by enabling public key authentication and encryption between access points (APs) and mobile nodes (MNs). In 802.1X, the port represents the association between MN and AP. There are three main components in the 802.1X authentication system: supplicant, authenticator, and authentication server (AS) Figure 1 depicts a typical 802.1X message exchange with both the supplicant PAE and authenticator PAE state transitions [19].

Diameter protocol is another framework for carrying authentication, authorization and accounting information between the network access server and the AAA Server [21]. Nowadays the application of RADIUS protocol is most widely used than Diameter, but for compatibility reasons the capacity of transition from one protocol to the other is indispensable.

The Remote Authentication Dial In User Service (RADIUS) protocol was originally defined to enable centralized authentication, authorization, and access control (AAA) for SLIP and PPP dial-up sessions [23].

2.2.3 IEEE 802.11i

To enhance the security performance in WLANs, IEEE launched IEEE 802.11i standard [25]. This standard allows wireless to have secure communication through the validation process, called authentication, applied to both user and device. The IEEE 802.11 Working Group has been working on MAC enhancement for several years. In May 2001, the MAC enhancement was split into different task groups. Task Group E (TGe) is responsible for quality of service (QoS). Task Group I (TGi) is working on security [25].

One of the major missions of 802.11 TGi is to define a robust security network (RSN). The definition of an RSN according to IEEE 802.11i draft [2] is a security network that only allows the creation of robust security network associations (RSNAs). That is, in an RSN the associations between all stations including APs are built on a strong association/authentication called an RSNA, which is also defined by the 802.11 TGi as: an RSNA depends on 802.1X to transport its authentication services and deliver key management services. A security association is defined as the context providing the state (cryptographic keys, counters, sequence spaces, etc.) needed for correct operation of the IEEE 802.11 cipher suites. RSNA includes a novel four-way handshake mechanism to provide robust session key management. By leveraging IEEE 802.1X, the four-way handshake, and the enhanced cryptographic algorithms, communication links in 802.11 wirelesses are securely protected.

2.2.3.1 The IEEE 802.11i Framework

The 802.11i standard defines two classes of security framework for 802.11 WLANs: RSN and pre-RSN. A station is considered RSN-capable equipment if it is capable of creating RSNAs.

Otherwise, it is pre-RSN equipment. A network that only allows RSNA in associations with RSN capable equipments is called an RSN security framework. A network that allows pre-RSNA associations between stations is called a pre-RSN security framework. The major difference between RSNA and pre-RSNA is in the four way handshake. If the four-way handshake is not included in the authentication/association procedures, stations are said to be pre-RSNA.

Pre-RSN: Pre-RSN security consists of two security subsystems, IEEE 802.11 entity authentication and WEP security.

The IEEE 802.11 entity authentication includes open system authentication and shared key authentication. In open system authentication, there is no authentication algorithm. A station is authenticated simply based on its identity. Shared key authentication, on the other hand, authenticates a station based on a secret key known to the authentication requester and responder. It requires the privacy mechanism to be implemented in WEP.

RSN: In addition to enhancing the security in pre-RSN, RSN security defines key management procedures for 802.11 networks. It also enhances the authentication and encryption in pre-RSN.

Authentication enhancement: 802.11i utilizes 802.1X for its authentication and key management services. It incorporates two components into the 802.11 architecture: the 802.1X port and authentication server (AS). The 802.1X port represents the association between two peers. There is a one-to-one mapping between the 802.1X port and the association. As discussed earlier, an 802.1X port will allow general traffic to pass only when the authentication is successfully completed.

The AS could be a standalone server or integrated into an AP. Although the protocol between the AS and AP is not specified by 802.11i, there should be a secure channel such as TLS (IETF RFC 2246) or IPSec (IETF RFC 2401) between the AP and AS. An EAP that supports mutual authentication should be used in an RSN. That is, the authentication requester and responder must be able to authenticate each other. EAP-MD5, for instance, cannot meet this requirement.

2.2.3.2 Key Management and Establishment

Two ways to support key distribution are introduced in 802.11i: manual and automatic key management. Manual key management requires the administrator to manually configure the key.

Automatic key management is available only in an RSNA. It relies on 802.1X to support key management services. More specifically, a fourway handshake is used to establish each transient key for packet transmission.

2.2.3.3 Encryption Enhancement

In order to enhance confidentiality, two advanced cryptographic algorithms are developed: Counter- Mode/CBC-MAC Protocol (CCMP) and Temporal Key Integrity Protocol (TKIP). In

RSN, CCMP is mandatory. TKIP is optional and recommended only to patch pre-RSNA equipment.

IEEE 802.11i specifies an RSN information element (RSN IE) that carries RSN security information including RSN capabilities, authentication, and cipher key selectors. An RSN IE could be used to distinguish pre-RSN stations and RSN-capable stations. RSN-capable stations shall include the RSN IE in beacons, probe response, association and reassociation request, and the second and third messages of the four-way handshake. On the other hand, there is no RSN IE in messages sent by pre-RSN stations. The RSN IE contains a list of authentication and cipher selector fields for communications. The Authentication and Key Management Suite Count indicates the number of authentication and key management suites contained in the Authentication and Key Management Suite List field. In the RSN Capabilities field, the requested or advertised capabilities are filled in. By using this field, the receiver can know the security mechanisms the sender supports or is requesting.

2.2.3.4 Authentication Enhancement

In the original 802.11 standard, a station should first associate with an 802.11 AP. It then is able to access the WLAN service. After finding an AP by receiving the Probe Response, the mobile station needs to proceed to the following two steps: 802.11 entity authentication and association. Before associating with an AP, the station needs to accomplish 802.11 entity authentications.

There are two authentication schemes: open system and shared key authentication. Open system authentication allows a station to be authenticated without having a correct WEP key. There are two message exchanges. The first message sending from supplicant (mobile station) to authenticator (AP) is used to expose the identity of the station. Based on the identity, the authentication result is sent from the authenticator back to the station. There is no authentication algorithm. In shared key authentication, there are four message exchanges. The first message containing the identity of the station is delivered from the station to the AP. The AP will then send a challenge has been discussed. The two message exchanges of flows 3 and 4 for open system authentication should not be replaced by the four message exchanges of shared key authentication.

IEEE 802.11i also specifies a more robust security framework utilizing 802.1X, a four-way handshake, and a group key handshake to authenticate and authorize stations. Please note that Figure. 2.3 depict the four-way handshake. After the station is authenticated successfully, the cryptographic keys are configured as well. The station is thus able to send and receive uncast and broadcast frames in a secure manner. Moreover, IEEE 802.11i also supports pre-authentication. A station could pre-authenticate with an AP before roaming. A station could initiate an EAPOL-Start message through the serving AP to inform the new AP to start the IEEE 802.1X authentication packet to the mobile station. The mobile station is required to encrypt the challenge packet using the shared WEP key and send the encrypted result back to the AP. If the challenge packet is encrypted correctly, the supplicant is authenticated successfully. The authentication result is sent to the station in the fourth message.

If the station is authenticated successfully, it proceeds to the 802.11 association. The mobile station should transmit an Association Request to the AP. The AP then sends back an Association Response to the station.

Figure 2.3 IEEE 802.11i enhancements

EAPOL-key (Key_info_Anonce)
EAPOL-key (Key_info_Anonce,MIC,RSN IE)
EAPOL-key (Key_info_Anonce,MIC,RSN IE)
EAPOL-key (Key_info_Anonce)
IEEE 802.11X
Supplicant
IEEE 802.11X
Authenticator
1. 802.11 probe request
2. 802.11 probe response
3. 802.11 open system
Authentication request
4. 802.11 open system
Authentication response
5. 802.11 Association request system
6. 802.11 Association response
7. IEEE802.1X Authentication
8. 4- way handshake
9. group key handshake
Encryption

EAPOL-key (Key_info_KeyID,KeyRSC,
MIC,GTK)
EAPOL-key (Key_info_MIC)

Shared key authentication in 802.11 is not adopted by 802.11i. Instead, it incorporates 802.1X as the authentication solution for the RSN. 802.1X is performed after 802.11 open system authentication and association. IEEE 802.1X provides a port-based network access control mechanism to protect against unauthorized access. Details of 802.1X

2.2.4 IEEE 802.11af

The requirements specification of 802.11af system is formed; the standardization process is using the principles of CR [26]. In another way, this standard is also called "Super Wi-Fi", or "White-Fi", "Super" - because of its cognitive properties, and "White" - due to work in a range of free TV WS frequencies. 802.11af is a modified 802.11 standard, which operate in a range of TV WS using the properties of CR. In this system, cognitive functions are supported using the channel power management (CMP) and mechanisms of dynamic station enablement (DSE), which controls the channel dependent stations (STAs) operating under the control of the enabling STA. In order to describe how the cognitive functions are implemented in this standard, it is necessary to consider the composition of the system. 802.11af includes three different STA types: fixed, enabling, and dependent STA. Fixed and enabling STAs are registered station that broadcast its registered location. The enabling STA is permitted to enable operation of unregistered STAs, i.e. dependent STAs. The enabling STA gets the available channel information from the TV WS database, and transmits the contact verification signal (CVS). The CVS is used for both establishing that the dependent STAs are still within the range of enabling STAs, as well as for checking the list of available channels. DSE allows dependent STAs use the available TV channels under the control of the enabling STA. Figure 2.4 illustrates DSE procedure of processing between enabling STA and dependent STA. In addition channel power management (CPM) is also used to update the list of available channels for work in basic service set (BSS), change a maximum transmission power or change the BSS operation in channel frequency and channel bandwidth, together with a maximum value transmission power [27]. There are two operating scenarios for 802.11af: first is shown on the figure 2.4, second is based on the access point (AP) communication to TV WS database through the so-called registered location secure server (RLSS). Depending on operating conditions, there may be two scenarios for deploying 802.11af standard.

2.2.4.1 The Standard Framework

This research describe the primitives and main mechanisms of the IEEE 802.11af standard, including the key architecture components, the communication flow and mechanisms utilized by the standard to satisfy different international regulations and introduce the entities that form an 802.11af network.

2.2.4.2 Components of the IEEE 802.11af Architecture

2.2.4.2.1 Geolocation Database (GDB)

The primary element and what mainly differentiates the IEEE 802.11af operation to other 802.11 standards is the GDB. The GDB is a database that stores by geographic location the permissible frequencies and operating parameters for WSDs to fulfill regulatory requirements. The GDBs are authorized and administrated by regulatory authorities; therefore the GDB's operation depends on the security and time requirements of the applied regulatory domain [28].

2.2.4.2.2 Registered Location Secure Server (RLSS)

The next architectural element in an IEEE 802.11af network is the Registered Location Secure Server (RLSS). This entity operates as a local database that contains the geographic location and operating parameters for a small number of basic service sets (BSSs). The RLSS distributes the permitted operation parameters to the APs and STAs within the BSSs under the RLSS control [28].

2.2.4.2.3 Geolocation Database Dependent (GDD)

The remainder elements in the IEEE 802.11af network are referenced by the term Geolocation Database Dependent (GDD), which specifies that their operation is controlled by an authorized GDB which assures these satisfy regulation requirements [28].

2.2.4.2.4 GDD Enabling Station

The GDD enabling station is the equivalent of the entity commonly known as the access point (AP). However, in the 802.11af standard this entity controls the operation of the stations (STAs) in its serving BSS. The GDD enabling STA can securely access the GDB to attain the operating frequencies and parameters permitted in its coverage region. With this information the GDD enabling STA has the authority to enable and control the operation of the STAs under its service, identified as GDD dependent STAs. Specifically, the parameters obtained from the GDB are represented through a white space map (WSM). The GDD enabling STA ensures to maintain and distribute a valid WSM. Additionally, the GDD enabling STA transmits a contact verification signal (CVS), for GDD dependent STAs to check validity of the WSM [28].

2.2.4.2.5 GDD Dependent Station

The GDD dependent station can be identified as the STAs in the BSS architecture. However, the 802.11af standard specifies that the operation of the STAs is controlled by the serving GDD enabling STAs. The GDD dependent STAs obtain the permitted operating frequencies and parameters in a form of a WSM from either the GDD enabling STA or RLSS. The validity of the WSM is confirmed through the CVS transmitted by the GDD enabling STA [28].

2.2.4.2.6 Registered Location Query Protocol (RLQP)

The Registered Location Query Protocol (RLQP) serves as the communication protocol between GDD enabling and GDD dependent STAs to share WSM and channel utilization [28]. This protocol enables the operation of the main mechanisms; used in the IEEE 802.11af standard. Through this communication the STAs can effectively select spectrum, power and bandwidth allowed by their regulation domain.

2.2.4.3 Communication Flow between Entities

The 802.11af standard defines the communication protocol between the GDD dependent STAs, GDD enabling STAs and RLSS. However, the communication flow between the GDB and the high level entities (RLSS and GDD enabling STAs) is outside the scope of the 802.11af protocol. The standard's mechanisms are independent of how this communication is performed, allowing regulators to select the communication protocol over the Internet’s infrastructure. Figure 2.5 illustrates two infrastructure BSSs containing all the components of the IEEE 802.11af architecture introduced in Section 2.2.4.1. As shown in Figure 2.5, the RLSS and GDD enabling STAs obtain white space availability through the Internet. Within the 802.11af scope, the RLSS only communicates with the GDD enabling STAs through infrastructure and operates bi-directionally. Finally, the GDD dependent STAs perform bi-directional, over-the-air communication with GDD enabling STAs [28].

2.2.4.4 802.11af Mechanisms

In this section the research present the mechanisms defined in the 802.11af standard and logical messages passed between the architecture entities to satisfy regulatory requirements.

2.2.4.4.1 Channel Availability Query (CAQ)

Through the CAQ procedure, STAs obtain the available radio frequencies that allow operation in their location, in form of a White Space Map (WSM). In the CAQ process the RLSS grants the WSM to the CAQ requesting STA. However in some regulatory domains the RLSS is required to access the GDB to obtain the channel availability information. The CAQ request may contain multiple device locations. The CAQ responding STA must restrict the WSM validity to either a unique device location or a bounded area of multiple locations [28].

The GDD dependent STA performs a CAQ request to a GDD enabling STA in three different cases. First, to remain in the GDD enable state after enablement times out. Second, the CAQ is required when a change in channel availability is indicated by the GDD enabling STA through a CVS. Third, if the GDD dependent STA has moved beyond the regulatory permitted distance [28].

2.2.4.4.2 Channel Schedule Management (CSM)

The GDD enabling STAs use the Channel Schedule Management (CSM) procedure to query a RLSS or other GDD enabling STAs to obtain white space channel schedule information. The channel schedule indicates a schedule change and consists of the start and ending times for the requested channels [28].

The GDD dependent STAs do not perform CSM requests. However, the GDD enabling STAs can transmit a CSM request to a RLSS or other GDD enabling STA (with GDB or RLSS access) to query the schedule information for white space channels in either TV channels or WLAN channels.

2.2.4.4.3 Contact Verification Signal (CVS)

The Contact Verification Signal is sent by a GDD enabling STA to serve two purposes. First, the transmission of the CVS establishes which GDD dependent STAs are within the reception range of a GDD enabling STA. Second, the CVS helps the GDD dependent STAs ensure operation under a valid white space map (WSM) and that it corresponds to the serving GDD enabling STA [28].

2.2.4.4.4 GDD Enablement

The GDD Enablement procedure allows a GDD enabling STA to form a network, satisfying regulation requirements under the control of a GDB [28]. A GDD enabling beacon signal is transmitted on available channels in the TVWS band by a GDD enabling STA to offer GDD enablement service. A GDD dependent STA upon receiving the GDD enabling signal can attempt enablement with the GDD Enablement Response frame. However, some regulatory domains require that prior to enablement the GDD enabling STA identifies with a GDB the requesting GDD [28].

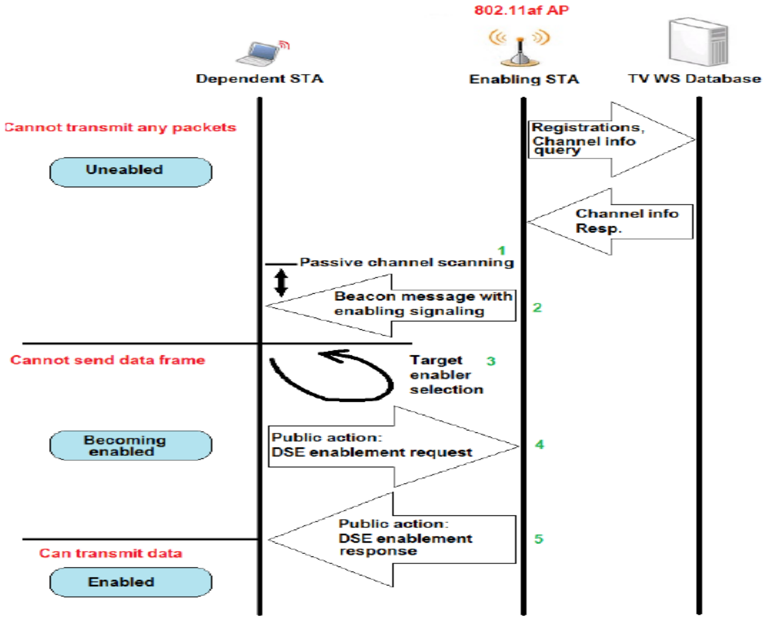


Figure 1: DSE processing procedure [10]

Figure 2.4 DSE Processing procedures in 11af

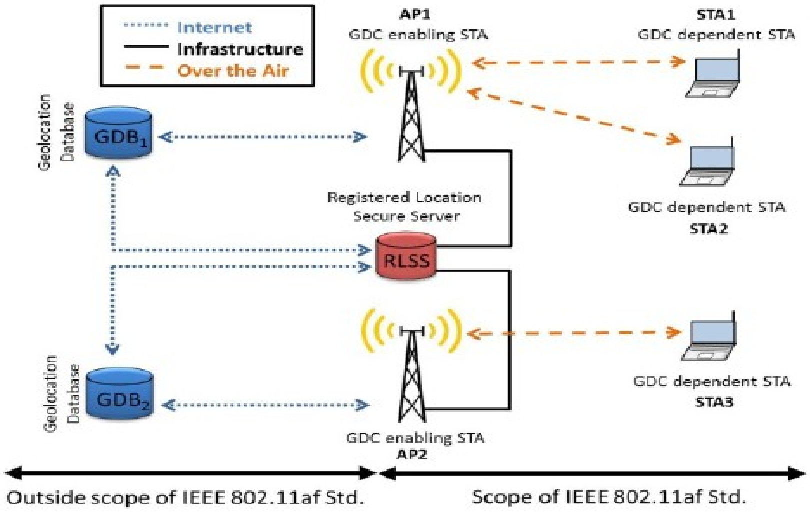


Figure 2. Example TVWS network including all 802.11af architecture entities [9].

Figure 2.5 example TVWS network including all 802.11af architecture entities

2.2.5 IEEE 802.15

The IEEE has approved the start of work on four projects concerning IEEE 802.15™ wireless personal area network (WPAN) standards [29]. These projects involve a wireless mesh topology standard for WPAN devices and alterations to the high rate WPAN standard so it supports new wireless multimedia uses more effectively. Two other projects were started for ultra-low power WPANs: one will create an alternate PHY and the other will correct and extend the base standard. IEEE P802.15.5™, “Recommended Practices for Mesh Topology Capability in Wireless Personal Area Networks (WPAN),” will provide an architectural framework for interoperable, stable and scalable wireless mesh topologies for WPAN devices. Mesh topologies can extend network coverage without increasing transmission power or receiver sensitivity. They can also improve reliability via route redundancy, easier network configuration and longer device battery life. IEEE P802.15.3b™, “Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amendment to MAC Sublayer,” will modify IEEE 802.15.3™ to improve the ease of implementation and interoperability. This will include minor optimizations while preserving backward compatibility. In addition, this amendment will correct errors, clarify ambiguities and add editorial clarifications. IEEE P802.15.4a™, “Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Alternate Physical Layer Extension for Low Rate Wireless Personal Area Networks (WPAN),” will provide an alternate WPAN PHY to meet evolving user needs for ultra-low complexity, ultra-low cost, ultra-low power WPAN communications. It will provide for precision ranging accurate to one meter or less, improved communication range, improved link robustness and the ability to support mobility. It also will continue to support coexisting networks of sensors, controllers and peripheral devices in multiple, compliant co-located systems. IEEE P802.15.4-REVb™, “Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPAN),” will revise the IEEE 902.15.4™ – 2003 standard to remove ambiguities.

2.2.5.1 IEEE802.15.4

The IEEE 802.15.4 specification outlines a new class of wireless radios and protocols targeted at low power devices, personal area networks, and sensor nodes. The specification includes a number of security provisions and options.

The growing importance of small and cheap wireless devices demands a common platform, so that the devices can communicate with each other and share components to lower costs. The 802.15.4 specification [29] describes wireless and media access protocols for personal area networking devices. The sensor network community has begun using these protocols as well. The protocols are intended for hardware implementation on a dedicated radio chip. The range of envisioned applications is broad, spanning wireless game controllers, environmental, medical, and building monitoring instruments, to heating and ventilation sensors [29].

2.2.5.2 Security Overview

A link layer security protocol provides four basic security services: access control, message integrity, message confidentiality, and replay protection. Access control means the link layer protocol should prevent unauthorized parties from participating in the network. Legitimate nodes should be able to detect messages from unauthorized nodes and reject them. Also, a secure network should provide message integrity protection: if an adversary modifies a message from an authorized sender while the message is in transit, the receiver should be able to detect this tampering. Including a message authentication code (MAC) with each packet provides message authentication and integrity. A MAC can be viewed as a cryptographically secure checksum of a message. Computing it requires authorized senders and receivers to share a secret cryptographic key, and this key is part of the input to the computation. The sender computes the MAC over the packet with the secret key and includes the MAC with the packet. A receiver sharing the same secret key recomputed the MAC and compares it with the MAC in the packet. The receiver accepts the packet if they are equal, and rejects it otherwise. Message authentication codes must be hard to forge without the secret key. Consequently, if an adversary alters a valid message or injects a bogus message, she will not be able to compute the corresponding MAC, and authorized receivers will reject these forged messages [29].

2.2.6 IEEE 802.16

The standard IEEE 802.16e [23] use the new privacy key management protocol PKMv2 to improve the security performance, in which the EAP [30] (Extensible Authentication Protocol) are introduced into IEEE 802.16e. By combining EAP and RSA, the PKMv2 has defined different authentication modes. According to the IEEE 802.16e PKMv2 there should be 5 kinds of authentication modes, according to the 8-bit binary value of the 'Authorization policy support' domain in SBC-REQ /SBC-RSP messages. The auth modes are single RSA, single EAP_based, RSA + authenticated_EAP, EAP + authenticated_EAP mode and RSA+EAP_based mode.

There are two problems with IEEE 802.16e to use it in the TVWS first problem is the protocol distance and the second one is this type of complexity authentication will become so difficult to implement in TVWS devices because this devices suppose to be a little capabilities.

Worldwide Interoperability for Microwave Access (WIMAX) is based on the IEEE802.16 Wireless Metropolitan Area Network (MAN) standard and it is a technology for providing last mile wireless broadband access as an alternative to cable and DSL (Digital Subscriber Line) [31]. The name WIMAX is defined by the WIMAX-forum which is a not-for-profit organization including more than 520 companies worldwide. WIMAX operates on multiple frequencies providing connection up to 40 Mb/s (single channel, line of sight) when fixed stations are used [31].

For mobile users WIMAX aim to provide connections up to 15Mb/s within a radius up to three kilometers [32]. WIMAX can be used in point-to-point, point-to-multipoint and mesh networks.

WIMAX security management is mainly defined in 802.16-2004 standard and some improvements have been introduced in 802.16e amendment. WIMAX security relies on an authenticated client-server key management protocol called PKM (Privacy Key Management) [31p.271]. Basic privacy is additionally strengthened by adding digital certificates- based authentication to key management protocol. The standard 802.16-2004 introduces the PKMv1 protocol as the method for key management. Later on PKMv1 protocol was improved by 802.16e amendment, which introduced PKMv2 protocol. This work concentrate on only these key management and authorization protocols and our aim is to introduce these techniques and evaluate and analyze possible weaknesses.

WiMAX, both physical and MAC layers have risk of threats like jamming [33] and denial of service respectively. But there are no efficient procedures to deal with threats posed at PHY layer of WiMAX so, the emphasis of WiMAX security is entirely at the MAC level [34]. MAC layer security threats and vulnerabilities of the WiMAX networks [35].

2.2.7 IEEE 802.19

The IEEE 802.19™ Wireless Coexistence Technical Advisory Group (TAG) has begun development of a recommended practice on methods for assessing the coexistence of wireless networks. This standard defined recommended wireless coexistence metrics and the methods for computing them, as well as various wireless coexistence scenarios. “Industry continues to develop new standards and specifications for wireless networks that operate in the same frequency bands as other wireless networks,” said Paul Nokolich, chair of the IEEE 802 (R) Local Area and Metropolitan Area Network Standards Committee [36]. “IEEE 802, for instance, has multiple working groups developing wireless networks standards for systems that share frequency bands. The recommended practice to be created by the IEEE 802.19 TAG will help IEEE 802 working groups and the rest of the industry assesses the performance of new wireless technologies and those now deployed in shared frequency bands." Steve Shellhammer, chair of the 802.19 Technical Advisory Group, adds that "The IEEE 802.19 TAG continues its work to meet the evolving needs of the public and industry. The IEEE has published a new standard for the coexistence in the TV white space among different or independently operated wireless networks. IEEE 802.19.1 is the "standard for TV white space (TVWS) coexistence methods." "IEEE 802.19.1 specifies radio technology independent methods for coexistence among dissimilar or independently operated wireless networks operating in TVWS. The standard is also intended to do the following:

- Leverage the cognitive radio capabilities of the TVWS devices, including geolocation awareness and access to information databases.
- Specify a coexistence discovery and information server, which gathers and provides coexistence information regarding TVWS networks.
- Specify a coexistence manager, which utilizes the information from the coexistence server in order to enhance the coexistence of the TVWS networks.
- Define common coexistence architecture and protocols, as well as several profiles to enable cost-efficient and flexible deployment of the coexistence system in various scenarios.

2.2.8 IEEE 802.22

IEEE 802.22 is the first standard protocol for enabling the use of the fallow TV bands by the infrastructure single-hop cognitive radio networks (CRNs) with the presence of one base station (BS) that performs spectrum management, which supports the provision of broadband fixed wireless data transmission in sparsely populated rural areas [37,38]. This standard on wireless regional area networks (WRANs) specifies the air interface including the cognitive medium access control layer (MAC) and physical layer (PHY).

The IEEE 802.22 security sub-layer1 architecture is shown in Figure 3.1, which provides the CPEs with security functions for the payloads and the MAC management messages across the WRAN TVWS. It achieves the security functionality by applying cryptographic transforms to

MAC Protocol Data Units (PDUs) carried across connections between the CPEs and BS. The security sub-layer 1 of the IEEE 802.22 standard has been designed with reference of the privacy key management version 1 (PKMv1) and the PKMv2 from IEEE 802.16-2009 standard [39, 40], which is correlated with the IEEE 802.22 operation and renamed the PKM protocol as the SCM protocol. The SCM protocol provides secure distribution of keying materials from the BS to the CPE and performs mutual authentication between the BS and the CPE. The SCM's authentication protocol establishes a shared secret between the CPE and the BS. The shared secret is then used to secure the subsequent SCM exchanges. The SCM supports elliptic curve cryptography-based authorization for authorizing the CPEs at the time of network entry. The main procedures carried out by the BS and the CPE to perform CPE network entry and initialization are as follows. Firstly, the CPE performs a test by itself and acquires the antenna gain information. Then, it performs sensing and synchronizing to the WRAN services, by which the CPE chooses a WRAN service and acquires the parameters of the downstream and upstream links from the selected WRAN service. Figure 3.2 shows the protocol architecture.

After the BS and the CPE perform an initial ranging, the CPE will precede with security capabilities negotiation. If all required basic capabilities are available in the CPE, the BS authenticates the CPE and performs the key exchange. Otherwise, the CPE cannot proceed to the registration phase.

The security suite consists of the authorization and authentication process. The authorization process is carried out when a CPE enters the network, to make sure that only the authorized device can access the network. The BS is capable of de-authorizing a CPE if the BS finds that the CPE does not contain valid AKs or it is generating spurious emissions. The authorization process also includes a mutual authentication process where both the BS and the CPE can authenticate each other. The security sub-layer 1 has a key management protocol by which the BS controls the process of the distribution of keying material to the CPEs. This keying material is used to protect the MAC management messages and may be optionally used to protect user data [44]. Additionally, the EAP-based authentication mechanism has been included to enhance the security function of the system. At the capabilities negotiation step of the network entry, if the CPE indicates that it will not support IEEE 802.22 security function, then the step of authorization and key exchange will be skipped. Otherwise, the BS considers the CPE as authenticated and authorizes the CPE to access the network. Without the authorization by the BS, the CPE will not be serviced, and further, the network entry of the CPE is de-authorized, and neither key exchange nor data encryption will be carried out. After the authorization, the traffic encryption keys will be exchanged by using the SCM protocol.

3.2.8.1 Comparison of 802.11af and 802.22 standards

Despite the fact that both standards operate in the TV WS range and that both of them have the properties of CR, in general they are very different [26]. Comparative analysis of the systems should be conducted in three planes: PHY layer feature, MAC layer feature and Cognitive feature. The results of the comparison are summarized in Tables 2.1.

2.2.8.1.1 Differences on PHY layer

In general, both the two standards use the same technology at PHY layer (Orthogonal Frequency Division Multiplexing (OFDM) modulation and Quadrature Phase-Shift Keying (QPSK), 16-QAM (Quadrature Amplitude Modulation), 64-QAM payload modulation, however Binary Phase-Shift Keying (BPSK) can be used only in 802.11af standard), but there is a difference in the total bandwidth, FFT size and error correcting code [26].

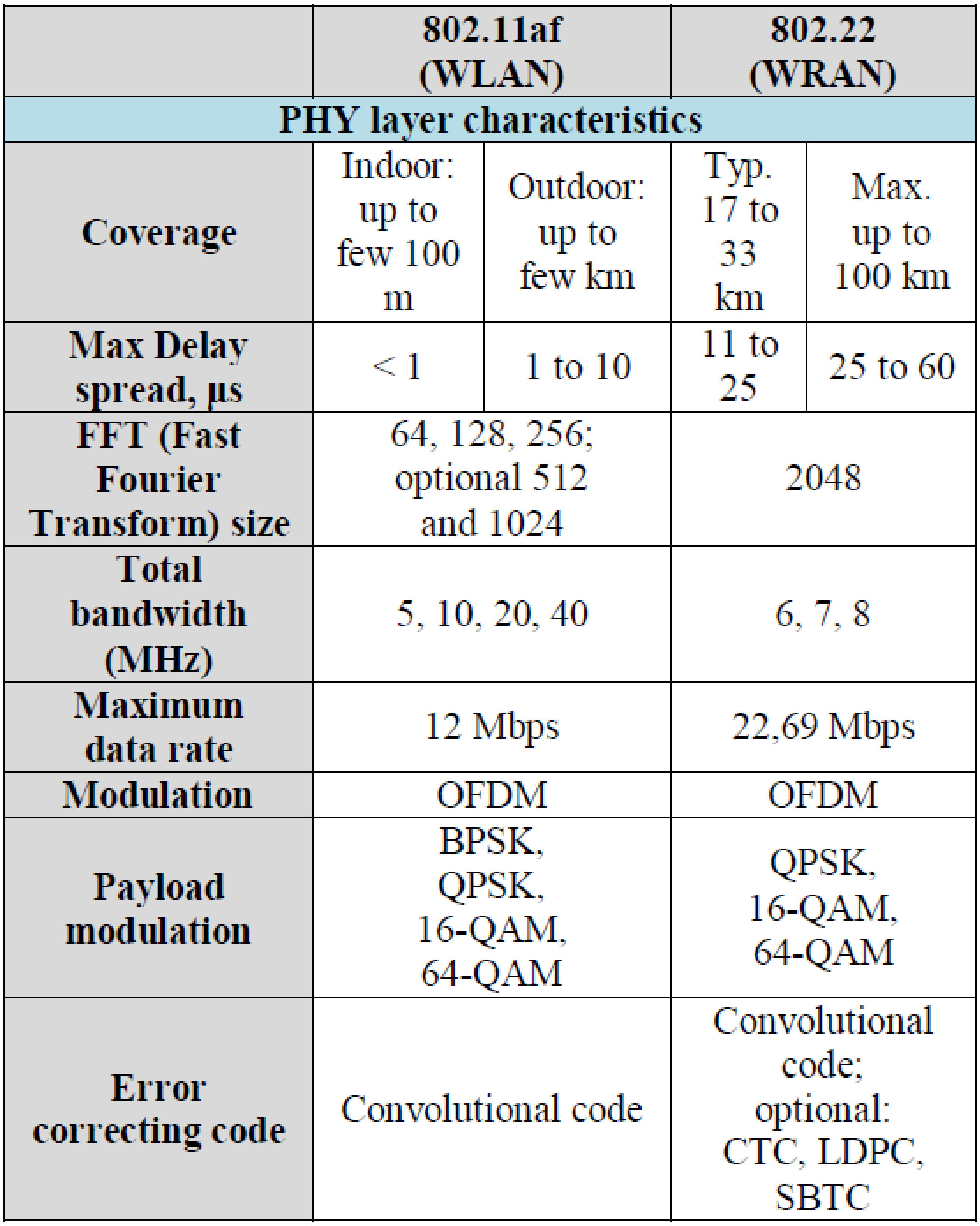
2.2.8.1.2 Differences on MAC layer

On MAC layer the 802.22 standard use Time Division Multiplex (TDM)-based access with PHY resources allocated on demand using Orthogonal Frequency Division Multiple Access (OFDMA, while 802.11af will use its Carrier Sense Multiple Access With Collision Avoidance (CSMA/CA)- based protocol [26]. Whereas 802.11af users may back off, when the medium is employed by 802.22 transmissions, the opposite can't be true, since the 802.22 devices do not need to listen before transmitting. The differences in MAC strategy can limit the effectiveness of non-cooperative listen-before-talk mechanism in achieving fairness in TV WS coexistence [28].

2.2.8.1.3 Differences on cognitive layer

In terms of cognitive origins of standards, the most important role is played by their cognitive properties [26]. As seen from Table 2.1, standard 802.11af has only the interface with the TV WS database, in contrast to standard 802.22 which employs larger set of cognitive properties.

Table 2.1: Comparison of parameters of 802.11af and 802.22 standards on PHY layer [68, 69, 70] 802.11af (WLAN) 802.22



	802.11af (WLAN)		802.22 (WRAN)	
PHY layer characteristics				
Coverage	Indoor: up to few 100 m	Outdoor: up to few km	Typ. 17 to 33 km	Max. up to 100 km
Max Delay spread, µs	< 1	1 to 10	11 to 25	25 to 60
FFT (Fast Fourier Transform) size	64, 128, 256; optional 512 and 1024		2048	
Total bandwidth (MHz)	5, 10, 20, 40		6, 7, 8	
Maximum data rate	12 Mbps		22,69 Mbps	
Modulation	OFDM		OFDM	
Payload modulation	BPSK, QPSK, 16-QAM, 64-QAM		QPSK, 16-QAM, 64-QAM	
Error correcting code	Convolutional code		Convolutional code; optional: CTC, LDPC, SBTC	

2.4 PAWS protocol

Internet Engineering Task Force (IETF) is developing a WG called Protocol to Access White Space database (PAWS) [7] with the goal of defining the device-database interface for TVWS database systems. Devices may be able to connect to the database directly or indirectly via the Internet or private IPnetworks. This interface needs to be: radio/air interface agnostic (802.11af, 802.16, 802.22, LTE etc)

PAWS pretend to specify both a database identification mechanism (how can a device know what database it has to connect to) and contents of the queries and responses (XML is an option). This protocol did not state any type of authentication procedure but just state that “This messaging between the device and the database needs to be secure (authentication, integrity of the content, prevent from man-in-the-middle attacks etc.), requiring some authentication and security measures” [7].

The PAWS protocol depends on tow layer authentication (HTTP/TLS and PAWS) this will become more complex and overhead. Figure 2.6 shows the PAWS protocol layer.

Figure 2.6 PAWS Protocol layers

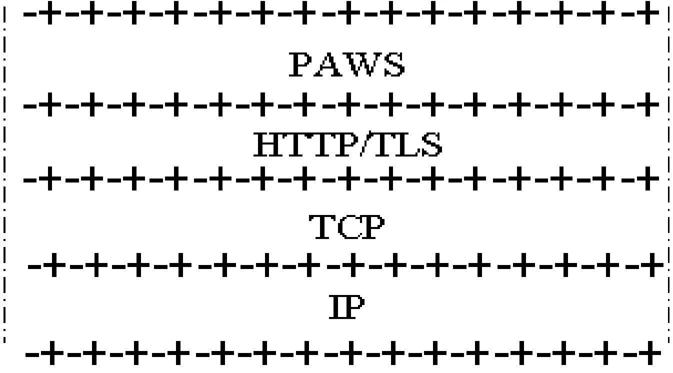


Figure3: PAWS Protocol layers

The protocol procedure is consisting of ten messages. Message one and tow to specify the device capability and message 3,4 for registration (if required) after the registration is completed the master send message 5 asking for the available channels he can use. Then the master send message 7 asking for device validation. (this is very important message, because this research uses this message to authenticate the user mode I). After the users select the channels they are willing to use, then the master send message 9 to notify the database about the usable channels. Figure 2.7 shows the protocol steps.

Figure 2.7: IETF protocol steps

White space
Database
Master Device
INIT_REQ
INIT_RESP
Registration_REQ
Registration_RESP
Device_validation_REQ
Device_validation_RESP
Spectrum_USE_RESP
Spectrum_USE_NOTIFY
AVAILABLE_CHANNELS_REQ
AVAILABLE_CHANNELS_RESP

2.5 Database Security

Database security is the mechanisms that protect the database against intentional or accidental threats. The responsibility to authorize use of the

DBMS usually rests with the Database Administrator (DBA), who must also set up individual user accounts and passwords using the DBMS itself [42].

2.5.1 Discretionary Access Control (DAC)

Most commercial DBMSs provide an approach to managing privileges that uses SQL called Discretionary Access Control (DAC). The SQL standard supports DAC through the GRANT and REVOKE commands. The GRANT command gives privileges to users, and the REVOKE command takes away privileges [42].

Discretionary access control, while effective, has certain weaknesses. In particular, an unauthorized user can trick an authorized user into disclosing sensitive data. For example,

an unauthorized user such as an Assistant in the *DreamHome* case study can create a relation to capture new client details and give access privileges to an authorized user such as a Manager without their knowledge. Clearly, an additional security approach is required to remove such loopholes, and this requirement is met in an approach called Mandatory Access Control (MAC).

2.5.2 Mandatory Access Control (MAC)

Mandatory Access Control (MAC) is based on system-wide policies that cannot be changed by individual users. In this approach each database object is assigned a *security class* and each user is assigned a *clearance* for a security class, and *rules* are imposed on reading and writing of database objects by users. The DBMS determines whether a given user can read or write a given object based on certain rules that involve the security level of the object and the clearance of the user. These rules seek to ensure that sensitive data can never be passed on to another user without the necessary clearance [42].

A popular model for MAC is called Bell–LaPadula model, which is described in terms of objects (such as relations, views, tuples, and attributes), subjects (such as users and programs), security classes, and clearances. Each database object is assigned a security class, and each subject is assigned a clearance for a security class. The security classes in a system are ordered, with a most secure class and a least secure class. There are four types of classes: top secret (TS), secret (S), confidential (C), and unclassified (U), and denote the class of an object or subject A as class (A). Therefore for this system, TS > S > C > U, where A > B means that class A data has a higher security level than class B data.

2.5.3 Comparing Discretionary Access Control and Mandatory Access Control

Discretionary access control (DAC) policies are characterized by a high degree of flexibility, which makes them suitable for a large variety of application domains [42].

The main drawback of DAC models is their vulnerability to malicious attacks, such as Trojan horses embedded in application programs. The reason is that discretionary authorization models do not impose any control on how information is propagated and used once it has been accessed by users authorized to do so. By contrast, mandatory policies ensure a high degree of protection—in a way, they prevent any illegal flow of information. Therefore, they are suitable for military and high security types of applications, which require a higher degree of protection.

However, mandatory policies have the drawback of being too rigid in that they require a strict classification of subjects and objects into security levels, and therefore they are applicable to few environments. In many practical situations, discretionary policies are preferred because they offer a better tradeoff between security and applicability.

2.5.4 Message Digest Algorithms and Digital Signatures

A message digest algorithm, or one-way hash function, takes an arbitrarily sized string (the *message*) and generates a fixed-length string (the *digest* or *hash*). A digest has two characteristics; first it should be computationally infeasible to find another message that will generate the same digest and secondly the digest does not reveal anything about the message [42].

A digital signature consists of two pieces of information: a string of bits that is computed from the data that is being ‘signed’, along with the private key of the individual or organization wishing the signature. The signature can be used to verify that the data comes from this individual or organization. Like a handwritten signature, a digital signature has many useful properties; first its authenticity can be verified, using a computation based on the corresponding public key. Secondly it cannot be forged (assuming the private key is kept secret). Thirdly it is a function of the data signed and cannot be claimed to be the signature for any other data; and lastly the signed data cannot be changed; otherwise the signature will no longer verify the data as being authentic.

Some digital signature algorithms use message digest algorithms for parts of their computations; others, for efficiency, compute the digest of a message and digitally sign the digest rather than signing the message itself.

2.5.5 Digital Certificates

A digital certificate is an attachment to an electronic message used for security purposes, most commonly to verify that a user sending a message is who he or she claims to be, and to provide the receiver with the means to encode a reply [42].

An individual wishing to send an encrypted message applies for a digital certificate from a Certificate Authority (CA). The CA issues an encrypted digital certificate containing the applicant’s public key and a variety of other identification information. The CA makes its own public key readily available through printed material or perhaps on the Internet.

The recipient of an encrypted message uses the CA’s public key to decode the digital certificate attached to the message, verifies it as issued by the CA, and then obtains the sender’s public key and identification information held within the certificate. With this information, the recipient can send an encrypted reply.

Clearly, the CA’s role in this process is critical, acting as a go-between for the two parties. In a large, distributed complex network like the Internet, this third-party trust model is necessary as clients and servers may not have an established mutual trust yet both parties want to have a secure session. However, because each party trusts the CA, and because the CA is vouching for each party’s identification and trustworthiness by signing their certificates, each party recognizes and implicitly trusts each other. The most widely used standard for digital certificates is X.509.

2.5.6 Kerberos

Kerberos is a server of secured user names and passwords (named after the three-headed monster in Greek mythology that guarded the gate of hell). The importance of Kerberos is that it provides one centralized security server for all data and resources on the network [42].

Database access, login, authorization control, and other security features are centralized on trusted Kerberos servers. Kerberos has a similar function to that of a Certificate server: to identify and validate a user. Security companies are currently investigating a merger of Kerberos and Certificate servers to provide a network-wide secure system. The weakness of the Kerberos is it is a centralized server and when it crash all the system will crash.

2.5.7 Secure Sockets Layer and Secure HTTP

Many large Internet product developers agreed to use an encryption protocol known as Secure Sockets Layer (SSL) developed by Netscape for transmitting private documents over the Internet [42]. SSL works by using a private key to encrypt data that is transferred over the SSL connection. Both Netscape Navigator and Internet Explorer support SSL and many Web sites use this protocol to obtain confidential user information, such as credit card numbers. The protocol, layered between application-level protocols such as HTTP and the TCP/IP transport-level. The protocol, is designed to prevent eavesdropping, tampering, and message forgery. Since SSL is layered under application-level protocols, it may be used for other application-level protocols such as FTP and NNTP.

Another protocol for transmitting data securely over the Web is Secure HTTP (S-HTTP), a modified version of the standard HTTP protocol. S-HTTP was developed by Enterprise Integration Technologies (EIT), which was acquired by Verifone, Inc. in 1995. Whereas SSL creates a secure connection between a client and a server, over which any amount of data can be sent securely, S-HTTP is designed to transmit individual messages securely. SSL and S-HTTP, therefore, can be seen as complementary rather than competing technologies. Both protocols have been submitted to the Internet Engineering Task Force (IETF) for approval as standards. By convention, Web pages that require an SSL connection start with (https **:)** instead of (http :). Not all Web browsers and servers support SSL/S-HTTP. Basically, these protocols allow the browser and server to authenticate one another and secure information that subsequently flows between them. A key component in the establishment of secure Web sessions using the SSL or S-HTTP protocols is the digital certificate, discussed above. Without authentic and trustworthy certificates, protocols like SSL and S-HTTP offer no security at all.

2.5.8 Secure Electronic Transactions and Secure Transaction Technology

The Secure Electronic Transactions (SET) protocol is an open, interoperable standard for processing credit card transactions over the Internet, created jointly by Netscape, Microsoft, Visa, Mastercard, GTE, SAIC, Terisa Systems, and VeriSign [42]. SET's goal is to allow credit card transactions to be as simple and secure on the Internet as they are in retail stores.

Certificates are heavily used by SET, both for certifying a cardholder and for certifying that the merchant has a relationship with the financial institution. While both Microsoft and Visa International are major participants in the SET specifications, they currently provide the Secure Transaction Technology (STT) protocol, which has been designed to handle secure bank payments over the Internet. STT uses DES encryption of information, RSA encryption of bankcard information, and strong authentication of all parties involved in the transaction.

CHAPTER THREE

LITERATURE REVIEW

AUTHENTICATION PROTOCOLS

3.1 Overview

Wireless Local Area Networks (WLAN) are widely used in our everyday life. Users are adopting the technology to save time and cost of running wires in providing high speed network access. The IEEE 802.11 is the most widely used WLAN standard, but it suffers from the weakness of its security protection [9].

The 802.11-1999 authentication mechanism works at the data link layer (MAC layer). Two authentication methods exist: open system authentication and shared key authentication [10]. Open system authentication is in principle a null authentication scheme and accepts anyone that requests authentication. Shared key authentication is a challenge-response authentication based on a shared secret. The user equipment (UE) sends an Authentication request to the Access Point (AP). The Access Point sends a chosen plaintext string to the station and the station responds with the WEP encrypted string. If the string is correctly encrypted the AP sends an authentication message to the UE to indicate that the authentication was successful. The standard allows for up to four keys in a cell but in practice all communication parties in the cell share the same key. The authentication is not mutual, only the UEs are authenticated.

Shared key authentication is very weak. An attacker that listens to a successful authentication exchange will have all elements that are needed to successfully perform an authentication of his/her own, even if the shared key is unknown. Today shared key authentication is not considered useful [11] [19].

3.2 TVWS Authentication Protocols

3.2.1. IEEE 802.11 WEP

Wired Equivalent Privacy (WEP) was an encryption algorithm designed to provide wireless security for users who implement 802.11 wireless networks. WEP uses the RC4 stream cipher, combining a 40-bit WEP key with a 24-bit random number known as an Initialization Vector (IV) to encrypt the data. The sender XORs stream ciphers with the actual data to produce cipher text. The packet, combined with the IV and with the cipher text that sent to the receiver. The receiver decrypts the packet using the stored WEP key and the attached IV [1].

WEP has several security issues, such as weak key usage, reuse of initial vectors, exposure to replay and packet forging and problems with the encryption algorithm. Other than that, key management and updating is poorly designed in WEP [2]. These keys are weak and can be cracked, even in few hours or minutes using freely available software. The ability to modify packets, even without knowing the encryption key allows an attacker to modify or alter packets undetectably.

WPA was introduced to solve the problems of WEP without changing the existing hardware. WPA keys can go up to 256 bits, but not transmitted over the air to protect against packet monitoring. Compared to RC4 encryption, TKIP encryption allows better message security with the assistance of Message Integrity Check (MIC) [3]. This avoids packet forging and removes replay attack by utilizing a new IV sequencing discipline. Meantime, re-keying mechanism invalidates reusing encryption and integrity keys by an attacker to decrypt the messages. Due to the weakness of encryption algorithms, WPA is vulnerable to key-stream recovery attack and message falsification. WPA2 in other words is vulnerable to Denial of Service (DoS) attacks, such as data flooding, frequency jamming and Layer 2 session hijacking. Additionally, the control packets are not protected and open to DoS attacks. Weak authentication for control frames makes the MAC addresses are possible to be spoofed. However, WPA and WPA2 provide considerably good security in today’s wireless networks even though they were already cracked [2]. Fast Initial Authentication (FIA) proposed by [4], they claimed that FIA can provide three services, first it can support for a large number of simultaneously entering mobile STAs in an ESS, secondly support for small dwell time (due to high velocity and small cell areas) in an Extended Service Area (ESA), thirdly it can provide secure initial authentication.

3.2.2 IEEE 802.1X

The IEEE 802.1X standard defines a mechanism for port-based network access control to provide compatible authentication and authorization mechanisms for devices interconnected by various 802 LANs. It could also be used to distribute security keys for 802.11 WLANs by enabling public key authentication and encryption between access points (APs) and mobile nodes (MNs). In 802.1X, the port represents the association between MN and AP. There are three main components in the 802.1X authentication system: supplicant, authenticator, and authentication server (AS) Figure 1 depicts a typical 802.1X message exchange with both the supplicant PAE and authenticator PAE state transitions.

Diameter protocol is another framework for carrying authentication, authorization and accounting information between the network access server and the AAA Server [11]. Nowadays the application of RADIUS protocol is most widely used than Diameter, but for compatibility reasons the capacity of transition from one protocol to the other is indispensable.

The Remote Authentication Dial In User Service (RADIUS) protocol was originally defined to enable centralized authentication, authorization, and access control (AAA) for SLIP and PPP dial-up sessions [7].

3.2.3 IEEE 802.11i

To enhance the security performance in WLANs, IEEE launched IEEE 802.11i standard [12]. This standard allows wireless to have secure communication through the validation process, called authentication, applied to both user and device [13].

Authentication should be done each time a user wishes to gain connection to the network. However the validation process will lead to a delay when the user moves from one access point (AP) to another in order to maintain its connection. As mentioned in [14] the delay should be minimized with a tolerance of less than or equal to 50 ms to prevent packet drop particularly in multimedia application.

However, IEEE 802.11i, which is relying on 802.1x and extensible authentication protocol (EAP) protocol to provide its authentication mechanism [15], requires a considerable amount of time to exchange the authentication’s packets.

802.1X ties a protocol called EAP (Extensible Authentication Protocol) to both the wired and wireless LAN media and supports multiple authentication methods, such as token cards, Kerberos, one-time passwords, certificates, and public key authentication.

WPA-PSK [2] is based on IEEE80 2.1X and EAP protocols, therefore it does not require any special hardware to work, any wireless card supporting IEEE 802.11i (WPA2) standard is suitable. WPA-PSK is implemented in such a way that both STA and AP can check that they’re agreeing on a non-forged RSN IE and therefore they are using the most secure available protocols. As in [17] indicated that access points (AP) vulnerabilities to DoS attacks by their experiments, and only a few AP devices have anti-DoS protection schemes.

Analyzing 802.11i with respect to several security considerations present in [18], pointing out that an adversary can launch DoS attacks by forging unprotected management frames and control frames. In [19] the authors summarize the current research findings and have a systematic analysis of 802.11i DoS attack threat in the physical layer and MAC layer.

Because of the openness of Internet, it is difficult to eliminate such kind threat completely. All the countermeasures aim to reduce the influence of DoS attacks on networks. Researchers propose many schemes against DoS attacks on authentication protocol, while most of them are not suitable for wireless resource-limited circumstance. Thus, how to enhance anti-DoS ability of 802.11i is an urgent issue.

In [20] the researcher are focuses mainly on request and authentication request flood DoS attacks. They proposed a new client-puzzle based DoS-resistant scheme of IEEE 802.11i wireless authentication protocol to improve the DoS-resistant ability of IEEE 802.11i wireless networks. The difference between their work and traditional client puzzle scheme is employing beacon frame to distribute the parameters of cryptographic puzzle on the basis of hash function. By listening on the wireless channels to get the AP's beacon frame, users construct a puzzle with the seed in the beacon frame and solve it by brute-force computation. The answers to the puzzle and other parameters constructing the puzzle are sent by authentication request. Whether providing the association to a station depends on the verification of puzzle by AP. This method keeps a good resource balance between the AP and stations, reducing the affection of resource depletion attack and the potential resource-exhausting in traditional client puzzle scheme.

There was considerable ongoing research to address the security issue in 802.11 WLANs. The vulnerabilities of the 802.11 management and media access services, and different types of Denial of-Service attacks possible on 802.11 networks are described in [31]. Where, he suggested the implementation and evaluation of non cryptographic counter measures that can be implemented in the firmware of the MAC hardware. Meritt Maxim et al [32] present a review of the threats that are unique to a wireless environment especially the problems that occur due to inter cell roaming. A detailed description of MiM attack and its ramifications have been discussed in [33], by suggesting the usage of a VPN (Virtual Private Network) and the necessity that all the traffic requires to pass through the VPN to a trusted, secure, wired network. Joshua Wright [34] reviews the techniques attackers utilize to disrupt wireless networks through MAC address spoofing.

A technical comparison between TTLS and PEAP has been done in [35]. TTLS has a number of slight advantages over PEAP and offer a slight degree of flexibility at the protocol level. Other comparison between the different EAP products has been presented in [36]. The use of mutual authentication would secure the wireless network during the phase of authentication between the AP and the Mobile clients. There is still need to provide secure wireless communication channel over the Internet using secure SSL according to [37].

3.2.4 IEEE 802.16

The standard IEEE 802.16e [23] use the new privacy key management protocol PKMv2 to improve the security performance, in which the EAP (Extensible Authentication Protocol) are introduced into IEEE 802.16e. EAP is an authentication framework widely used in WLANs [24]

WiMAX, both physical and MAC layers have risk of threats like jamming [25] and denial of service respectively. But there are no efficient procedures to deal with threats posed at PHY layer of WiMAX so, the emphasis of WiMAX security is entirely at the MAC level [26]. MAC layer security threats and vulnerabilities of the WiMAX networks [27] are discussed below:

There are numerous considerable deficiencies of 802.16 security implemented at the MAC layer. In order to establish secure connections between BS and SS, 802.16 [21] utilize a sequential two-way communication for controlling, authorization, and authentication. There are many problems faced during the connection face. One major problem is that, while setting up the primary connection, management messages by MAC are launched in plain-text and are not well authenticated. Therefore there is a strong possibility that they get hacked and can give way to other attacks. Second problem is that, 802.16 [22] uses X.509 certificate, the standard for Primary Key Identification (PKI), to identify a legitimate SS. a SS's certificate is provided by the manufacture and persistent on the machine making it possible for the attacker to steal it and exploits it.

WiMAX supports mutual authentication based on the generic EAP [28] and also supports its variants like EAP- TLS (transport layer security) (X.509 certificate based) and EAP- SIM.

Implementation of the security solutions against most of the threats related to MAC layer proposed in [29]. They claimed that threats like Rouge BS and Replay Attacks have totally been removed. And also provide the protection of the data by using their algorithms.

In [30] proposed three mechanisms to reduce the authentication cost of the WiMAX Network entry process which in turn enhances seamless handoff. These mechanisms can be implemented separately or combined to cross cut the desired level of authentication cost.

3.2.5 IEEE 802.22

A well-known issue in modern wireless communications is spectrum scarcity. To solve the dilemma between the increasing bandwidth demands and the actual underutilization of spectrum resource [40], the Federal Communications Commission [41] has allowed unlicensed users to opportunistically access the temporarily unoccupied television (TV) bands, namely TV white spaces (TVWSs) [42]. IEEE 802.22 is the first standard protocol for enabling the use of the fallow TV bands by the infrastructure single-hop cognitive radio networks (CRNs) with the presence of one base station (BS) that performs spectrum management, which supports the provision of broadband fixed wireless data transmission in sparsely populated rural areas [4,5]. This standard on wireless regional area networks (WRANs) specifies the air interface including the cognitive medium access control layer (MAC) and physical layer (PHY).

In [45] they discovered that the authentication protocol specified in IEEE 802.22 is vulnerable to the interleaving attacks and cannot achieve mutual key confirmation. It is worth noting that this protocol violates the first principle of Anderson and Needham [46], which is ‘sign before encrypting’. The motivation behind this principle is that the signature cannot provide assurance that the signer knows the plaintext in the encrypted message. In this protocol, the CPE can get an AK, which could be used to communicate with BS, but the CPE cannot know whether it is indeed assigned by the BS or whether the BS knows the key. What is worse, an adversary could remove the signature of BS and replace it with the adversary’s own signature. Hence, in this authentication protocol, it is vulnerable to interleaving attacks [47] because of the omission of the identity. And so [45] proposed an Enhanced Certificate-based Authentication scheme (ECA) has been proposed to overcome the vulnerability of the authentication scheme in IEEE 802.22 standard with much less requirements on the computation and communication resources.

3.2.5.1. Non-cognitive security mechanisms

The IEEE 802.22 security sub-layer1 architecture is shown in Figure 3.1, which provides the CPEs with security functions for the payloads and the MAC management messages across the WRAN TVWS. It achieves the security functionality by applying cryptographic transforms to

MAC Protocol Data Units (PDUs) carried across connections between the CPEs and BS. The security sub-layer 1 of the IEEE 802.22 standard has been designed with reference of the privacy key management version 1 (PKMv1) and the PKMv2 from IEEE 802.16-2009 standard [7], which is correlated with the IEEE 802.22 operation and renamed the PKM protocol as the SCM protocol. The SCM protocol provides secure distribution of keying materials from the BS to the CPE and performs mutual authentication between the BS and the CPE. The SCM's authentication protocol establishes a shared secret between the CPE and the BS. The shared secret is then used to secure the subsequent SCM exchanges. The SCM supports elliptic curve cryptography-based authorization for authorizing the CPEs at the time of network entry. The main procedures carried out by the BS and the CPE to perform CPE network entry and initialization are as follows. Firstly, the CPE performs a test by itself and acquires the antenna gain information. Then, it performs sensing and synchronizing to the WRAN services, by which the CPE chooses a WRAN service and acquires the parameters of the downstream and upstream links from the selected WRAN service.

After the BS and the CPE perform an initial ranging, the CPE will precede with security capabilities negotiation. If all required basic capabilities are available in the CPE, the BS authenticates the CPE and performs the key exchange. Otherwise, the CPE cannot proceed to the registration phase.

The security suite consists of the authorization and authentication process. The authorization process is carried out when a CPE enters the network, to make sure that only the authorized device can access the network. The BS is capable of de-authorizing a CPE if the BS finds that the CPE does not contain valid AKs or it is generating spurious emissions. The authorization process also includes a mutual authentication process where both the BS and the CPE can authenticate each other. The security sub-layer 1 has a key management protocol by which the BS controls the process of the distribution of keying material to the CPEs. This keying material is used to protect the MAC management messages and may be optionally used to protect user data

[16]. Additionally, the EAP-based authentication mechanism has been included to enhance the security function of the system. At the capabilities negotiation step of the network entry, if the CPE indicates that it will not support IEEE 802.22 security function, then the step of authorization and key exchange will be skipped. Otherwise, the BS considers the CPE as authenticated and authorizes the CPE to access the network. Without the authorization by the BS, the CPE will not be serviced, and further, the network entry of the CPE is de-authorized, and neither key exchange nor data encryption will be carried out. After the authorization, the traffic encryption keys will be exchanged by using the SCM protocol. Figure 3.2 shows the SCM control management structure.

Figure 3.1 IEEE 802.22 security sub-layers architecture

Convergence Sub-layer

Mac Common Part Sub-layer

Security Sub-layer1

Security Sub-layer 2

Physical layer

M
A
C
P
H
Y

Figure 3.2 shows the SCM control management structure.

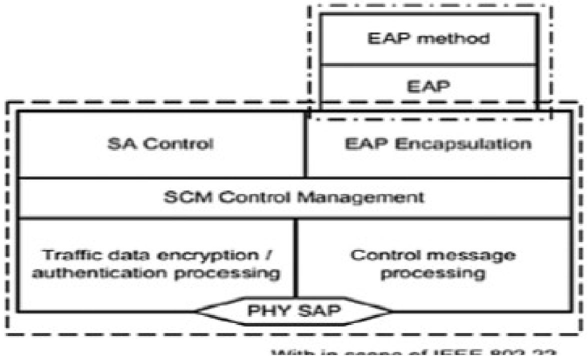

EAP method
EAP
SA Control
EAP Encapsulation
SCM Control Management
Traffic data encryption / authentication processing
Control message processing
PHY SAP
With in scope of IEEE 802.22

Figure 3.1: protocol architecture of IEEE 802.22

3.2.6 PAWS protocol

PAWS is a protocol whereby a master device requests a schedule of available spectrum at its location (or location of its Slave Devices) before it (they) can operate using those frequencies. Some rule sets require a master device to send its registration information to the database in order to establish certain operational parameters [51]. The database may implement device registration as a separate device registration request, or as part of the spectrum availability request. If the database does not implement a separate device registration request, it must return an error with the unimplemented code in the error-response message. The device registration request procedure is shows in Figure 3.3

Registration Request
Registration Response
Master Device
Spectrum database
Figure 3.3: device registration request

3.2.6 .1Device Validation/ authentication

According to PAWS a Slave Device needs a Master Device to ask the Database on its behalf for available spectrum. Depending on the rule set, the Master Device also must validate with the Database that the Slave Device is permitted to operate. When the rule set allows a Master Device to "cache" the available spectrum for a period of time, the Master Device may use the simpler Device Validation component, instead of the full Available Spectrum Query component, to validate a Slave Device.

When validating one or more Slave Devices, the Master Device sends the Database a request that includes the device identifier -- and any other parameters required by the rule set -- for each Slave Device. The Database MUST return a response with an entry for each device to indicate whether it is permitted to use the spectrum. A typical sequence for using the Device Validation request is illustrated in Figure 5, where the Master Device already has a valid set of available spectrum for Slave Devices. Note that the communication and protocol between the Slave Device and Master Device is outside the scope of PAWS protocol.

3.2.6 .2 Using HTTPS over TLS

PAWS use the "HTTP over TLS" as transfer’s mechanism for transferring the data. TLS provides message integrity and confidentiality between the master device and the database, but it needs special adaptation like use of recommended cipher suites and modes of operation. Consequently, the improvement of PAWS security depends on prior relationship between a database and device. In some cases the server may require client authentication, as described in the "Transport Layer Security (TLS) Protocol" [RFC5246], to authenticate the device. When client authentication is required, the database must specify, by prior arrangement, acceptable root Certificate Authorities (CAs) to serve as trust anchors for device certificates. The Database and devices should support "Stateless TLS Session

Resumption" [RFC5077] to enable databases to handle large numbers of requests from large numbers of devices.

A PAWS’s request message is carried in the body of an HTTP POST request and PAWS’s response message is carried in the body of an HTTP response [51]. Authentication process between master and slave devices and is outside the PAWS protocol.

The database authenticates its identity, either as a domain name or IP address, to the Master Device by presenting a certificate containing that identifier as a "subjectAltName" and the client must be able to validate the certificate. In particular, the validation path of the certificate must end in one of the client’s trust anchors, even if that trust anchor is the Database certificate itself.

Although client authentication may be required by specific regulatory domains, it is not required for the core PAWS. TLS provides client authentication when its require by the database but with three conditions, first the database must nominate acceptable Certificate Authorization (CAs) and the master device must have a certificate rooted at one of those CAs. The second condition the TLS client authentication procedure only determines that the device has a certificate chain rooted in an appropriate CA (or a selfsigned certificate), the database would not know what the client identity ought to be, unless it has some external source of information. Lastly authentication schemes are secure only to the extent that secrets or certificates are kept secure. When there are a vast number of deployed devices using PAWS, the possibility that device keys will not leak becomes small. Implementations should consider how to manage the system in the eventuality that there is a leak.

3.3 Non IEEE Authentication Protocols

3.3.1 IPSec

The IPSec system is a set of protocols that facilitate the creation and maintenance of secure IP channels called Security Associations (SAs). This is optional for IPv4, but it is an integral component of IPv6. IPSec consists of three main protocols; Authentication Header (AH), Encapsulating Security Protocol (ESP), and Internet Key Exchange (IKE). Consequently, high speed key exchange is a fundamental requirement in order to support IPSec for applications requiring high speed connections [4]. IPSec is based on a key exchange protocol to make an automatic establishment IKEv2 of security associations (SA). Each SA is maintained between two or more entities which describe the algorithms, keys and other security parameters to be used. To maintain a SA, two phases are required by IKEv2. Phase 1 performs mutual authentication between two parts and establishes an IKE_SA, whereas Phase 2 executes the creation of IPSec_SA between the same pairs. This presents challenges for the implementation of IKEv2 on wireless environments by considering the processor cost and bandwidth limitation. So, there is need to develop a lightweight IKE which can be easily deployed in the target network while maintaining security properties [3].

3.3.2 Fast Initial Authentication (FIA) Proposed Solutions

There are already some proposed solutions that could possibly mitigate the WEB challenges, most of them rely on the existing authentication mechanisms and try to reduce the number of exchanged packets by modifying the 802.1x/EAP authentication process. It is globally agreed that Wi-Fi enabled handsets are much more than the ones that can support 802.1x/EAP. Although some of them support Extensible Authentication Protocol (EAP), most of the Wi-Fi networks still use IEEE 802.11 except the enterprise networks.

Thus, the solution describes a way to improve IEEE 802.11 authentication. There were no significant improvements in generic WLAN access as far as the security in initial authentication is considered. More specifically, there is lot of doubt about the existence of OSA which is considered to be a pre-RSNA authentication process and not acceptable anymore in contemporary wireless networks. The solution of piggybacking authentication information onto association Request/Response messages is also proposed. Finally, another proposal was to append the upper layer information on association Request/Response messages in order to speed up the link establishment process.

The first solution is promising with response to the authentication delay and more or less should be incorporated in the next standard. The only reason that OSA still exists is the backward compatibility with IEEE 802.11 state machine [4]. The second solution seems capable of improving the whole authentication process, though it does not seem to provide a fine grained and performance-wise acceptable solution towards more effective authentication. Finally, the third solution does not really improve the authentication process itself; rather, it is an intermediate approach to accelerate the link establishment delay. By the time mentioned, WEP and WPA security was already broken [6]. Consequently, there is a demanding requirement for security in contemporary wireless networks. Despite that, EAP authorization framework is not in the scope of this research.

To a chive this three goals this method needs some modifications to the 802.11 standards which can possibly be integrated to or form a new amendment of the existing standard.

An efficient approach to FIA is identifying Host Identity Protocol HIP-WEB which is a light-weight authentication and key management protocol on 802.11 wireless networks. HIP-WPA utilizes HIP as a key management scheme which was initially designed to provide end-to-end authentication and key establishment. HIP introduces a new namespace for host identifiers. Thus, host identity can be represented either by a Host Identifier (HI) or by a Host Identity Tag (HIT).

HI is a public key of an asymmetric key-pair. However, HI is not suitable to serve as a packet identifier since the length of the public key can vary. HIP Base Exchange (HIP-BEX) uses a Sigma-compliant 4-way handshake in order to establish a Diffie-Hellman (DH) key exchange and a pair of IPsec Encapsulated Security Payload (ESP) Security Associations (SAs) between two entities; the Initiator (I) and the responder (R) [5].

Host Identity Protocol Diet Exchange (HIP-DEX) is a secure Authentication and Key Management (AKM) scheme that fits into many constrained applications, due to enhanced security it provides with Elliptic Curve Cryptography (ECC) and comparatively less overhead [7]. The HIP-DEX module was developed in C++ with the support of OpenSSL version 1.0.1c which includes ECC point multiplication for ECDH handshake [8].

There are two basic security standards to enable WLANs authenticate their user, namely open system and shared key [12]. However, none of these standards were able to protect the communications. Several holes have been found and attacks have been launched against these standards. In an open system, there is no mechanism that available for user to authenticate access point (AP), the user has to trust the AP and ignore the possibility of a fake AP with the same service set identifier (SSID) to the AP that he wants to connect. In 2000[13] was proven that shared key, also called Wired Equivalent Privacy (WEP), present a null authentication process. The presence random 24-bit string called the initialization vector (IV) along with pseudorandom number generator (PNRG) to generate a challenge text [13] which was at the beginning intended to provide strong authentication mechanism had become a hole for security. In [14] several attacks against WEP key implementation were explored as well, such as FMS (Fluhrer, Mantin and Shamir) attack in 2001, KoreK attack in 2004, and PTW attack by Tews, Weinmann, and Pyshkin in 2007.

3.3.3 LEAP

LEAP is Cisco's lightweight EAP, which is widely deployed in today's WLANs. (Cisco Systems, 2003). With this method, the RADIUS server sends an authentication challenge to the client, the client uses a one-way hash of the user-supplied password to fashion a response to the RADIUS server. Using information from its user database, the RADIUS server creates it is own response and compares that to the response from the client. When the RADIUS server authenticates the client, the process repeats in reverse, enabling the client to authenticate the RADIUS server. After the completion of this process, an EAP success/failure message is sent to the client and both the RADIUS server and the client derive the dynamic WEP key.

LEAP's use of unencrypted challenges and responses does leave it open to online (active) and offline (passive) dictionary attacks.

Unlike online attacks, offline attacks are not easily detected. With Cisco's LEAP, security keys change dynamically with every communication session, preventing an attacker from collecting the packets required to decode data. (Cisco Systems, 2002).

3.3.4 TLS

Transport layer security (TLS) protocol is based on SSL v3.0, which is used in most web browsers for secure web transactions. SSL was developed by the Netscape Communications Corporation in 1994 (Cisco Systems, 2003) to secure transactions over the World Wide Web. Soon after, the Internet Engineering Task Force (IETF) began work to develop a standard protocol that provided the same functionality. They used SSL 3.0 as the basis for that work, which became the TLS v1.0 protocol.

TLS provides a very secure mutual authentication protocol that overcomes the shortcomings of the password-based and challenge-based methods. The TLS protocol is composed of two layers: the TLS record protocol and the TLS handshake protocol (Ma and Cao, 2003).

3.3.5 EPA-TLS

Is considered as the best available (security wise) authentication method for WLANs but the main concern with this method is that it requires Public Key Infrastructure (PKI) because it uses a digital certificate to authenticate the server to the client; the server requires the client to send it a digital certificate if it wishes to authenticate the client.

3.3.6 TTLS

To overcome complications associated with the use of PKI in the client, tunnelled transport layer security (TTLS) was developed. EAP-TTLS offers strong security during authentication while accommodating existing end-user working methods (user ID/password), thus avoiding the complexities of PKI in the client's site.

3.3.7 PEAP

Protected EAP (PEAP) is an authentication type that is designed to allow hybrid authentication. While for server-side authentication PEAP employs PKI, for client-side authentication, PEAP can use any other EAP authentication type.

3.4 KEY Management

Cryptographic key establishment is a fundamental requirement of secure communication that supports confidentiality and authentication services, and is crucial for preserving user privacy [49] [50] [51] [52]. Achieving fast and reliable key agreement between wireless communication parties using a shared channel is challenging but desired due to its efficiency. The procedures like symmetric [55, 56] and public key infrastructures are covered by a lot of studies. The key generation and key distribution have three drawbacks. In network security, symmetric key encryption methods are commonly used in order to generate and exchange a secret key between the sender and the receiver. The main drawbacks of this method are that, the attackers might access the transmitted data and use it to obtain the key and even generate new keys. Also, the process of generating the key is inefficient and time consuming. In the next section the research present the literature about these problems and the efforts has been done to solve them. IEEE 802.22 is the first standard protocol for enabling the use of the fallow TV bands by the infrastructure single-hop cognitive radio networks (CRNs) [50, 51]. For this reason most of the researcher studies the key generation and key distribution from this point of view. There is a few research conducted to solve these problems in TVWS database systems. One recent trend in this regard is to allow two parties to build keys separately using inherent wireless channel properties [51].

3.4.1 KEY Distribution and Data Exchange

EAP-TTLS-ISRP method proposed in [57], which embeds the transmission of security messages in a secure tunnel. This authentication method is proposed for a single EAP based authentication to achieve both user and device authentications between Mobile Station (MS) and Authentication Server (AS) by using strong and fast authentication methods.

To perform the user authentication and key exchanges, we use EAP-ISRP method which is one of the strongest password based methods; and it has been improved in [58] to overcome the problems of EAP-SRP, such as the overhead by reducing the no of exchanged messages.

The most sensitive protocol against key generation time is real time communications applications such as streamed media, which requires secure concurrent connections, are driving the need for high-speed key exchange [60]. KEEP is a fast secret key extraction protocol proposed in [54], which uses a validation recombination mechanism to obtain consistent secret keys from CSI measurements of all subcarriers.

In [62] propose a time constraint three-party encrypted key exchange protocol in which both two parties/clients only act as the roles within the relationship of exchanging secure messages constrained by the intersection time periods. The technique of time constraint is a trusted server generates a serial of secure exchanging messages, which are used to generate multiple session keys by each party individually. Each session key corresponding to one time period which is belonging to the intersection set from both two parties' requesting time bounds. Only both two authorized parties/clients can communicate to each other within the intersection set of time bounds. It can get better performance on when three parties do many times for traditional 3PEKE protocol. The problem with this method is uses a trusted third party to generate the key which is not applicable in TVWS. **,** Elliptic Curve Diffie-Hellman (ECDH) Algorithm key agreement scheme is employed with smaller key sizes proposed by [63]. They claimed that it result in faster computations. Their algorithm is used to split exponents for fast exponentiation has been implemented to speed up and increase the randomness of key generation. A fast symmetric key distribution technique with additional security services is presented by [64]. The aim of their proposed technique is to improve the conventional Needham and Schroeder five-message protocol in four aspects to reduce the key generation time.

In [57] present EMBGK - Energy and Mobility based Group Key to reduce the end to end delay. This method is suitable for mobile nodes and group key generation.

ECC based key management is used to further strengthen the symmetric block cipher. This mechanism proposes the faster computation of the algorithm with smaller key size. High level security of ECDH based security has the difficulty of discrete logarithm problem for breaking the keys [61]. This study proposed, implementation of ECDH key exchanging mechanism in real time using the open source PBX software Asterisk and IP phone has been carried out.

3.4.2 KEY Generation Time

Generating symmetric keys individually on different communication parties without key exchange or distribution is desirable but challenging [54]. They propose a fast secret key extraction protocol, called KEEP. KEEP uses a validation recombination mechanism to obtain consistent secret keys from CSI measurements of all subcarriers [54]. Self-Organizing Maps (SOM) method proposed. This method is simple to apply, but it takes time to generate the map and also the changing in the map is not easy.

In [61] they specify the necessary to improve the strengthening of initial key exchanging mechanism. ECDH is another key agreement protocol that allows two parties to generate a shared secret key that can be used for private key algorithms. In this system, both parties exchange some public information to each other. Here the Public information is elliptic curve parameter, domain value, Public key. Using this public data and their own private data these two parties calculates the shared secret value. Any third party cannot calculate the shared secret from the available public information without knowing the private data value.

3.4.3 Generating New Key Methods

is when the attacker knows the key; they will be able to generate more key once the life time of the existing key is expired. Most of the security protocols consider the security to be vulnerable when the key is broken, consequently that particular key will be disabled or cancelled. However, it is important to ensure that the attackers are unable to use that key to generate more keys. In [65] described a method to generate a numbers of random keys, but they didn’t explain a method to change the key incase of the attacker break the key.

CHAPTER FOUR

METHODOLOY

4.1 Overview

The name geolocation database is used to emphasize the importance of geographical information in the controlling of the utilization of white space spectral resources. One of the main roles of the geolocation database [52, 53] is to protect incumbent systems, search for available white space frequencies for white space devices, and possibly also control the interference between them.

The accuracy and precision of database algorithms are essential in determining frequency channel and transmitting power. The closer is the database output to optimal value for the given location input; the better is the white space utilization. The optimal value means that the white space communications uses the maximum allowable transmission power, while incumbent systems can still be operated normally [54].

Security issues in geolocation database must be taken into account from the different point-of-view than in traditional wireless communication. This is due to the Internet access between the WSD and the database. The device and the database must perform mutual authentication. Which means the database has to know if the device is allowed to access white space, and the device has to know which databases are certified by regulatory authorities? Another security issue is, the data transfer has to be encrypted and the integrity of geolocation data has to be secured. Moreover the database may be also a target for Denial of Service (DoS) attack. If information security fails, it can cause severe interference to incumbent systems, in addition to white space network, due to the incorrect or inaccurate information on the allowed white space areas or maximum transmitting powers.

In the literature, the vulnerability of TV broadcast network in the case mis-behaving TV white space system has been a concern as the TV is the main source of information distribution in crisis situations. A main consideration in publications with security considerations for cognitive radios and dynamic spectrum access have considered DoS attacks towards secondary networks, and also secondary network as a tool for DoS against primary networks [55-59]. In [59] study includes also the analysis for white space system susceptibility for man-in-the-middle attack. In [55] analyses the fair distribution of spectrum resources between white space devices have been performed.

To overcome these security issues the mutual authentication is prevent unauthorized users to utilize the network service. Also the authentication prevents the DOS attack and the misused of the available channels. Another advantages of the authentication protocol is allows the users and the Database Server DS to generate and exchange a shared secrete key in secure manner, to use it in encrypt/decrypt the transmission data.

Wireless technology is nowadays on high demand and this makes it hard to secure the communications, and so the geolocation database has become the best way of accessing the free channels. The database is used to store user's data and all the available channels and the information related to these channels such as frequencies, interference and authorizations.

The Protocol to Access White Space (PAWS) was defined by Internet Engineering Task Force (IETF), to be the first standard protocol for TVWS. Also cognitive radio systems is the wireless regional area network operating in television white space (TVWS) spectrum, which has been specified by IEEE 802.22 standard [50]. As mentioned in chapter 3 all the available authentication's protocols apply the authentication from different type of views, either from network view or from database point of view. The next section demonstrates the proposed mutual authentication protocol which utilizes the availability of the database and the network link together in one protocol to authenticate the entire link between the users and the DS. This protocol includes the authentication process and key management process in terms of key generation and key distribution. Finally the new proposed protocol introduces new integrated security framework that ensures closing the gap between security sublayer at lower layers and upper layers at session layer and above, by utilizes the availability of the database and uses it in the authentication protocol. The new protocol is supported with all recommended functions from various standards. Figure 4.1 shows the proposed protocol structure.

4.2 The Proposed Authentication Protocol

The Database Server (DS) has all the information about the master devices (mode II device) and the user's devices (mode1 device) and this information is stored in the database. To utilize this database in the security authentication a modification to the database should be made by adding a new column and put a random number in this column [1] as depicted in table 4.1. Then the protocol divided the authentication into two phases: Phase1 the Master authenticates itself with the Database Server (DS), and the second phase the mdoe1 device (user device) authenticate itself with the DS through the master (mode II device) as depicted in figure 4.2.

Figure 4.1: a protocol to authenticate the entire link between CPE and DS

CPE Mode I

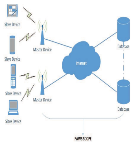
BS

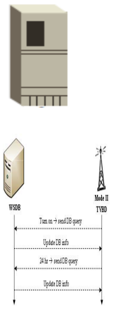
Covered by database security
Covered by IEEE protocols
The protocol must authenticate the entire link
Authentication security

Backbone Channel
Or Internet access

Table 4.1: the database table

Column Name	Type	Inf0_Number
DeviceType	Fixed	9658551
DeviceName	String	7435679
DeviceSerialNumber	String	62248537





Phase1: Master authentication
Phase2: User authentication
Database Server
Master Mode II
User Mode I

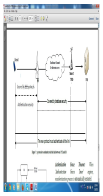
Figure 4.2: The proposed protocol structure

4.2.1: Phase1 Authentication between the Master and DS

Phase1 authentication process contain 12 steps, from step one to step 8 the master should complete the authentication with the DS, starts with sends its certificate to the DS and immediately sends a registrations request message (Message 1 and 2). When DS receive these messages it verify the master certificate and if it is accepted then the server replies by sending its certificate (Message 3) and pick one of the random number from the master's table as a server's challenge question and send it back to the master (Message 4). When the master receive these messages it first verify the server's certificate and if it is accepted then the master replies with confirmation message (Message 5).The confirmation message is divided into two parts; the first part is the answer's of the server's challenge question by find the value of the random number from the master's table, and in the second part the master must pick one of the random number from the DS’s table as a master’s challenge question and send it back to the DS figure 4.3 show the message 5 structure. The DS’s table and master’s table are shared only between the master device and the server device, so when the DS receive these messages it first verify the answer of the server’s challenge question and if it is true that means the identification of the master is verified because there is no one can get the correct answer except the master itself. So the DS replies with conf_reply message (Message 6) which answer the master’s challenge question from the DS’s table. When the master receive this message it verify the answer of the challenge question and if the answer is true that means the server’ identification is verified (because the DS’s table is shared only between the master and the DS) then the master replies with Ser-Auth-success(Message 7), and the DS replies with Auth-comp-success fully message(Message 8). Figure 4.4 shows the protocol steps.

Answer server’s challenge question
Master’s challenge question
Figure 4.3: message 5 the confirmation message from master to DS

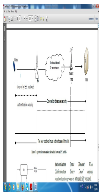
At this point the master and the DS are mutual authenticated, then the master send Ava_Chanell-req message (Message 9). And also send a list of allowable user’s request message message10 (the list of the users that the master is allows to authenticate). The last message is very important to complete phase2 authentication. And the DS replies with the available channels that the master can use in this area (Message 11) and a list of the registered users the master is allow to authenticate them (Message 12) Figure 4.6 shows this steps.


Backbone Channel
Or Internet access
Covered by IEEE protocols
Authentication security
Covered by database security
The new protocol must authenticate all the link

1
2
3
4
5
6
7
8

Figure 4.4: master authentication steps

After message 12 is received successfully this means the master is mutual authenticated with the database server and be able to authenticate some users (in the list) and the users will start to authenticate themselves with DS through the master. Figure 3.5 shows the flow chart of the authentication process.



9
10
11
12

Figure 4.6: list of avl_userTo Authenticat Request

MSG1: master DC & REGREQ message

MSG5: available channel request & user’s list
MSG2: server DC & challenge number

MSG6: available channel & list of users resp
MSG3: the value of challenge question & challenge number for the server

MSG4: authentication success & the value of the challenge question

Figure 4.5: master server authentication flowchart

Start

Server accept master's DC
Master accept server's DC
The master value is correct
The sever value is correct

END

Yes
Yes
Yes
Yes
No
No
No
No
Master sends MSG1

Server sends MSG2

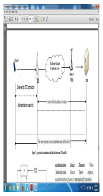
Server sends MSG4

Master sends MSG3

Master sends MSG5

Server sends MSG6

-


Covered by IEEE protocols
Authentication security
Covered by database security
The new protocol must authenticate all the link

Authentication Grace Timeout: When 'Authentication Grace Timer' expires, reauthentication process is automatically restarted.

9
10
11
12
Figure 4.6: list of avl_userTo Authenticat Request

4.2.2: Phase2 Authentication between the Master and Users

When the mobile Subscriber (MS) sends an authentication request message (Message 1) to the master then the authentication in this phase can run into two cases

Case1: If the user device mode1 is already registered in the list that the master was received in message 12, then the master replies by sending message 1.1 which contain the master’s certificate and the master’s challenge question to the user, and continue the same authentication process as in phase 1.

Case2: if the user is not in the list (the master was received in message 12) then the master must sends user’s authentication request message to the DS (Message 2). And the DS replies with the user’s data to the master (Message 3). Then the master updates it’s database and then forward the challenge question to the user (Message 4). After this the master and the user can continue the authentication process as in phase1. Figure 4.7 show the cases.

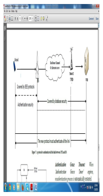


Figure 4.7: shows user case authentication

1.1

1
2
3
4

After these procedures are completed that means the user becomes securely mutual authenticated with the DS and this achieve the first and second goals of this research. And now the user and DS must continue in the authentication protocol to generate and exchange a shared secrete key to use it in the encryption/decryption of the transfer's messages during the communication.

4.3 Key Management Method

As stated in chapter one the key management process will start after the authentication process to complete the authentication protocol. And the main drawbacks in the available key management process are, the master/DS and the user exchange some data that the attackers might access and use it to obtain the key, and even to generate new keys when the key life time is expired. Also, the process of generating the key is inefficient in time consuming.

To overcome these problems this section presents two solutions for the key management method. The first solution generate one key in the session and the second solution is aims to generate a number of reliable keys based on mathematical calculations without exchanging any data has a relation with the key generation, and the calculation time must be fast.

4.3.1 Generate a Single Key Solution

This section presents the proposed new key generation protocol. This protocol aims to generate a secure key with different size, based on mathematical calculations. The basic idea of the key generation protocol depend on the pre-shared secrete key. So the proposed protocol tried to utilized this key and extended it into large size big than the key length.

When the master want to generate and exchange the key with the Database Server (DS), he just generate two random numbers (L, KL) the number L specify to the DS how many bits need to be shift from the left side of the pre-shared key, and KL specify the key length. Then the master sends both numbers in key generation message (key_genMsg) to the DS. If the key life time (T) and the grace time (G) is not fixed, so the master has to specify them in the key_genMsg before sending the message.

key_genMsg = (L, KL, T, G)

When the DS receive this message just shift L bits from the left side of the pre-shared and calculate the key uses the KL and setup the key life time and the grace time. And send successful message as response, and the master reply by sends Ack message.

The last two messages must not encrypt by the key to avoid man in the middle (MitM) attack. Before the key life time is expired the master must sends new key_genMsg to generate a new key. If the server suspect there is an attacker's message then the DS discard the available key and send a request changing key message, and the master must reply by sending new key_genMsg. Figure 4.8 specify the key generation scenario.

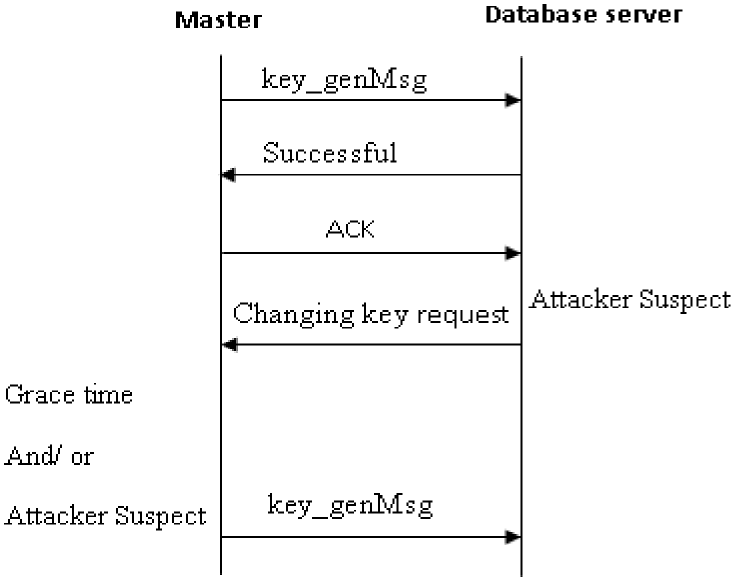


Figure 3 shows the protocol process

Figure 4.8 generate a single key protocol

The generation of this key will be very fast because shifting the bits from the pre-shard key will not take time so this protocol will be faster than the available protocols.

4.3.1.1 An Example of Generating a Single Key

To explain this idea supposes that the pre-shard key size is 512 bit. When the master want to generate a new key it just need to generate a random number (L) and specify the key length (KL) less than 512. Now supposes that L =35 and KL =64. Let the key life time (T =24 hours, and Grace time G = 1 hour) so

key_genMsg = (35, 64, 24, 1)

When the DS receive this message it shift 35 bits (start from 0) from the left side of the pre-share key and start to get the key 64 bits starting from bit number 35 until the bit number 98 and then setup the key life time and the grace time.

35
98
L
Key
0
34
Pre-shared Key

Figure 4.9 explain the key generation example.

511

Figure 4.9 explain the key generation example.

4.3.2 Generate a Number of Keys

To generate a number of key this protocol runs in two steps

4.3.2.1 First Step (Initialization Step)

In the first step which called pre-establish configuration, the DS/master and the mode I device must agree with the following agreements:

- Store a message with length (m) instead of pre-share key.
- The number of keys (n) they are willing to generate in the session.

- The length of the keys (L), this research considers the keys lengths are equal.
- The length of m must be with the following condition $m >> n*L$.
- The keys must divided into (i) groups, with the following condition $n_1+n_2+n_3….n_i = n$.**4.3.2.2The Protocol Steps**

When the master (mode II) wants to generate the keys he must generate a random numbers L_1, L_2, L_3 with the following condition: $L_1, L_2, L_3 < m$. L_1, L_2, L_3 tells the database server to shift (L_1) bits from the left side and start to calculate the n1 numbers of keys, and shift (L_2) bits from left side to calculate n_2 numbers of keys, and shift (L_3) bits from right to calculate the n_3 numbers of keys, with the following formula: the first key (k_1) in group n_1 keys start after shifting (L_1) bits from the left side of the message with the length of L, and the first key in group n_2 start after shift L_2 bits from the left side and with the same length and so on.

Suppose the number of keys groups are 3, so the calculation message will be like this

$$\mathit{CalcMessage} = ((L_1, n_1), (L_2, n_2), (L_3, n_3))$$

At this point the master has n numbers of keys. So he has to select one key to be the first key to use and also specify the life time T and the grace time GT for this key.

$$\mathit{keySelectMessage} = (N, T, GT)$$

Where N is the number of the key the master is selected, T is the life time, and GT is the grace time.

When the protocol is started and after the authentication process is completed the master need to send both CalcMessage and keySelectMessage in one message called KeyGenMessage like this:

$$\mathit{KeyGenMessage} = ((L_1, n_1), (L_2, n_2), (L_3, n_3), N, T, GT)$$

When the database server receives this message he should replies by sending KeyGenMessage-accepted as an acknowledgment of success full key generation.

So before the life time is finished the master should send only a new keySelectMessage with the new values of N and T and GT (if the life time and grace time are not constant). If the T and GT are constant the master should send only the new key number he wants to chose, unless he wants to change the life time and/or the grace time. During the key life time the database server can send change- key- request message in case of attacker suspect. And the master should response with keySelectMessage.

4.3.2.3 Protocol Analysis

When the master sends keySelectMessage that means the master avoid to send any data the attackers can use it to generate the key, and this way solve the first problem in key generation. This protocol can be broken under one difficult condition, when the attackers can get both the message m and the keySelectMessage and this very difficult because the message m will never transmit in the network, so the attacker need to get inside the server 's database or the master's database to get this message. But the protection of the data inside the devices is outside of the protocol scope. If the attacker can break the keySelectMessage he can't drive the (m) message because there is no relation or formula to calculate m. This means the second problem in key generation is solved.

So suppose the attacker can break the key (by luck or any way) he can use this key if and only if the life time is not finished yet and in the same time the master is not working during this period of time. In the first case (life time not finished) the attacker will not know the life time and even if he knows he can not do any things about this and he can not generate another key. For the second case (the master is not working) when the master is working the database server will receive two messages in the same time with different properties and the server will suspect there is an attacker, so the database server will response with changing- key- request message sand will conceder this key is not valid and this force the attacker to stop using the key, because the master can generate a new key but the attacker can not. The last problem solved in this method is the time to generate the key becomes very short, because it merely shift the bits in the message m and get the key, there no difficult calculation. So the suggestion is to minimize the key life time as much as possible because we have so many keys and it's very easy to generate another set of keys.

4.3.2.4 An example for This Method

Suppose we have a key with length of (L = 16) bits and we want to generate (n= 10) keys.

The first step is to generate m message of 512 bits so that

m>> n * L

This information shall be distributed between the master and the database server before the protocol start with the following steps:

1. The master generate randomly n1, n2, n3, L1, L2, L3 and N

Suppose that n1 =3, n2=5, n3=2, L1=55, L2 = 247, L3=42, N =7, life time T =24 hour and grace time GT = 30 (T and GT is not a big issue in our protocol) so that

N < n and n1+n2+n3 =n and L1, L2, L3 <L.

2. The master sends KeyGenMessage to the database server

KeyGenMessage = ((55, 3), (247, 5), (42, 2), 7, 24:00:00, 00:30:00)

Then both the master and the database server will calculate the keys, as the algorithm stated K1 start after shifting L1 bit and with the length of L. Figure 4.10 shows the key map inside the m message and figure 4.11 explain the algorithm steps.

L1
K1

K2
K3
n1
0
55
70
86
102
L3
K10
K9
n3
451
465
480
512
L2
K4
K5
K8
n2
247
279
263
295
K6
K7
311
327

Figure 4.10: the keys generated

Key management algorithm

<u>Notice</u> *: Step1, the Specification of the values of m, n, L is outside the protocol*

1. Set the value of message (m), the number of keys (n), length of k (L) with the condition that

$$m >> n*L$$

2. The master generate random numbers for N,n1,n2,n3,l1,l2,l3 with the following conditions

i- N <= n

ii- n1+n2+n3 =n

iii- L1,L2,L3 < m

iv- L,$L_{2\ \text{is}}$ starts from the left side , L_3 starts from the right side

3. The master specifies the life time (T) and the grace time (GT) if they are not specified before the protocol started.

4. The master sends the KeyGenMessage to the database server.

5. the database server send ACK message

6. If suspicious accurse then the database server must send key changing request and the master should reply by sending keySelectMessage.

7. If the life time is finished before the master sends KeySelectMessage then the master should re-authenticate.

8. If the master uses all the n keys go to step2.

Figure 4.11 specify the key management algorithm

CHAPTER FIVE
RESULTS AND DISCUSSION

5.1 Overview

This chapter specifies the simulation topology, environment and setup. This research uses two types of simulation's environment. The first simulator is the AVISPA project simulator which is simulator specialist in security protocols measurement. The second simulator is OMNeT ++ 4.6 which uses to measure the simulation time in the proposed protocol and compare with the IEEE 802.22 protocol.

5.2 The AVISPA project

AVISPA stands for **A**utomated **V**alidation of **I**nternet **S**ecurity **P**rotocols and **A**pplications.

AVISPA aims at developing a push-button, industrial-strength technology for the analysis of large-scale Internet security-sensitive protocols and applications. This technology will speed up the development of the next generation of network protocols, improve their security, and therefore increase the public acceptance of advanced, distributed IT applications based on them.

The AVISPA will achieve this by advancing specification and deduction technology to the point where industry protocols can be specified and automatically analyzed. A central aim of the project is then to integrate this technology into a robust automated tool, tuned on practical, large-scale problems, and migrated to standardization bodies, whose protocol designers are in dire need of such tools [60].

5.2.1 Modeling Security Protocols

Protocol specification languages have evolved from low-level generic languages such as TLT [73], TLA [66], and PROMELA [74]. These languages are not specialized to security protocols, but can be applied to many types of concurrent systems. They require the modeller to explicitly specify the behavior of channels, encryption, message composition and decomposition, and many other things related to security protocols. Due to their generality they can be applied to any protocol, but they are unsuitable for general use because of the time it takes to specify protocols, and also because they are not optimized to handle protocol analysis. This creates complex models with enormous numbers of states, and makes the analysis of large protocols with these tools difficult.

A number of different formal techniques and logics have been applied specifically to the domain of formal security protocol analysis, and tools have been developed based on these logics. State space exploration is a technique which can be used to explore all possible paths through a state space defined by a model of the protocol under analysis. Another technique which has been applied to the analysis of security protocols is the Burrows, Abadi and Needham (BAN) logic [75]. This logic is based on the beliefs of principals and inference rules related to these beliefs, for example, if a principal A believes that only B and itself know of a shared key K, and A receives a message encrypted with K, then A will believe the message was actually from B. BAN logic provides a very high-level view of a protocol. It has been used successfully to identify a number of attacks, however, [76] claims that the BAN logic is flawed and discusses two protocols which, when modelled correctly in BAN logic, were incorrectly found to be secure. [77] counter-claims that the two protocols were modelled incorrectly, and that the BAN logic is not flawed.

Inductive theorem proving techniques [78, 79] are based on sets of rules for extending sequences of events. These rules represent the actions of both honest participants and of the intruder. Authentication and secrecy goals are then expressed as properties of these sequences, and are proved using induction.

The general purpose theorem prover Isabelle [80] is used to make the theorem proving process more automated, but these tools are still interactive and require a high level of expertise. Strand spaces [65] are a graph-theoretic approach to representing security protocols. They are closely related to inductive theorem proving techniques, but provide a simpler, more intuitive model and more precise results.

In recent years new specification languages have been developed which are specialized to the domain of security protocols. They provide constructs which allow modellers to easily specify things like messages send and receive actions, encryption, and a variety of other capabilities. Some examples of these languages are: Casper [81], CAPSL [82], and HLPSL1.0 [83].

HLPSL1.0 and CAPSL are both languages based on the Alice-Bob notation described earlier. Both languages are based on the idea of a high-level language which is translated into a lower level language for analysis by a number of tools. The Casper approach does not support non-atomic keys. More importantly, Casper is geared towards finite state model checking, and requires restrictive assumptions to be made about the system. For example, the maximum depth of messages must be specified. CAPSL is designed to be a common specification language which can be used by a number of tools. Unfortunately, the language is limited in some ways. For example, it can not express a situation where a principal receives a message which it can not immediately decrypt, and it is restricted to secrecy and authentication goals. The HLPSL1.0 language can provide precise results in authentication and key management protocol analyzed. This is the reason why this research is focus on HLPSL because it provides precise result in authentication and key management which is the problem stamen of the research.

5.2.1.1 The AVISPA Tools (Simulation Environment)

AVISBA is now a commonly used verification tool for cryptographic protocols. The main advantage of this tool is the ability to use different verification techniques on the same protocol specification. The protocol designer interacts with the tool by specifying a security problem (ie a protocol paired with a security property that the protocol is expected to achieve) in the High-Level Protocol Specification Language (HLPSL).The HLPSL is an expressive, modular, role-based, formal language that is used to specify control-flow patterns, data-structures, alternative intruder models and complex security properties, as well as different cryptographic primitives and their algebraic properties.

The features of HLPSL

1. High level language which means easy to write and easy to understand.
2. The protocol designer can specify the protocol specification and the protocol’s roles.
3. Rich of tools to measure the security issues such as authentication, integrity and so on.

These features make HLPSL well suited for specifying modern, industrial-scale protocols.

The protocol designer write the code in HLPSL language then, the AVISPA tools consist of a translator from the High Level Specification Language (HLSPL) into the Intermediate Format (IF), and four back-ends with which to analyze the generated IF specification.

Each of the four back-ends may make use of a further translator in order to convert the IF file into the tool's individual specification language. The four back-ends are the On-the Fly Model Checker (OFMC), SAT based model checker (SATMC), Constraint Logic Attack Searcher (CL-ATSE) and Tree Automata based automatic approximations for the analysis of Security Protocols (TA4SP). All four back-ends must be able to parse the generic IF file and must comply with a standard output format. This makes automated test runs of all four tools over a large number of specifications simpler, and will soon allow a graphical front-end to parse results for visual display.

Figure 5.1 depicts the overall architecture of the system including SPAN. The initial development of the SPAN tool was done in collaboration with Olivier Heen and Olivier Courtay of Thomson R&D France.

SPAN helps in interactively producing Message Sequence Charts (MSC) which can be seen as an "Alice & Bob" trace from an HLPSL specification. SPAN can represent one or more sessions of the protocol in parallel according to the information given in the role environment. Then, MSCs are produced interactively with the user. SPAN also includes the possibility to check the values, at every moment, of the variables of each principal: the user chooses the variables of each role he wants to monitor.

The three modes of SPAN are

• Protocol Simulation for simulating the protocol and build a particular MSC corresponding to the HLPSL specification;

• Intruder Simulation for simulating the protocol with an active/passive intruder;

HLPSL
IF
OFMC
CL
SATMC
TA4SP

Figure 5.1 AVISBA system architecture and SPAN

SPAN

• Attack Simulation for automatic building of MSC attacks from the output of either OFMC or CL-ATSE tools.

5.2.1.2 The High Level Protocol Specification Language (HLSPL)

HLPSL [71, 85] is the protocol specification language of the AVISPA project. It was designed to be a fast, easy to use protocol specification language which would be accessible to protocol engineers who might not necessarily be well versed in formal methods.

HLPSL is based on some of the concepts of TLA [69]. The semantics of HLPSL are wholly defined using TLA. HLPSL provides a flexible, theoretically sound protocol specification language which is sufficiently high level to be accessible, yet expressive enough to be able to model most security protocols.

The HLPSL language supports branching, non-determinism, multiple layers of encryption, functions, sets, role composition, and even allows the intruder to participate in protocol sessions as a legitimate player.

Each HLSPL specification is made up of basic roles and compositional roles. Basic roles define the initial knowledge and the behavior of each of the participants. Compositional roles are used to instantiate the roles with values and to define protocol sessions.

Each basic role contains a parameter list which describes the initial knowledge it must be instantiated with. A basic role also has a played by parameter, which is used to instantiate the role with a player.

Each basic role also contains a list of local variables and an initialization section, and finally, a list of transitions. A basic role is used to define the behavior of an honest participant. This is done using a list of transitions. Each transition has a left hand side which describes what must be true for the transition to be enabled, and a right hand side which defines the consequences of that transition being fired. An example of a HLPSL transition is in Figure 5.2.

```
1. State = 0 /\ RCV(Text) =|>
State' = 1 /\ SND(Text)
```

Figure 5.2: An example of a HLPSL transition

In this example the transition is fired if the variable State is equal to 0 *and* the message Text is received on the channel RCV. This transition is only triggered when the message received on the channel RCV is equal to the value of the variable Text. If the transition is fired, the value of State is changed to 1. The syntax State' (spoken as .state-prime.) refers to the value of State immediately after the transition is complete. This syntax for referring to the new value of a variable is taken from TLA. As well as updating the value of State, a message containing the value of Text is sent on a channel called SND. Note that the variables State, RCV, SND, and Text all need to be declared and given appropriate types.

The different types available in HLPSL are described in [5]. They include support for messages, agents, keys, nonces, natural nonces, and Dolev-Yao channels. Figure 4.3 is a more advanced example of a HLPSL transition. It demonstrates concatenation, encryption, nonce generation, and the binding of values received in messages to variables.

```
1. State = 1 /\ RCV({Text.Val'}KeyA) =|>
State' = 2 /\ SND({Text.Nonce'}KeyA)
```

Figure 5.3: A more advanced HLSPL transition

When an intruder is under concern, after each step of protocol execution, SPAN shows the current intruder knowledge and proposes to construct and send malicious messages from this knowledge. Message patterns are proposed to the user conjointly with intruder data, relevant, pattern structure and type. The tool can save and load execution traces corresponding to the execution of the protocol supervised by the user. The MSC can be exported in postscript format or PDF format. Finally, SPAN comes with a local version of the web interface of AVISPA that supports the editing of protocol specifications, allows the user to select and configure the back-ends integrated into the tool and launch the three different kind of animations: protocol simulation (with no intruder), intruder simulation (build your own attacks by hand), attack simulation (load attacks found by OFMC/CL-ATSE in the simulation).

The HLPSL is an expressive, modular, role-based, formal language that is used to specify control flow patterns, data-structures, alternative intruder models and complex security properties, as well as different cryptographic primitives and their algebraic properties.

We give a flavor of HLPSL using the specification of the Needham-Schroeder Public Key protocol. Here is an example of a basic role declaration extracted from the HLPSL specification of this protocol. Figure 5.4 and 5.5 specify a part of the master role and the server role

```
role master (M, S: agent, Km, Ks: public_key, SND, RCV: channel (dy))
          played_by M def=
          local State : nat, Nm, Ns: text
          init State := 0
         transition
               0. State = 0 /\ RCV(start) =|>
                 State’:= 2 /\ Nm’ := new() /\ SND({Nm’.A}_Ks)
              2. State = 2 /\ RCV({Nm.Ns’}_Km) =|>
```

```
                   State’:= 4 /\ SND({Ns’}_Ks)
end role
```

Figure 5.4 specify the master role

```
role Server (S, M: agent, Km, Ks: public_key, SND, RCV: channel (dy))
            played_by B def=
            local State : nat, Nm, Ns: text
            init State := 1
           transition
                   1. State = 1 /\ RCV({Nm’.M}_Ks) =|>
                       State’:= 3 /\ Ns’ := new() /\ SND({Nm’.Ns’}_Km)
                   3. State = 3 /\ RCV({Ns}_Ks) =|> State’:= 5
end role
```

Figure 5.5 specify the server role

Then, roles are composed together in sessions where the knowledge shared between the roles (public keys for instance) are made explicit.

```
role session(M, S: agent, Km, Ks: public_key) def=
     local SM, RM, SS, RS: channel (dy)
    composition
    master(M,S,Km,Ks,SM,RM) /\ server (M,S,Km,Ks,SS,RS)
end role
```

Finally, the environment used for protocol execution is defined, where ’i’ denotes the intruder. The environment also defines the initial knowledge of the intruder and the initial setting for the sessions, i.e. how many sessions are run and who run them.

```
role environment() def=
     const a, b, c, d : agent,
     km, ks, ki, kc, kd : public_key,
     intruder_knowledge = {m, s, km, ks, kc, kd, ki, inv(ki)}
    composition
     session(a,b,ka,kb) /\ session(c,d,kc,kd) /\ session(a,i,ka,ki)
end role
```

In the example above, four honest agents are defined, namely a,b,c, and d, and the intruder knows all the public keys as well as its own private key inv(ki).

The HLPSL language has been designed to support temporal logic style security goals; however the AVISPA tools do not yet provide this support. Goals are currently specified as macros. There are three types of goal macros available: secrecy, strong authentication, and weak authentication. These three goals can be used to capture the requirements of many protocols, but are not sufficient to model many others. The specification of goals is HLPSL is done in the goal section. Figure 5.6 is an example of two HLPSL goals.

```
goal
Server authenticates master on Msg
Secrecy of Msg
end goal
```

Figure 5.6: HLPSL goals

5.2.1.3 Formal Analysis of Security Protocols

Formal analysis has been used with some success to verify the correctness of security protocols. A specification language is used to describe the protocol and its security requirements, and a model checker is then used to verify that the security requirements of the protocol are met. This approach is in some ways superior to human examination because it has the advantage of an exhaustive search through all possible ways in which the protocol and the intruder can behave, and all the ways in which concurrent sessions of a protocol can interact and interfere with each other.

A number of different formal techniques and logics have been applied to the domain of formal security protocol analysis. Most commonly, state space exploration techniques have been used to explore all possible paths through a state space defined by a model of the protocol under analysis. Some other techniques which have been applied to the problem include the Burrows, Abadi and Needham (BAN) logic [63], which is based on the beliefs of principals and inference rules related to these beliefs, inductive theorem proving techniques [64], and the graph-theoretic strand space model [65], [66], provides a good summary of the current state of formal analysis of security protocols.

The tools used for analyzing security protocols are quite varied. Some general purpose tools for describing concurrent systems have been adapted to the purpose, while there are also a number of tools specialized to the field of protocol analysis.

Formal security protocol specification languages have previously been limited in a number of ways. Expressiveness, scalability and ease of use have all been identified as limitations of the current generation of tools. This is changing as new analysis techniques and specification languages are developed.

5.2.2 Proposed Simulation Setup

In the proposed protocol as specified in chapter four, the simulation is divided into two phases. Phase one the authentication between the master and the database (DS) server, so the first simulation program define the master, users and DS as TVWS devices and defined all the roles and the states for each one and the simulation environment. After the master complete authentication process with the DS then the Phase two starts to authenticate the user with the master mode. At the result of the simulation specify that this protocol is saved, and when changing the number of users from one user to 80 users, gets the same result the protocol is saved, and then polite a graph as in figure 5.11. Appendix A contains the simulation code.

The second simulation program defines the master, users and DS as TVWS devices and defined all the roles and the states for each one and the simulation environment. The key generation methods is defined inside the roles and the simulation result shows that this method is saved, end then collect the result at the end of the simulation and polite a graph as in figure 5.12. Appendix B contains the simulation code.

5.3 OMNeT ++

Computer simulation has become a popular methodology for performance study of computer and telecommunication networks. This popularity results from the availability of various sophisticated and powerful simulation

software packages, and also because of the flexibility in model construction and validation offered by simulation. While various network simulators
exist for building a variety of network models, choosing a good network simulator is very important in modeling and performance analysis of wireless networks. A good simulator is one that is easy to use; more flexible in model development, modification and validation; and incorporates appropriate analysis of simulation output data, pseudo-random number generators, and statistical accuracy of the simulation results.

OMNeT++ is becoming one of the most popular network simulators because it has all the features of a good simulator. The use of discrete event simulation packages as an aid to modeling and performance evaluation of computer and telecommunication networks, including wireless networks has grown in recent years [86, 87]. This popularity is due to the availability of sophisticated simulation packages and low-cost powerful personal computers (PCs), but also because of the flexibility in rapid model construction and validation offered by simulation. This research uses OMNeT++ simulator to compare between the proposed protocol and IEEE 802.22 in terms of execution time.

OMNeT++ itself is not a simulator of anything concrete, but rather provides infrastructure and tools for writing simulations. One of the fundamental ingredients of this infrastructure is component architecture for simulation models. Models are assembled from reusable components termed modules. Well-written modules are truly reusable, and can be combined in various ways.

Modules can be connected with each other via gates (other systems would call them ports), and combined to form compound modules. The depth of module nesting is not limited. Modules communicate through message passing, where messages may carry arbitrary data structures. Modules can pass messages along predefined paths via gates and connections, or directly to their destination; the latter is useful for wireless simulations. For example modules may have parameters that can be used to customize module behavior and/or to parameterize the model's topology. Modules at the lowest level of the module hierarchy are called simple modules, and they encapsulate model behavior. Simple modules are programmed in C++, and make use of the simulation library.

OMNeT++ simulations can be run under various user interfaces. Graphical, animating user interfaces are highly useful for demonstration and debugging purposes, and command-line user interfaces are best for batch execution. The simulator as well as user interfaces and tools are highly portable. They are tested on the most common operating systems (Linux, Mac OS/X, Windows), and they can be compiled out of the box or after trivial modifications on most Unix-like operating systems.

5.3.1 Modeling Concepts (Simulation Environment)

An OMNeT++ model consists of modules that communicate with message passing It has a generic architecture, so it can be used in various problem domains: modeling communication networks, protocol modeling, queuing networks, multiprocessors and other distributed hardware systems, in general, modeling and simulation of any system where the discrete event approach is suitable, and can be conveniently mapped into entities communicating by exchanging messages. OMNeT++ itself is not a simulator of anything concrete, but rather provides infrastructure and tools for writing simulations. One of the fundamental ingredients of this infrastructure is component architecture for simulation models.

5.3.2 Topology Description Method (Simulation Setup)

The Topology for this simulation is specified in figure 5.7 where the DS and the master are wired connected and the user connect with the master through wireless connection. To run the simulation OMNeT++ offers three file to configure the scenario described as follow:

5.3.2.1 Ned file

The user describes the structure of a simulation model in the NED language. NED stands for Network Description. NED lets the user declare simple modules, and connect and assemble them into compound modules. The user can label some compound modules as networks; that is, self-contained simulation models. Channels are another component type, whose instances can also be used in compound modules. An example of NED language file:

```
network Network
{
    simple TVWSD
          {
            parameters:
                          @display("i=block/routing");
            gates:
                    input in [];  // declare in[] and out[] to be vector gates
                   output out[];
         }
network UMauthe
       {
         submodules:
                           Mode[2]: TVWSD;
                         server: TVWSD;
               connections:
                            Mode[1].out++ --> {  delay = 100ms; } --> server.in++;
                            Mode[1].in++ <-- {  delay = 100ms; } <-- server.out++;
       }
```

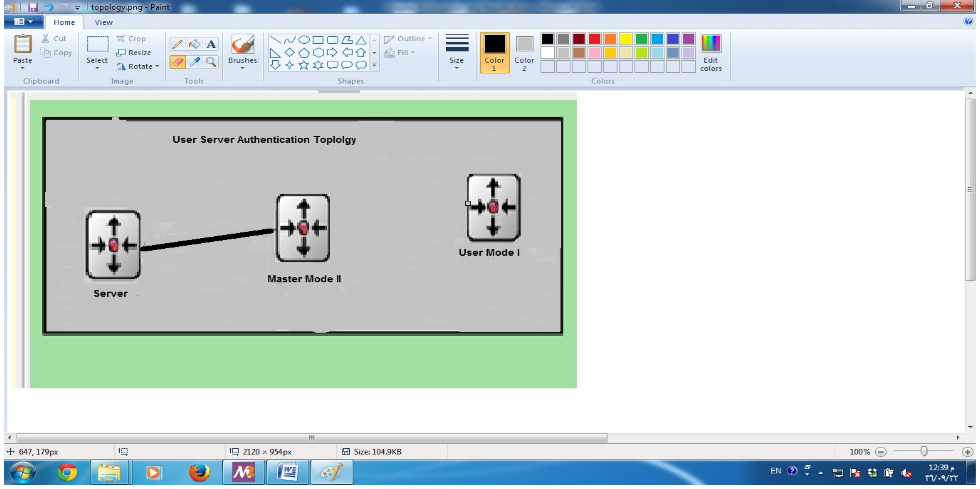
The above code defines a network type named TVWS. Note that the NED language uses the familiar curly brace syntax, and “//” to denote comments. It describe the research simulation topology as in figure 5.7

5.3.2.2 Programming the Algorithms (C++ File)

The simple modules of a model contain algorithms as C++ functions. The full flexibility and power of the programming language can be used, supported by the OMNeT++ simulation class library. The simulation programmer can choose between event-driven and process-style description, and freely use object-oriented concepts (inheritance, polymorphism etc) and design patterns to extend the functionality of the simulator.

The simulation uses C++ code to define the protocol scenario and run the simulation, figure 5.8 specify snapshot of the simulation program written in C++ language and calculate the result of

Figure 5.7 Simulation scenario



the simulation for both the proposed new protocol and IEEE 802.22 protocol and compare between them in terms of the authentication time and key generation and exchange time.

```
cMessage *TVWSD::generateNewMessage()
{
    char msgname[20];
    if (strcmp("mode[0]", getName()) ==0)
      {
        sprintf(msgname, "REG_RESP  Welcome %d", seq);
        cMessage *msg = new cMessage(msgname);
       }
    if (umsgtype = 1)
      {
        cMessage *msg = new cMessage("REG_RESP ");
        sprintf(msgname, "REG_RESP  -%d", seq);
       }
    return msg;
}
void TVWSD::sendCopyOf(cMessage *msg)
{ int k =0;
    // Duplicate message and send the copy.
    cMessage *copy = (cMessage *) msg->dup();
    send(copy, "out",k);
   handleMessage(copy);
}
void TVWSD::handleMessage(cMessage *msg)
{
    if (strcmp("mode[1]", getName()) ==0)
    {
         Message arrived.
        EV << "Message " << msg << " arrived.\n";
        delete msg;
       bubble("This is  master!");
       //forwardMessage(msg);
    }
     else
    {
        // We need to forward the message.
        //bubble("This is  user!");
        forwardMessage(msg);
    }
}
```

Figure 5.8 snapshot of the simulation code

Figure 5.13, 5.14 shows the simulation results for both the authentication protocol and key management method respectively.

```
[General]
network = sim
[Config UMauthe]
network = UMauthe
 sim.host1.limit = 80
*.*.servername = "DS"
*.*.masterask = "ModeII"
*.*.userID = "1222535"
```

Figure 5.9 example of omnetpp.ini file

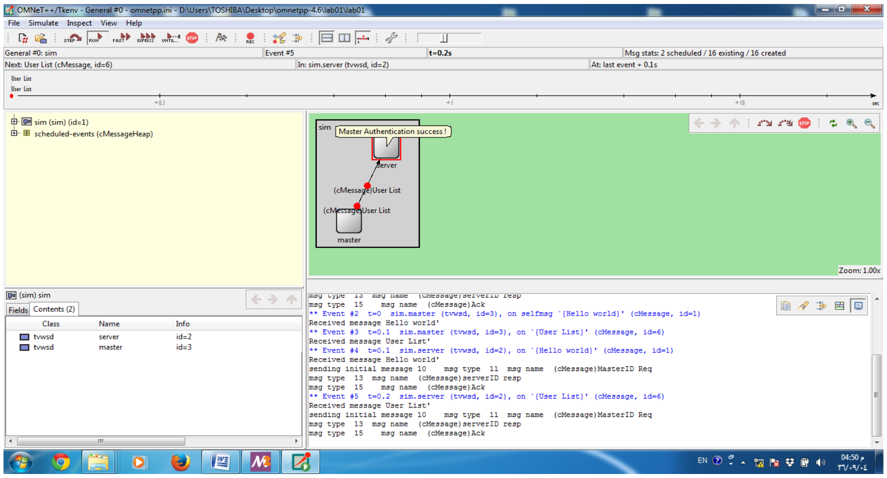
5.3.2.3 The simulation control (omnetpp.ini)

This file uses to define the values of the variables and the name of the running network in case of some networks topology is specified in the same directory. Figure 5.9 shows an example of omnetpp file

5.3.2.4 User Interfaces

The primary purpose of user interfaces is to make the internals of the model visible to the user, to control simulation execution, and possibly allow the user to intervene by changing variables/objects inside the model. This is very important in the development/debugging phase of the simulation project. Equally important, a hands-on experience allows the user to get a feel of the model's behavior. The graphical user interface can also be used to demonstrate a model's operation.

The same simulation model can be executed with various user interfaces, with no change in the model files themselves. The user would typically test and debug the simulation with a powerful graphical user interface, and finally run it with a simple, fast user interface that supports batch execution. Figure 5.10 shows the graphical user interface for OMNeT simulation.



General #0: sim | Event #5 | t=0.2s | Msg stats: 2 scheduled / 16 existing / 16 created

Next: User List (cMessage, id=6) | In: sim.server (tvwsd, id=2) | At: last event + 0.1s

sim (sim) (id=1)
scheduled-events (cMessageHeap)

(sim) sim

Fields | Contents (2)

Class	Name	Info
tvwsd	server	id=2
tvwsd	master	id=3

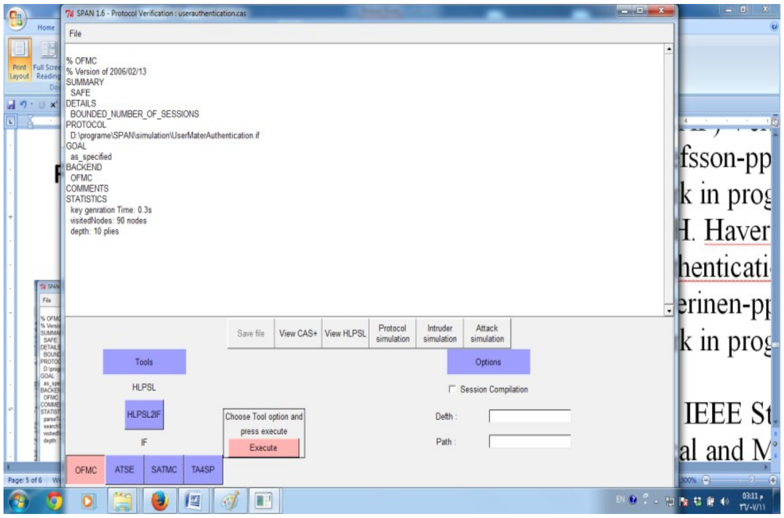
```
msg type  13  msg name  (cMessage)serverID resp
msg type  15    msg name  (cMessage)Ack
** Event #2  t=0  sim.master (tvwsd, id=3), on selfmsg `{Hello world}' (cMessage, id=1)
Received message Hello world'
** Event #3  t=0.1  sim.master (tvwsd, id=3), on `{User List}' (cMessage, id=6)
Received message User List'
** Event #4  t=0.1  sim.server (tvwsd, id=2), on `{Hello world}' (cMessage, id=1)
Received message Hello world'
sending initial message 10    msg type  11  msg name  (cMessage)MasterID Req
msg type  13  msg name  (cMessage)serverID resp
msg type  15    msg name  (cMessage)Ack
** Event #5  t=0.2  sim.server (tvwsd, id=2), on `{User List}' (cMessage, id=6)
Received message User List'
sending initial message 10    msg type  11  msg name  (cMessage)MasterID Req
msg type  13  msg name  (cMessage)serverID resp
msg type  15    msg name  (cMessage)Ack
```

Figure 5.10 shows the graphical user interface for OMNeT simulation

5.4 Results and Discussion

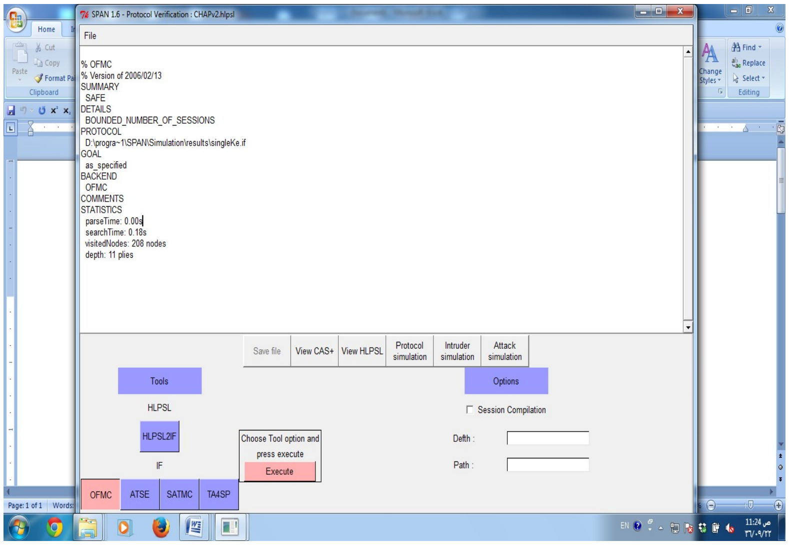
The HLPSL simulation results as in figure 3.11 show the authentication process is saved. Figure 5.12 and 5.13 shows that the generation of single key and generating a number of keys is secure. The OMNET++ 4.6 simulation used to compare between the proposed protocol and IEEE 802.22 protocol in terms of delay. Figure 3.14, shows that the authentication process speeds in the new protocol is equal to the IEEE 802.22 protocols when the users are less than 50. And after 50 users the difference in delay is not negligible. Figure 3.15 specify that the authentication procedures and key management is very fast than PKMv2.

Figure 5.11: The authentication simulation result



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  D:\programe\SPAN\simulation\UserMaterAuthentication.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  key genration Time: 0.3s
  visitedNodes: 90 nodes
  depth: 10 plies
```

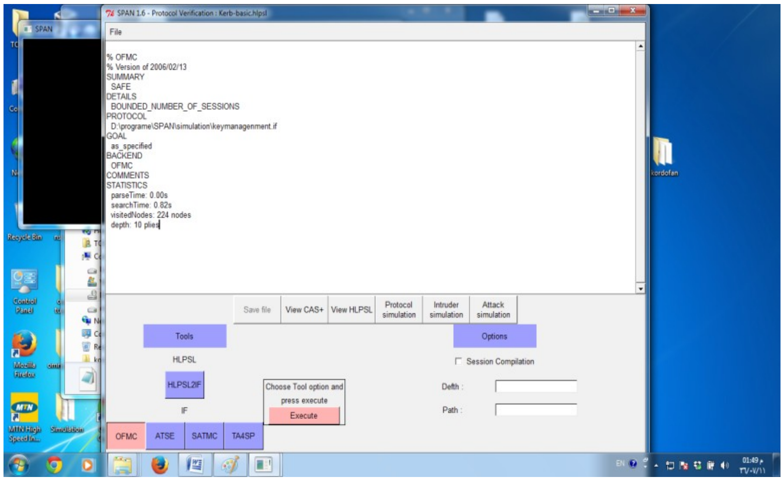
Saved



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  D:\progra~1\SPAN\Simulation\results\singleKe.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.18s
  visitedNodes: 208 nodes
  depth: 11 plies
```

Figure 5.12 the single key generation’s result
Save

Figure 5.13: The Number of Key generation result



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  D:\programe\SPAN\simulation\keymanagenment.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.82s
  visitedNodes: 224 nodes
  depth: 10 plies
```

SAFE

Figure 5.14: Authentication delay

Figure 5.15: key management comparisons protocols

CHAPERT SIX

Conclusion and Recommendations

6.1 Conclusions

A well-known issue in modern wireless communications is spectrum scarcity. To solve the dilemma between the increasing bandwidth demands and the actual underutilization of spectrum resource, the Federal Communications Commission (FCC) has allowed unlicensed users to opportunistically access the temporarily unoccupied television (TV) bands, namely TV white spaces (TVWSs) on the basis of noninterference of the licensed users.

The need for wireless spectrum is growing fast due to the success of smart phones and tablets. Users demand wireless access everywhere and all the time. Spectrum shortage forces to utilize that scarce resource more efficiently. One of the most prominent approaches is dynamic spectrum use. TV white space technology implementation is the first step in the direction of collective spectrum use.

Many standards agreed two ways for spectrum sharing, either by spectrum sensing and/or spectrum database. TVWS unlicensed can use cognitive radio (CR) techniques for sharing the spectrum. These techniques are similar to Wi-Fi techniques but the difference is CR is cover wide area, but still lack of coordinator and centralized device to avoid the interference.

The other approach is to develop a database of available channels in every area, and have each TV White Space device contact the database, provide its location, and be assigned spectrum that is available in that area. The advantage of this approach is that TV White Space devices are simpler, do not make erroneous decisions, and can be built without the expensive logic required to track its location, resulting in devices estimated to cost about the same as today’s Wi-Fi access points.

The name geolocation database is used to emphasize the importance of geographical information in the controlling of the utilization of white space spectral resources. With geolocation databases the security issues must be taken into account from the different point-of-view than in traditional wireless communication. This is due to the nature of the TVWS which uses the network link's security and database's security. The security threats could be such as man-in-the-middle attack, Denial of Service (DoS) attack and/or misuse of the available channels, which lead to interference and decrease the utilization of the available spectrums.

To overcome these security problems the device and the database must perform mutual authentication. The mutual authentication is a process which allows both database server (DS) and the users to authenticate each other. In the other word the database has to know if the device is allowed to access white space. And in the same time, the device has to know which databases are certified by regulatory authorities. Naturally, data transfer has to be encrypted and the integrity of geolocation data has to be secured, so the mutual authentication must perform key management process in terms of key generation and key exchange to encrypt/decrypt the transfer's data. In network security, symmetric key encryption methods are commonly used in order to generate and exchange a secret key between the sender and the receiver. The main drawbacks of this method are that, sender and receiver exchange a data the attackers might access and use it to obtain the key, and even more they can generate new keys when the life time of the available key is expired . Also, the process of generating the key is inefficient and time consuming.

Internet Engineering Task Force (IETF) is developed a Protocol to Access White Space database (PAWS) with the aims of defining the device-database interface for TVWS database systems. Devices may be able to connect to the database directly or indirectly via the Internet or private IPnetworks. There is no intent to restrict the protocol to any particular set of authorities.

PAWS use the "HTTP over TLS" as transfer’s mechanism for transferring the data. TLS provides message integrity and confidentiality between the master device and the database, but it needs special adaptation like use of recommended cipher suites and modes of operation. In some cases the server may require client authentication, as described in the "Transport Layer Security (TLS) Protocol", to authenticate the device. When client authentication is required, the database must specify, by prior arrangement, acceptable root Certificate Authorities (CAs) to serve as trust anchors for device certificates. The Database and devices should support "Stateless TLS Session Resumption" to enable databases to handle large numbers of requests from large numbers of devices. In terms of key management PAWS uses secure channel for communication, which is highly secure but very expensive when comparing with other methods that generate and exchange shared secrete keys.

IEEE 802.22 is defined as the first wireless protocol for cognitive radio in wireless regional area network (WRAN). The security sublayer defined in 802.22 provides confidentiality, authentication, and data integrity services by applying cryptographic transformations to MAC data units carried across connections between CPEs and the BS. The security sublayer has two components: an encapsulation protocol and a Privacy Key Management (PKM) protocol.

These protocols apply the authentication from network point of view and applied at the network layer. The database’s authentication protocols are performs the authentication in terms of user names and password, which applies at session layer and above.

This research demonstrates the proposed mutual authentication protocol which utilizes the availability of the database and the network link together in one protocol to authenticate the entire link between the users and the DS. This protocol includes the authentication process and key management process in terms of key generation and key distribution. Finally the new proposed protocol introduces new integrated security framework that ensures closing the gap between security sublayer at lower layers and upper layers at session layer and above, by utilizes the availability of the database and uses it in the authentication protocol. The new protocol is supported with all recommended functions from various standards of IEEE.

The simulation results show that the new protocol is secure and high performance than IEEE 802.22 protocol.

6.2 Recommendation

The proposed protocol study the mutual authentication fixed master mode device so the future work can be done when the master move during the user authentication process from DS area to a new area, that means the master need to authenticate again with the new DS before authenticate the user. In this case so many problems can be study such as the unexpected user’s authentication delay, the user may not registered with the new DS, and the user decision to authenticate with other master mode.

In case of key generation and key distribution the proposed protocol did not study the way of how to exchange the message (m) and conceder this step out of the protocol scope, so the future research can be conducted on this issue. Another future work can focus on group key generation and distribution.