

**ENHANCING SECURITY OF WIMAX COMMUNICATION  
SYSTEM USING DIFFIE HELLMAN ALGORITHM**

**STEPHEN OCHIENG OGUTA**

**MASTER OF SCIENCE**

**(Telecommunication Engineering)**

**JOMO KENYATTA UNIVERSITY OF  
AGRICULTURE AND TECHNOLOGY**

**2018**

**Enhancing Security of Wimax Communication System Using Diffie Hellman  
Algorithm**

**Stephen Ochieng Oguta**

**A thesis submitted in partial fulfillment for the degree of Master of Science in  
telecommunication Engineering in the Jomo Kenyatta University of  
Agriculture and Technology**

**2018**

## **DECLARATION**

This thesis is my original work and has not been presented for a degree in any other University

Signature..... Date .....

**Stephen Ochieng Oguta**

This Thesis Has Been Submitted For Examination With Our Approval As University Supervisors.

Signature..... Date .....

**Prof. Stephen. Musyoki, PhD**

**Technical University of Kenya**

Signature..... Date .....

**Dr. Kibet. Langat, PhD**

**JKUAT, Kenya**

## **Dedication**

I dedicate this work to God and my family. You are a strong pillar in this journey of life.

## **Acknowledgement**

My heartfelt gratitude goes to God for his grace at such a time as this. He has been Ebenezer. I also thank my supervisors, Prof. S. Musyoki and Dr. K. Langat for their guidance and presence during the preparation of this thesis. Their dedication and guidance was very helpful in this journey. I also appreciate my family for their encouragement and endless support over the years.

My classmates too were very wonderful people. God bless you all.

## TABLE OF CONTENTS

<b>DECLARATION.....</b>	<b>ii</b>
<b>Dedication .....</b>	<b>iii</b>
<b>Acknowledgement .....</b>	<b>iv</b>
<b>TABLE OF CONTENTS .....</b>	<b>v</b>
<b>LIST OF FIGURES .....</b>	<b>vii</b>
<b>LIST OF ABBREVIATIONS .....</b>	<b>viii</b>
<b>ABSTRACT.....</b>	<b>x</b>
<b>CHAPTER ONE .....</b>	<b>1</b>
<b>INTRODUCTION.....</b>	<b>1</b>
1.1 Background Information .....	1
1.2 Problem Statement .....	2
1.3 Justification .....	4
1.4 Objectives.....	4
1.4.1 Main Objective .....	4
1.4.2 Specific objectives .....	4
<b>CHAPTER TWO .....</b>	<b>5</b>
<b>LITERATURE REVIEW .....</b>	<b>5</b>
2.1 Security Weaknesses in WiMAX System.....	6
2.1.1 The man-in-the-middle attack .....	6
2.1.2 Water Torture Threats .....	7
2.1.3 Jamming Attacks .....	7
2.1.4 Masquerade Attack .....	7
2.1.5 Denial-of-service (DoS) .....	8
2.2 PKMv1 Authentication Protocol.....	8

2.3 PKMv2 Authentication Protocol .....	9
2.4 Diffie Hellman Protocol .....	11
<b>CHAPTER THREE.....</b>	<b>16</b>
<b>METHODOLOGY .....</b>	<b>16</b>
3.1 Proposed Protocol .....	16
3. 2 Modeling an intruder.....	18
<b>CHAPTER FOUR.....</b>	<b>21</b>
<b>RESULTS AND DISCUSSIONS.....</b>	<b>21</b>
4.1 Legitimate BS case .....	21
4.2 Rogue BS case.....	26
4.3 Validation of the DH protocol.....	32
<b>CHAPTER FIVE .....</b>	<b>33</b>
<b>CONCLUSION .....</b>	<b>33</b>
5.1 Conclusion.....	33
5.2 Future work .....	34
<b>REFERENCES.....</b>	<b>35</b>
<b>APPENDICES .....</b>	<b>39</b>

## LIST OF FIGURES

<b>Figure 1.1:</b> Rogue and legitimate BS [18] .....	3
<b>Figure 2.1:</b> WIMAX Network .....	5
<b>Figure 2.2:</b> WiMAX Architecture.....	6
<b>Figure 2.3:</b> PKMv 1 diagram .....	9
<b>Figure 2.4:</b> PKMv 2 diagram .....	10
<b>Figure 2.5:</b> DH flow Diagram.....	13
<b>Figure 3.1:</b> Implementation process chart.....	17
<b>Figure 4.1:</b> Request for SS entry.....	21
<b>Figure 4.2:</b> Requests for BS Entry .....	22
<b>Figure 4.3:</b> Successful authentication .....	23
<b>Figure 4.4:</b> Differing Authentication keys .....	24
<b>Figure 4.5:</b> AK obtained at second attempt .....	25
<b>Figure 4.6:</b> Rogue BS requests SS for unique number .....	26
<b>Figure 4.7:</b> First Rejection of BS .....	27
<b>Figure 4.8:</b> Second Attempt Rejection.....	28
<b>Figure 4.9:</b> Third attempt rejection .....	29
<b>Figure 4.10:</b> Blocking of the network.....	30

## **LIST OF ABBREVIATIONS**

<b>AAA</b>	Authorization, Authentication and Accounting
<b>AES</b>	Advanced Encryption Standard
<b>AH</b>	Authentication Header
<b>AK</b>	Authentication Key
<b>AP</b>	Access Point
<b>ASN</b>	Access Service Network
<b>ASN-GW</b>	Access Service Network Gateway
<b>BER</b>	Bit Error Rate
<b>BS</b>	Base Station
<b>CIA</b>	Confidentiality Integrity and Availability
<b>CMAC</b>	Cipherbased MAC
<b>CSN</b>	Connectivity Service Network
<b>DES</b>	Data Encryption Standard
<b>DH</b>	Diffie Hellman
<b>DoS</b>	Denial of Service
<b>EAP</b>	Extensible Authentication Protocol
<b>EAP-AKA</b>	EAP-Authentication and Key Agreement
<b>GSM</b>	Global Systems for Mobile Communication
<b>GW</b>	Gateway
<b>HA</b>	Home Agent
<b>HMAC</b>	Hashbased Message Authentication Code
<b>IEEE</b>	Institute of Electrical and Electronic Engineers

<b>IETF</b>	Internet Engineers Task Force
<b>INE</b>	Initial Network Entry
<b>KEK</b>	Key Encryption Key
<b>MAC</b>	Medium Access Control
<b>MAN</b>	Metropolitan Area Network
<b>MITM</b>	Man-In- The-Middle
<b>MS</b>	Mobile Station
<b>NAP</b>	Network Access Point
<b>LoS</b>	Line of Sight
<b>PHY</b>	Physical Layer
<b>PKM</b>	Public Key Management
<b>PF</b>	Proportionate Fair
<b>RSA</b>	Rivest, Shamir, and Adelman
<b>SA</b>	Security Association
<b>SAID</b>	SA Identifier
<b>SC</b>	Single Carrier
<b>SS</b>	Subscriber Station
<b>TEK</b>	Traffic Encryption Key
<b>TDD</b>	Time Division Duplex
<b>VoIP</b>	Voice over Internet Protocol
<b>WiMAX</b>	Worldwide Interoperability Microwave Access
<b>WMAN</b>	Wireless Metropolitan Area Network

## **ABSTRACT**

The Worldwide Interoperability for Microwave Access (WiMAX) is a new technology was recently rolled out. WiMAX defines Privacy Key Management (PKM) protocol in the security sub-layer, which assures the security of connections access in WiMAX channel. PKM protocol has two goals, one is to provide the authorization process and the other is to secure distribution of keying data from the Base Station (BS) to Mobile Station (MS). PKM uses X.509 certificates and symmetric cryptography to secure key exchange between an SS and a BS. Currently, there are two versions of PKM. The PKMv1 process involves a one sided authentication while PKMv2 allows for mutual authentication but after transfer of vital management information. The BS network authenticates the (Subscriber station) SS but the SS has no capacity to authenticate a BS. As a result, a rogue BS can successfully enter the network of a SS without prevention. The rogue BS can then tap all the unencrypted management messages. This constitutes a major security flaw. In this research, a modification of the Diffie-Hellman (DH) key exchange protocol was done to mitigate the man-in-the middle attack in WiMAX by modeling using the Dev C++. The DH protocol is appropriate for enhancing security because of its mutual authentication capabilities. This protocol uses a unique algorithm whose solution must be obtained by both the SS and the BS for communication to be allowed. From the simulation, the DH protocol only allows a genuine BS to access the SS network after successful mutual authentication while a rogue BS is denied access into the network even after several attempts on the algorithm. The DH protocol therefore enhances the security in the SS of the WIMAX network. The results show that no management information is passed in the process of DH algorithm solution. Consequently, safety is enhanced when the SS and the BS solve the algorithm.

## **CHAPTER ONE**

### **INTRODUCTION**

#### **1.1 Background Information**

WiMAX-802.16 is an emerging standard that offers broadband wireless access with high bandwidths and transmission rates [1]. With the deployment of wireless communication in recent years, security issues in wireless networks also become a growing concern [14] [15]. Privacy or confidentiality is fundamental for secure communication, which provides resistance to interception and eavesdropping. Message authentication provides integrity of the message and sender authentication, corresponding to the security attacks of message modification and impersonation. Message replay attack is one of the most common attacks on authentication and authenticated key establishment protocols [16]. If the messages exchanged in an authentication protocol do not carry appropriate freshness identifiers, then an adversary can easily get himself authenticated by replaying messages copied from a legitimate authentication session. Man-in-the-middle attack is another classic attack and is generally applicable in a communication protocol where mutual authentication is absent. Other familiar attacks include parallel session attack, reflection attack, interleaving attack, attack due to type flaw, attack due to name omission, and attack due to misuse of cryptographic services. [3]. In order to prevent forgery or replay attack mutual authentication is always required for any wireless medium.

PKM v1 and PKM v2 have been used in WiMAX for security purposes [1]. The above versions only secure the data being transferred. It also secures the BS. The MS/SS is however left vulnerable to rogue BS [16]. The rogue BS can tap the management messages before the actual passing of transmitted data. DH algorithm is proposed to curb effects of the rogue BS [15]. In this

research, DH protocol introduces mutual authentication prior to exchange of any management equipment information. Both the BS and the MS must authenticate each other. Even though PKMv2 previously used allows for mutual authentication, network management information can easily be tapped by a rogue BS. This research illustrates an enhanced security by rejecting a BS that fails to arrive at the right AK after five attempts.

This research presents an analysis of the security threats to WiMAX security that reflects the most recent work of the IEEE and WiMAX Forum and answers the following questions -

What are the Vulnerabilities and Security threats of the WiMAX Technology?

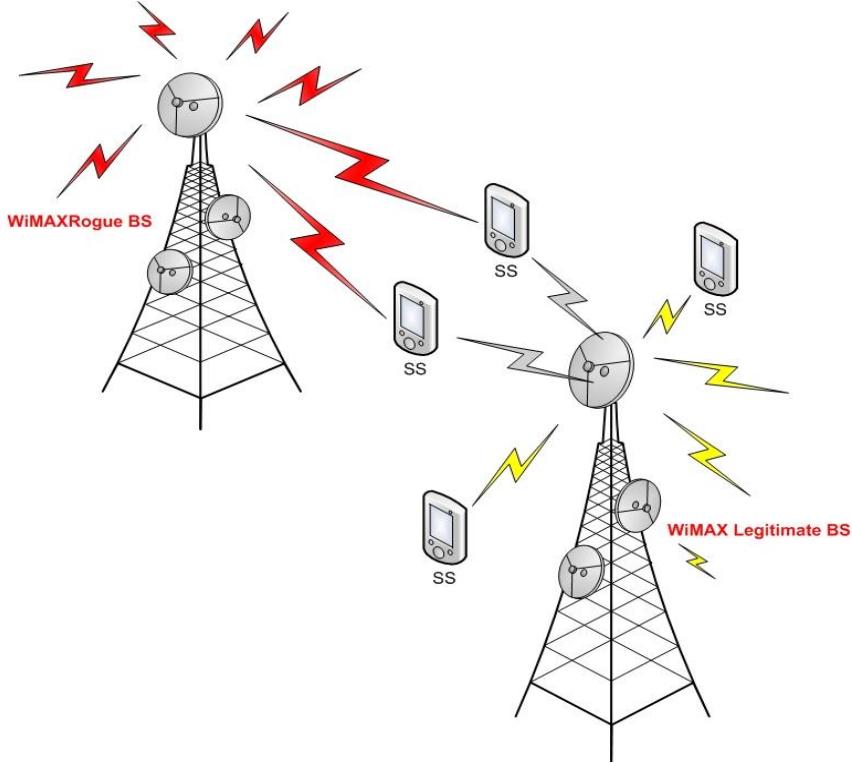
What are the security threats at the Physical Layer and at the MAC layer?

What are the possible solutions that can be achieved from WiMAX Mesh networks?

How can the solution improve security?

## **1.2 Problem Statement**

Security being the major motivation of this research, a more detailed knowledge and understanding of how security issues and concerns occur in wireless networks is needed. The mode through which the wireless networks transmit wireless signals makes it more difficult to protect, thereby making the network vulnerable to attacks [4] [10]. Figure 1.1 below shows how a SS can experience communication both from a legitimate and rogue BS.



**Figure 1.1: Rogue and legitimate BS [18]**

Failure to address the CIA results to wireless traffic, entity resources and services being compromised by unauthorized users [11]. Even if WiMAX technology has complex authentication and authorization methods and a very strong encryption technique it is still vulnerable to different attacks or threats like jamming and MITM [8]. Diffie Hellmann protocol algorithm will be implemented in this research with an aim of introducing mutual authentication between the BS network and SS network prior to the exchange of any management information. The PKMv2 failed to carry out mutual authentication between the SS and the BS prior to exchange of vital management information. As a result, a rogue BS could tap SS information to use for attack. DH introduces mutual authentication that makes the SS safe.

### **1.3 Justification**

The MITM attack fools legitimate stations participating in a communication process into operating as if they are still communicating with each other while disrupting the efficient functioning of the network. Communication in WiMAX can be disrupted as a result of such attacks. WiMAX is selected for this research because it is a recent technology and is presently being rolled out in many parts of the world because of its broadband capacities. This technology provides an environment for many gadgets to communicate. Figure 1.1 gives an illustration of how the many SS can be attacked. A rogue BS can pose as a genuine BS to fool the SS equipments. DH protocol is consequently relevant in WiMAX since it allows for mutual authentication prior to the exchange of sensitive network information. The protocol enables two entities with no prior knowledge of each other to establish a shared key. This process takes place without including any management information in the communication.

### **1.4 Objectives**

#### **1.4.1 Main Objective**

This research aims at enhancing the security of WIMAX by implementing modified Diffie Hellmann protocol.

#### **1.4.2 Specific objectives**

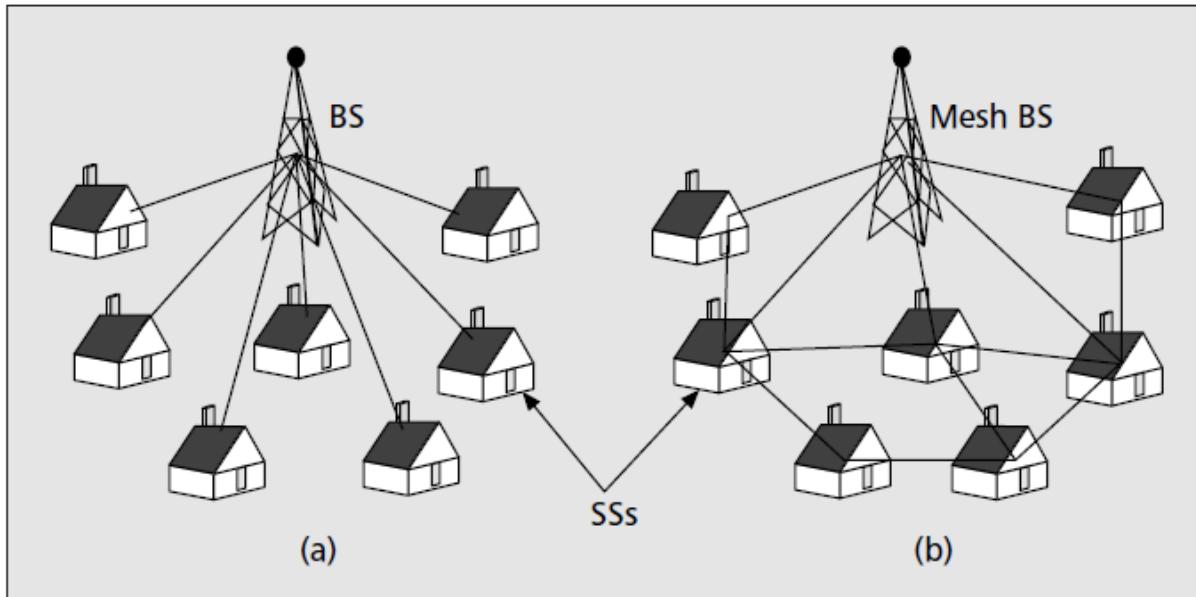
Specific objectives include:

- a) To modify DH algorithm to implement in WiMAX security layer to keep away MITM
- b) To simulate and implement DH algorithm using Dev C++
- c) To validate the enhanced security after implementing DH

## CHAPTER TWO

### LITERATURE REVIEW

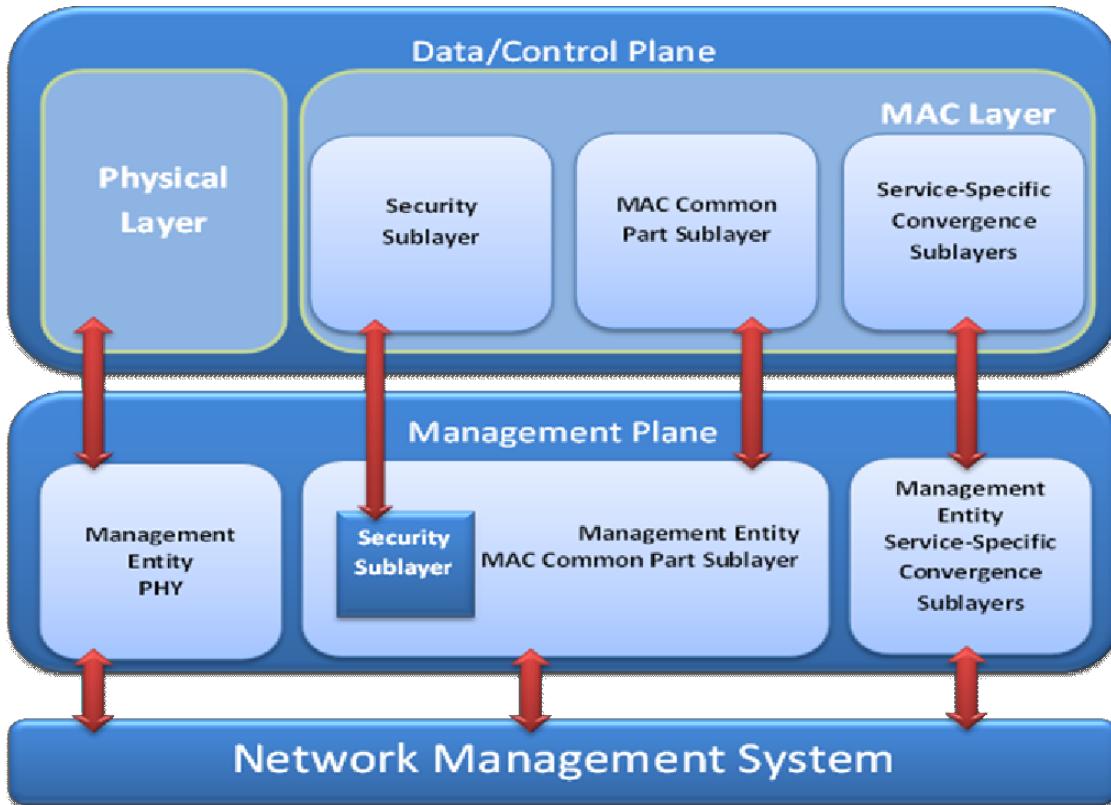
Worldwide Interoperability of Microwave Access, also known as the IEEE 802.16 protocol, is the latest innovation in wireless networks. Figure 2.1 below illustrates a WiMAX network. The SS communicate with the BS via wireless connection. The BS is then connected to the core network through long distance communication links like WiMAX, fiber optics or satellite. The BS and the SS networks have authentication capabilities that help the terminals get connected and pass information the core networks.



**Figure 2.1: WIMAX Network**

Two other recognized standards in IEEE 802.11 standard family are IEEE 802.11a and IEEE 802.11g [2] [16]. Though they provide high speed WLAN standard, the exposure area is limited. IEEE 802.16 is a standard providing broadband access as an alternative to cable connection [7]

[23]. With the support for Mesh networking, WiMAX systems can be easily configured as a Wireless Metropolitan Area Networks (WMAN) [4] [9] [14]. The WiMAX protocol architecture is structured into two major layers (see Fig. 2.2): - the MAC layer and the PHY layer. MAC layer contains 3 sub layers [2] [3] [4].



**Figure 2.2: WiMAX Architecture**

## 2.1 Security Weaknesses in WiMAX System

### 2.1.1 The man-in-the-middle attack

The man-in-the-middle attack (MITM) is a form of active eavesdropping in which the attacker makes independent connections with the communication systems and passes messages between them [4], and in this manner, making the systems believe they are traosacting to each other. The

attacker must be able to intercept all messages going between the two SS and BS in the WiMAX network and inject new ones, which is straightforward in many circumstances [7]. Success in this process is very possible when the attacker seamlessly impersonates the communicating parties.

### **2.1.2 Water Torture Threats**

This is a threat in communication systems that target the battery life. The intention of the attacker is to drain the power of the communicating system so as to keep it out of proper functioning [4]. Additionally, an attacker with a properly positioned Radio Frequency (RF) receiver can interrupt the messages sent through wireless channel [7]. As the physical location of the attacker is not an issue, management messages are more at risk in the WiMAX system.

### **2.1.3 Jamming Attacks**

Wireless networks use common transmission medium. An attacker with a communication transceiver intercepts a transmission, injects disruptive packets, and blocks the legitimate transmission [6]. Jammers disturb communication by generating noise across the entire bandwidth near the transmitting and receiving nodes. The classification of jamming attacks plays an important role not only to differentiate them from each other but also to identify different network performance degradation phenomena like network congestion or channel fading.

### **2.1.4 Masquerade Attack**

A masquerade attack is an attack that uses a falsified identity to gain unauthorized access to SS information in WiMAX through legitimate access identification. If an authorization process is not fully protected, it can become extremely vulnerable to a masquerade attack [7]. Masquerade attacks can be perpetrated using stolen passwords and logons, by locating gaps in programs, or by finding a way around the authentication process [8]. Masquerade attackers can have a full smorgasbord of cybercrime opportunities if they've gained the highest access

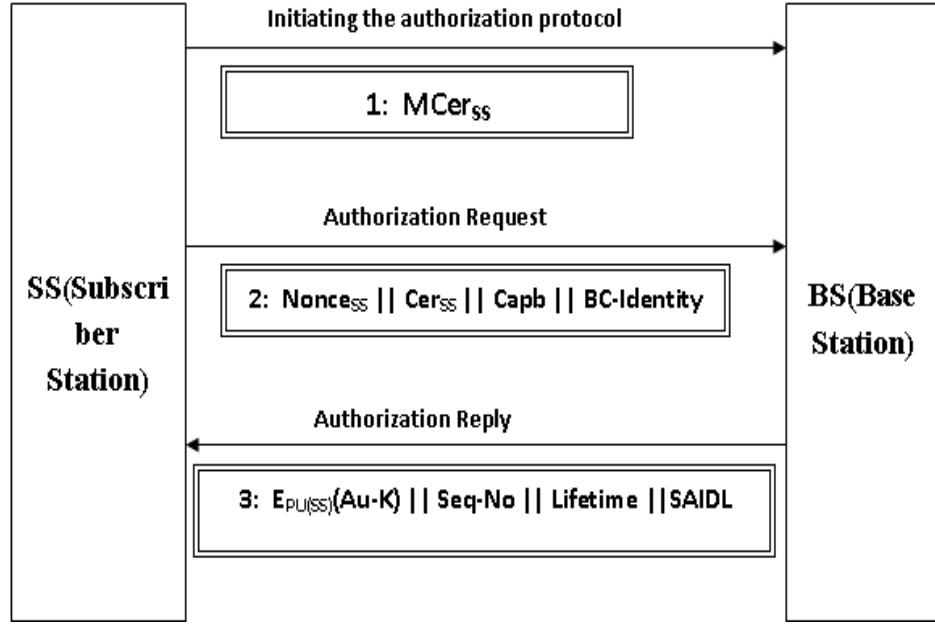
authority to a business organization. A standard strategy to resist this kind of attack is to create innovative algorithms in WiMAX that can efficiently detect the suspicious actions, which could result in the detection of imposters.

### **2.1.5 Denial-of-service (DoS)**

In wireless WiMAX, a denial-of-service is an attempt to make a machine or network resource unavailable to its intended users [8]. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of efforts to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet. DoS (Denial of Service) attacks are sent by one person or system [5]. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root nameservers. These are mostly WiMAX networks. DoS threats are also common in business, and are sometimes responsible for wireless network attacks.

### **2.2 PKMv1 Authentication Protocol**

SS uses the Authentication Information Message, to push its X.509 certificate which identifies its manufacturer to BS as shown in figure 2.3 [1] [20]. BS uses this certificate to decide whether SS is a trusted device. This version only allows authentication of the SS but not the BS. For this reason, a rogue BS can take advantage of this weakness and launch attacks on the SS.



**Figure 2.3: PKMv 1 diagram**

Key for figure 2.3

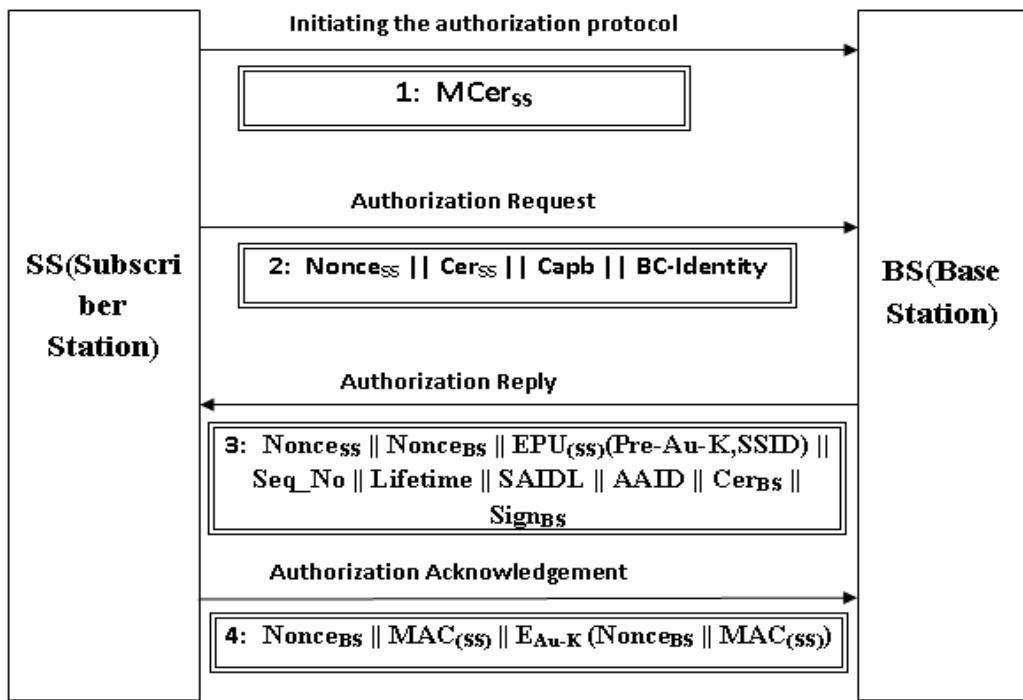
$MCer_{ss}$ -	Manufacturer's certificate of SS
$\text{Nonce}_{ss}$	A number chosen by SS for identification purposes
$Cer_{ss}$	Certificate belonging to SS
$Capb$	Security capabilities of SS
$BC\text{-Identity}$	Security capabilities of SS
$E_{pu(ss)}(Au\text{-K})$	Authentication key features
$\text{Seq\_NO}$	Sequence number
$\text{Lifetime}$	lifetime of the AK
$\text{SAIDL}$	Security Association Identity

### 2.3 PKMv2 Authentication Protocol

Wimax updated to a new version (PKMv2) of the protocol that caters for the shortcomings of the first version. PKMv1 does not have a capacity for mutual authentication [20]. This lack of

mutual authentication by PKMv1 makes the SS prone to attack from a rogue BS. In this second version, SS and BS authenticate each other. They exchange communication which comprises of unique numbers selected, manufacture and equipment certificates and pre authentication keys. Even though mutual authentication is present in PKMv2, WiMAX still has a risk from the leaking management information [4] [24].

PKMv2 has a weakness in allowing management information to be passed through the insecure network prior to mutual authentication as shown in Figure 2.4. This management information like the manufacture certificates can be tapped by a rogue BS to launch an attack.



**Figure 2.4: PKMv 2 diagram**

$MCer_{ss}$  - Manufacturer's certificate of SS

$\text{Nonce}_{ss}$  - A number chosen by SS for identification purposes

$\text{Nonce}_{BS}$  - A number chosen by BS for identification purposes

$Cer_{ss}$  - Certificate belonging to SS

Cer <sub>BS</sub>	Certificate belonging to BS
Capb	Security capabilities of SS
BC-Identity	Security capabilities of SS
E <sub>pu(ss)</sub> (Au-K)	Authentication key features
Seq-NO	Sequence number
Lifetime	lifetime of the AK
SAID	Security Association Identity
MAC <sub>(ss)</sub>	MAC address of SS

## 2.4 Diffie Hellman Protocol

Diffie-Hellman key exchange (DH) is a cryptographic protocol that allows two parties that have no prior knowledge of each other to establish together a shared secret key over an insecure communications channel [21] [23]. Then they use this key to encrypt subsequent communications. The scheme was first published publicly by Whitfield Diffie and Martin [25].

The PKMv2 by itself does not provide authentication of the communicating parties and is thus susceptible to a man-in-the-middle attack [20]. A method to mutually authenticate the communicating parties to each other is generally needed to prevent this type of attack [2]. As shown in Figure.2.4, SS sends a request message to the BS that includes the certificate [17]. BS then responds to the challenge. Communication is only allowed when a common answer is obtained between the SS and BS.

The Diffie-Hellman key exchange protocol [14] [15] [16] originally supports unauthenticated key agreements between stations wishing to communicate. The basic version of the Diffie-Hellman protocol is implemented as described below:

Let

$$PkMS = GNb \bmod P \quad 2.1$$

$$PkBS = GNa \bmod P \quad 2.2$$

Where:

PkMS is the mobile Station's public key

PkBS is the base Station's public key

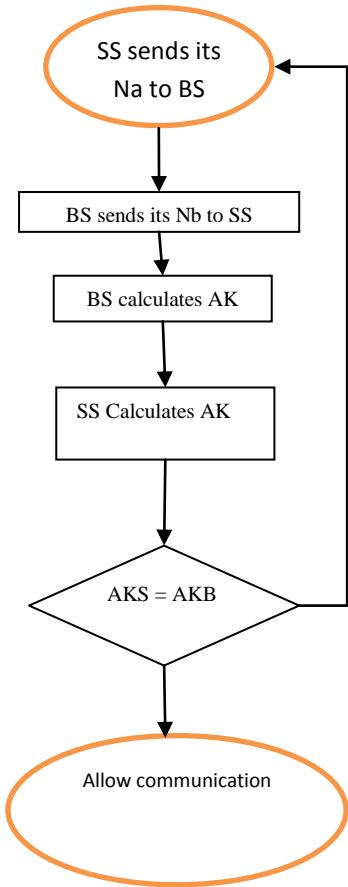
G and P are global variables called primes numbers

G is a primitive root of P.

'Na' and 'Nb' are the private keys of the MS and the BS respectively.

In the basic version of DH illustrated in figure 2.5, after the respective exchange of the public keys, the MS and the BS calculate the shared encryption key as shown in the equations 2.1 and 2.2. In order to implement mutual authentication, AS sends Na to BS, BS calculates AKB [11] [18]. BS then sends another unique number Nb to SS. Similarly, SS calculates AKS. If AKS is equal to AKB, AS believes this message sent by BS [8]. The AK in both SS and BS is calculated as follows:

$$AK = GNb \bmod P = GNa \bmod P \quad 2.3$$



**Figure 2.5: DH flow Diagram**

## 2.5 Related Work

PKMv1 authentication protocol has been used to boost security in the WIMAX network [1]. This protocol uses X.509 to decide whether a subscriber station is trusted. This protocol uses messages exchanged between the BS and the SS to carry out authentication. Mutual authentication is missing in this PKMv1 and this makes the SS vulnerable to rogue BS attacks [20]. The latest standard, IEEE 802.16e-2005, includes a new version (PKMv2) of the protocol that caters for the shortcomings of the first version. PKMv1 does not have a capacity for mutual

authentication [4] [12] [17]. The PKMv2 has mutual authentication that takes place after exchange of essential management information. A rogue station can use the management information from the BS to launch an attack on SS [12]. For this reason, DH protocol is suggested to create mutual authentication prior to the exchange of management information. Diffie-Hellman is not an encryption mechanism that encrypts data. Instead, it is a method to securely exchange the keys that encrypt data [26]. Diffie-Hellman accomplishes this secure exchange by creating a shared secret between two devices [27]. The shared secret then encrypts the symmetric key. Previous encryption versions used symmetric keys which were to be known by both communicating devices [26]. However, the process of transferring the key from one device to another was difficult because of MITM. It was very difficult to achieve secure transmission of the encryption key [26]. Advanced DH targeted asymmetric key transmission where secret and public information would be transferred between the two SS and BS [4] without fear of attacker. MITM could not utilize the public key because it only became useful after obtaining solution to the algorithm.

## **2.6 Summary**

PKMv1 was used in WIMAX to ensure that the SS is authenticated through the exchange of X.509 certificates. This first version lacks mutual authentication. PKMv2 was then applied in the WIMAX network to ensure that mutual authentication is done to book safety. In this second version of PKM, the management information from the BS and the SS are exchanged between the two stations in the mutual authentication process and this creates a security risk. A rogue base station can tap the management information and consequently pose as a genuine BS to

launch attacks on SS. DH protocol boosts security by introducing mutual authentication between two sites that have no prior knowledge of one another.

## CHAPTER THREE

### METHODOLOGY

#### 3.1 Proposed Protocol

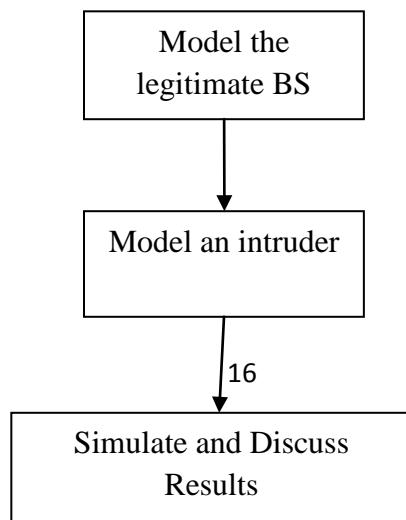
Figure 3.1 illustrates the implementation procedure of the proposed protocol. The legitimate BS is modeled using the Dev C ++ program. The modulus function illustrated in the equation 2.3 is the one programmed as illustrated in Appendix One.

$$AK = GNb \bmod P = GNa \bmod P$$

3.1

In the above equation, AK is the authentication key obtained by the SS and the BS. The equation is known only by the SS and the BS but not any rogue BS. In the first code, the formula is loaded into the SS and the BS. The two parties communicate one to another in an attempt to solve the algorithm  $AK = GNa \bmod P$ .

Equation 2.3 illustrates the DH algorithm to be solved by SS and BS to come up with a shared authentication key. The algorithm is programmed as illustrated in appendix one and results obtained. An intruder BS is then modeled by introducing an error in the DH protocol. This second algorithm is then programmed as illustrated in appendix two and results analyzed. The flow chart of this implementation is as shown below.



### **Figure 3.1: Implementation process chart**

Proposed Protocol

A modified version is used for the purposes of this simulation as shown below

$$S = B^a \bmod p = A^b \bmod p \quad 3.2$$

In the above equation, SS and BS selects a number a and b. These numbers are used in stage one calculation. The solution obtained in stage one is sent across the unsecured network. In the code, stage one calculation is obtained as follows:

SS selects 6 and BS selects 15.

B and P are set at 5 and 23 respectively.

The initial stage of calculation is

$$S = 5^6 \bmod 23 = 8$$

At the same time, the BS selects its number 15 and calculates as follows

$$S = 5^{15} \bmod 23 = 19$$

After stage one calculation the results of SS and BS are passed through the unsecure environment to BS and SS respectively. The obtained result is further used in the calculation to obtain the actual AK.

The code runs these calculations and transfers the answer from one station to another for stage two computations. In this case, SS receives number 19 from BS which also receives 8.

The second scenario is as follows:

$$AK_s = 19^6 \bmod 23 = 2$$

$$AK_b = 8^{15} \bmod 23 = 2$$

In the above scenario, the AK obtained by the SS and the BS are the same. This is the procedure that takes place for the case of a genuine BS. The code is made to allow up to five trials to cater for errors in integer selection. A primitive integer is used for this programming because of its ease of computation and simplicity. Integers also come in whole numbers.

### **3. 2 Modeling an intruder**

The initial knowledge of the Intruder is made up of the identities of the participating principals in the protocol run and its own identity and generic data, developed from scratch. The modeled Intruder process with MITM capabilities intercepts all the sent messages in the protocol run so as to increase its knowledge for attack launch. The results generated from the respective nonces and cryptographic function as a measure of the principal's validity, are easily intercepted by the Intruder since any measure of confidentiality to the critical information is lacking. This information can be used by the Intruder to fool the legitimate principals that they are actually communicating with each other yet they are exchanging messages with the Intruder and thus the MITM attack. Authentication will fail in the presence of an intruder. More time will be taken to accomplish authentication. The system will be set to limit the number of times an authentication request is sent from one station.

Parameters of interest during simulation include

- ❖ Time taken for authentication
- ❖ Number of attempts of authentication
- ❖ The number of successful authentications
- ❖ The number of failed authentications

The MITM intruder is modeled by the code in 7.2

The intruder modeling is done by introducing an error in the code as shown below

$$S = cB^a \bmod p = cA^b \bmod p \quad 3.3$$

The above equation is almost similar to the DH algorithm but has a bug

The algorithm computations for the intruder take the form illustrated below

$$S = cB^a \bmod p = cA^b \bmod p$$

Where  $c = 3$ ,  $p = 23$  and base  $g = 5$ .

The initial entry is

$$S = 3 * 5^6 \bmod 23 = 24$$

$$B = 5^{15} \bmod 23 = 57$$

Second stage calculation comes to

$$AK_s = 3 * 57^6 \bmod 23 = 27$$

$$AK_b = 3 * 24^{15} \bmod 23 = 3$$

This means that the results obtained by the SS and the BS are different. This difference means that the BS has no knowledge of the algorithm. Even if the simulation allows for five trials, the BS would never arrive at the accurate solution. Introduction of the error is the modification on the DH algorithm. The code is set to have five runs to cater for errors in integer selection. Dev

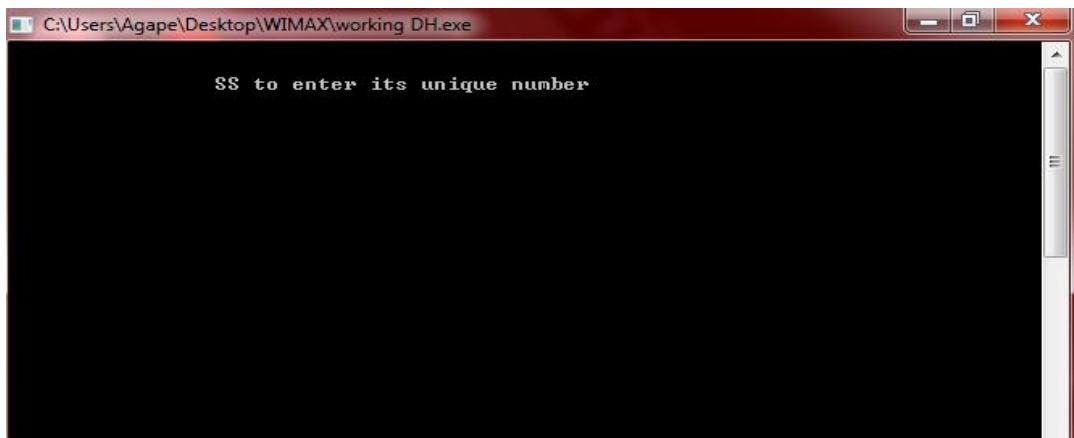
C++ is used in this implementation because it is easy to access and has DH features that can run a mutual authentication prior to exchange of the essential management information certificates. Running the dev C++ code is free because of its open source nature. Wireless simulator is another implementation option but has cost implementations. A comparison is done between the PKMv2 that of a successful DH authentication on the code and the results discussed as illustrated in the next chapter.

## CHAPTER FOUR

### RESULTS AND DISCUSSIONS

#### 4.1 Legitimate BS case

The modeled BS responded to the algorithm in a series of communication illustrated in the following result displays. In figure 4.1 illustrated below, the algorithm provokes the Subscriber station to enter its first unique number.



**Figure 4.1: Request for SS entry**

This number is the first nonce selected. The DH algorithm initiates the communication between the SS and the BS. The BS and the SS must solve equation 2.3 also written below:

$$AK = GNb \bmod P = GNa \bmod P \quad 4.1$$

As previously stated, the algorithm is known only to genuine systems. This can be loaded at the point of manufacture. An attacker is not able to get the right solutions to the algorithm. Such an attacker is therefore completely locked out of the network. Only a genuine SS and BS can get the right solution to the algorithm.

```
SS to enter its unique number  
2  
the first SS computation result is  
4  
BS to enter its first unique number
```

**Figure 4.2: Requests for BS Entry**

The SS selects 2 as shown in figure 4.2 above. The selected number is used to calculate the SS side of equation 4.1. The SS gets 4 as the computation result. The SS then sends a request to BS to send its first unique number. The number 4 is obtained as a result of correct computation of the DH algorithm by the SS. This number is not a final DH solution but would be used for subsequent application.

```
C:\Users\Agape\Desktop\WIMAX\working DH.exe

SS to enter its unique number
2
the first SS computation result is
4
BS to enter its first unique number
3

the BS computation result is:
6
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
1

The authentication key obtained from BS is:
1
value is the same mutual authentication has been obtained
continue with communication the network is secure

Process exited after 91.97 seconds with return value 0
Press any key to continue . . .
```

**Figure 4.3: Successful authentication**

In figure 4.3, the BS sends number 3 as its first unique number. The received number is used to compute equation 4.1. Number 6 is the result. The results from SS and BS are passed across the insecure network to compute the AK of the DH algorithm in equation 2.1. In the PKMv2, the BS sends management information to the SS for verification. This management information like the manufacturer certificates are tapped by the rogue BS for launch of attack at a later time. In the DH protocol however, the BS sends a number which is a computation result. A rogue BS can see the number but make no sense of it because it lacks knowledge of the DH algorithm to get a solution. Both SS and BS arrive at the AK the result. The algorithm then queries whether the AKs obtained are equal. For equal AKs, the network is made accessible for communication. Differing results bring about a repeat of the authentication process. Figure 4.3 illustrates that both the SS and the BS obtain a result of 1. These two stations arrive at the same number as a

proof that they have a legitimate DH protocol installed in them. The protocol then allows access to the network after agreement on the Authentication key. A rogue base station will never get the right solution to the DH algorithm and this means that the network is secure. In the event that the right BS selects a wrong number, it has a chance to reenter another number for authentication.

```
C:\Users\Agape\Desktop\WIMAX\working DH.exe

SS to enter its unique number
2
the first SS computation result is
4
BS to enter its first unique number
32

the BS computation result is:
-2
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
4

The authentication key obtained from BS is:
-2
mutual authentication not obtained. try again
SS to enter its unique number for second attempt
-
```

**Figure 4.4: Differing Authentication keys**

This figure 4.4 illustrates the DH algorithm solution. SS and BS select their unique numbers which are used to solve the algorithm. 2 and 32 are the SS and BS selected numbers. SS and BS obtain 4 and -2 as the first computation results. AKs result obtained are 4 and -2. It means that SS and BS authentication keys are not the same. DH algorithm rejects access requests and allows

the SS and the BS to solve the algorithm a second time. This is the first illustration of a case when a legitimate BS misses to gain access to the SS network. The DH protocol grants a second chance to the BS to reenter its unique number to be solved by the algorithm. The choice of the unique number must fit within the set boundaries of the DH algorithm. Missing the right unique number can be corrected in subsequent entry permissions.

```
C:\Users\Agape\Desktop\WIMAX\working DH.exe

the authentication key obtained from SS is:
4

The authentication key obtained from BS is:
-2
mutual authentication not obtained. try again
SS to enter its unique number for second attempt
2

the first SS computation result is
4
BS to enter its second unique number for attempt two
5

the BS computation result is:
3
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
2

The authentication key obtained from BS is:
2
value is the same mutual authentication has been obtained continue
with communication the network is secure

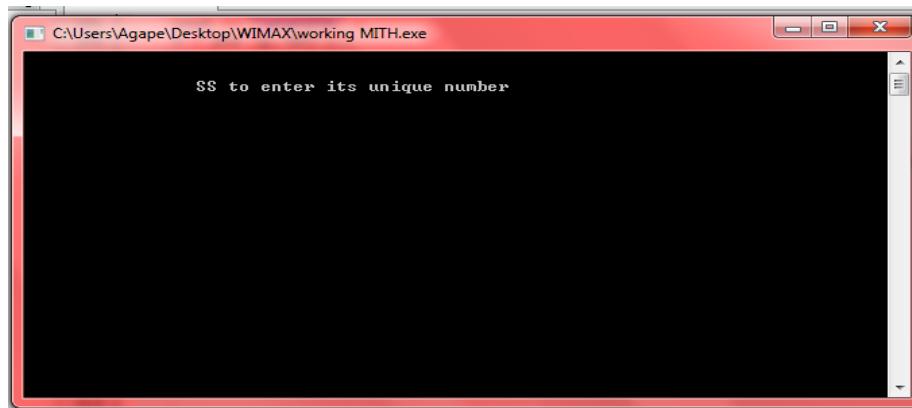
Process exited after 35.55 seconds with return value 0
Press any key to continue . . .
```

**Figure 4.5: AK obtained at second attempt**

This figure 4.5 illustrates a second attempt by the SS and the BS to solve the DH algorithm. The AKs obtained is 2 and is similar for both SS and BS. Access to the network is allowed. In the event that a legitimate BS selects a unique number that is not in the range of the DH protocol, a second chance is granted. The algorithm follows the same procedure to solve the DH protocol. In

this case, both SS and the BS arrive at 2. Access to the network is granted because of the rhyme of the AK obtained by the two stations. Further, smaller time is taken to solve the algorithm. A legitimate BS takes very short time to get the right authentication key.

#### 4.2 Rogue BS case



**Figure 4.6: Rogue BS requests SS for unique number**

This figure 4.6 illustrates the beginning of another authentication process between SS and MITH (rogue BS). At this point, the SS responds to the inquiry without knowing that an attack is being launched. The only salvation would be on the basis of accurate solution of the DH algorithm.

```
C:\Users\Agape\Desktop\WIMAX\working MITH.exe
SS to enter its unique number
2
the first SS computation result is
4
BS to enter its first unique number
5

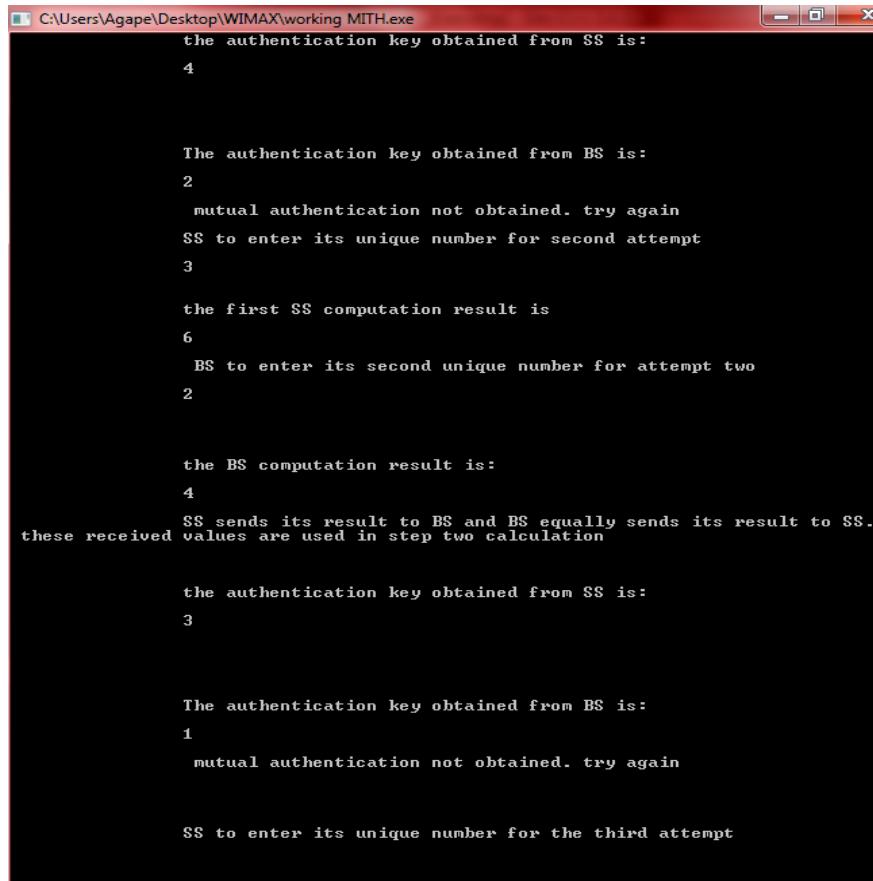
the BS computation result is:
3
SS sends its result to BS and BS equally sends its result to SS..
these received values are used in step two calculation

the authentication key obtained from SS is:
4

The authentication key obtained from BS is:
2
mutual authentication not obtained. try again
SS to enter its unique number for second attempt
```

**Figure 4.7: First Rejection of BS**

In this figure 4.7, the BS and the SS obtain different AK keys and try a second solution to the DH algorithm. In the first attempt, BS obtains 2 while the SS obtains 4 as the AK. The protocol requests for a second selection of unique numbers after failure in the first attempt. The room for trial is given to cater for errors that could be made by a legitimate BS in the selection of the unique numbers.



The screenshot shows a terminal window titled 'working MITH.exe' with the path 'C:\Users\Agape\Desktop\WIMAX'. The window displays a text-based communication between two entities, likely a Station (SS) and a Base Station (BS), using the Diffie-Hellman (DH) protocol for key exchange.

```
C:\Users\Agape\Desktop\WIMAX\working MITH.exe
the authentication key obtained from SS is:
4

The authentication key obtained from BS is:
2
mutual authentication not obtained. try again
SS to enter its unique number for second attempt
3

the first SS computation result is
6
BS to enter its second unique number for attempt two
2

the BS computation result is:
4
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
3

The authentication key obtained from BS is:
1
mutual authentication not obtained. try again

SS to enter its unique number for the third attempt
```

**Figure 4.8: Second Attempt Rejection**

The second solution to the algorithm also reveals different AKs as shown in figure 4.8. SS gets 3 while BS gets 1. Access to the network is not granted because of the difference in results. The algorithm allows a third attempt at the solution. The DH protocol only grants access to the SS network upon the realization of same AK by the two stations. For this case, the SS network is still secure because only a legitimate BS can crack the right solution to the DH protocol. A third chance is allowed just in case a legitimate BS could have selected a wrong unique number.

```
C:\Users\Agape\Desktop\WIMAX\working MITH.exe
4

The authentication key obtained from BS is:
6
Warning! mutual authentication not obtained. Two more attempts
remaining! try again

SS to enter its unique number for fourth attempt
5

the first SS computation result is
3
BS to enter its unique number for attempt four
7

the BS computation result is:
5
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
1

The authentication key obtained from BS is:
-2
Mutual authentication not obtained. Last Trial
SS to enter its unique number for Last attempt
-
```

**Figure 4.9: Third attempt rejection**

The AKs obtained at third attempt are not similar. DH requests for a fourth and second last attempt. The algorithm issues a warning to the authenticating parties that only two attempt are remaining as illustrated in figure 4.9. On the fourth trial, a wrong set of AKs is obtained. The last trial is allowed. This result illustrates that a rogue BS has a lot of difficulty in getting solution to an authentication key that is similar to that of the SS. The DH protocol is set to help only legitimate BS to gain access to the SS network. A rogue BS takes a lot of time in getting the solutions besides having several attempts at the algorithm.

```
C:\Users\Agape\Desktop\WIMAX\working MITH.exe

the authentication key obtained from SS is:
1

The authentication key obtained from BS is:
-2
Mutual authentication not obtained. Last Trial
SS to enter its unique number for Last attempt
4

the first SS computation result is
2
BS to enter its unique number for Last attempt
8

the BS computation result is:
4
SS sends its result to BS and BS equally sends its result to SS.
these received values are used in step two calculation

the authentication key obtained from SS is:
2

The authentication key obtained from BS is:
-2
NETWORK BLOCKED!!!

Process exited after 84.65 seconds with return value 0
Press any key to continue . . .
```

**Figure 4.10: Blocking of the network**

The AKs obtained are 2 and -2. The network is blocked at this fifth non successful attempt on the algorithm. The third and the fourth attempts to solve the AK were not successful. The MITH obtains wrong solution despite trying for the fourth time. The DH algorithm blocks the network after the fifth attempt as illustrated in figure 4.10. No matter how many times the MITH attempts the DH; it is difficult to obtain matching AKS.

As previously stated, the algorithm is known only to genuine systems. This can be loaded at the point of manufacture. An attacker is not able to get the right solutions to the algorithm. Such an attacker is therefore completely locked out of the network. Only a genuine SS and BS can get the right solution to the algorithm. The DH protocol results are transmitted across the un-secured

network to the other communicating party. There is no danger of sending the unique numbers because only the legitimate BS will have the ability to get right solution to the algorithm. The DH protocol inquires to establish if the AK results obtained by the SS and the BS are the same [8]. If so, the network is cleared for transfer of essential contents. If not, the network is blocked and a warning issued to the SS. It should be noted that most threats are always targeted towards the SS. This is because the SS is the end client. Such end clients can be banks, schools and other business organizations.

The DH mutual authentication protocol can be designed to allow for several trials before ultimate termination of the network. This means that a genuine BS might fail to select proper numbers in its first attempt. The concept of several trials has been used in many authentication protocols [12]. The phone SIM cards for example have up to three chances for authentication. The DH protocol in this thesis has been set to allow for five attempts. The selected numbers must be primitive integers. In the event that mismatching AK solutions are obtained, the DH protocol resets again and starts the process all over again. The SS and the BS are allowed to enter other numbers for further trials.

A Man in the middle attack can be modeled by a slight distortion of the DH algorithm. Figure 4.6 illustrates the entry point of an attacker. The mutual authentication procedure is the same as a genuine set up. The attacker BS protocol queries the SS demanding for its first nonce. The SS is thus provoked to enter its first number choice to be used for the DH algorithm calculations. The MITH always appears genuine on the surface. No one can detect that a signal is from an attacker. The DH protocol calculation comes in to help detect fraud. No matter how many times a rogue

BS tries to solve the DH algorithm, it cannot get a matching authentication key as the subscriber station.

#### **4.3 Validation of the DH protocol**

For validation purposes, a comparison is made between the successful DH protocol solution in figure 4.3 and that of authentication procedure of PKMv2 in figure 2.4 [8]. The figure 4.3 illustrates a mutual authentication in which no management information is being required in the authentication process. The SS and the BS only require the ability to come up with the right solution to the DH algorithm in order to authenticate each other. On the other hand, figure 2.4 shows a mutual authentication process of PKMv2 in which management information is being transferred even before mutual authentication is done. In this PKMv2 system, SS details like manufacturer's certificate ( $MCer_{ss}$ ), Subscriber certificate ( $Cer_{ss}$ ) and Base station certificate ( $Cer_{BS}$ ) are exchanged in a non secured environment [12]. These management information can easily be tapped by rogue BS to launch attacks on the network at a later time. It is for this reason that DH protocol has been implemented in this research to enable mutual authentication without the use of management information. In figure 4.3, the SS and the BS only exchange unique numbers that make no sense to any eavesdropper in the network. The Unique numbers are then used to calculate the AK for both the BS and the SS. Successful calculation of the AK results into mutual authentication

## CHAPTER FIVE

### CONCLUSION

#### **5.1 Conclusion**

With the deployment of wireless communication in recent years, security issues in wireless networks also become a growing concern. Privacy or confidentiality is fundamental for secure communication, which provides resistance to interception and eavesdropping. Message authentication provides integrity of the message and sender authentication, corresponding to the security attacks of message modification and impersonation. Even if WiMAX technology has complex authentication and authorization methods and a very strong encryption technique, it is still vulnerable to different attacks or threats.

The results show that the DH protocol works to keep away any intruder in the network. In this research, the DH succeeds in keeping the management information away from rogue BS. Previously, PKMv2 carried out authentication but the management information was left to leak into unsecure network. This research ensures that mutual authentication is done but without involving the management information. The WiMAX network is consequently made more secure when the management information is omitted from the system.

Diffie Hellmann protocol algorithm introduces mutual authentication between the BS and SS prior to the exchange of any management information. WiMAX is selected for this research because it is a recent technology and is presently being rolled out in many parts of the world because of its broadband capacities. This technology provides an environment for many gadgets to communicate. A rogue BS can pose as a genuine BS to fool the SS equipments. DH protocol is consequently relevant in WiMAX since it allows for mutual authentication prior to the exchange of sensitive network information.

DH protocol alone may not provide sufficient security for the Wimax network. Other cryptographic methods like RSA (Rivest, Shamir, and Adelman) and PKMv2 can be used to strengthen the security in the network. DH has the advantage of carrying out mutual authentication prior to the exchange of the management information details. Disruption of communication by an attacker often results into great loses in businesses. Network security is therefore very important.

## **5.2 Future work**

Most authentication keys get cracked five to ten years after implementation in a system. Future research should aim at improving the AK used in DH to help maintain security. Research should also target more advanced Security measures in WIMAX such as attacks launched through files and internet document transmissions. Future works should focus on the use of DH in the wireless sensor network gateways to enhance security.

## REFERENCES

- [1] E. Yuksel," Analysis of the PKMv2 Protocol in IEEE 802.16e-2005 Using Static Analysis", Technical Paper at University of Denmark, pp, 45-54, Feb 2007.
- [2] N. Li," Research on Diffie-Hellman Key Exchange Protocol", *IEEE transactions on security*, vol.23, no 5, 2010.
- [3] L. Han,"a threat analysis of the Extensible Authentication Protocol", *IETF, RFC 4286*, April 2006.
- [4] J. Hur, H. Shim, P. Kim, H. Yun, N. Oak song, "security considerations for handover schemes in mobile WiMAX networks", *Wireless Communications and Networking Conference*, April 2008. Las Vegas, USA WCNC 2008.
- [5] A. Taha, A. Abdel hamid, A. Tahar," formal verification of IEEE 802.16 security sub layer using scyther tools", ESRGroups France, 2009.
- [6] Z. You, X. Xie, W. Zheng," Verification and Research of a WiMAX authentication protocol Based on SSM", ICETC, Shanghai, China, 2010.
- [7] A, Deininger, S, Kiyomoto, Security Vulnerabilities and Solutions in Mobile WiMAX, *International Journal of Computer Science and Network Security*, vol. 7 no. 11, November 2007, 7-15.
- [8] M. Bogdanoski, P.Latkoski, A.Risteski, B.Popovski," IEEE 802.16 Security Issues: A Survey", *16<sup>th</sup> Telecommunication forum*, Belgrade Serbia, 25<sup>th</sup> -27<sup>th</sup> November, 2008.
- [9] H. Tseng, R. Hong, W. Yang,"A chaotic maps-base key agreement protocol that preserves user anonymity", *IEEE International Communication Conference*, 2009.

- [10] M. Shaikhan, A. Sobhani, M. E. Kalantari," Modification of Mobile Web Shopping Protocol Using GAA and Analysis by Colored Petri Nets", *Science Academy Transactions on Computer and Communication Networks*, 2011.
- [11] K. Jensen, Thesis: " Coloured Petri Nets and CPN Tools for Modeling and Validation of Concurrent Systems", Department of Computer Science, University of Aarhus, 2008.
- [12] J. Huang, C. Tser," Secure Mutual Authentication Protocols for Mobile Multi-hop Relay WiMAX Networks against Rogue Base/Relay Stations" *Journal of Electrical and Computer Engineering*; NY, USA, vol 11, pg, 121-128, (2011).
- [13] S .Sidharth, M. P. Sebastian," A Revised Secure Authentication Protocol for IEEE 802.16 (e)", *International Conference on Advances in Computer Engineering*, Bangalore, India, pp 34-42, 2010.
- [14] M. Holbal, T, Welzer," An Improved Authentication Protocol Based on One-Way Hash Functions and Diffie-Hellman Key Exchange", *International Conference on Availability, Reliability and Security*, Fukuoka, Japan, 2009.
- [15] F. Leu, Y. Huang, C. H.Chiu," Improving security levels of IEEE802.16e authentication by Involving Diffie Hellman PKDS", *International Conference on Complex, Intelligent and Software Intensive Systems*, Tunghai, Taiwan, pp 67-74, 2010.
- [16] E. Liu, K. Huang and L. Jin,"the design of trusted access scheme base on identity for WiMAX network" *IEEE computer society (International Workshop on Education Technology and Computer Science)*, Tehran, Iran, 2009.
- [17] M. Bogdanoski, P. Latkoski, A. Risteski, and B. Popovski, "IEEE 802.16 Security Issues: A Survey," in *16th Telecommunications forum TELFOR 2008*, Belgrade, Serbia, 2008.

- [18] M. Barbeau, "WiMAX/802.16 Threat Analysis," in *Proceedings of ACM Q2SWinet'05*, Montreal, Quebec, Canada, 2005, pp. 8-15.
- [19] K. C. Chen, J. Boberto and B. De Marca, Mobile WiMAX. John Wiley & Sons Ltd, 2008.
- [20] T. Han, N. Zhang, K. Liu, B. Tang, and Y. Liu, "Analysis of Mobile WiMAX Security: Vulnerabilities and Solutions," in *IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2008, NC, USA, pp. 828-833.
- [21] E.M. Clarke, O. Grumberg, and D.A. Peled, Model Checking in wireless networks. The MIT press, 1999.
- [22] A. M. Taha, A.T. Abdel, and S.Tahar, "Formal Verification of IEEE 802.16 Security Sublayer Using Scyther Tool," in *IEEE International Conference on Network and Service, N2S '09*, 2009, Amsterdam, Netherlands, pp. 1-5.
- [23] P. Narayana et al., "Automatic Vulnerability Checking of IEEE 802.16 WiMAX Protocols through TLA+," in *Proceedings of the 2nd IEEE Workshop on Secure Network Protocols*, November, 2006, Warsaw, Poland, pp. 44-49.
- [24] R. K. Guha, Z. Furqan, and S. Muhammad, "Discovering Man-In-The-Middle attacks in authentication protocols," in MILCOM 2007, Orlando, FL, October 29-31, 2007.
- [25] B. Diffie and M. Hellman, "An overview of Public Key Cryptography", in *IEEE communication magazine*, November 1978, vol 16, no. 6.
- [26] M. Nidhal and N. Achir. "A diffie Hellman Key generation for secure communications", *Consumer Communications & Networking Conference (CCNC)*, Las Vegas, USA, 2016 13<sup>th</sup> Annual conference.

[27] P. Joshi and M. Verma. Secure authentication approach using Diffie-Hellman key exchange algorithm for WSN, *International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, Kumaracoil, India, 2015.

## Appendices

### Appendix One: The Diffie Hellman mutual authentication code

```
#include <iostream>
#include <math.h>
using namespace std;

int main ()

{

cout << "\n\n\t\t" ;
int g, a, p, b, b2, A, X, B, Y, C, D, E, F, K1, K2 ;
cout << "SS to enter its unique number\n\n";
g = 5;
p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture
cout << "\n\n\t\t" ;
cin >> a ; // the SS selects its unique number for computation
cout << "\n\n\t\t\t\t" ;
A = pow (g, a);
cout << "the first SS computation result is \n\n\t\t" ;
B = A % p ;
cout << B ; // this is the value obtained by the SS
cout << "\n\n\t\t";
```

```

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its first unique number \n\n\t\t" ;
cin >> b ; // BS selects its unique number for computation

cout << "\n\n" ;

C = pow (g, b) ;

cout << "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t";
cout << D ;
cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;
cout << K1 ;
cout << "\n\n" ;

F = pow (C, a);

cout << "\n\n" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

```

```

cout << "The authentication key obtained from BS is: \n\n\t\t" ;
cout << K2;
cout << "\n\n\t\t" ;

if ( K1 == K2)
{
    cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}
else
{
    cout << " mutual authentication not obtained. Try again\n\n\t\t" ;
    cout << "SS to enter its unique number for second attempt \n\n\t\t";
    g = 5;
    p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

    cin >> a ; // the SS selects its unique number for computation

    cout << "\n\n\t\t" ;

    A = pow (g, a);

    cout << "the first SS computation result is \n\n\t\t" ;

    B = A % p ;

    cout << B ; // this is the value obtained by the SS

    cout << "\n\n\t\t";
    // X is then sent by the SS across the network to the BS.
}

```

```

cout << " BS to enter its second unique number for attempt two\n\n\t\t" ;
cin >> b2 ; // BS selects its unique number for computation

cout << "\n\n" ;

C = pow (g, b2) ;

cout << "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t";

cout << D ;

cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n" ;

F = pow (C, a);

cout << "\n\n" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

```

```

cout << K2;

cout << "\n\n\t\t" ;

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " mutual authentication not obtained. try again\n\n\n\n\t\t" ;

cout << "SS to enter its unique number for the third attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for attempt three \n\n\t\t" ;

```

```

cin >> b2 ; // BS selects its unique number for computation

cout << "\n\n\t\t" ;

C = pow (g, b2) ;

cout << "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t" ;

cout << D ;

cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a);

cout << "\n\n\t\t" ;

K1 = E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

cout << K2;

```

```

cout << "\n\n\t\t" ;

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with

communication the network is secure\n\n\t\t" ;

}

else

{ cout << " Warning! Mutual authentication not obtained. Two more attempts remaining!

try again\n\n\n\t\t" ;

cout << "SS to enter its unique number for fourth attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at

the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for attempt four\n\n\t\t" ;

```

```

cin >> b2 ; // BS selects its unique number for computation

cout << "\n\n\t\t" ;

C = pow (g, b2) ;

cout << "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t" ;

cout << D ;

cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

cout << K2;

```

```

cout << "\n\n\t\t" ;

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " Mutual authentication not obtained. Last Trial \n\n\t\t" ;

cout << "SS to enter its unique number for Last attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for Last attempt \n\n\t\t" ;

cin >> b2 ; // BS selects its unique number for computation

```

```

cout << "\n\n\t\t" ;
C = pow (g, b2) ;

cout << "\n\n\t\t" ;
D = C % p;

cout << "the BS computation result is: \n\n\t\t";
cout << D ;
cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;
K1 = E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;
cout << K1 ;
cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;
K2 = F % p ;

cout << "\n\n" ;
cout << "The authentication key obtained from BS is: \n\n\t\t" ;
cout << K2;
cout << "\n\n\t\t" ;

```

```
if ( K1 == K2)

{

cout << "value    is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " NETWORK BLOCKED!!! \n\n\t\t" ;

}

}
```

## Appendix Two

### MITH code

```
#include <iostream>

#include <math.h>

using namespace std;

int main ()

{

cout << "\n\n\t\t" ;

int g, a, p, b, b2, A, X, B, Y, C, D, E, F, K1, K2 ;

cout << "SS to enter its unique number\n\n";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at

the point of manufacture

cout << "\n\n\t\t" ;

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t\t\t";

// X is then sent by the SS across the network to the BS.
```

```

cout << " BS to enter its first unique number \n\n\t\t" ;
cin >> b ; // BS selects its unique number for computation

cout << "\n\n" ;

C = pow (g, b) ;

cout << "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t";

cout << D ;

cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = a * E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n" ;

F = pow (C, a);

cout << "\n\n" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

```

```

cout << K2;

cout << "\n\n\t\t" ;

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " mutual authentication not obtained. try again\n\n\t\t" ;

cout << "SS to enter its unique number for second attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its second unique number for attempt two\n\n\t\t" ;

```

```

cin >> b2 ; // BS selects its unique number for computation

cout << "\n\n" ;

C = pow (g, b2) ;

cout <<      "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t";

cout << D ;

cout <<      "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = a * E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n" ;

F = pow (C, a);

cout << "\n\n" ;

K2 = F % p ;

cout << "\n\n\t\t" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

cout << K2;

```

```

cout << "\n\n\t\t" ;

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with

communication the network is secure\n\n\t\t" ;

}

else

{ cout << " mutual authentication not obtained. try again\n\n\n\n\t\t" ;

cout << "SS to enter its unique number for the third attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at

the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for attempt three \n\n\t\t" ;

cin >> b2 ; // BS selects its unique number for computation

```

```

cout << "\n\n\t\t" ;
C = pow (g, b2) ;

cout << "\n\n\t\t" ;
D = C % p;

cout << "the BS computation result is: \n\n\t\t";
cout << D ;
cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;
K1 = a * E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;
cout << K1 ;
cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;
K2 = F % p ;

cout << "\n\n\t\t" ;
cout << "The authentication key obtained from BS is: \n\n\t\t" ;
cout << K2;
cout << "\n\n\t\t" ;

```

```

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " Warning! mutual authentication not obtained. Two more attempts remaining!
try again\n\n\n\t\t" ;

cout << "SS to enter its unique number for fourth attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for attempt four\n\n\t\t" ;

cin >> b2 ; // BS selects its unique number for computation

```

```

cout << "\n\n\t\t" ;
C = pow (g, b2);

cout << "\n\n\t\t" ;
D = C % p;

cout << "the BS computation result is: \n\n\t\t";
cout << D ;
cout << "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;
K1 = a * E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;
cout << K1 ;
cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;
K2 = F % p ;

cout << "\n\n\t\t" ;
cout << "The authentication key obtained from BS is: \n\n\t\t" ;
cout << K2;
cout << "\n\n\t\t" ;

```

```

if ( K1 == K2)

{

cout << "value is the same mutual authentication has been obtained continue with
communication the network is secure\n\n\t\t" ;

}

else

{ cout << " Mutual authentication not obtained. Last Trial \n\n\t\t" ;

cout << "SS to enter its unique number for Last attempt \n\n\t\t";

g = 5;

p = 7; // the SS and BS agree to use same values of g and p. such an agreement is set at
the point of manufacture

cin >> a ; // the SS selects its unique number for computation

cout << "\n\n\t\t" ;

A = pow (g, a);

cout << "the first SS computation result is \n\n\t\t" ;

B = A % p ;

cout << B ; // this is the value obtained by the SS

cout << "\n\n\t\t";

// X is then sent by the SS across the network to the BS.

cout << " BS to enter its unique number for Last attempt \n\n\t\t" ;

cin >> b2 ; // BS selects its unique number for computation

cout << "\n\n\t\t" ;

```

```

C = pow (g, b2) ;

cout <<      "\n\n\t\t" ;

D = C % p;

cout << "the BS computation result is: \n\n\t\t";

cout << D ;

cout <<      "\n\n\t\t" ;

cout << "SS sends its result to BS and BS equally sends its result to SS. these received
values are used in step two calculation \n\n\t\t" ;

E = pow (D, a) ;

cout << "\n\n\t\t" ;

K1 = a * E % p ;

cout << "the authentication key obtained from SS is: \n\n\t\t" ;

cout << K1 ;

cout << "\n\n\t\t" ;

F = pow (C, a);

cout << "\n\n\t\t" ;

K2 = F % p ;

cout << "\n\n" ;

cout << "The authentication key obtained from BS is: \n\n\t\t" ;

cout << K2;

cout << "\n\n\t\t" ;

```

```
if ( K1 == K2)

{

cout << "value    is the same mutual authentication has been obtained continue with

communication the network is secure\n\n\t\t" ;

}

else

{

cout << " NETWORK BLOCKED!!! \n\n\t\t" ;

}

}

}
```

## **Appendix Three**

### **List of publications**

1. Mr. S. Oguta, Dr. S. Musyoki and Dr. K. Langat. (2014). SECURITY ANALYSIS OF WIMAX TECHNOLOGY 8th Egerton University International Conference, 26 to 28th March 2014.
2. Mr. S. Oguta, Dr. S. Musyoki and Dr. K. Langat. (2014). Diffie Hellman Application in Wimax Security. The 2014 Annual International SRI Conference May, 5-8<sup>th</sup> 2014.
3. Stephen Ochieng Oguta, Proff. S. Musyoki , Dr. K. Langat. Mutual Authentication in Wimax Security Using Diffie Hellman: IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) e-ISSN: 2278-1676, p-ISSN: 2320-3331, Volume 11, Issue 2 Ver. I (Mar. – Apr. 2016), PP 42-46.